

Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Государственное учреждение
«Администрация Парка высоких технологий»

Открытые семантические технологии проектирования интеллектуальных систем

OSTIS-2014

Open Semantic Technologies for Intelligent Systems

МАТЕРИАЛЫ
IV МЕЖДУНАРОДНОЙ
НАУЧНО-ТЕХНИЧЕСКОЙ КОНФЕРЕНЦИИ

(Минск, 20–22 февраля 2014 года)

УДК 004.822+004.89-027.31
ББК 32.813-02+73
О-83

Редакционная коллегия :

*В. В. Голенков (отв. ред.), Л. С. Глоба, Н. А. Гулякина, И. В. Ефименко, О. П. Кузнецов, Б. М. Лобанов,
Д. Ш. Сулейманов, А. А. Харламов, В. Ф. Хорошевский*

Организаторы конференции:

Учреждение образования «Белорусский
государственный университет информатики и
радиоэлектроники»
Российская ассоциация искусственного интеллекта
Государственное учреждение «Администрация Парка
высоких технологий» (Республика Беларусь)
Научно-технологическая ассоциация «Инфопарк»
Объединённый институт проблем информатики
Национальной академии наук Беларуси
Тверской государственный технический университет
Научно-исследовательский институт «Прикладная
семиотика» АН РТ
Институт информатизации образования Российской
академии образования
Международная ИТ-компания «Itransition»
Компания «Melesta»
Компания «Qulix Systems»

Digital-агентство «ARTOX media»
Компания «Речевые Технологии»
Компания «ВирусБлокАда»
Иностранное общество с ограниченной
ответственностью «Седон БЛР»
ООО «ИксБи Софтваре»
ЧП «ЭктСистемс»
СООО «Ясо Бэкап»
ООО «Европейский центр разработки программного
обеспечения»
ООО «АйтиРекс Групп»
ООО «Айпрувд софтвер»
ООО «Прикладные системы»
Иностранное ЧУП по оказанию услуг «ЗуСофт»
ЧУП по оказанию услуг «Фингерз медиа»

Техническая и информационная поддержка:

Международный журнал «Программные продукты и
системы»
Научный журнал «Информатика»
Научно-практический журнал для специалистов
«Электроника ИНФО»

Научно-практический журнал «Речевые технологии»
Научный журнал «Онтология проектирования»
Журнал «Программные системы и вычислительные
методы»
Научно-практический журнал «IT-Бел»

*Издание осуществлено по заказу государственного учреждения
«Администрация Парка высоких технологий»*

О-83 Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2014) : материалы IV междунар. науч.-техн. конф. (Минск, 20–22 февраля 2014 года)/ редкол. : В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2014. – 576 с.
ISBN 978-985-543-034-7.

Сборник включает прошедшие рецензирование статьи IV международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем».

Сборник предназначен для преподавателей высших учебных заведений, научных сотрудников, студентов, аспирантов, магистрантов, а также для специалистов предприятий в сфере проектирования интеллектуальных систем.

Материалы сборника одобрены Программным комитетом OSTIS-2014 и печатаются в виде, представленном авторами.

**УДК 004.822+004.89-027.31
ББК 32.813-02+73**

ISBN 978-985-543-034-7

© УО «Белорусский государственный
университет информатики
и радиоэлектроники», 2014

ОРГАНИЗАЦИОННЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Батура Михаил Павлович	ректор Белорусского государственного университета информатики и радиоэлектроники
Живицкая Елена Николаевна	проректор по учебной работе и менеджменту качества Белорусского государственного университета информатики и радиоэлектроники по учебной работе и менеджменту качества
Кузнецов Александр Петрович	проректор по научной работе Белорусского государственного университета информатики и радиоэлектроники по научной работе
Никульшин Борис Викторович	проректор по учебной работе и информатизации Белорусского государственного университета информатики и радиоэлектроники по учебной работе и информатизации
Шилин Леонид Юрьевич	декан факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники
Голенков Владимир Васильевич	заведующий кафедрой интеллектуальных информационных технологий Белорусского государственного университета информатики и радиоэлектроники
Гракова Наталья Викторовна	ассистент кафедры интеллектуальных информационных технологий Белорусского государственного университета информатики и радиоэлектроники
Сушко Оксана Ренгольдovна	начальник патентно-информационного отдела Белорусского государственного университета информатики и радиоэлектроники
Титович Анна Францевна	директор центра международного сотрудничества Белорусского государственного университета информатики и радиоэлектроники
Лихачевский Дмитрий Викторович	начальник Управления подготовки научных кадров высшей квалификации Белорусского государственного университета информатики и радиоэлектроники
Боярко Алла Викторовна	руководитель пресс-службы Белорусского государственного университета информатики и радиоэлектроники

ORGANIZATIONAL COMMITTEE

Mikhail Batura	Rector, Belarusian State University of Informatics and Radio Electronics
Elena Zhivitskaya	Vice-Rector for Academic Work and Quality Management, Belarusian State University of Informatics and Radio Electronics
Alexander Kuznetsov	Vice-Rector for Research and Development, Belarusian State University of Informatics and Radio Electronics
Boris Nikulshin	Vice Rector for Education and Information, Belarusian State University of Informatics and Radio Electronics
Leanid Shylin	Dean of the Faculty of Information Technologies and Control, Belarusian State University of Informatics and Radio Electronics
Vladimir Golenkov	Head of Informational Intelligent Technologies Department, Belarusian State University of Informatics and Radio Electronics
Natalia Grakova	Assistant of Intellectual Information Technologies Department, Belarusian State University of Informatics and Radio Electronics
Oksana Sushko	Head of Patent Information, Belarusian State University of Informatics and Radio Electronics
Anna Titovich	Director of the Center for International Cooperation, Belarusian State University of Informatics and Radio Electronics
Dmitry Likhachevsky	Head of training highly qualified scientific personnel, Belarusian State University of Informatics and Radio Electronics
Alla Boyarko	Head of Press Service, Belarusian State University of Informatics and Radio Electronics

ПРОГРАММНЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

д.т.н., проф.	Кузнецов Олег Петрович - ПРЕДСЕДАТЕЛЬ	Заведующий лабораторией, Институт проблем управления РАН, г. Москва, Россия
к.т.н., доц.	Боргест Николай Михайлович	Профессор, Самарский государственный аэрокосмический университет имени Академика С.П.Королева, г. Самара, Россия
д.т.н., проф.	Борисов Аркадий Николаевич	Профессор, Рижский технический университет, Институт информационных технологий, г. Рига, Латвия
д.т.н., проф.	Гаврилова Татьяна Альбертовна	Заведующая кафедрой информационных технологий в менеджменте, Высшая Школа менеджмента СПбГУ, г. Санкт-Петербург, Россия
д.т.н., проф.	Глоба Лариса Сергеевна	Заведующая кафедрой, Национальный технический университет Украины «Киевский политехнический институт», г. Киев, Украина
д.т.н., проф.	Голенков Владимир Васильевич	Заведующий кафедрой интеллектуальных информационных технологий, Белорусский государственный университет информатики и радиоэлектроники, г. Минск, Беларусь
д.т.н., проф.	Головко Владимир Адамович	Заведующий кафедрой интеллектуальных информационных технологий, Брестский государственный технический университет, г. Брест, Беларусь
д.фил.н., проф.	Гордей Александр Николаевич	Директор, Республиканский институт китаеведения имени Конфуция БГУ г. Минск, Беларусь
д.т.н., с.н.с.	Грибова Валерия Викторовна	Заведующая лабораторией интеллектуальных систем, Федеральное государственное бюджетное учреждение науки Институт автоматизации и процессов управления Дальневосточного отделения Российской академии наук, г. Владивосток, Россия
к.ф.-м.н., доц.	Гулякина Наталья Анатольевна	Заместитель заведующего кафедрой интеллектуальных информационных технологий, Белорусский государственный университет информатики и радиоэлектроники г. Минск, Беларусь
д.т.н., проф.	Еремеев Александр Павлович	Заведующий кафедрой прикладной математики, Национальный исследовательский университет «МЭИ», г. Москва, Россия
к.фил.н.	Ефименко Ирина Владимировна	Заместитель декана факультета филологии, НИУ ВШЭ; Ведущий научный сотрудник, ЦИАС ИСИЭЗ НИУ ВШЭ г. Москва, Россия
д.т.н., проф.	Заболеева-Зотова Алла Викторовна	Профессор кафедры САПРиПК, Волгоградский государственный технический университет, г. Волгоград, Россия
к.т.н., с.н.с.	Загоруйко Юрий Алексеевич	Заведующий лабораторией, Институт систем информатики им. А.П. Ершова СО РАН, г. Новосибирск, Россия
д.ф.-м.н., проф.	Клещев Александр Сергеевич	Главный научный сотрудник, Федеральное государственное бюджетное учреждение науки Институт автоматизации и процессов управления Дальневосточного отделения Российской академии наук, г. Владивосток, Россия
д.мед.н., проф.	Кобринский Борис Аркадьевич	Руководитель, Медицинский центр новых информационных технологий, г. Москва, Россия

д.п.н., проф.	Козлов Олег Александрович	Заместитель директора института, Учреждение Российской Академии Образования «Институт Информатизации Образования», г. Москва, Россия
д.т.н., проф.	Колесников Александр Васильевич	Профессор кафедры компьютерного моделирования и информационных систем, ФГАОУ ВПО Балтийский федеральный университет имени И.Канта, г. Калининград, Россия
д.т.н., проф.	Комарцова Людмила Георгиевна	Профессор кафедры «Компьютерные системы и сети», Московский Государственный технический университет им. Н.Э. Баумана (Калужский филиал), г. Калуга, Россия
д.т.н., проф.	Курейчик Виктор Михайлович	Заместитель руководителя по научной и инновационной деятельности, Таганрогский кампус ЮФУ, г. Таганрог, Россия
д.т.н., с.н.с	Ландэ Дмитрий Владимирович	Заведующий отделом, Институт проблем регистрации информации Национальной академии наук Украины, г. Киев, Украина
д.т.н., с.н.с.	Лобанов Борис Мефодьевич	Главный научный сотрудник, Объединённый институт проблем информатики НАН Беларуси, г. Минск, Беларусь
к.ф.-м.н.	Лукашевич Наталья Валентиновна	Ведущий научный сотрудник, Московский государственный университет имени М.В.Ломоносова, г. Москва, Россия
д.т.н., проф.	Массель Людмила Васильевна	Главный научный сотрудник, заведующая лабораторией, Институт систем энергетики им. Л.А. Мелентьева СО РАН, г. Иркутск, Россия
к.т.н.	Найденова Ксения Александровна	Старший научный сотрудник, Военно-медицинская академия им. С. М. Кирова, г. Санкт-Петербург, Россия
к.т.н., доц.	Невзорова Ольга Авенировна	Заместитель директора по научной работе, Научно-исследовательский институт «Прикладная семиотика» АН РТ, г. Казань, Россия
д.ф.-м.н., проф.	Осипов Геннадий Семенович	Заместитель заведующего кафедрой «Математические методы системного анализа», Национальный исследовательский университет «Высшая школа экономики», г. Москва, Россия
д.т.н., проф.	Палюх Борис Васильевич	Заведующий кафедрой Информационных систем, Тверской государственный технический университет, г. Тверь, Россия
д.т.н., проф.	Петровский Александр Александрович	Заведующий кафедрой ЭВС, Белорусский государственный университет информатики и радиоэлектроники, г. Минск, Беларусь
д.т.н., проф.	Петровский Алексей Борисович	Заведующий лабораторией, ФГБУН Институт системного анализа Российской академии наук, г. Москва, Россия
к.ф.-м.н., проф.	Плесневич Геральд Станиславович	Профессор кафедры «Информационные технологии», Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «МАТИ – Российский государственный технологический университет имени К.Э. Циолковского», Г. Москва, Россия
к.т.н., доц.	Родченко Вадим Григорьевич	Доцент кафедры программного обеспечения интеллектуальных и компьютерных систем, Учреждение образования «Гродненский государственный университет имени Янки Купалы», г. Гродно, Беларусь

к.фил.н., доц.	Рычкова Людмила Васильевна	Заведующий кафедрой общего и славянского языкознания, Гродненский государственный университет имени Янки Купалы, г. Гродно, Беларусь Профессор кафедры прикладной лингвистики, Белорусский государственный университет, г. Минск, Беларусь
д.т.н., проф.	Сидоркина Ирина Геннадьевна	Декан факультета информатики и вычислительной техники, Поволжский государственный технологический университет, г.Йошкар-Ола, Республика Марий Эл, Россия
д.т.н.	Смирнов Сергей Викторович	Директор, ведущий научный сотрудник, заведующий лабораторией анализа и моделирования сложных систем, Институт проблем управления сложными системами Российской академии наук, г. Самара, Россия
д.ф.-м.н., проф.	Соловьёв Сергей Юрьевич	Профессор, Московский государственный университет имени М.В.Ломоносова, г. Москва, Россия
д.т.н., проф.	Соснин Петр Иванович	Заведующий кафедрой «Вычислительная техника», Ульяновский государственный технический университет, г. Ульяновск, Россия
д.т.н., проф.	Стефанюк Вадим Львович	Главный научный сотрудник, Институт проблем передачи информации РАН, г. Москва, Россия
д.т.н., Академик АН РТ, проф.	Сулейманов Джавдет Шевкетович	Заведующий кафедрой, Казанский (Приволжский) федеральный университет, г. Казань, Россия
к.т.н., доц.	Тарасов Валерий Борисович	Доцент кафедры компьютерных систем автоматизации производства, Московский государственный технический университет им. Н.Э.Баумана, г. Москва, Россия
д.э.н., проф.	Тельнов Юрий Филиппович	Заведующий кафедрой, Московский государственный университет экономики, статистики и информатики, г. Москва, Россия
д.ф.-м.н., проф.	Тузиков Александр Васильевич	Генеральный директор, Объединенный институт проблем информатики НАН Беларуси, г. Минск, Беларусь
д.т.н.	Харламов Александр Александрович	Старший научный сотрудник, Институт высшей нервной деятельности и нейрофизиологии РАН, г. Москва, Россия
д.т.н., проф.	Хорошевский Владимир Федорович	Заведующий сектором, ВЦ им. А.А. Дородницына РАН Главный научный сотрудник, ЦИАС ИСИЭЗ НИУ ВШЭ, г. Москва, Россия
Академик НАН Беларуси, д.т.н., проф.	Чернявский Александр Фёдорович	Заведующий кафедрой интеллектуальных систем, Белорусский государственный университет, г. Минск, Беларусь
д.т.н., проф.	Шарипбаев Алтынбек Амирович	Директор научного центра «Искусственный интеллект», Евразийский национальный университет имени Л.Н.Гумилева, г. Астана, Казахстан
к.т.н., с.н.с., доц.	Щербак Сергей Сергеевич	Доцент, Кременчугский национальный университет имени Михаила Остроградского, г. Кременчуг, Украина
PhD, Associate Professor	WANG Lipo	School of Electrical and Electronic Engineering, Division of Information Engineering, Singapore

PROGRAM COMMITTEE

Borgest N.	c. of t.s., RF ass. proff.	Nevzorova O.	c. of t.s., RF ass. proff.
Borisov A.	d. of t.s., Latvia proff.	Osipov G.	d. of ph.-m.s., RF proff.
Chernyavsky A.	NAS Belarus academician	Palyukh B.	d. of t.s., RF proff.
Efimenko I.	c. of phyl.s., RF	Petrovsky A.	d. of t.s., RB proff.
Eremeev A.	d. of t.s., RF proff.	Petrovsky A.	c. of t.s., d. of t.s., RF proff.
Gavrilova T.	d. of t.s., RF proff.	Plesniewicz G.	c. of ph.-m.s., RF
Globa L.	d. of t.s., Ukraine proff.	Rodchenko V.	c. of t.s., RB ass. proff.
Golenkov V.	d. of t.s., RB proff.	Rychkova L.	c. of phyl.s., RB ass. proff.
Golovko V.	d. of t.s., RB proff.	Sharipbaev A.	d. of t.s., Kazakhstan proff.
Gribova V.	d. of t.s., RF	Shcherbak S.	c. of t.s., Ukraine ass. proff.
Guliakina N.	c. of t.s., RB ass. proff.	Sidorkina I.	d. of t.s., RF proff.
Hardzei A.	d. of phyl.s., RB proff.	Smirnov S.	d. of t.s., RF proff.
Kharlamov A.	d. of t.s., RF	Soloviev S.	d. of ph.-m.s., RF proff.
Khoroshevsky V.	d. of t.s., RF proff.	Sosnin P.	d. of t.s., RF proff.
Kleshev A.	d. of t.s., RF proff.	Stefanuk V.	d. of t.s., RF proff.
Kobrinskiy B.	d. of med.s., RF	Suleymanov D.	AS Tatarstan academician, KF
Kolesnikov A.	d. of t.s., RF proff.	Tarasov V.	c. of t.s., RF ass. proff.
Komartsova L.	d. of t.s., RF	Telnov Yu.	d. of e.s., RF proff.
Kozlov O.	d. of teach.s., RF proff.	Tuzikov A.	d. of ph.-m.s., RB proff.
Kureychik V.	d. of t.s., RF proff.	Wang Lipo	PhD, Singapore ass. Proff
Kuznetsov O.	d. of t.s., RF proff.	Zaboleeva-Zotova A.	d. of t.s., RF
Lande D.	d. of t.s., Ukraine	Zagorulko Yu.	c. of t.s., RF ass. proff.
Lobanov B.	d. of t.s., RB proff.		
Loukachevitch N.	c. of t.s., RF		
Massel L.	d. of t.s., RF proff.		
Naidenova K.	c. of t.s., RF		

СОДЕРЖАНИЕ

ПРЕДИСЛОВИЕ	23
О Викторе Владимировиче Мартынове	25
УСК МАРТЫНОВА – ТРИДЦАТЬ ЛЕТ СПУСТЯ Ефименко И.В., Хорошевский В.Ф.	29
ОТ СЕМАНТИЧЕСКОГО КОДА К КОГНИТИВНОЙ ЛИНГВИСТИКЕ, СЕМИОТИКЕ И ИНФОРМАТИКЕ: НАСЛЕДИЕ В.В.МАРТЫНОВА Тарасов В.Б	39
ТЕОРИЯ АВТОМАТИЧЕСКОГО ПОРОЖДЕНИЯ АРХИТЕКТУРЫ ЗНАНИЙ (ТАПАЗ- 2) И ДАЛЬНЕЙШАЯ МИНИМИЗАЦИЯ СЕМАНТИЧЕСКИХ ИСЧИСЛЕНИЙ Гордей А.Н.	49
СТРУКТУРИЗАЦИЯ СМЫСЛОВОГО ПРОСТРАНСТВА Голенков В. В., Гулякина Н.А.	65
ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС ИНТЕЛЛЕКТУАЛЬНОЙ МЕТАСИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ Корончик Д. Н.	79
БАЗА ЗНАНИЙ ИНТЕЛЛЕКТУАЛЬНОЙ МЕТАСИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ Гракова Н.В., Давыденко И.Т., Русецкий К.В.	83
МАШИНА ОБРАБОТКИ ЗНАНИЙ ИНТЕЛЛЕКТУАЛЬНОЙ МЕТАСИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ Шункевич Д.В.	93
ПОСТРОЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ, ОСНОВАННЫХ НА ЗНАНИЯХ, С ПОВТОРНЫМ ИСПОЛЬЗОВАНИЕМ КОМПОНЕНТОВ Борисов А.Н.	97
О РЕШЕНИИ ЗАДАЧ СНЯТИЯ ОМОНИМИИ ПРИ РАСПОЗНАВАНИИ И СИНТЕЗЕ РЕЧИ Лобанов Б.М., Житко В.А.	103
СИТУАЦИОННОЕ УПРАВЛЕНИЕ И СЕМАНТИЧЕСКОЕ МОДЕЛИРОВАНИЕ В ЭНЕРГЕТИКЕ Массель Л.В., Массель А.Г.	111
О ФОРМАЛИЗАЦИИ СЕМАНТИКИ ОБЛАСТЕЙ ЗНАНИЙ В ИНФОРМАЦИОННЫХ И ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ НА ОСНОВЕ ОНТОЛОГИЙ Загоруйко Ю.А., Загоруйко Г.Б.	117
КАКОЙ ДОЛЖНА БЫТЬ ПАРАДИГМА РЕШЕНИЯ ЗАДАЧ НА ОСНОВЕ БАЗ ЗНАНИЙ? Грибова В.В., Клещев А.С.	131
ПРОЕКТИРОВАНИЕ ИНЖЕНЕРНЫХ РАСЧЕТОВ ДЛЯ ПОРТАЛОВ ЗНАНИЙ Глоба Л.С., Новогрудская Р.Л.	137

ОДИН ЯЗЫК ДЛЯ СПЕЦИФИКАЦИИ ОНТОЛОГИЙ Плесневич Г.С.	143
ПЕРСОНАЛЬНАЯ ОНТОЛОГИЯ ПРОФЕССИОНАЛЬНОГО ОПЫТА Соснин П.И.	147
РОЛЬ ОНТОЛОГИИ В ПРОЕКТИРОВАНИИ ИНФОРМАЦИОННЫХ СИСТЕМ Боргест Н.М.	155
МОДЕЛИРОВАНИЕ ДИНАМИКИ ПРОЦЕССОВ ПОСЛЕДОВАТЕЛЬНОСТЬЮ ОДНОРОДНЫХ СЕМАНТИЧЕСКИХ СЕТЕЙ НА ОСНОВЕ АНАЛИЗА ПОСЛЕДОВАТЕЛЬНОСТИ ОПИСЫВАЮЩИХ ПРОЦЕСС ТЕКСТОВЫХ ВЫБОРОК Харламов А.А., Ермоленко Т.В., Жонин А.А.	161
ПРАВИЛА ФОРМИРОВАНИЯ ТЕРМИНОЛОГИЧЕСКИХ КЛАСТЕРОВ Мальковский М.Г., Соловьев С.Ю.	169
НАПОЛНЕНИЕ СЕМАНТИЧЕСКИХ СЛОТОВ РЕЛЯЦИОННО-СИТУАЦИОННОГО ФРЕЙМА НА ПРИМЕРЕ ТАТАРСКИХ СИНТАКСЕМ Сулейманов Д.Ш., Гатиатуллин А.Р.	173
СЕМАНТИЧЕСКИЕ ТЕХНОЛОГИИ ДЛЯ МАТЕМАТИЧЕСКИХ ЭЛЕКТРОННЫХ РЕСУРСОВ Невзорова О.А.	179
ПОДХОДЫ К ОПИСАНИЮ ОТНОШЕНИЯ ЧАСТЬ-ЦЕЛОЕ В ОНТОЛОГИЧЕСКИХ РЕСУРСАХ Лукашевич Н.В.	185
УСЛОВНО-РЕФЛЕКТОРНАЯ ОСНОВА ЗАПОМИНАНИЯ ЗНАНИЙ Стефанюк В.Л	191
САПР ТП С РАЗВИВАЮЩЕЙСЯ БАЗОЙ ЗНАНИЙ Бурдо Г.Б., Палюх Б.В., Воробьева Е.В.	195
ПРИМЕНЕНИЕ КГТВ-АЛГОРИТМА ДЛЯ НАУЧНЫХ ТЕКСТОВ Ландэ Д.В., Снарский А.А., Ягунова Е.В.	199
ПОСТРОЕНИЕ ИНТЕГРАЛЬНОГО ПОКАЗАТЕЛЯ ОЦЕНКИ ПЕРСПЕКТИВНОСТИ ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА Лобанов В.Н., Петровский А.Б.	205
НЕКОТОРЫЕ ЗАКОНОМЕРНОСТИ И ОБЪЕКТИВНЫЕ ОГРАНИЧЕНИЯ РЕАЛИЗАЦИИ АЛГОРИТМОВ СЕМАНТИЧЕСКОЙ ОБРАБОТКИ НА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ С МАССОВЫМ ПАРАЛЛЕЛИЗМОМ Вереник Н.Л., Сейткулов Е.Н., Гирель А.И., Татур М.М.	211
ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ УПРАВЛЕНИЯ НА БОРТУ ОТЕЧЕСТВЕННЫХ СПУТНИКОВ: ФАНТАСТИКА ИЛИ РЕАЛЬНОСТЬ Димитров Д.М., Данилин Н.С., Сабиров И.Х.	217
ОНТОЛОГИЯ РОССИЙСКОГО СЕГМЕНТА МЕЖДУНАРОДНОЙ КОСМИЧЕСКОЙ СТАНЦИИ И ЕЁ ПРАКТИЧЕСКОЕ ИСПОЛЬЗОВАНИЕ В ИНТЕЛЛЕКТУАЛЬНЫХ АЭРОКОСМИЧЕСКИХ ПРИЛОЖЕНИЯХ Вакурина Т.Г., Котеля В.В., Лахин О.И., Матюшин М.М., Скобелев П.О.	221

МОДЕЛИ ОЦЕНКИ УБЕЖДЕННОСТИ ОБ АДЕКВАТНОСТИ ПРЕДСТАВЛЕНИЙ О ПРЕДМЕТНОЙ ОБЛАСТИ В ЗАДАЧАХ ВЫБОРА	227
Виноградов Г.П., Филатова Н.Н.	
ИССЛЕДОВАНИЕ НЕЙРОСЕТЕВЫХ АЛГОРИТМОВ ОБУЧЕНИЯ В ИНТЕЛЛЕКТУАЛЬНЫХ ЭВОЛЮЦИОННЫХ СИСТЕМАХ	233
Комарцова Л.Г., Лавренков Ю.Н., Антипова О.В.	
СЕМАНТИЧЕСКИЕ АССОЦИАТИВНЫЕ РЕСУРСНЫЕ СЕТИ С ПЛЮРАЛЬНЫМИ ВЕРШИНАМИ	239
Жилякова Л.Ю.	
КОГНИТИВНОЕ МОДЕЛИРОВАНИЕ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ (СЕМИОТИЧЕСКИЙ ПОДХОД)	243
Кулинич А.А.	
МАТРИЧНОЕ ПРЕДСТАВЛЕНИЕ КОНЕЧНЫХ ПРЕДИКАТОВ ДЛЯ АВТОМАТИЗАЦИИ ЛОГИКО-СЕМАНТИЧЕСКОГО АНАЛИЗА	251
Зуенко А.А.	
КОНСТРУКТИВНАЯ ПРОПОЗИЦИОНАЛЬНАЯ ЛОГИКА С НЕПАРАДОКСАЛЬНОЙ ИМПЛИКАЦИЕЙ	257
Сметанин Ю.М.	
СМЫСЛОВОЕ СОДЕРЖАНИЕ ПОСЛЕДОВАТЕЛЬНОСТИ ПРОСТЫХ СУБЪЕКТНО- ПРЕДИКАТНЫХ СУЖДЕНИЙ	263
Сметанин Ю.М.	
ЛОГИКО-ЛИНГВИСТИЧЕСКАЯ МОДЕЛЬ ИДЕНТИФИКАЦИИ СЕМАНТИЧЕСКИХ ОТНОШЕНИЙ СУЩНОСТЕЙ СРЕДСТВАМИ АЛГЕБРЫ КОНЕЧНЫХ ПРЕДИКАТОВ	267
Хайрова Н.Ф., Узлов Д.Ю., Шаронова Н.В.	
СПИРАЛЬНАЯ КОГНИТИВНАЯ МЕТАДИНАМИКА: ИССЛЕДОВАНИЕ МАСШТАБИРУЕМОСТИ ДИНАМИЧЕСКИХ ПРОЦЕССОВ	271
Прокопчук Ю.А.	
СИТУАЦИОННЫЙ ПОЛИГОН КАК ИНСТРУМЕНТ СИТУАЦИОННОГО УПРАВЛЕНИЯ В ЭНЕРГЕТИКЕ	277
Массель А.Г., Иванов Р.А.	
ЭКСПЕРТНЫЕ СИСТЕМЫ БЕЗ ЭКСПЕРТОВ	281
Клещев А.С., Смагин С.В.	
СОДЕРЖАНИЕ СИСТЕМНОГО АНАЛИЗА ПРИ АВТОМАТИЗАЦИИ ИНТЕЛЛЕКТУАЛЬНОЙ ДЕЯТЕЛЬНОСТИ НА УРОВНЕ ОТРАСЛИ	285
Клещев А.С., Шалфеева Е.А.	
ИНТЕРНЕТ-КОМПЛЕКС ДЛЯ ПРЕОБРАЗОВАНИЙ КЛАССОВ СЕМАНТИЧЕСКИХ СЕТЕЙ	291
Тимченко В.А.	
ИЕРАРХИЯ НЕЧЕТКИХ ПОНЯТИЙ ДЛЯ ОПИСАНИЯ ИСТОРИЧЕСКОЙ ЛИЧНОСТИ	297
Глоба Л.С., Островский С.А., Попова М.А., Стрижак А.Е., Терновой М.Ю.	

СРЕДСТВА КОГНИТИВНОЙ ГРАФИКИ В ИНТЕЛЛЕКТУАЛЬНЫХ ОБУЧАЮЩЕ-ТЕСТИРУЮЩИХ СИСТЕМАХ	303
Янковская А.Е., Ямшанов А.В., Кривдюк Н.М.	
ТТР-СИСТЕМА: ИНТЕГРАЦИЯ ЕСТЕСТВЕННОГО ЯЗЫКА И ИЗОБРАЖЕНИЙ	309
Курбатов С.С., Литвинович А.В., Лобзин А.П., Хахалин Г.К.	
РАЗБОР ГРАФИЧЕСКИХ ИЗОБРАЖЕНИЙ ОНТОЛОГИЧЕСКИМ КОМПОНЕНТОМ ГИБРИДНОЙ СИСТЕМЫ РАСПОЗНАВАНИЯ	315
Хахалин Г.К., Курбатов С.С., Лобзин А.П.	
СЕМАНТИЧЕСКОЕ КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ ЕСТЕСТВЕННО-ЯЗЫКОВОГО ИНТЕРФЕЙСА ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ	321
Елисеева О.Е.	
МЕТАД ПАБУДОВЫ КАМПАНЕНТАЎ СІНТЭЗУ МАЎЛЕННЯ ПА ТЭКСЦЕ ДЛЯ НАТУРАЛЬНА-МАЎЛЕНЧАГА ІНТЭРФЕЙСА ПРЫ ДАПАМОЗЕ NOOJ	333
Гецэвіч Ю.С., Скопінава А.М., Окрут Т.І.	
ЗНАНИЕ-ОРИЕНТИРОВАННЫЕ СРЕДСТВА ПОДДЕРЖКИ СЕМАНТИЧЕСКОГО ПОИСКА В WEB	339
Рогушина Ю.В.	
ИСПОЛЬЗОВАНИЕ ОНТОЛОГИЧЕСКОГО ПОДХОДА К СЕМАНТИЧЕСКОМУ ПОИСКУ В СРЕДЕ ИНТЕРНЕТА ВЕЩЕЙ	353
Рогушина Ю., Гладун А., Андрушеви А., Курбатский А.	
УНИВЕРСАЛЬНЫЙ СЕМАНТИЧЕСКИЙ КОД КАК ТЕХНОЛОГИЯ АБСТРАГИРОВАНИЯ ДЛЯ РЕШЕНИЯ НЕЧЕТКО ПОСТАВЛЕННЫХ ТЕХНИЧЕСКИХ ЗАДАЧ	357
Канделинский С. Л., Бойко И. М.	
ПРИМЕНЕНИЕ ОНТОЛОГИЙ В ПРОЕКТНОМ УПРАВЛЕНИИ	363
Веремьев В.Л., Горовая Д.О.	
ОНТОЛОГИЧЕСКИЙ ПОДХОД К ПОСТРОЕНИЮ СИСТЕМЫ ИМИТАЦИИ, НАСТРАИВАЕМОЙ НА КОНКРЕТНУЮ ПРЕДМЕТНУЮ ОБЛАСТЬ	369
Замятина Е.Б., Михеев Р.А.	
ИНТЕГРАЦИЯ СИСТЕМ МОДЕЛИРОВАНИЯ НА ОСНОВЕ DSM-ПЛАТФОРМЫ С ИСПОЛЬЗОВАНИЕМ ОНТОЛОГИЙ	375
Сухов А.О., Лядова Л.Н., Замятина Е.Б.	
АРХИТЕКТУРА СИСТЕМЫ ОБРАБОТКИ ПУБЛИКАЦИЙ ПО ТЕМАТИКЕ МОДЕЛИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ	381
Ланин В.В., Чугунов А.П.	
ОНТОЛОГИЧЕСКИ-ОРИЕНТИРОВАННАЯ МОДЕЛЬ КЛАССИФИКАЦИЙ ТЕКСТОВЫХ ДОКУМЕНТОВ	385
Наместников А.М., Субхангулов Р.А.*	
СОСТОЯНИЕ, ТЕНДЕНЦИИ И КОНЦЕПЦИЯ РАЗВИТИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ТЕХНОЛОГИЙ В ЗАЩИТЕ ИНФОРМАЦИИ	391
Вишняков В.А.	

РАЗРАБОТКА ГИБРИДНОЙ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ОЦЕНКИ ЗНАНИЙ СТУДЕНТОВ	395
Козлов О.А., Михайлов Ю.Ф.	
ИСПОЛЬЗОВАНИЕ АДАПТИВНЫХ СЕМАНТИЧЕСКИХ МОДЕЛЕЙ ДЛЯ ПРЕДСТАВЛЕНИЯ И СТРУКТУРИЗАЦИИ ЗНАНИЙ В СИСТЕМАХ ОБРАЗОВАТЕЛЬНОГО НАЗНАЧЕНИЯ	399
Шихнабиева Т.Ш.	
КОНЦЕПЦИЯ ГЕНЕЗИСА САМОРАЗВИВАЮЩЕГОСЯ ИНТЕЛЛЕКТА Ж. ПИАЖЕ: МЕТОДОЛОГИЯ И ПРИНЦИПЫ МОДЕЛИРОВАНИЯ	403
Старжинский В.П.	
ПРОЕКТИРОВАНИЕ ЭЛЕКТРОННЫХ ОБУЧАЮЩИХ СИСТЕМ	411
Ниязова Р.С., Алсеитова А.Т.	
СЕМАНТИКА И ОНТОЛОГИИ В ЕСТЕСТВЕННЫХ И ИСКУССТВЕННЫХ СИСТЕМАХ	415
Сторож В.В.	
ИСПОЛЬЗОВАНИЕ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ ПРОГНОЗИРОВАНИЯ СУДЕБНЫХ РЕШЕНИЙ	423
Климов В.С., Новикова А.О., Данко А.И.	
ОПЫТ ЭКСПЕРИМЕНТАЛЬНОЙ РАЗРАБОТКИ ПРОГРАММНОЙ СРЕДЫ С БАЗОЙ ЗНАНИЙ НА СЕМАНТИЧЕСКИХ СЕТЯХ	427
Катаев В.А.	
РАСПОЗНАВАНИЕ БОТОВ В СОЦИАЛЬНЫХ СЕТЯХ	431
Кипаева Е.В., Кириченко М.И., Орлова Ю.А., Заболеева-Зотова А.В.	
АВТОМАТИЗАЦИЯ СЕМАНТИЧЕСКОГО АНАЛИЗА НОВОСТНЫХ ИНТЕРНЕТ-ТЕКСТОВ	435
Солошенко А.Н., Розалиев В.Л., Заболеева-Зотова А.В.	
АГЕНТНЫЙ ПОДХОД К РАЗРЕШЕНИЮ НЕ-ФАКТОРОВ В ЗАДАЧЕ СЕМАНТИЧЕСКОГО СРАВНЕНИЯ ИМЕН СУЩНОСТИ ПРИКЛАДНОГО ПРОГРАММНОГО РЕШЕНИЯ	439
Бердник В.Л., Заболеева-Зотова А.В.	
ОПРЕДЕЛЕНИЕ КЛЮЧЕВЫХ ПАРАМЕТРОВ ЛИЦА ЧЕЛОВЕКА ДЛЯ ПОСТРОЕНИЯ ОБЪЕМНОЙ МОДЕЛИ	443
Климов А.С., Фоменкова М.А., Орлова Ю.А., Розалиев В.Л.	
ПРИНЦИПЫ ИНТЕГРАЦИИ СОДЕРЖИМОГО RDF-ХРАНИЛИЩ В ПРОЕКТ OSTIS	447
Каешко А.И., Колб Д.Г.	
ПРИМЕНЕНИЕ МЕТОДА СЕМАНТИЧЕСКОГО ДИФФЕРЕНЦИАЛА ДЛЯ ОЦЕНКИ ПОКАЗАТЕЛЕЙ КАЧЕСТВА КОНВЕРСИИ ГОЛОСА	453
Захарьев В.А., Петровский А.А.	
КАМΠΑНАЕНТ ДЛЯ АНАЛІЗУ ЧАСЦІН МОВЫ БЕЛАРУСКІХ СЛЮЎ ВА ЎМОВАХ АБМЕЖАВАНЫХ СІСТЕМНЫХ РЭСУРСАЎ	457
Шчурко М.Л., Гецэвіч Ю.С., Пакладок Д.А.	

АНАЛИЗ И ВЕРИФИКАЦИЯ РАСКРАШЕННЫХ СЕТЕЙ ПЕТРИ РЕАЛЬНОГО ВРЕМЕНИ С ПОДДЕРЖКОЙ ЛОГИКИ АЛЛЕНА Еремеев А.П., Королев Ю.И.	461
МЕТОДЫ И АЛГОРИТМЫ НАХОЖДЕНИЯ СТЕПЕНЕЙ ОБОСНОВАНИЯ В СИСТЕМАХ АРГУМЕНТАЦИИ Моросин О.Л.	465
ОНТОЛОГИИ ЖИЗНЕННОГО ЦИКЛА СЛОЖНОЙ ТЕХНИЧЕСКОЙ СИСТЕМЫ Тарасов В.Б., Федотова А.В., Черепанов Н.В.	471
АНАЛИЗ ДИНАМИКИ ИЗМЕНЕНИЙ НАУЧНОЙ ОБЛАСТИ МЕТОДАМИ ОНТОЛОГИЧЕСКОГО ИНЖИНИРИНГА Лещева И.А., Лещев Д.В.	483
ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ В ДИАГНОСТИКЕ ИШЕМИЧЕСКИХ АТАК Головкин В.А., Войцехович Г.Ю., Апанель Е.Н., Масыкин А.С.	487
ОСОБЕННОСТИ ПОСТРОЕНИЯ ИНФОРМАЦИОННЫХ ПРОСТРАНСТВ ПРОМЫШЛЕННЫХ ПРЕДПРИЯТИЙ НА ОСНОВЕ СВЯЗАННЫХ ДАННЫХ Галушка И.Н., Щербак С.С.	491
МЕТОД ЦЕЛЕНАПРАВЛЕННОГО ПОИСКА СВЯЗАННЫХ ДАННЫХ В УСЛОВИЯХ СТРУКТУРНОЙ НЕОПРЕДЕЛЕННОСТИ Тертышный В.А., Шаповал И.С., Щербак С.С.	495
ПРАКТИЧЕСКИЕ АСПЕКТЫ ТЕСТИРОВАНИЯ РАСПРЕДЕЛЕННЫХ ПРИЛОЖЕНИЙ НА ОСНОВЕ СВЯЗАННЫХ ДАННЫХ Солошич С.Н.	499
ПРОГРАММНЫЙ ИНСТРУМЕНТАРИЙ ДЛЯ РАЗРЕШЕНИЯ МОРФОЛОГИЧЕСКОЙ МНОГОЗНАЧНОСТИ В ТАТАРСКОМ ЯЗЫКЕ Сулейманов Д.Ш., Гильмуллин Р.А., Гатауллин Р.Р.	503
ВОЗМОЖНОСТЬ АВТОМАТИЧЕСКОГО ОПРЕДЕЛЕНИЯ АДРЕСАТА НА ОСНОВЕ СЕМАНТИКО-СИНТАКСИЧЕСКИХ ОСОБЕННОСТЕЙ ТЕКСТА Глазкова А.В.	503
СОЗДАНИЕ ХРАНИЛИЩА НАУЧНЫХ ПУБЛИКАЦИЙ С ПРИМЕНЕНИЕМ МЕТОДОВ КОРПУСНОЙ ЛИНГВИСТИКИ Елохов Е.С., Югов А.С., Ланин В.В.	513
О ПОДХОДЕ К ПОДБОРУ DSL НА ОСНОВЕ АНАЛИЗА КОРПУСА СПЕЦИАЛИЗИРОВАННЫХ ДОКУМЕНТОВ Валеев М.Т., Елохов Е.С., Узунова Е.Н., Югов А.С.	517
РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОГО СЕРВИСА АНАЛИЗА ПРЕДЛОЖЕНИЙ НА РЫНКЕ НЕДВИЖИМОСТИ Анисимова Т.В., Нестеров Р.А., Печенежский А.Б.	523
ПОТРЕБНОСТЬ В РЕЧЕВЫХ ФУНКЦИЯХ ИНТЕЛЛЕКТУАЛЬНЫХ САПР Абрамов Г.А., Боргест Н.М., Коровин М.Д.	527

РЕАЛИЗАЦИЯ ОНТОЛОГИЧЕСКОЙ МУЛЬТИАГЕНТНОСТИ ПРЕДМЕТНОЙ ОБЛАСТИ СРЕДСТВАМИ РЕЛЯЦИОННОЙ СУБД НА ПРИМЕРЕ ЗАЧИСЛЕНИЯ АБИТУРИЕНТОВ В УНИВЕРСИТЕТЫ РОССИИ	531
Боргест Н.М., Лысаковский И.А.	
ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ВСПОМОГАТЕЛЬНЫХ АЛГОРИТМОВ ДЛЯ ПОСТРОЕНИЯ КЛАСТЕРНЫХ СТРУКТУР	537
Горюнов В.А., Жукевич А.И., Родченко В.Г.	
ЛОГИЧЕСКАЯ И ЛИНГВИСТИЧЕСКАЯ СЕМАНТИКА НА ОСНОВЕ ИДЕЙ Г.ФРЕГЕ И В.В. МАРТЫНОВА	543
Демиров В.В.	
СЕМАНТИЧЕСКИЕ МОДЕЛИ ИНТЕЛЛЕКТУАЛЬНОЙ ОЦЕНКИ ЗНАНИЙ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ	547
Кабенов Д.И., Разахова Б.Ш.	
АДАПТИВНЫЙ ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЕЙ	553
Барлыбаев А.Б.	
ВИРТУАЛЬНЫЙ ПОМОЩНИК АБИТУРИЕНТА ПРИ ВЫБОРЕ СПЕЦИАЛЬНОСТИ	559
Барлыбаев А.Б., Нургазинова Г.Ш.	
СОЗДАНИЕ ОНТОЛОГИЧЕСКОЙ МОДЕЛИ НАЛОГОВОЙ СИСТЕМЫ КАЗАХСТАНА	565
Шарипбаев А.А., Омарбекова А.С., Турмаганбетова Ш.К.	
АВТОРСКИЙ УКАЗАТЕЛЬ	569

TABLE OF CONTENTS

FOREWORD	23
About Viktor Martynov	25
MARTYNOV'S USC – 30 YEARS LATER Efimenko I.V., Khoroshevsky V.F.	29
FROM SEMANTIC CODE TO COGNITIVE LINGUISTICS, SEMIOTICS AND INFORMATICS: ON MARTYNOV'S SCIENTIFIC HERITAGE Tarassov V. B.	39
THEORY OF THE KNOWLEDGE ARCHITECTURE AUTOMATIC FORMATION (TKAAF-2) AND FURTHER MINIMIZATION OF SEMANTIC CALCULUS Hardzei A.	49
STRUCTURIZATION OF SENSE SPACE Golenkov V.V., Guliakina N.A.	65
THE USER INTERFACE OF THE INTELLIGENT DESIGN SUPPORT METASYSTEM FOR INTELLIGENT SYSTEMS Koronchik D. N.	79
THE KNOWLEDGE BASE OF THE INTELLIGENT DESIGN SUPPORT METASYSTEM FOR INTELLIGENT SYSTEMS Grakova N.V., Davydenko I.T., Rusetski K.V.	83
THE KNOWLEDGE PROCESSING MACHINE OF THE INTELLIGENT DESIGN SUPPORT METASYSTEM FOR INTELLIGENT SYSTEMS Shunkevich D.V.	93
ONTOLOGY-BASED INTELLIGENT SYSTEM CONSTRUCTION THROUGH COMPONENT REUSE Borisov A.N.	97
SOLVING DISAMBIGUATION IN SPEECH RECOGNITION Lobanov B.M., Zhitko V.A.	103
CONTINGENCY MANAGEMENT AND SEMANTIC MODELING IN ENERGY SECTOR Massel L.V., Massel A.G.	111
ONTOLOGY-BASED FORMALIZATION OF SEMANTICS OF KNOWLEDGE AREAS IN INFORMATION AND INTELLIGENT SYSTEMS Zagorulko Yu.A. , Zagorulko G.B.	117
WHAT SHOULD BE THE PARADIGM OF TASK SOLVING FOR THE KNOWLEDGE- BASED SYSTEMS? Gribova V, Kleshev A.	131
THE DEVELOPMENT OF ENGINEERING CALCULATION FOR THE KNOWLEDGE PORTALS Globa L.S., Novogrudskaya R.L.	137

A LANGUAGE FOR SPECIFYING ONTOLOGIES Plesniewicz G.S.	143
A PERSONAL ONTOLOGY OF PROFESSIONAL EXPERIENCE P.I. Sosnin	147
ROLE OF ONTOLOGIES IN INFORMATION SYSTEMS DESIGN Borgest N.M.	155
MODELING OF PROCESS DYNAMICS BY SEQUENCE OF HOMOGENOUS SEMANTIC NETWORKS ON THE BASE OF TEXT CORPUS SEQUENCE ANALYSIS Kharlamov A.A. , Yermolenko T.V., Zhonoin A.A.	161
RULES FOR TERMINOLOGICAL CLUSTERS CREATIONS Malkovsky M.G., Soloviev S.Y.	169
FILLING OF SEMANTIC SLOTS IN THE RELATIONAL-SITUATIONAL FRAME ON THE EXAMPLE OF TATAR SINTAXEMES Suleymanov D. Sh., Gatiatullin A. R.	173
SEMANTIC TECHNOLOGIES FOR MATHEMATICAL RESOURCES Nevzorova O.A.	179
DESCRIPTION OF PART-WHOLE RELATIONS: APPROACHES IN ONTOLOGICAL RESOURCES Loukachevitch N.V.	185
CONDITIONAL-REFLEXIVE BASES FOR REMEMBERING OF KNOWLEDGE Stefanuk V.L.	191
METHODOLOGICAL BASES OF CONSTRUCTION CAD TA WITH DEVELOPING THE KNOWLEDGE BASE Burdo G.B., Palyukh B.V., Vorobyeva E.V.	195
APPLICATION OF THE CHVG-ALGORITHM FOR SCIENTIFIC TEXTS Lande D.V. , Snarskii A.A., Yagunova E.V.	199
CONSTRUCTION OF AN INTEGRAL INDEX FOR ASSESSING PROSPECTS OF COMPUTING COMPLEX Lobanov V.N., Petrovsky A.B.	205
SOME REGULARITIES AND OBJECTIVE LIMITATIONS WHEN IMPLEMENTING SEMANTIC PROCESSING ALGORITHMS ON COMPUTER SYSTEMS WITH MASSIVE PARALLELISM Verenik N.L., Seitkulov Y.N., Girel A.I., Tatur M.M.	211
ON-BOARD INTELLIGENT CONTROL SYSTEMS OF RUSSIAN SATELLITES: SCIENCE FICTION OR REALITY. Dimitrov D.M., Danilin N.S., Sabirov I.Kh.	217
ONTOLOGY OF ISS RUSSIAN SEGMENT AND ITS PRACTICAL USE IN INTELLECTUAL AEROSPACE APPLICATIONS Vakurina T., Kotelya V., Lakhin O., Matushin M., Skobelev P.	221

VALUATION MODELS CONVICTION ABOUT THE ADEQUACY OF THE PRESENTATION OF THE SUBJEC AREA IN THE PROBLEMS OF CHOICE Vinogradov G.P., Filatov N.N.	227
RESEARCHING OF NEURAL LEARNING ALGORITHMS IN INTELLIGENT EVOLUTION SYSTEMS Komartsova L.G., Lavrenkov Ju.N., Antipova O.V.	233
SEMANTIC ASSOSIATIVE RESOURCE NETWORKS WITH PLURAL VERTICES Zhilyakova L.Yu.	239
COGNITIVE MODELLING IN THE CONDITIONS OF UNCERTAINTY (THE SEMIOTICS APPROACH) A. A. Kulinich	243
MATRIX REPRESENTATION OF A FINITE PREDICATES FOR AUTOMATION OF LOGICAL AND SEMANTIC ANALYSIS Zuenko A.A	251
CONSTRUCTIVE PROPOSITIONAL LOGIC WITH IMPLICATION UNPARADOXICAL Smetanin Yu. M.	257
SEMANTIC CONTENT OF A SEQUENCE OF SUBJECTPREDICATE JUDGMENTS Smetanin Yu. M.	263
LOGICAL-LINGUISTIC MODEL FOR IDENTIFICATION OF SEMANTIC RELATIONSHIPS BETWEEN ENTITIES ON BASE OF ALGEBRA OF FINITE PREDICATES Khairova N., Uzlov D., Sharonova N.	267
SPIRAL COGNITIVE METADYNAMICS: DYNAMIC PROCESS SCALABILITY ANALYSIS Prokopchuk I.A.	271
SITUATIONAL MODELING PLATFORM AS A TOOL OF CONTINGENCY MANAGEMENT IN ENERGY SECTOR Massel A.G., Ivanov R.A.	277
EXPERT SYSTEMS WITHOUT EXPERTS Kleshev A.S., Smagin S.V.	281
SYSTEM ANALYSIS CONTENTS FOR INTELLIGENT ACTIVITY AUTOMATION AT BRANCH LEVEL Kleshev A. , Shalfeeva E.	285
THE INTERNET SOFTWARE TOOL FOR SEMANTIC NETWORK CLASSES TRANSFORMATION Timchenko V.A.	291
FUZZY CONCEPT HIERARCHY DEVELOPMENT FOR HISTORICAL PERSONALITY DESCRIPTION Globa L.S., Ostrovskiy S.A., Popova M.A., Stryzhak O.Y., Ternovoy M.Y.	297

APPLICATION OF COGNITIVE GRAPHICS TOOLS BASED ON THE 3-SIMPLEX IN INTELLIGENT TRAINING-TESTING SYSTEMS Yankovskaya A.E., Yamshanov A.V., Krivdyuk N.M.	303
TEXT-TO-PICTURE SYSTEM: INTEGRATION OF NATURAL LANGUAGE AND IMAGES Kurbatov S.S., Litvinovich A.V., Lobzin A.P., Khakhalin G.K.	309
PARSING OF GRAPHIC IMAGES ON THE BASIS OF APPLIED ONTOLOGY IN HYBRID SYSTEM OF RECOGNITION Khakhalin G.K., Kurbatov S.S., Lobzil A.P.	315
SEMANTIC CONCEPTUAL DESIGN OF NATURAL LANGUAGE INTERFACES FOR INTELLIGENT SYSTEM Yeliseyeva O.E.	321
METHOD OF CONSTRUCTING TEXT-TO-SPEECH COMPONENTS FOR NATURAL LANGUAGE INTERFACES WITH THE HELP OF NOOJ Hetsevich Yu.S., Skopinava A.M., Okrut T.I.	333
KNOWLEDGE-ORIENTED MEANS OF SEMANTIC SEARCH INTO THE WEB Rogushina J.	339
USE OF THE ONTOLOGICAL APPROACH TO SEMANTIC SEARCH IN THE ENVIRONMENT OF THE INTERNET OF THINGS Rogushina J., Gladun A., Andrushevich A., Kurbatski A.	353
UNIVERSAL SEMANTIC CODE AS AN ABSTRACTING TECHNOLOGY FOR FUZZY ENGINEERING PROBLEM SOLVING Kandelinsky S.L., Boyko I.M.	357
ONTOLOGIES FOR PROJECT MANAGEMENT Veremyev V.L., Gorovaya D.O.	363
ONTONTOLOGICAL APPROACH TO A SIMULATION SYSTEM CONSTRUCTING FOR THE PARTICULAR SUBJECT AREA Zamyatina E.B., Mikheev R.A.	369
INTEGRATION OF MODELING SYSTEMS ON THE BASIS OF DSM-PLATFORM WITH ONTOLOGIES Sukhov A.O., Lyadova L.N., Zamyatina E.B.	375
AN ARCHITECTURE OF THE SCIENTIFIC PUBLICATIONS PROCESSING SYSTEM Lanin V.V., Chugunov A.P.	381
ONTOLOGICALLY-ORIENTED MODEL OF CLASSIFICATIONS OF TEXT DOCUMENTS Namestnikov A.M., Subkhangulov R.A.	385
STATE, TRENDS AND CONCEPTION DEVELOPMENT OF INTELLIGENCE TECHNOLOGIES IN INFORMATION DEFENSE Vishniakou U.A.	391

DEVELOPMENT OF A HYBRID INTELLECTUAL SYSTEMS FOR SOLVING THE PROBLEM OF ASSESSING KNOWLEDGE STUDENTS Kozlov O.A., Mikhailov J.F.	395
USING ADAPTIVE SEMANTIC MODEL FOR THE PRESENTATION AND THE STRUCTURING OF KNOWLEDGE AMONG THE EDUCATIONAL APPOINTMENTS Shicknabieva T.SH.	399
CONCEPT GENESIS SPONTANEOUS INTELLIGENCE J. PIAGET'S METHODOLOGY AND PRINCIPLES of MODELING Starzhinsky V.P.	403
DESIGNING E-LEARNING SYSTEMS Niyazova R.S., Alseitova A.A.	411
SEMANTICS AND ONTOLOGY IN NATURAL AND ARTIFICIAL SYSTEMS Storozh V.V.	415
USE OF ARTIFICIAL INTELLIGENCE FOR PREDICTION OF JUDGMENTS Klimov V.S., Novikova A.O., Danko A.I.	423
EXPERIENCE OF EXPERIMENTAL DEVELOPMENT OF SOFTWARE ENVIRONMENT WITH THE KNOWLEDGE BASE ON SEMANTIC NETWORKS Kataev V.A.	427
RECOGNITION BOTS IN SOCIAL NETWORKS Kipaeva E.V., Kirichenko M.I., Orlova Yu.A., Zaboleeva-Zotova A.V.	431
AUTOMATION OF SEMANTIC ANALYSIS OF INTERNET NEWS TEXTS Soloshenko A.N., Rozaliev V.L., Zaboleeva-Zotova A.V.	435
AN AGENT-BASED APPROACH FOR RESOLVING NON-FACTORS IN THE PROBLEM OF SEMANTIC COMPARISON OF ESSENCE NOTIONS IN APPLIED PROGRAM SOLUTION Berdnik V.L., Zaboleeva-Zotova A.V.	439
DETERMINATION OF FACE KEY POINTS FOR 3D-MODEL CONSTRUCTION Klimov A.S., Fomenkova M.A., Orlova Yu.A., Rozaliev V.L.	443
BASIC PRINCIPLES OF INTEGRATION OF RDF-STORAGE CONTENT INTO PROJECT OSTIS Kayeshko A.I., Kolb D.G.	447
SEMANTIC DIFFERENTIAL METHOD FOR THE VOICE CONVERSION QUALITY TESTING Zahariev V.A. , Petrovsky A.A.	453
COMPONENT FOR PART-OF-SPEECH TAGGING OF BELARUSIAN WORDS WITH LOW SYSTEM REQUIREMENTS Shchurko M.L., Hetseovich Y.S., Pakladok D.A.	457
ANALYSIS AND VERIFICATION OF REAL-TIME COLORED PETRI NETS WITH SUPPORT OF ALLEN'S LOGIC Eremeev A.P., Korolev Y.I.	461

METHODS AND ALGORITHMS FOR CALCULATING JUSTIFICATION DEGREES IN ARGUMENTATION SYSTEMS Morosin O.L.	465
LIFECYCLE ONTOLOGIES FOR COMPLEX TECHNICAL SYSTEMS Tarassov V.B., Fedotova A.M., Cherepanov N.V.	471
ANALYSIS OF TRENDS IN SCIENTIFIC DOMAINS BY ONTOLOGICAL ENGINEERING Leshcheva I.A., Leshchev D.V.	483
NEURAL NETWORKS USAGE IN ISCHEMIC ATTACKS DIAGNOSTICS Golovko V.A., Vaitsekhovich H.Y., Apanel E.N., Mastykin A.S.	487
PECULIARITIES OF LINKED DATA INFORMATIONAL SPACE DEVELOPMENT FOR INDUSTRIAL ENTERPRISES Galushka I.N., Shcherbak S.S.	491
METHOD OF LINKED DATA SEARCH IN CONDITIONS OF STRUCTURAL UNCERTAINTY Tertishniy V.A., Shapoval I.S., Shcherbak S.S.	495
PRACTICAL ASPECTS TESTING DISTRIBUTED APPLICATIONS BASED ON THE RELATED DATA Soloshych S.N.	499
SOFTWARE TOOL FOR MORPHOLOGICAL DISAMBIGUATION IN THE TATAR LANGUAGE Suleymanov D.Sh., Gilmullin R.A., Gataullin R.R.	503
THE CAPABILITY OF AUTOMATIC TEXT ADDRESSEE RECOGNITION BASED ON SYNTACTIC AND SEMANTIC FEATURES Glazkova A.V.	503
PUBLICATIONS REPOSITORY CREATION BASED ON CORPUS LINGUISTICS TECHNIQUES Elokhov E.S., Yugov A.S., Lanin V.V.	513
AN APPROACH TO THE SELECTION OF DSL BASED ON CORPUS OF DOMAIN-SPECIFIC DOCUMENTS Valeev M.T., Elokhov E.S., Uzunova E.N., Yugov A.S.	517
INTELLIGENT SERVICE FOR ANALYSIS OF REAL ESTATE MARKET OFFERS Anisimova T.V., Nesterov R.A., Pechenezhskiy A.B.	523
THE NEED FOR VOICE CONTROL IN INTELLECTUAL CAD Abramov G.A., Borgest N.M., Korovin M.D.	527
REALIZATION ONTOLOGICAL MULTIAGENT OF SUBJECT AREA USING RDBMS BY EXAMPLE ADMISSION IN RUSSIAN UNIVERSITIES Borgest N.M., Lysakovkiy I.A.	531
SOFTWARE IMPLEMENTATION OF SUBSIDIARY ALGORITHMS FOR CONSTRUCTING CLUSTER STRUCTURES Harunou V.A., Zhukevich A.I., Rodchenko V.G.	537

LOGICAL AND LINGUISTIC SEMANTICS BASED ON THE IDEAS OF G. FREGE AND V.V. MARTYNOV Demirov V.V.	543
SEMANTIC MODELS OF KNOWLEDGE INTELLECTUAL ASSESSMENT IN NATURAL LANGUAGE Kabenov D.I., Razahova B.Sh.	547
ADAPTIVE USER INTERFACES Barlyabyev A.B.	553
THE VIRTUAL ASSISTANT FOR APPLICANT IN CHOOSING OF THE SPECIALTY Barlyabyev A.B. , Nurgazinova G.Sh.	559
CREATING AN ONTOLOGICAL MODEL FOR THE TAX SYSTEM IN KAZAKHSTAN Sharipbaev A., Omarbekova A., Turmaganbetova S.	565
AUTHOR INDEX	569

ПРЕДИСЛОВИЕ

Основным практическим результатом исследований в области искусственного интеллекта является создание не только интеллектуальных систем, но и технологий, обеспечивающих быстрое и качественное построение таких систем. Разработка указанных технологий требует решения следующих задач:

- чёткого выделения логико-семантического уровня интеллектуальных систем, который абстрагируется от всевозможных вариантов технической реализации этих систем (в том числе и от использования принципиально новых компьютеров, ориентированных на аппаратную поддержку интеллектуальных систем);
- разработки онтологии проектирования интеллектуальных систем и унификации описания логико-семантических моделей интеллектуальных систем;
- обеспечения платформенно независимого характера логического проектирования интеллектуальных систем, результатом которого является унифицированное описание логико-семантических моделей проектируемых интеллектуальных систем;
- использования методики компонентного проектирования интеллектуальных систем, в основе которой лежит постоянно пополняемая библиотека многократно используемых компонентов интеллектуальных систем (многократно используемых подсистем, знаний, машин и операций обработки знаний, компонентов пользовательских интерфейсов);
- обеспечения семантической совместимости многократно используемых компонентов интеллектуальных систем, входящих в состав указанной библиотеки.

Основной целью ежегодных конференций OSTIS (Open Semantic Technology for Intelligent Systems) является создание условий для расширения сотрудничества различных научных школ, вузов и коммерческих организаций, направленного на разработку и применения комплексной массовой и постоянно совершенствуемой технологии компонентного проектирования интеллектуальных систем.

IV Международную научно-техническую конференцию «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2014), организаторы посвящают 90-летию Виктора Владимировича Мартынова, который внёс большой вклад в формализацию семантики. Следовательно, основанные вопросы, рассматриваемые на конференции, касаются формализации семантики и, в частности разработки, формальных моделей онтологий.

Конференция OSTIS-2014 открывает цикл научных мероприятий приуроченных к 50-летию основания Белорусского государственного университета информатики и радиоэлектроники.

Председатель Программного комитета конференции OSTIS-2014.
Председатель Совета Российской ассоциации искусственного интеллекта
Кузнецов Олег Петрович

FOREWORD

Creation not only intelligent systems but also technologies that ensure fast and efficient construction of intelligent systems are the main practical result of research in artificial intelligence. Development of these technologies requires the following tasks:

- precise separation of logical and semantic level of intelligent systems, which is abstracted from various versions of the technical implementation of these systems (including the use of innovative computer-based hardware support for Intelligent Systems);
- developing an ontology design of intelligent systems and unifying description of logical and semantic models of intelligent systems;
- provide a platform independent nature of the logical design of intelligent systems, which result is a unified description of logical and semantic models of an intelligent systems;
- use of a component design methodology of intelligent systems, which is based on permanently increasing library of reusable components of intelligent systems (reusable subsystems, knowledge, machine and operations of knowledge processing, user interface components)
- ensuring semantic compatibility of reusable components of intelligent systems that make up the library

Creating the conditions for the expansion of cooperation between different scientific schools, universities and business organizations, aimed on development and using of comprehensive mass and continuously improved component design technology of intelligent systems is the main purpose of annual conferences OSTIS (Open Semantic Technology for Intelligent Systems).

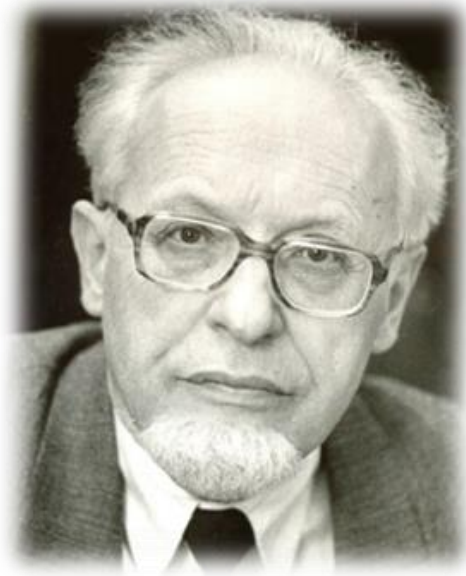
IV International scientific and technical conference “Open Semantic Technology for Intelligent Systems” (OSTIS-2014) is dedicated to 90-th anniversary of V.V. Martynov, who made a great contribution into semantic formalization. Therefore, the main questions on the conference are related to semantic formalization and, in particular, developing formal anthology models.

OSTIS-2014 conference opens a circle of scientific actions, which are dedicated to 50-th anniversary of Belarusian State University of Informatics and Radioelectronics.

Programme Committee Chair.

Chairman of the Council of the Russian Association for Artificial Intelligence

Kuznetsov Oleg Petrovich



Виктор Владимирович Мартынов родился 25 января 1924 года в г. Одесса. В 18 лет ушел на фронт и воевал простым пехотинцем – им написаны трогательные воспоминания о юном солдате среди ужасов войны. В 1948 году В.В. Мартынов окончил Одесский университет, в 1951 – аспирантуру по славистике при Львовском университете. В 1952-1960 гг. был заведующим кафедрой иностранных языков Одесского университета, а с 1960 г. Переехал в Минск и работал в Институте языкознания Академии наук БССР, где с 1962 по 1990 гг. заведовал отделом общего и славянского языкознания. В 1969 г. Виктор Владимирович защитил докторскую диссертацию по теме «Славянско-германское лексическое взаимодействие древнейшей поры. К проблеме прародины славян», в 1971 стал профессором.

Доктор филологических наук профессор Мартынов В.В. автор более 20 книг и брошюр, более 200 статей. Его ранние научные интересы были связаны с литературоведением и анализом художественного текста. В частности, В.В. Мартынову принадлежит ряд серьезных исследований о творчестве польского поэта Юлиуша Словацкого.

Три темы наиболее близки профессору В.В. Мартынову. Первая касается белорусистики. С 1969 г. Мартынов – редактор и соавтор многотомного «Этымалагічнага слоўніка беларускай мовы» (т.1-7, 1978-1991). Материалы трех кандидатских диссертаций по диалектной лексикографии, подготовленных под научным руководством В.В. Мартынова, обобщены в коллективной монографии «Лексіка Палесся ў прасторы і часе» (1971). Два другіх коллектива, которые он возглавлял, подготовили монографии по современному белорусскому языку «Марфемныя дыстрыбуцыя ў беларускай мове. Дзеяслоў» (1967) и «Словаўтваральная сістэма сучаснай беларускай мовы» (семантико-словообразовательный анализ имен), а также новый тип словарей: «Канкарданс беларускай мовы XIX ст.» и «Канкардансы беларускіх пісьменнікаў. Канкарданс Кузьмы Чорнага» (не издан).

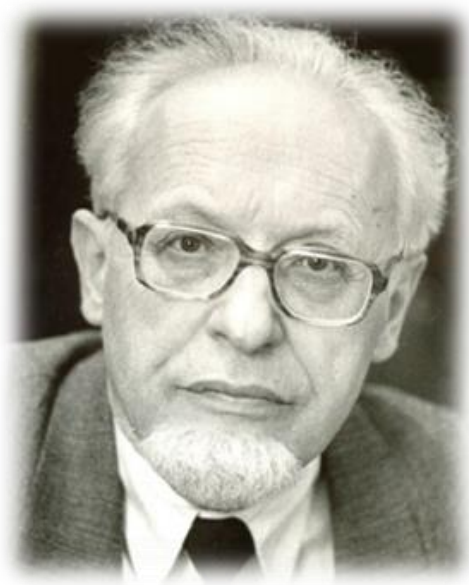
Вторая тема – славистическая, - выявление глоттогенеза и онтогенеза славян. Виктор Владимирович – признанный авторитет и классик в области компаративистики и славистики. В монографиях «Славянско-германское лексическое взаимодействие древнейшей поры. К проблеме прародины славян» (1963), «Язык в

пространстве и времени. К проблеме глоттогенеза славян» (1983), и ряде других, более поздних книг и статей, в докладах на Международных съездах славистов в Софии, Варшаве, Любляне, Киеве, Братиславе В.В. Мартынов снова и снова обращается к вопросам происхождения славянской лексики, славяно-неславянских контактов, становления праславянского контактов, становления праславянского языка в лингвокультурном аспекте.

Третья тема, которой Виктор Владимирович занимался с особым вдохновением, связана с информатикой, кибернетикой, искусственными системами и технологиями. В 60-е годы XX столетия научной среде возникло понимание того, что без структурализации знаний, без их формального представления и преобразования создать искусственный интеллект невозможно. Однако добиться этого интуитивными эвристическими приемами не удалось, а как применить к столь сложной проблематике дедуктивные методы никто не знал. До появления работ Виктора Владимировича Мартынова погружение знаний в формализмы представления оставалось за границами рационального познания мира. Опираясь на взаимосвязь лингвистики с семиологией, праксикологией и теорией информации, ученый предложил способ «исчисления языковых смыслов». В монографии «Кибернетика. Семиотика. Лингвистика» (1966) им излагаются основы и перспективы дедуктивной семиологии, а его следующая большая работа «Семиотологические основы информатики» (1974) уже содержит прототип универсального семантического кода как средства снятия неопределеннозначности естественного языка. Первую версию УСК, описанную в одноименной монографии (1977), автор совершенствует в ряде последующих публикаций, в том числе в монографиях «Основы семантического кодирования. Опыт представления и преобразования знаний» (2001) и «В центре сознания человека» (2009). Неиссякаемый интерес к проблемам непротиворечивого описания естественного языка В.В. Мартынов проявляет и как руководитель центра «Семантика», и как председатель белорусского общества искусственного интеллекта.

Виктор Владимирович Мартынов занесен в «Книгу почета» Соединенных штатов Америки (1989). В 1990 г. Ему присвоено звание Заслуженный деятель науки БССР. В 1993г. он объявлен Международным биографическим центром в Кембридже «Человеком века» и удостоен самой престижной «Награды XX столетия за достижения», которая присуждается раз в столетие.

Под руководством профессора В.В. Мартынова защищено 17 кандидатских диссертаций и 4-е докторские.



Viktor Vladimirovich Martynov was born January 25, 1924 in Odessa. When Martynov was 18 years old, he went to the front and fought like simple infantryman. The touching memories of a young soldier of the horrors of war have been written by him. In 1948 Martynov graduated from the University of Odessa, in 1951 - postgraduate studies in Slavic Studies at Lvov University. In 1952-1960 Viktor was the head of the Department of Foreign Languages at Odessa University, and in 1960 he moved to Minsk and worked at the Institute of Linguistics of the Academy of Sciences of Belarus, where from 1962 to 1990 headed the Department of General and Slavic linguistics. In 1969 Viktor defended his doctoral thesis on "Slavic-German lexical interaction oldest times. On the problem of the ancestral home of the Slavs ", in 1971 he became a professor.

Doctor of Philology, professor Viktor Vladimirovich Martynov is the author of over 20 books and pamphlets, more than 200 articles. His early research interests were related to the study of literature and analysis of literary text. In particular, Martynov owns a number of serious research on the works of Polish poet Juliusz Slovák.

Three themes are the closest to professor Martynov. The first concerns Belarus Science. Since 1969 Martynov is the editor and co-author of a multivolume "Etymological dictionary of the Belarusian language" (v.1-7, 1978-1991). Materials of three master's theses on the dialect lexicography, prepared under the supervision of Martynov, summarized in the monograph "Lexicon Polesse in space and time" (1971). Two other teams, which he headed, prepared monographs on contemporary Belarusian language, "Morphemic distribution in the Belarusian language. The verb" (1967) and "Word-formation system of the modern Belarusian language" (semantic analysis derivation of names), as well as a new type of dictionaries "Kankardans Belarusian language XIX century" and "Kankardanses of Belarusian writers. Kankardans of Kuzma Chorny" (not published).

The second theme is Slavonic: the identification glottogenic and ontogeny of Slavs. Viktor Martynov is a recognized authority in the field of Classics and Comparative Literature and Slavic Studies. Monographs "Slavic- German lexical interaction oldest times. On the problem of the ancestral home of the Slavs" (1963), "Language in time and space. On the problem glottogenic of the Slavs"(1983), and a number of other more recent books and articles, Martynov again addresses issues of Slavic origin vocabulary, non-Slavic-

Slavic contacts, becoming the pre-Slavic contacts, becoming the pre-Slavic language in the linguistic-cultural aspect in his reports on the International Congress of Slavic in Sofia, Warsaw, Ljubljana, Kiev, Bratislava.

The third topic, which Victor Vladimirovich engaged with especial inspiration, is connected with informatics, cybernetics, artificial systems and technologies. In 60-s years of XX century the scientific community found out, that without knowledge structuralizing, formal representation and conversion artificial intelligence cannot be created. However, an attempt to achieve this with intuitive heuristic methods was failed, and no one knew, how to use deductive method to such a difficult problem. Before the appearance of V. V. Martynov's works, knowledge's dive into representation formalisms remained outside the rational cognition of the world. Basing on the relationship between linguistics, semiology, praxiology and information theory, scientist suggested a method "of calculating linguistic meanings". In the monograph "Cybernetics. Semiotics. Linguistics" (1966) he sets out the fundamentals and prospects of deductive semiology and his next big job "Semitologic Foundations of Computer Science" (1974) already contains the prototype of a universal semantic code as a mean of resolving uncertainty in meanings of natural language. The first version of the USC, described in the monograph of the same name (1977), the author develops a number of subsequent publications, including monographs "Basics semantic encoding. Experience of representation and transformation of knowledge". (2001) and "In the center of human consciousness" (2009). V.V. Martynov demonstrated inexhaustible interest in the problems of consistent description of natural language as a head of the center "Semantics", and as a chairman of the Belarusian Society of Artificial Intelligence.

Viktor Martynov is recorded in the Book of Honor of the United States of America (1989). In 1990 he was awarded the title of Honored Worker of Science of BSSR. In 1993 the International Biographical Centre in Cambridge declared him as a "Man of the century" and awarded the most prestigious "Award for achievements of the XX century", which is awarded once a century.

Under the guidance of Viktor Martynov 17 PhD and 4 doctoral thesis were protected.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 519.767.2

УСК МАРТЫНОВА – ТРИДЦАТЬ ЛЕТ СПУСТЯ

Ефименко И.В.^{*}, Хорошевский В.Ф.^{**}

^{} Центр информационно-аналитических систем ИСИЭЗ НИУ ВШЭ,
г. Москва, Россия
iefimenko@hse.ru*

*^{**} Вычислительный центр им. А.А. Дородницына РАН,
Центр информационно-аналитических систем ИСИЭЗ НИУ ВШЭ
г. Москва, Россия
khor@ccas.ru, vkhoroshevsky@hse.ru*

В работе дается ретроспективный анализ систем представления знаний семейства УСК (Универсальный Семантический Код), разработанных в конце 70-х, начале 80-х годов прошлого века известным советским ученым В.В. Мартыновым. Показано, что УСК-3 Мартынова в определенном смысле опередил существующие в то время подходы к представлению лингвистических знаний и определил вектор использования методов и средств компьютерной лингвистики в практически значимых приложениях. Представлена попытка интерпретации идей В.В. Мартынова, положенных в основу УСК-3, с позиций современных достижений в области компьютерной лингвистики и искусственного интеллекта.

Ключевые слова: понимание естественного языка; семантический код; язык описания и исчисления смыслов; псевдофизическая логика; онтологическое моделирование предметной области.

Вместо введения

В 2014 году исполняется 90 лет со дня рождения Виктора Владимировича Мартынова – ученого, внесшего существенный вклад в исследования на стыке лингвистики с неклассическими логиками и в становление научного направления «представление и обработка знаний».

В.В. Мартынов окончил Одесский университет в 1948 году, в 1951 году – аспирантуру по славистике при Львовском университете, в 1952-1960 гг. был заведующим кафедрой иностранных языков Одесского университета, а в 1960 г. переехал в Минск, где много лет работал в Институте языкознания АН БССР. Доктор филологических наук, профессор В.В. Мартынов автор 20 книг и более чем 200 статей.

Три научные области были наиболее близки В. В. Мартынову. Первая – это славистика, вторая – компаративистика и, в частности, глоттогенез и онтогенез славян. Третья, которой Виктор Владимирович занимался с особым вдохновением, связана с искусственным интеллектом и информационными технологиями, где в 60-е годы XX века формировалось новое научное направление

– представление и обработка знаний.

С позиций сегодняшнего дня можно сказать, что основная заслуга В.В. Мартынова как ученого состоит в том, что ему удалось, опираясь на взаимосвязь лингвистики с теорией информации, предложить новый способ «исчисления языковых смыслов», а затем и универсальный семантический код как средство снятия «неопределеннозначности» естественного языка при построении формальных моделей предметных областей.

Настоящая работа посвящена обсуждению семейства языков представления знаний УСК, а изложение организовано следующим образом. Сначала дается краткое авторское обоснование того, почему и каким образом классический филолог и специалист в области компаративистики пришел к проблеме представления и обработки знаний в парадигматике неклассических логик, затем приводится ретроспективный анализ систем представления знаний семейства УСК Мартынова (Универсальный Семантический Код), которое в значительной мере опередило существующие в то время подходы к представлению лингвистических знаний и использованию методов и средств компьютерной лингвистики в практически значимых приложениях. В заключительной части

работы представлена попытка интерпретации идей В.В. Мартынова, положенных в основу УСК-3, с позиций современных достижений в области компьютерной лингвистики и искусственного интеллекта.

1. От компаративистики к представлению и обработке знаний

В области лингвистики и филологии в сфере интересов В. В. Мартынова входил широкий набор направлений: литературоведение и культурология, сравнительно-историческое языкознание, славистика и востоковедение. В. В. Мартынов внес существенный вклад в развитие дисциплин, находящихся на стыке гуманитарных и точных / инженерных наук, в рамках исследований по формализации семантики и лингвистическим аспектам искусственного интеллекта.

Однако, как представляется, основным направлением, в контексте которого следует рассматривать историю создания универсального семантического кода, является сравнительно-историческое языкознание, или компаративная лингвистика и, в частности, исследования В. В. Мартынова в области глоттогенеза славян и славяно-неславянских контактов ([Мартынов, 1963; Мартынов, 1983; Мартынов, 2003] и др.).

В силу необходимости работы с большими объемами разнородного языкового материала компаративистика предусматривает развитие формальных методов описания, цель которых – обеспечить возможности сравнения языков и обоснования гипотез об их происхождении и родстве, а также в той или иной степени реконструировать праязык. Одним из важных результатов при этом является классификация языков.

Как отмечает С. А. Старостин, выдающийся специалист в области сравнительного языкознания, «возникшая в начале XIX в. и с тех пор неуклонно развивавшаяся как в отношении предмета изучения, так и в отношении методологии, компаративистика послужила основой для методологии всего современного языкознания, явившись, по сути, главной стимулирующей силой для выхода общей лингвистики на новый, научный этап своего развития. К настоящему времени компаративистика является полноценной научной дисциплиной, оперирующей строго формализованными методами (в том числе компьютерными) с целью проникновения вглубь истории языков и реконструкции все более и более отдаленных от современности праязыков человечества» [Старостин, 2007].

Таким образом, можно отметить две важнейшие черты компаративистики, в которых, вероятно, следует искать истоки интереса В. В. Мартынова к проблематике создания универсального семантического кода: во-первых, это поиск единой основы естественных языков (общее ядро, единый

праязык, универсальные лингвистические явления) и, во-вторых, создание формализованных средств их описания.

Как представляется, исследования по компаративистике, с одной стороны, заложили основы «персональной траектории» В. В. Мартынова, результатом которой стало создание универсального семантического кода, с другой стороны – во многом определили области приложения УСК. Можно утверждать, что формальное языконезависимое представление лингвистических знаний является одной из ключевых задач для подавляющего большинства приложений, связанных с синтезом или анализом естественного языка. Однако именно для решений в области взаимодействия человека и компьютера, которые, прежде всего, интересовали В. В. Мартынова, это является наиболее очевидным. Подходы, основанные на универсальном представлении лингвистических знаний, сыграли важную роль и в развитии других типов приложений – например, в сфере машинного перевода, о чем будет сказано позднее.

2. Семейство УСК – ретроспективный анализ

2.1. Общие замечания

Как отмечается в работах [Поспелов, 1981; Поспелов, 1986], которые, по нашему мнению, сыграли значительную роль в формировании взглядов В.В. Мартынова на проблему создания универсального семантического кода, при использовании ЕЯ в качестве основы для построения языка представления знаний в нем выделяются следующие классы элементов (слов и словосочетаний), играющие функциональную роль в представлении знаний: **понятия**, **имена** и **отношения**.

При этом для любого естественного языка характерно наличие слов и словосочетаний, определяющих **понятия-классы**, обладающие определенными свойствами (например, «статья», «лаборатория»), имена служат для идентификации элементов, входящих в понятие-класс (например, понятие-класс «вычислительный центр» в качестве элементов может содержать понятия-классы «лаборатория» с определенным номером, который играет роль имени, а лаборатории в качестве своих элементов могут содержать отделы или сектора, тоже снабженные именами. В отличие от понятий-классов **понятия-процессы** описывают группы однородных процессов (например, «учебная нагрузка»). По своей функциональной роли близки к ним **понятия-состояния**, примерами которых могут быть словосочетания «нормальный режим», «компьютерная сеть работает» и т. п. Для идентификации понятий-процессов и понятий-состояний, как и в случае понятий-классов, могут использоваться имена. Важно, что для естественных

языков множества понятий и имен потенциально бесконечны.

Отношения служат для установления связей на множестве понятий или идентифицированных понятий. При этом уже сама идентификация реализуется с помощью специального отношения «называться». Существует гипотеза о конечности множества различных, не сводимых друг к другу, отношений для естественных языков (базовых отношений) и сводимости других отношений, присутствующих в ЕЯ-текстах, к комбинации базовых. Поэтому при построении моделей представления знаний для реальных предметных областей можно всегда считать, что мы имеем дело с конечным числом различных понятий, имен и отношений.

С помощью понятий, имен и отношений можно описывать ситуации, имеющие место в предметной области. Для этого в терминальный словарь (T) ЯПЗ вводятся три рассмотренных функциональных класса: понятия (V), имена (I) и отношения (R)

$$T = V \cup I \cup R, \text{ где}$$

$$V = B \cup D \cup G;$$

$$B = \{b_1, b_2, \dots, b_n\};$$

$$D = \{d_1, d_2, \dots, d_m\};$$

$$G = \{g_1, g_2, \dots, g_k\};$$

$$I = \{i_1, i_2, \dots, i_l\};$$

$$R = \{(\cdot), \rho, r_1, \dots, r_q\},$$

и строится совокупность следующих синтаксических правил, определяющих правильно построенные формулы (ППФ) в системе представления знаний:

1. Любой элемент, кроме элементов множества R , является ППФ.
2. Если α — любой элемент из V и β — любой элемент из I , то тройка $(\alpha\beta)$ есть ППФ.
3. Если δ и γ — любые элементы из V , то тройка $(\delta\gamma)$, где r — любой элемент из R , кроме скобок и ρ , есть ППФ.
4. Если ε и χ суть ППФ, то тройка $(\varepsilon\chi)$ есть ППФ.
5. Других ППФ нет.

Для примера рассмотрим следующую ЯПЗ-запись, удовлетворяющую введенным выше обозначениям, и попробуем восстановить ЕЯ-текст, соответствующий этой записи:

$$((\langle\langle\text{судно}\rangle\rangle r \langle\langle\text{№1}\rangle\rangle) r 3 (\langle\langle\text{причал}\rangle\rangle r \langle\langle\text{№8}\rangle\rangle)) \& \\ ((\langle\langle\text{груз}\rangle\rangle r \langle\langle\text{лес}\rangle\rangle) r 12 (\langle\langle\text{процесс}\rangle\rangle r \langle\langle\text{погрузка}\rangle\rangle))$$

Две тройки, входящие в первую «большую» скобку, соединенные отношением $r3$, имеющим значение «быть в окрестности», дают фразу «Судно №1 находится около причала №8». Следующая «большая» скобка соответствует фразе «Идет погрузка леса», непосредственная расшифровка которой даст предложение «Груз по имени лес участвует в процессе ($r12$) по имени погрузка». Но смысл этой фразы тождествен смыслу ранее приведенной фразы, которая по-русски звучит

естественнее. Таким образом, приведенная выше формальная запись может быть заменена одной фразой «Судно №1 находится под погрузкой леса у причала №8».

Не менее, а быть может более важной проблемой при создании ЯПЗ, является установление смысловой эквивалентности различных ППФ, которая обычно решается с помощью специальных ограничений на синтез ППФ, поскольку некоторые отношения не могут произвольно вводиться между понятиями произвольной природы. Проблема эта известна в структурной лингвистике как проблема построения языка семантических представлений (СЕМП), к которому предъявляются следующие требования. Если ЕЯ-фраза, по мнению его носителей, имеет смысл, то в СЕМП этой фразе должно соответствовать, по крайней мере, одно представление и обратно — если ЕЯ-фраза, по мнению его носителей, смысла не имеет, то в СЕМП для нее не должно найтись ни одного представления. Если две ЕЯ-фразы, по мнению его носителей, имеют совпадающий смысл, то в СЕМП им должно соответствовать либо одно общее представление, либо различные представления, но такие, что с помощью системы формальных преобразований фраз в СЕМП они переводятся одно в другое. Таким образом, построение СЕМП предполагает наличие эффективной процедуры перевода фраз естественного языка в СЕМП и эффективную процедуру установления эквивалентности записей в СЕМП.

И, наконец, в рамках семантического представления смысла существует еще одна важная проблема — проблема установления противоречивости описаний. По существу, ее решение сводится к построению СЕМП второго уровня, где вместо семантических правил соединения слов во фразе (и в дополнение к ним) разрабатываются семантические правила соединения фраз в тексте.

Таким образом, даже на уровне достаточно простых моделей представления смысла в те годы, когда В.В. Мартыновым создавались ЯПЗ семейства УСК, имелись серьезные научные проблемы, как в лингвистике, так и в области представления знаний. И, если учесть, что в реальных моделях представления знаний требовалось в дополнение к понятиям, именам и отношениям учитывать еще, как минимум, наличие в ЕЯ императивов, квантификаторов, модификаторов, а также модальностей и оценок, станет ясно, что задача создания универсального семантического кода для представления смысла текстов на естественном языке была чрезвычайно сложной. Справедливости ради отметим, что данная проблема не решена в общем случае и до сих пор, через 30 лет после того, как ее решением занялся В.В. Мартынов.

В процессе исследований В.В. Мартыновым предлагались различные версии УСК [Мартынов, 1977], но последним и, по-видимому, наиболее проработанным в этом семействе ЯПЗ является

УСК-3 [Мартынов, 1984]. Как представляется авторам настоящей работы, именно в процессе создания этого языка В.В. Мартынов окончательно определил основные концепции всех систем представления знаний семейства УСК, которые состоят в следующем:

- Язык представления знаний должен быть языком формального описания и исчисления смыслов и, следовательно, псевдофизическим языком и псевдофизической логикой одновременно.
- УСК должен быть языком полной экспликации смысла (т.е. каждый комбинаторный тип цепочки элементов должен иметь один и только один смысл).
- УСК должен быть языком универсальной канонизации (т.е. ограничения, накладываемые на его систему, не должны зависеть от того, какой фрагмент реального мира описывается).
- УСК должен быть языком неконвенционального представления семантики (т.е. его цепочкам семантика должна не приписываться, а выводиться из аксиом-универсалий).
- УСК должен строиться как система, способная понимать мир (т.е. формировать новые понятия и строить гипотезы о причинах и следствиях ситуаций, причем и то, и другое должны реализовываться на базе формальных преобразований цепочек).

Как следует из приведенных выше требований, уже в начале 80-х годов прошлого века В.В. Мартыновым была поставлена сверх сложная и, вместе с тем, чрезвычайно актуальная проблема создания формальной модели представления знаний, в рамках которой были бы интегрированы не только методы собственно описания смысла ЕЯ-текстов, но и средства формального описания человеческих рассуждений о мире.

2.2. УСК: «чужой среди своих»

Как указывалось выше, В. В. Мартынов внес вклад в развитие целого ряда направлений в сфере филологии и лингвистики. Основными работами В. В. Мартынова, которые изучаются лингвистами, можно считать труды по компаративистике и славистике. В частности, классической является уже упоминавшаяся выше монография «Язык в пространстве и времени. К проблеме глоттогенеза славян». Однако необходимо отметить, что непосредственно УСК не нашел широкого признания и применения среди лингвистов, чему может быть дано следующее объяснение.

Для того, чтобы описать естественноречевые выражения, порождаемые носителями, с помощью УСК, требуется построить сложные, многокомпонентные цепочки, что ограничивает возможности прикладного применения соответствующего формализма. Кроме того, при таком описании существует риск стирания значимых семантических различий между близкими, но не идентичными фрагментами

естественно-языковых текстов (см. пример в [Мартынов, 1984]) – как в части естественно-языковых цепочек, так и в части описаний на УСК: «Высказывание Главк обеспечивает завод оборудованием реально означает Главк держит оборудование на заводе. SAOÖ обозначает X держит Y (в себе). SAÖO – X держит себя в Z-е (\equiv X находится в Z-е). SAÖO – X держит себя в себе (\equiv X есть). Последняя интерпретация, вероятно, нуждается в некотором содержательном комментарии. Если мы интерпретировали X держит себя в Z-е как X находится в Z-е, то X держит себя в себе сначала интерпретируется как X находится в себе, а потом как X есть, потому что быть значит –находиться где-то (покоиться)», а находится где-то в любом случае означает –находиться в себе»).

Что касается использования УСК в области лингвистической теории, то здесь наблюдается ряд расхождений с общепринятыми методами и принципами описания языка, а также, в некотором роде, с имеющимися задачами.

С одной стороны, подход, лежащий в основе УСК, в частности, идея «разложения» языковых примеров на элементарные семантические единицы, соответствует традициям формальной семантики (см. пример в [Мартынов, 1984] – интерпретацию высказывания «Носильщик везет чемодан в тележке» как «Носильщик держит чемодан в тележке и перемещает тележку», т.е., в терминах УСК: X держит Y в Z-е и перемещает Z). Идея трансформации одних цепочек УСК в другие также перекликается с идеями перехода от глубинной структуры к поверхностной в генеративной лингвистике. Не противоречит лингвистическим традициям и тот факт, что УСК предназначен, прежде всего, для описания нарративных предложений. Так, например, известный российский лингвист, автор работ по компаративистике и происхождению языка С. А. Бурлак отмечает: «Если вы посмотрите на знаменитые лингвистические примеры, они все до единого – нарративные тексты: «Фермер убил утенка», «Бесцветные зеленые идеи яростно спят», «Глокая куздра штеко будланула бокра и курдячит (или «кудрячит») бокренка». И никто не начал свою теорию с предложений типа «Дай, пожалуйста!» или, скажем, «Марш отсюда!»». Говорится, что такие предложения неполные, что они особые. А «правильные», «настоящие» – это именно нарративные предложения, то есть, по сути, комментарии» (см. стенограмму лекции на <http://polit.ru/article/2008/11/07/lang/>).

С другой стороны, задача канонизации естественно-языковых высказываний, являющаяся основой подхода к описанию языка с использованием УСК, по сути своей предполагает маргинальный характер тех выражений, которые должны быть канонизированы. Это, как и ограниченный состав атомарных элементов в УСК, изначально является препятствием для описания естественного языка на уровне, необходимом для развития лингвистической теории. Впрочем, это

отмечается самим автором: «Необходимость канонизации естественного языка для диалога человек – ЭВМ объясняется тем, что естественный язык в его полном виде нельзя сделать понятным для машины» [Мартынов, 1984]. В некотором роде, в УСК сделана попытка стереть границы между лексической и грамматической (в частности, синтаксической) составляющими языка и описать их с использованием единого формализма, а также создать «универсальное синтаксическое описание». Между тем, известно, что описание синтаксиса является одной из самых сложных задач и для компаративистики (например, при реконструкции праязыка), где, как указывалось выше, могут быть в определенном смысле найдены истоки УСК, и для прикладной лингвистики, к которой относятся работы по УСК как таковые. В частности, полномасштабное синтаксическое описание считается камнем преткновения в области извлечения информации из текстов (Information Extraction), машинного перевода и других задач автоматической обработки естественного языка.

В области анализа (и синтеза) естественного языка «драйвером» создания УСК послужила, прежде всего, задача человеко-машинного общения, диалогового взаимодействия с ЭВМ. В частности, В. В. Мартынов исследует работы А. П. Ершова, И. А. Мельчука, А. С. Нариньяни, Э. В. Попова по указанной проблематике. И несмотря на то, что УСК не стал общепринятым способом описания естественного языка в лингвистике, в т. ч. прикладной, можно констатировать, что В. В. Мартынов сформулировал целый ряд важных принципов и разработал методологию и инструментарий, позволяющие взглянуть на естественный язык с новых позиций, важных для решения целого ряда прикладных задач.

Одной из таких задач является машинный перевод естественного-языковых текстов. Детальный анализ методов и средств машинного перевода, в том числе, с использованием универсальных семантических языков (что наиболее близко к идее УСК), выходит за рамки настоящей работы. Здесь целесообразно отметить лишь то, что подходы, основанные на использовании единого формального описания, к которому приводится текст на языке-источнике (задача анализа) и на основе которого порождается представление на языке-цели, стали исторически первыми в прикладной области «машинный перевод». Так, считается, что сама идея автоматического перевода зародилась в XVII веке, когда Рене Декарт предложил универсальный язык, в котором один символ выражает эквивалентные идеи, формулируемые на различных естественных языках. В XX веке универсальные описания, промежуточные между языком-источником и языком-целью, активно использовались в лингвистических подходах, основанных на правилах (Rule-based approaches). Однако на современном этапе развития систем машинного перевода ведущими можно считать подходы, использующие статистические методы и

инструменты корпусной лингвистики, в частности, подходы, которые относятся к классу Example-based и которые основаны на применении параллельных корпусов. В частности, такого рода подход используется переводчиком Google. Однако это не означает, что Rule-based подходы, опирающиеся на промежуточную формальную спецификацию, потеряли свою актуальность. Поэтому наиболее перспективными на современном этапе представляются гибридные методы, сочетающие достоинства лингвистических и статистических подходов к машинному переводу.

2.3. УСК: «свой среди чужих»

Приведенные во введении к настоящей работе краткие библиографические сведения о научном пути профессора В.В. Мартынова показывают, что по своему образовательному и научному background-у это был филолог, который в процессе работы «перешел» в лагерь специалистов по искусственному интеллекту и компьютерной лингвистике. Следует отметить, что в те годы, когда само научное направление представления и обработки знаний только формировалось, это было не единичным случаем, а скорее естественным следствием междисциплинарности новой науки. И появление в научном сообществе математиков и программистов специалистов из области филологии, психологии и других направлений воспринималось с энтузиазмом, который активно поддерживался Научным советом по искусственному интеллекту АН СССР, в рамках междисциплинарных проектов «Диалог» и «Диалог-2», а позже и в рамках международных рабочих групп РГ-18 и РГ-22. Общеизвестными «центрами притяжения» в новом научном сообществе были такие научные лидеры, как профессор Д.А. Поспелов и член-корр. РАН А.Е. Кибрик, академик Г.С. Поспелов и академик А.П. Ершов, в общении с которыми формировались и активно работали молодые коллективы из разных республик и городов нашей страны. Естественным образом «вписался» в это научное сообщество и В.В. Мартынов, который сконцентрировался на проблеме языков, с помощью которых интеллектуальные устройства могли бы вести диалог с человеком.

При этом в дискуссиях о языке, которые велись в то время (нужно ли учить ЭВМ понимать естественный язык или следует разрабатывать искусственный язык, удобный для общения ЭВМ с человеком), В.В. Мартынов четко стал на позицию применения ограниченных вариантов естественных языков с основным упором на канонизацию языка общения, т. е., на спецификации того, какие синтаксические и лексические средства языка можно использовать, а какие нельзя в соответствии с некоторой теорией эффективного представления знаний.

Необходимость канонизации естественного языка для диалога человек — ЭВМ обосновывалась В.В. Мартыновым тем, что естественный язык в его

полном виде нельзя сделать понятным для машины уже в силу принципиальной семиологической эллиптичности и многозначности (омонимичности) ЕЯ-фраз, что ведет к неэксплицированности их смысла на уровне структурных описаний. При этом правильно отмечалось, что лишь незначительная часть информации, извлекаемой адресатом из ЕЯ-сообщения, содержится в самом сообщении, а большая ее часть восстанавливается на основе коллективного и индивидуального опыта человека (пресуппозиции), что подтверждалось и специалистами по искусственному интеллекту. Так, например, Р. Шейк и Р. Абельсон в своей, теперь уже классической, работе [Шенк и др., 1975] отмечали, что «исследователи понимания естественного языка в течение некоторого времени уже чувствуют, что проблема может быть решена в той мере, в какой мы способны характеризовать наши знания о мире...».

Как следствие, в качестве единственного средства преодоления барьеров на пути создания языка общения человека с ЭВМ В.В. Мартынов видел канонизацию ЕЯ, которая может быть ориентирована на проблему либо на сам язык, т. е. проводится в соответствии с семиологической теорией эффективного представления мира без его ограничения. Следует отметить, что примерно такая же позиция была в то время и других исследователей [Ершов, 1982].

В.В. Мартынов в своей монографии [Мартынов, 1984] отмечал, что «лингвисты, работающие в области семантического синтаксиса, и кибернетики, занятые проблемами искусственного интеллекта, по существу, ведут исследования в одном направлении, но, поскольку при этом используется различная терминология, в известной мере дублируют друг друга. То, что лингвисты называют пресуппозицией в широком смысле данного понятия (без различия ее видов), в теории ИИ оказывается моделью мира, или представлением знаний. Если лингвисты сомневаются насчет того, считать ли пресуппозицию лингвистической или экстра лингвистической категорией, то кибернетики уверены в экстра лингвистичности модели мира. Вернее, такого вопроса у них просто не возникает»... В действительности это утверждение было не вполне верным даже в то время, а сейчас, по-видимому, корректнее говорить о интеграции лингвистических и внелингвистических подходов к представлению и обработке знаний.

Вместе с тем, можно согласиться с В.В. Мартыновым в том, что в большинстве систем искусственного интеллекта «лингвистический процессор действует только на входе, т. е. является устройством, выполняющим перевод с ограниченного естественного языка на внутренний язык машин, на котором информация обрабатывается внутренними процессорами. Внутренний язык машин, хотя и назван языком, в действительности рассматривается как объект не лингвистический, а логический, как система

исчислений... В системах естественного интеллекта дело обстоит иначе. Язык информации, поступающей на вход, и внутренний язык совпадают. Естественный язык одновременно выполняет функции описания феномена и рассуждения о нем. Если канонизированный естественный язык или искусственный представить в виде формальной системы, то он сможет стать языком описания и исчисления смыслов, или псевдофизическим языком и псевдофизической логикой. Иными словами, отчужденное от человека кибернетическое устройство получит инструмент и с его помощью будет описывать реалии и ситуации, в которых эти реалии могут находиться (псевдофизический язык), вести рассуждения по поводу возможных изменений ситуаций (псевдофизическая логика)».

Целесообразность разработки подобного языка для В.В. Мартынова и многих других специалистов подкреплялось результатами исследований и разработок в области ИИ. Например, работами по ситуационному управлению [Поспелов, 1981], где явно указывалось, что существующие в то время «интеллектуальные системы должны, но пока не умеют: а) формировать модель мира в виде многоуровневых обобщенных знаний о классах объектов и ситуаций; б) устанавливать полные ассоциативные связи между классами объектов и ситуаций; в) строить формальную классификацию задач; г) принимать решения». А основная причина этих недостатков виделась исследователями в отсутствии эффективных способов «вложения семантических знаний в формализмы представления» [Попов и др., 1976]. При этом фактически речь шла о том, как регулярно соотносить план выражения и план содержания внутреннего языка ЭВМ, который для этого должен стать семантически мощным полифункциональным языком описания и исчисления смыслов. На разработке именно такого языка и сконцентрировался В.В. Мартынов.

Как представляется авторам настоящей работы, большое влияние на разработку семейства языков УСК оказал один из активно использовавшихся в то время в теории ситуационного управления для представления знаний язык RX-кодов [Скореходько, 1968]. В алфавите языка RX-кодов использовались два типа символов, — **X** (названия предметов) и **R** (названия отношений), — и ряд специальных обозначений. Для терминов языка было характерно, во-первых, ступенчатое кодирование, что давало возможность экономного и максимального приближения к полному выражению релевантных семантических характеристик, а, во-вторых, использование отношений, как в синтагматике, так и в парадигматике, что имитировало формирование новых понятий посредством разноструктурной модификации старых. В-третьих, сами отношения в языке RX-кодов образуются как производные путем сочетания элементарных. Эти и ряд других характеристик языка RX-кодов превратили его в весьма эффективный, по тому времени, инструмент

информационного поиска. Однако, поскольку язык создавался именно для целей информационного поиска, он не мог претендовать на роль полифункционального языка представления и обработки знаний. Именно на этом позиционировались языки семейства УСК Мартынова и, в частности, обсуждаемый ниже УСК-3.

Теории УСК предшествует логическая теория отношений, поэтому часть базовых предложений УСК формулируется в терминах последней. При этом вводятся три группы базовых предложений: о структуре отношений; о свойствах отношений; семиологические. Первые две группы объединяются как логические и, таким образом, противопоставлены третьей.

Для каждого отношения предполагается класс элементов, составляющих область отношения, и класс элементов, образующих противообласть (конверсную область) отношения. Отношения, согласно А. Тарскому, делятся по структуре на три и только три класса: первой степени (элемент-элемент) — XY ; второй степени (элемент-отношение) — XYR ; третьей степени (отношение—отношение) — $XYR(WR)$.

В соответствии с числом элементов или отношений, выступающих в роли элементов, отношения делятся на бинарные и n-арные, а свойства отношений сводятся к трем типам: рефлексивность, симметричность и транзитивность. Классы отношений, упорядоченные по их свойствам, противопоставлены друг другу по

- рефлексивности—нерефлексивности — антирефлексивности,
- симметричности—несимметричности—антисимметричности,
- транзитивности—нетранзитивности—антитранзитивности.

Дополнительно к базовым предложениям из области теории отношений вводятся базовые предложения из модальной логики и теории множеств.

Утверждения могут получать теоретико-множественную интерпретацию. Тогда утверждение с квантором общности интерпретируется посредством полного, или универсального, множества, а утверждение с квантором существования интерпретируется посредством частичного, или парциального, множества отдельных объектов, входящих в предметную область, на которую распространяется данное утверждение.

При переходе от логической базы теории УСК к семиологической логическую символику В.В. Мартынов заменяет семиологической. При этом элемент класса элементов, составляющих область отношения, обозначается как S (субъект, левая маргинальная позиция), элемент класса элементов, образующих противообласть отношения,

обозначается как O (объект, правая маргинальная позиция), а отношение S к O выражается как A (акция, центральная позиция). Такого рода цепочка (SAO) называется ядерной. Кроме единичных элементов (S, A, O) в состав цепочки могут входить множественные элементы (S), (A), (O).

Таким образом, бинарное отношение между элементами S и O ядерной цепочки определяется наличием специального элемента A, задающего данное отношение, называемое эксплицитным. В n-арном отношении, сводимом к бинарным, не всякое из них эксплицитно. Отношение, определяемое отсутствием специального элемента, задающего это отношение, называется имплицитным. В соответствии с определением не может быть ядерной цепочки, состоящей из более чем одного эксплицитного отношения или из одного имплицитного отношения.

В случае незамещенности позиции ее элементом она замещается элементом S. Незамещенность позиции O в цепочке $SA\bar{O}$ (черта над позицией означает ее незамещенность) указывает на то, что позицию O замещает S, и, следовательно, элемент A задает отношение S к самому себе (рефлексивность). Рефлексивность S в ядерной цепочке означает отношение S к самому себе, что получает в семиологической терминологии название регрессивной доминации (в отличие от прогрессивной или собственно доминации). Незамещенность позиции A ($S\bar{A}O, S\bar{A}\bar{O}$) не интерпретируется в терминах теории отношений, поскольку в этом случае S замещает элемент, который сам задает отношение между S и O, а позиция S в ядерной цепочке не может быть незамещенной. Имплицитное отношение как дополнительное к эксплицитному вводится при расширении ядерной цепочки путем увеличения числа позиций одного из ее элементов ($SAO \Rightarrow SAOO$; $SAO \Rightarrow SSAO$; $SAO \Rightarrow SAAO$). Так как S всегда доминирует над остальными элементами, доминация сохраняется и после расширения. В этом случае S первое (S ядерной цепочки) доминирует над вторым S, возникшим в результате увеличения числа позиций.

Увеличение числа позиций в результате введения в ядерную цепочку имплицитной доминации называется мультипликацией позиции. Цепочка с тернарным отношением и мультипликацией позиции называется расширенной, а возникшие в результате позиции — вторичными. Расширенные цепочки, как и ядерные, являются отношениями первой степени, или эксплицитными отношениями элемент-элемент. Не может быть расширенных цепочек с мультипликацией обеих маргинальных позиций ($*SSAOO, *SSAAOO$). Приведенные выше определения специфицируют возможность следующих типов расширенных цепочек: SSAO, SAAO, SAOO, SSAO, SAAO и только их.

Вторичные позиции расширенной цепочки могут быть замещены цельнооформленными цепочками

(S(SAO)AO, SA(SAO)O). В результате в маргинальных позициях возникнут отношения второй степени — имплицитные отношения элемент-отношение.

Цепочками, воплощающими отношения второй степени, могут быть только расширенные. В цепочках с отношением второй степени отношения занимают только вторичные позиции. Цельнооформленные цепочки, занимающие вторичные позиции расширенных цепочек, сами подчиняются тем же правилам порождения, которым подчиняются ядерные и расширенные цепочки.

Отношения третьей степени, или отношения отношений, воплощаются в сложные цепочки, все позиции в них замещаются цельнооформленными цепочками (ядерными и расширенными). Сложные цепочки представляют собой цепочки типа (SAO)A(SAO) (SAO).

Бинарные отношения типа *(SAO)A(SAO) и *(SAO) (SAO) A (SAO) (SAO) невозможны, что объясняется их сводимостью к ядерным цепочкам и противоречит определению последних. Невозможно также сведение этих цепочек к расширенным, ибо последним не свойственно эксплицитное отношение отношений и замещение первичных позиций цельнооформленными цепочками.

Сложная цепочка не сводима к имплицитному бинарному отношению отношений типа (SAO)(SAO). Такого рода выражение не составляет цепочки. Оно не может быть ядерной цепочкой, поскольку невозможна ядерная цепочка, состоящая из одного имплицитного отношения и воплощающая бинарное отношение отношений.

Структура цепочек, замещающих позиции сложной цепочки, определяется следующими базовыми предложениями: а) цепочки в позиции субъекта реализуются всегда как цепочки типа SSAO или S(SAO)AO; б) правая часть сложной цепочки состоит из последовательности двух эквивалентных цепочек или цепочек, различающихся по знаку. На основании этого определяются следующие три типа сложных цепочек: (SSAO)A(SAO) (SAO), (SSAO)A - (SAOO) (SAOO) и (SSAO) A(SSAO) (SSAO) или соответственно с S (SAO)AO в левой части. Сложные цепочки с незамещенной позицией одного из объектов невозможны, а производные цепочки строятся на основе доминанции смежных маргинальных элементов в порядке их следования слева направо.

На основании предшествующих предложений определяется порядок диффузных преобразований для цепочек типа SAOO (исключая тавтологии): $SAOO \Rightarrow SAO\bar{O} \Rightarrow SA\bar{O}O \Rightarrow SA\bar{O}\bar{O}$.

Вторым (вслед за диффузией) вводится преобразование, которое представляет собой превращение позиции элемента S^2 в позицию элемента O^2 , или замену второго субъекта вторым

объектом при сохранении всех остальных позиций и их элементов. Такое преобразование В.В. Мартынов называет транспозицией (переносом элементов вторичных позиций вместе с самими позициями). Первичные позиции и их элементы остаются при этом неизменными. Транспозиция асимметрична. Второй субъект заменяется вторым объектом, но обратное неверно. Транспозиция с сохранением позиций невозможна.

Транспозиция включается в качестве начального преобразования в порядок диффузных преобразований. Поскольку $SSAO \Rightarrow SAOO$ асимметрично, устанавливается следующий порядок преобразований цепочек УСК: $SSAO \Rightarrow SAO\bar{O} \Rightarrow SA\bar{O}O \Rightarrow SA\bar{O}\bar{O}$.

Выше мы определили способы порождения и преобразования цепочек УСК по В.В. Мартынову и тем самым дали описание его синтаксиса в статике и динамике. Тезаурус основных понятий УСК-3 представлен на Рисунке 1.

Аналогичным образом описывается и его семантика. При этом в качестве метаязыка используется канонизированный вариант русского языка.

Не имея возможности в данной работе полностью описать УСК-3, авторы отсылают заинтересованных читателей к монографии [Мартынов, 1984], а ниже приводят несколько примеров семантических представлений ЕЯ-фраз с помощью УСК-выражений, взятых из этой работы.

Так, например, в силу рассмотренных выше свойств УСК-выражений, цепочка SAO интерпретируется как «X преобладает над Y-ом», цепочка $SA\bar{O}$ — как «X преобладает над собой», цепочка $S\bar{A}O$ — как «X постоянно преобладает над Y-ом», а цепочка $S\bar{A}\bar{O}$ — как «X постоянно преобладает над собой».

Сложнее интерпретируются расширенные цепочки. Так, например, фразе «Директор пьет кофе в кабинете» в действительности должна интерпретироваться двумя фразами: «Директор пьет кофе» & «Директор находится в кабинете» («Директор держит себя в кабинете»). УСК-цепочка, соответствующая первой фразе, очевидна. Второй же фразе соответствует цепочка $SA\bar{O}O$, которая интерпретируется как «X держит себя в Z-е».

И, наконец, совсем сложной и далеко не всегда соответствующей ожиданиям носителей языка (в данном случае русского) является следующая интерпретация цепочки $(SSAO)A(S\bar{A}\bar{O}O)(S\bar{A}\bar{O}O)$: «X посредством Y-а воздействует на Z, в результате чего сначала Z является частью V, потом Z является частью V». В.В. Мартынов в данном случае рассматривает несколько ступеней канонизации исходного текста. Так, на второй ступени представленная выше формулировка преобразуется в следующую: «X посредством Y-а воздействует на Z, в результате чего Z сохраняется как часть V \Rightarrow X препятствует Z-у перестать быть частью V».

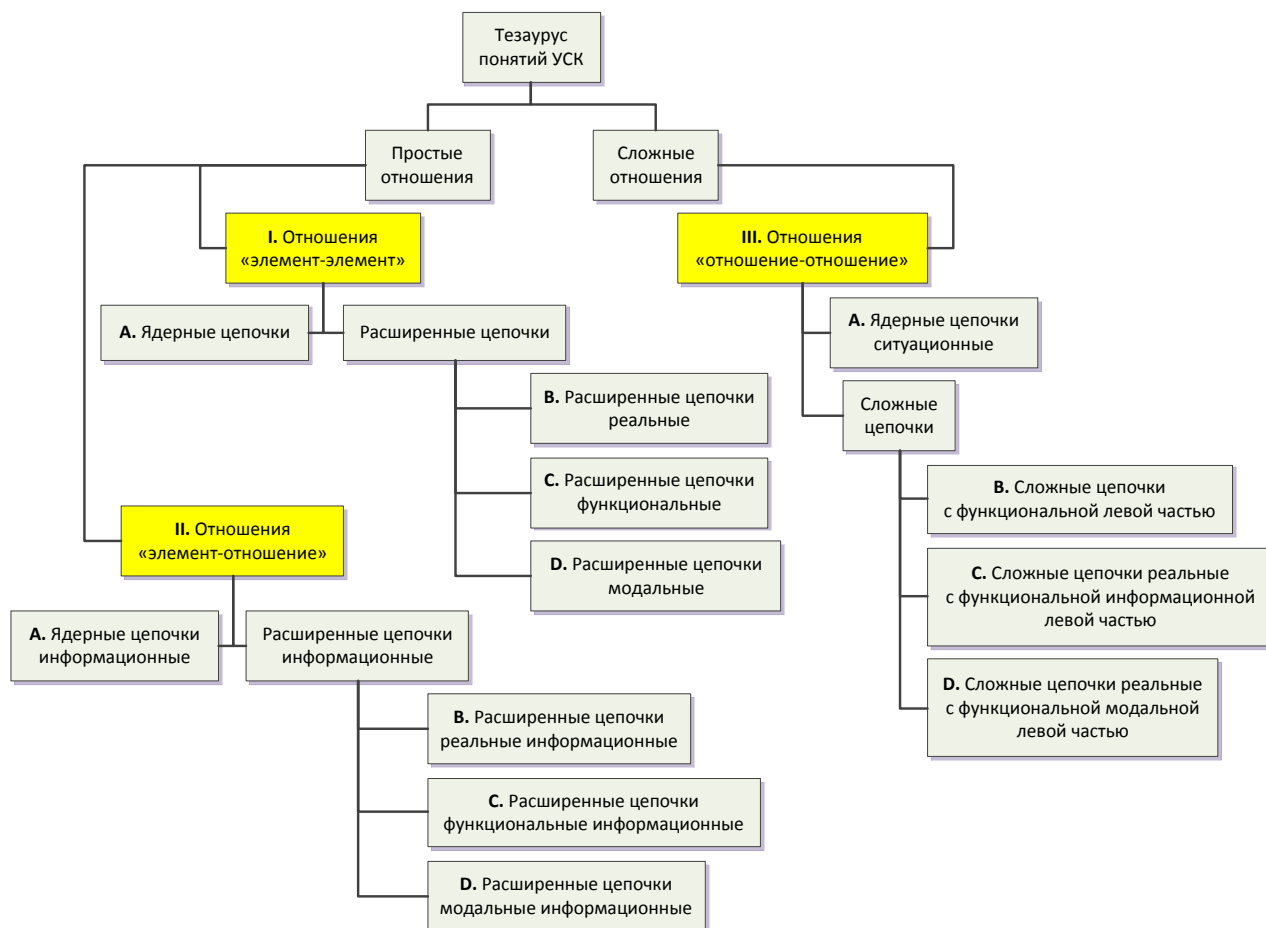


Рисунок 1 – Тезаурус основных понятий УСК-3

Как следует из приведенных выше примеров, задача преобразования ЕЯ-текста в УСК-представление является, в свою очередь, достаточно сложной. Именно это, на наш взгляд, стало еще одним препятствием на пути практического использования УСК в реальных системах того времени.

Заключение

В настоящей работе дан ретроспективный анализ систем представления знаний семейства УСК, разработанных в конце 70-х, начале 80-х годов прошлого века известным советским ученым В.В. Мартыновым.

С позиций сегодняшнего дня можно констатировать, что основные идеи, положенные в основу универсального семантического кода, актуальны до сих пор.

В первую очередь это идея единого языка и для представления, и для манипулирования знаниями. Не менее важно и то, что такой язык должен базироваться на идеях псевдофизических логик. И, наконец, идея о том, что семантика цепочек языка должна выводиться из системы аксиом.

В настоящее время часть этих идей уже воплощена в современных системах представления знаний. В частности, это идея использования

псевдофизических логик в качестве базиса для представления знаний, которая предполагает формальную спецификацию семантики выражений ЯПЗ [Поспелов, 1986]. Вместе с тем, идея единого языка для представления и манипулирования знаниями в настоящее время еще не нашла своего воплощения, а технологическая целесообразность привела к тому, что эти две компоненты явно разделяются – для представления знаний используются онтологические модели и языки типа OWL, а для манипулирования знаниями – специальные языки запросов к базам знаний типа SPARQL и/или языки машин вывода на правилах типа Rule ML.

Несколько иначе в настоящее время видится и перевод с ЕЯ на язык представления знаний. Если в концепции В.В. Мартынова для этого предполагалось использовать специальные табличные формы и словари канонизированного русского языка [Мартынов, 1984], то в современных системах понимания ЕЯ акцент делается на извлечении информации из текстов и трансформации полученных результатов в системы взаимосвязанных онтологий [Хорошевский, 2008; Efimenko, et al., 2009; Хорошевский, 2009; Хорошевский, 2010; Хорошевский, 2012].

Однако общее направление универсализации семантических представлений на основе

формальных систем представления и манипулирования знаниями по-прежнему развивается и уже приносит практически значимые результаты.

Библиографический список

- [Efimenko, et al., 2009] Efimenko I., Minor S., Starostin A., Drobazko G., Khoroshevsky V. Generating Semantic Content for the Next Generation Web, Chapter in Monograph "Semantic Web", Publisher IN-TECH, 2009, ISBN 978-953-7619-33-6.
- [Ершов, 1982] Ершов А. П. К методологии построения диалоговых систем: феномен деловой прозы // Вопросы кибернетики. Общение с ЭВМ на естественном языке. М., 1982.
- [Мартынов, 1963] Мартынов, В.В. Славяно-германское лексическое взаимодействие древнейшей поры (к проблеме прародины славян). Минск, 1963.
- [Мартынов, 1977] Мартынов, В.В. Универсальный семантический код (Грамматика. Словарь. Тексты). – Минск: «Наука и техника», 1977.
- [Мартынов, 1983] Мартынов, В.В. Язык в пространстве и времени. К проблеме глоттогенеза славян. М., 1983.
- [Мартынов, 1984] Универсальный семантический код: УСК-3 / Мартынов В.В.; – Минск: «Наука и техника», 1984.
- [Мартынов, 2003] Мартынов, В.В. Кельто-славянские этноязыковые контакты // Мовознаўства. Літаратура. Культаралогія. Фалькларыстыка: даклады беларускай дэлегацыі на XIII Міжнародным з'ездзе славістаў, Любляна, 2003 / НАН Беларусі камітэт славістаў. Мінск, 2003.
- [Попов и др., 1976] Попов Э.В., Фирдман Г.Р. Алгоритмические основы интеллектуальных роботов и искусственного интеллекта. – М., 1976.
- [Поспелов, 1981] Поспелов Д.А. Логико-лингвистические модели в системах управления. – М., 1981.
- [Поспелов, 1986] Поспелов Д. А. Ситуационное управление: теория и практика, М., Наука, 1986.
- [Скороходько, 1968] Скороходько Э.Ф. Информационно-поисковая система БИТ. – Киев, 1968.
- [Старостин, 2007] Старостин, С.А. Труды по языкознанию. - М., 2007. - С. 770-778.
- [Хорошевский, 2008] Хорошевский, В.Ф. Пространства знаний в сети Интернет и Semantic Web (Часть 1) / В. Ф. Хорошевский // Искусственный интеллект и принятие решений. - 2008. - № 1. - С.80-97.
- [Хорошевский, 2009] Хорошевский В.Ф. Пространства знаний в сети Интернет и Semantic Web (Часть 2) // Искусственный Интеллект и Принятие решений, № 4, 2009, стр. 15-36.
- [Хорошевский, 2010] Хорошевский В.Ф. Извлечение информации из текстов на конференциях серии Диалог: взгляд соседа по лестничной клетке //Труды международной конференции «Диалог–2010». Компьютерная лингвистика и интеллектуальные технологии. М.: 2010.
- [Хорошевский, 2012] Хорошевский В.Ф. Пространства знаний в сети Интернет и Semantic Web (Часть 3) // Искусственный Интеллект и Принятие решений, № 1, 2012, стр. 3-38.
- [Шенк и др., 1975] Шенк Р., Абельсон Р. Сценарии, планы, знания. // В сборнике трудов IV Международной объединенной конференции по искусственному интеллекту. –М., 1975.

MARTYNOV'S USC – 30 YEARS LATER

Efimenko I.V. *, Khoroshevsky V.F. **

** Center for Information Intelligence Applications
of Institute for Statistical Studies and Economics
of Knowledge, NRU HSE*

Moscow, Russia

iefimenko@hse.ru

*** Institution of Russian Academy of Sciences
Dorodnicyn Computing Centre of RAS,
Center for Information Intelligence Applications of
Institute for Statistical Studies and Economics of
Knowledge, NU HSE*

Moscow, Russia

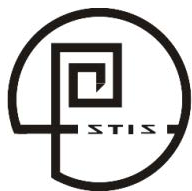
khor@ccas.ru, vkhoroshevsky@hse.ru

The paper represents a retrospective analysis of knowledge representation systems of the so called USC (i.e. universal semantic code) family. This formalism was developed in 1970-80s by a Soviet scientist Victor V. Martynov.

It is stated and shown that one of the most sophisticated versions of USC, which is USC-3, was ahead of many contemporarily existing approaches to linguistic knowledge representation in a number of ways. Martynov was among first evangelists of using methods and tools of computational linguistics in human-computer interaction applications. His approach has set a framework for following works.

The paper attempts to interpret Martynov's ideas which form the basis of the USC third version within a context of currently dominating paradigms of computational linguistics and artificial intelligence.

Keywords: natural language processing; semantic code; formal languages; calculation of meanings; pseudophysical logics; ontology engineering



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:658.51

ОТ СЕМАНТИЧЕСКОГО КОДА К КОГНИТИВНОЙ ЛИНГВИСТИКЕ, СЕМИОТИКЕ И ИНФОРМАТИКЕ: НАСЛЕДИЕ В.В.МАРТЫНОВА

Тарасов В.Б

*ФГБОУ ВПО «Московский государственный технический университет им. Н.Э.Баумана»
г. Москва, Россия*

Vbulbov@yahoo.com

Работа посвящена анализу ряда составляющих научного наследия выдающегося советского и белорусского ученого-лингвиста Виктора Владимировича Мартынова, пионера семиологического подхода в искусственном интеллекте, инициатора построения формальных лингвистических моделей с помощью абстрактных алгебр, автора знаменитого языка представления знаний «Универсальный семантический код», одного из родоначальников ряда новых «наук-перекрестков»: когнитивной лингвистики, когнитивной семиотики, когнитивной информатики.

Ключевые слова: искусственный интеллект; представление знаний; семиотика; лингвистика; исчисление смыслов; универсальный семантический код; когнитивная семиотика; когнитивная информатика.



Участники учредительного съезда Советской ассоциации искусственного интеллекта

Введение

В январе 2014г. исполнилось 90 лет со дня рождения выдающегося советского и белорусского ученого, основоположника одной из ведущих научных школ формализованной семантики, заслуженного деятеля науки Белорусской ССР, доктора филологических наук, профессора Виктора Владимировича Мартынова. Круг научных интересов и достижений В.В.Мартынова был весьма широк: он простирался от лингвистики, литературоведения, культурологии, этимологии, славистики, востоковедения до кибернетики, информатики и искусственного интеллекта (ИИ).



Виктор Владимирович Мартынова (1924-2013)

Виктор Владимирович был участником учредительного съезда Советской ассоциации искусственного интеллекта, состоявшегося в мае 1989г. в Коломне, на котором он был избран членом научного совета САИИ. К этому времени он был хорошо известен в сообществе специалистов по ИИ как создатель нового метода и языка представления знаний, названного «Универсальный семантический код» (УСК), автор замечательных книг и брошюр, среди которых можно указать монографии: «Кибернетика, семиотика, лингвистика» [Мартынов, 1966]; «Семиологические основы информатики» [Мартынов, 1974]; «Универсальный семантический код. Грамматика, Словарь, Тексты» [Мартынов, 1977]; «Универсальный семантический код: УСК-3» [Мартынов, 1984], «Универсальный семантический код: УСК-4» [Мартынов, 1988].

Присущие В.В.Мартынову дух новаторства, широкая, поистине энциклопедическая эрудиция, умение работать «на стыке наук», в частности, лингвистики и семиологии, праксеологии и информатики, кибернетики и искусственного интеллекта, привели к появлению «исчисления языковых смыслов» и развитию семиологического направления в ИИ. Здесь следует отметить две его основополагающие статьи: «Семиологические

проблемы искусственного интеллекта» [Мартынов, 1978] и «Построение и реализация алгоритма планирования, основанного на УСК» [Гуминский, 1985], публикации: "Человеко-машинные методы решения задач в системе УСК» [Мартынов, 1979] и «Об использовании аппарата УСК-3 при формировании алгоритма решения интеллектуальных задач» [Мартынов, 1983], лекцию «Семантическое кодирование для представления и преобразования знаний» [Мартынов, 1990].

В 1993г. за научные открытия В.В.Мартынов был объявлен Международным биографическим центром в Кембридже «Человеком века» и удостоен очень престижной «Награды XX столетия за достижения», которая присуждается раз в столетие.

К сожалению, после распада СССР ранее проводившиеся в рамках САИИ совместные научные работы между российскими специалистами по ИИ и белорусскими учеными – представителями школы В.В.Мартынова – были приостановлены. Сегодня молодое поколение российских ученых мало знает об идеях и работах В.В.Мартынова, хотя уже в XXI-м веке им были написаны и опубликованы фундаментальные монографии «Основы семантического кодирования. Опыт представления и преобразования знаний» [Мартынов, 2001] и «В центре сознания человека» [Мартынов, 2009]. Последняя книга Виктора Владимировича, подводящая итоги его научной деятельности, содержит также новые идеи по развитию и будущей интеграции информационных технологий, когнитивных и лингвистических наук. По сути, в ней впервые рассмотрены основы когнитивной лингвистики, когнитивной семиотики, когнитивной информатики, а также показаны неразрывные связи между этими научными направлениями (хотя сам автор и не использует эти термины).

В настоящей обзорной статье автор стремится после рассмотрения основных вех жизненного и научного пути Виктора Владимировича Мартынова описать становление и развитие теоретических идей и приложений универсального семантического кода, а также раскрыть роль В.В. Мартынова в создании современного симбиоза лингвистических, когнитивных, математических наук.

1. Этапы жизненного и научного пути В.В.Мартынова

Виктор Владимирович Мартынов родился 25 января 1924г. Участник Великой отечественной войны: ушел на фронт в возрасте 18 лет. В составе 3-го Украинского фронта участвовал в боях за освобождение Молдавии, Югославии, Австрии, Румынии, Болгарии и Венгрии. Награжден орденом Отечественной войны 2-й степени.

В 1948-м году В.В.Мартынов окончил Одесский университет, а в 1951-м году – аспирантуру по славистике при Львовском университете. В период

с 1952-го по 1960-й год был заведующим кафедрой иностранных языков Одесского университета.

В 1960-м году переехал в Минск и стал работать в Институте языкознания Академии наук БССР, где в течение почти 30 лет с 1962-го по 1990-й год руководил отделом общего и славянского языкознания. С 1992-го года был профессором кафедры общего языкознания Минского лингвистического университета и руководителем научно-исследовательского центра «Семантика».

В 1969-м году Виктор Владимирович защитил докторскую диссертацию в области лингвистики, а в 1971-м стал профессором.

Интерес к лингвистическим и семиотическим проблемам кибернетики и информатики возник у В.В.Мартынова еще в 1960-е годы. В своей первой монографии по этой междисциплинарной тематике «Кибернетика, семиотика, лингвистика» [Мартынов, 1966] он рассмотрел основы и перспективы дедуктивной семиологии, а его следующей книге, вышедшей в свет 40 лет назад [Мартынов, 1974], уже были затронуты семиологические проблемы информатики и содержалось описание прототипа УСК. В книге «Универсальный семантический код. Грамматика, Словарь, Тексты» [Мартынов, 1977] была представлена первая версия языка УСК, который ученый развивал на протяжении всей жизни (всего было создано 6 версий УСК).

Близкое знакомство В.В.Мартынова с Г.С.Поспеловым, Д.А.Поспеловым, В.Ф.Хорошевым и другими ведущими учеными в области ИИ состоялось в 1970-е годы на очень популярных тогда больших междисциплинарных научных конференциях, таких как Всесоюзные симпозиумы по кибернетике в Грузии [Гааза-Рапопорт, 1991] и Международные симпозиумы по ИИ (см. [Мартынов, 1983]). Так образовались важные научные связи, особенно укрепившиеся после Учредительного съезда САИИ в Коломне и 2-й конференции по искусственному интеллекту в Минске в 1990г. Тогда они казались незыблемыми.

2. В.В.Мартынов: от семиологического подхода к междисциплинарной методологии в искусственном интеллекте

2.1 Семиологический подход в ИИ и лингвистические модели знаний

В своих работах [Мартынов, 1974 и 1978] Виктор Владимирович выделяет два основных пути развития искусственного интеллекта: бионический и семиологический. Чрезвычайно популярный в настоящее время бионический подход опирается на идею практического применения в искусственных системах тех биологических механизмов, которые природа «отработала» в ходе эволюции живых организмов. Здесь речь идет о механизмах адаптации и коллективного поведения, методах

эволюционного моделирования и бионических алгоритмах, подходах роевого и эмергентного интеллекта.

Семиология (термин Ф.де Соссюра [Соссюр де, 2004]) или *семиотика* (по Дж.Локку и Ч.Пирсу [Пирс, 1999] есть наука о знаковых системах. Родоначальники теории знаковых систем исходили из разных предпосылок: так де Соссюр рассматривал ее как науку, изучающую знаковые системы в контексте социальной жизни [Соссюр де, 2004], а Пирс определял ее как логику в самом общем смысле, формальную доктрину знаковых систем [Пирс, 1999]. Однако все они были едины в том, что единицей любой семиотической системы является знак. Любой знак имеет три аспекта: синтаксис, семантику и прагматику или в других терминах: план выражения, план содержания и план значения. Таким образом, любой язык как семиотическая система функционирует и эволюционирует в трехмерном пространстве (синтактика, семантика, прагматика).

«Отцом-основателем» семиотического подхода в ИИ по праву считается Д.А.Поспелов [Поспелов, 1970, 1976 и 1999]. Так еще в работе [Поспелов, 1970] им сформулировано следующее важное утверждение: условием создания технических устройств, способных к решению творческих задач, является формирование внутри такого устройства системы (иерархии) знаковых систем. Как известно, даже отдельные простейшие знаки не находятся в точном соответствии с реальными предметами. Однако структура знаковой системы подобна структуре отношений между предметами реального мира. Такую систему Д.А.Поспелов рассматривает как систему знаков первого уровня.

Но, кроме знаков, значениями которых выступают предметы или явления реального (или моделируемого) мира, можно рассматривать знаки знаков (метазнаки), значениями которых служат знаки семиотической системы первого уровня. Такую систему следует отнести к знаковой системе второго уровня. Путем индукции нетрудно ввести системы знаков любого *k*-го уровня.

Иерархия знаковых систем имеется у человека (например, система знаков естественного языка служит для знаковой системы математики системой предшествующего уровня).

Согласно Ю.М.Лотману, основными вопросами для любой семиотической системы являются, во-первых, отношение к миру, лежащему за ее пределами, а, во-вторых, отношение статики к динамике [Лотман, 2010]. Развитие этих идей в русле искусственного интеллекта привело Д.А.Поспелова к созданию логико-лингвистических моделей представления знаний, введению так называемых псевдофизических логик для систем ИИ и определению семиотической системы как расширения формальной системы с помощью правил изменения ее компонентов (см. [Поспелов, 1986; Тарасов, 2006]).

В свою очередь, профессор В.В.Мартынов предложил свой вариант семиологического подхода в ИИ, основанный на понимании семиотической системы как линейно упорядоченной совокупности знаков и модели «универсального семантического кода» (УСК) для представления знаний в искусственных интеллектуальных системах на ограниченном естественном языке. По его мнению, произошедший в 1970-1980-е годы перенос центра тяжести исследований в ИИ с эвристик на представление знаний есть не что иное, как переход к принципам структурной семантики.

Независимо от исследуемого объекта все рассуждения о нем неизбежно строятся на языке. Язык – это система знаков, служащая средством человеческого общения, мышления и выражения. Язык возникает тогда, когда система сигналов превращается в систему знаков. С помощью языка осуществляется познание мира, в языке объективизируется самосознание личности.

В [Мартынов, 1978] указано, что центральными проблемами ИИ являются задача разработки конструктивных лингвистических моделей знаний и задача построения языков диалога «человек-компьютер». В этом контексте естественный язык (ЕЯ) представляет собой мощнейший инструмент моделирования, наиболее удобное и совершенное средство коммуникации, универсальный аппарат выражения знаний о мире. В то же время ЕЯ в полном объеме не может быть использован как язык дружественного интерфейса «человек-компьютер». Это обусловлено рядом серьезных причин.

Естественный язык обладает асимметрией. С ней связана принципиальная неполнота парадигмы слова, выражающаяся в полисемии и синонимии. В результате одно и то же содержание предстает в разных формах. Особенно большие трудности возникают в связи с постоянным развитием языка и внутренним взаимовлиянием его компонентов.

Как указывает В.В.Мартынов в [Мартынов, 2001], все признают, что между синтаксическими структурами и семантикой нет одно-однозначных отношений. Однако далеко не все понимают, что между ними нет и одно-многозначных, и даже много-многозначных (полиморфных) отношений. Для ЕЯ характерна не многозначность, а скорее неопределеннозначность. Неопределеннозначность выражений ЕЯ часто связана с его эллиптичностью, т.е. с пропуском важных уточняющих элементов высказывания. Нередко о существовании эллипсиса можно догадаться исключительно из контекста.

Таким образом, в ИИ нам необходим стандартизированный в соответствии с некоторым ограниченным числом правил вариант сужения ЕЯ. В стандартизированном или канонизированном естественном языке должна быть эксплицирована глубинная семантика, непосредственно связанная с мышлением и позволяющая давать интерпретацию предложения независимо от особенностей конкретного ЕЯ. Согласно В.В.Мартынову, именно

отсутствие такой экспликации лишило грамматику Монтегю возможности стать эффективным средством ИИ.

Канонизация ЕЯ может проводиться на базе двух противоположных стратегий: а) стратегии, ориентированной на конкретную предметную область; б) стратегии, инвариантной по отношению к предметной области. Для В.В.Мартынова больший интерес представляет второй случай, когда для синтеза любой «семантической микросхемы» мы должны определить систему базовых единиц – примитивов.

2.2 Некоторые методологические аспекты работ В.В.Мартынова

При развитии семиологического подхода к ИИ с целью создания языка представления знаний и диалога типа «Универсальный семантический код» (УСК) профессор В.В.Мартынов разработал свой вариант открытой, системной, междисциплинарной методологии, которая схематически изображена на рисунке 1. Опишем подробнее ее составляющие. Для этого приведем обоснование необходимости построения УСК как многофункционального языка. Профессор В.В. Мартынов начинает с определения проблемы в целом и общих требований к УСК [Мартынов, 1974, 1977, 2001].



Рисунок 1 – К методологии научного исследования В.В. Мартынова

Нам нужен реальный язык общения, диалога человека с компьютером. Этот язык следует строить на основе наиболее общих закономерностей ЕЯ. Необходимо предварительное выделение таких лингвистических универсалий. Как это сделать? Здесь профессор В.В.Мартынов ссылается на такие малоизвестные для большинства лингвистов авторитеты как Д.Гильберт и А.Тарский.

Еще в самом начале XX-го века Д.Гильберт раскрыл понятие «метаязыка» и построил новую аксиоматику для геометрии Эвклида, а в 1930-е годы А.Тарский высказал предложение о

целесообразности аксиоматизации языка, построении аксиом ЕЯ на основе множества высказываний с последующим определением основных правил, с помощью которых из аксиом будут получены теоремы.

Следуя этим идеям, В.В.Мартынов показывает необходимость для создания УСК привлечь в лингвистику аксиоматический(дедуктивный) метод, т.е. такого способ построения научной теории, при котором в ее основу кладутся некоторые исходные предложения, называемые аксиомами, а все остальные предложения получаются как следствия из этих аксиом. Иными словами, надо заменить простое представление примитивов их исчислением с разграничением языковых и метаязыковых элементов.

В общем случае, исчисление – это формальный аппарат оперирования со знаками и их сочетаниями определенного вида. Исчисление как дедуктивная система предполагает способ задания множества путем указания исходных элементов, аксиом исчисления и правил вывода.

Разрабатываемый язык должен представлять собой некоторую обозримую семантическую сеть и удовлетворять важному условию минимальной семантической удаленности соседних вершин. Сама семантическая сеть должна быть представима алгебраическими средствами.

В [Мартынов, 2001] ставится задача построения алгоритма (модели), позволяющего заменить представление примитивов языка их исчислением. Она решается автором с помощью абстрактных алгебр Линденбаума и Лукасевича путем построения классификатора примитивов (т.е. по сути, онтологии верхнего уровня) в виде ориентированного графа и разработки изоморфного ему векторного представления. Векторные модели вместе с моделями на основе графов позволяют получить систему примитивов в виде вложенных баз знаний, причем множество примитивов не постулируется, а рекурсивно исчисляется.

Итак, в методологии В.В.Мартынова во-первых речь идет об интеграции методов и моделей лингвистики (структурной и комбинаторной семантики), семиотики, психологии (психологии памяти и психологии мышления), информатики «внутри квадрата» на рисунке 1. Так в информатике под кодом понимается универсальный способ отображения информации, задаваемый через соответствие между элементами сообщений и сигналами, с помощью которых они фиксируются. Код часто определяется как совокупность правил или ограничений, обеспечивающих требуемое функционирование некоторой знаковой системы. Семантический код – это языковой знак, который представляет собой производную единицу, состоящую из более простых единиц – семантических множителей. При семантическом кодировании информации придается определенный смысл.

Очень интересным представляется также сопоставление генетического и языкового кода [Мартынов, 2001 и 2009]. Среди всех систем передачи информации только генетический код и языковой код базируются на использовании таких дискретных компонентов, которые сами по себе не имеют смысла, но служат для построения минимальных единиц, имеющих смысл.

Несомненным достоинством подхода В.В.Мартынова является опора на экспериментальные модели психологии памяти, в частности, на результаты И.Хофмана, согласно которому информация в кратковременной памяти может храниться только в форме внутреннего семантического кода [Хофман, 1986]. Отсюда вытекают следующие положения: 1) в основе человеческого мышления лежит наш внутренний семантический код, который конструирует общую картину мира; 2) этот код принадлежит подсознанию и должен передаваться по наследству, т.е. быть включенным в семантический код; 3) в плане построения УСК требуется создание моделей внутреннего семантического кода.

Во-вторых, при разработке моделей УСК Виктор Владимирович изучил ряд разделов современной математики (абстрактную алгебру, математическую логику, аксиоматическую геометрию, теорию графов) и успешно применил их методы и подходы.

А в третьих, профессору В.В.Мартынову удалось вскрыть глубокие внутренние связи между лингвистикой и абстрактной алгеброй, семиотикой и геометрией. Справедливости ради, надо отметить, что еще Ч.С.Пирс писал заметки о возможности представления языка в алгебраической записи или геометрическом изображении, а В.В.Налимов рассмотрел функции математики как языка природы и науки. При этом структура «чистой» математики уподобляется им грамматике языка прикладной математики: если математика в прикладных задачах играет роль языка, то математические структуры этого языка естественно рассматривать как его грамматику [Налимов, 1979].

С иерархией языков связано понятие метаязыка. Так язык математики, используемый для описания прикладных задач, выступает в роли метаязыка по отношению к языку, на котором ранее формулировались и обсуждались эти задачи. Например, язык математической статистики стал метаязыком по отношению к теории измерений. Аналогично В.В.Мартынов считает, что лингвистическая семантика и логическая семантика находятся на разных уровнях исследования: первая относится к языку, а вторая – к метаязыку.

Связь между лингвистикой и математикой он раскрывает в [Мартынов, 2001], показывая, что лингвистика и «чистая» математика – единственные две науки, в которых объект и инструмент исследования фактически совпадают. В области лингвистики в последнее время бурно развивается

типология универсалий. Речь идет о так называемых абсолютных универсалиях, т.е. закономерностях, характерных для всех ЕЯ и описываемых в виде «во всех языках есть А». Наличие таких универсалий в некоторой (например, семиотической) системе означает, что она способна выполнять роль, подобную роли ЕЯ.

Аналогичное явление мы наблюдаем в области алгебры которая стала универсальной алгеброй, определяемой как множество элементов (объектов), на котором задано множество операций любой природы.

Построив алгебру семантического кода, профессор В.В.Мартынов наглядно раскрыл связи между лингвистическими универсалиями и абстрактными алгебрами.

Следуя Н.А.Васильеву и Я.Лукаевичу, широко использовавшим геометрические аналогии для построения неаристотелевых логик, и развивая тезис Л.Витгенштейна о том, что область наших первичных пространственных представлений лежит в основе логических построений языка, В.В. Мартынов ввел геометрические представления в семиотику. Уже в своей ранней книге [Мартынов, 1966] он утверждал, что семиотика должна строиться как дедуктивная наука и как своеобразная геометрия знаковых систем. Объясняя эту мысль, он сравнивал первую и вторую сигнальные системы

Сигнальная система животных ситуационно связана: получаемый сигнал и соответствующее действие животного практически совпадают во времени. Человек же оперирует тремя обобщенными временами: прошлым, настоящим и будущим. Он может анализировать прошлое, наблюдать настоящее и планировать будущее, что нашло отражение в чрезвычайно богатой знаковой системе, которая ситуационно свободна. Это служит обоснованием положения о том, что геометрия второй сигнальной (знаковой) системы есть геометрия времени [Мартынов, 1966], поскольку знаки как семиотические единицы имеют протяженность и направленность во времени.

В качестве базовой структуры примитивов для УСК берется четверка (X, Y, Z, W) , где некто X посредством инструмента Y воздействует на объект Z , в результате чего получается продукт W . Сам В.В.Мартынов говорил о близости этой структуры к идеям *праксеологии* – варианта общей теории систем Т.Котарбинского, который писал: «при выполнении любой работы всегда налицо агент действия, его цель, инструмент действия и некоторый продукт труда [Котарбинский, 1963].

3. Универсальный семантический код: лингвистические основы, принципы построения, структура и тенденции развития

Еще в 1960-е годы в научной среде

специалистов по ИИ возникло понимание того, что без структуризации знаний, без создания удобных и наглядных средств их формального представления и преобразования создать системы искусственного интеллекта невозможно. В работах [Мартынов, 1990 и 2009] предложена общая классификация моделей и средств представления знаний, приведенная в таблице 1.

Таблица 1 – Средства представления знаний

Критерии Классы	Собственные средства представления семантики	Собственные средства преобразования семантики
Информационные носители	—	—
Языки семантического представления	+	—
Логические системы	—	+
Универсальный семантический код	+	+

Первый класс моделей условно назван «информационными носителями». Сюда относятся фреймы и семантические сети. По мнению В.В. Мартынова, их нельзя называть языками, поскольку они не имеют собственных средств представления и преобразования знаний. Высказывания на естественном языке, погруженные в семантическую сеть или фреймовую систему, остаются такими же высказываниями на ЕЯ. Поэтому такие удобные для компьютерной обработки модели не снимают общих трудностей представления и преобразования знаний.

Ко второму классу относятся языки семантического представления. В основе этих языков лежат некоторые семантические примитивы, которые формируют семантическую запись высказывания. Здесь в первую очередь могут быть названы модель концептуальной зависимости Р.Шенка и модель «смысл-текст» И.А.Мельчука. Примитивы данных моделей не претендуют на полноту, независимость и непротиворечивость, так как эти модели разрабатывались эмпирическим путем. К этим же проектам примыкают примитивы Л.Вежицкой, ролевая грамматика, основанная на теории глубинных падежей Ч.Филлмора, «силовая динамика» Л.Талми. В целом, языки семантического представления имеют собственные (хотя и довольно ограниченные) средства представления знаний, но лишены развитых средств их обработки.

Третий класс образуют логические средства, в частности, исчисление предикатов первого порядка, модальные логики. Основной их недостаток заключается в отсутствии собственных средств семантического представления. Понимание этого недостатка привело к созданию псевдофизических логик и других специальных логик для ИИ. Тем не менее, неясно, как на основе псевдофизических логик (их общее количество не ограничено, а соотнесение пока не проведено) строить целостное представление мира, и как обеспечить полноту этого представления.

Четвертый класс формируют языки типа «Универсальный семантический код» (УСК). Сама идея создания языка типа семантического кода была впервые высказана известным французским этнографом К.Леви-Строссом. Подобные языки не зависят от того, какой фрагмент мира описывают, и отличаются полной экспликацией смысла (каждый комбинаторный вариант цепочки элементов имеет один и только один смысл). Еще одним важным требованием является способность формировать новые понятия и строить гипотезы о причинах и следствиях ситуации. Все это реализуется в УСК путем формальных преобразований цепочек символов. Таким образом, языки типа УСК представляют собой дедуктивные системы, семантика которых не задается, а исчисляется. В итоге УСК располагает собственными средствами представления и преобразования семантики.

В качестве минимальной или ядерной лингвистической цепочки В.В.Мартынов берет трехчленную структуру «субъект (*S*) - действие (акция *A*)-объект (*O*), т.е. цепочку *SAO*. Здесь акция *A* есть глагольный узел (или предикатное ядро), выступающий по Теньеру как центр предложения. Этот центральный элемент цепочки характеризует отношение доминирования субъекта над объектом, или иначе выражает любую форму воздействия субъекта на объект; чаще всего, это перемещение объекта, включая удержание, или преобразование объекта, включая его создание или уничтожение.

С одной стороны, выбор позиционной цепочки данного вида в качестве базового смыслового компонента обоснован бинарностью отношения, а с другой стороны, он объясняется лингвистическими универсалиями: трансформацией актива в пассив и конверсного преобразования доминирующего и доминируемого элементов.

В данном контексте сам язык ограничивается тройкой «имя 1 – глагол – имя 2», а все остальное относится к метаязыку, организующему основной текст. К числу таких метаязыковых элементов относятся местоимения и числительные, а в чистом виде, предлоги и союзы. Местоимения выполняют в основном указательные и заместительные функции, числительные – порядковые и количественные. Союзы осуществляют временные преобразования, а предлоги – пространственные.

Практически любой языковой текст может быть построен на основе стандарта организованных цепочек с расширением их посредством актуаторов для центра и модификаторов для крайних членов (маргиналий). При этом модификаторы задаются прилагательными, а актуаторы – наречиями. Поэтому в качестве базовых операций автором УСК рассмотрены такие операции как: конверсия ядерной цепочки, т.е. обмен элементами *S* и *O*, стоящими в маргинальных позициях; варианты композиции (мультипликация) элементов в крайних позициях, например, параллельная S_1/S_2 или последовательная S_1S_2 .

Канонизация ЕЯ у В.В.Мартынова начинается с учета эллипсисов. В УСК неопределеннозначность, вызываемая эллипсисами, может сниматься с помощью презумпций (горизонтальный контекст) и пресуппозиций (вертикальный контекст). Здесь термин «презумпция» означает языковую картину мира, а «пресуппозиция» – общую модель мира, отражающую индивидуальный или коллективный опыт. Так презумпции воздействия и намерения, выражаемые ниже словами в скобках, например, «художник задумал (создать) оригинальную картину» или «отец садится за стол (намереваясь) обедать», позволяют установить утраченные в процессе синтаксической конденсации (сжатия) категории воздействия и потенциального действия. Понимание фраз не ЕЯ обеспечивается так называемой пресуппозицией, т.е. по сути всем предшествующим индивидуальными коллективным опытом. Пресуппозиция когнитивно предшествует высказыванию, так как это – предположение о знаниях собеседников.

Расширение ядерной цепочки в цепочку с двумя субъектами (S_1-S_2-A-O) или цепочку с двумя объектами ($S-A-O_1-O_2$) позволяет реализовать другие стратегии семантической конденсации. В УСК были также введены кванторы и модальные операторы.

В общем случае УСК описывается алгеброй

$$A = \langle M, \rightarrow, \neg \rangle,$$

где *M*- множество, а \rightarrow , \neg – суть бинарная операция импликации и унарная операция отрицания. При этом обе этих операции являются составными, образованными как композиции элементарных импликаций и отрицаний соответственно.

В монографии Д.А.Поспелова [Поспелов, 1986] дан краткий обзор УСК и указаны основные принципы его создания: 1) полная экспликация смысла; 2) независимость от предметной области; 3) недоговорной характер семантики метауровня, жестко задаваемой семантикой простых ядерных конструкций и синтаксисом производной цепочки; 4) Неполнота правил преобразования цепочек друг в друга.

Всего было реализовано шесть версий универсального семантического кода: 1) УСК-1 (1974г.); 2) УСК-2 (1977 г.); 3) УСК-3 (1983 г.); 4) УСК-4 (1988 г.); 5) УСК-5 (1996 г.); 6) УСК-6 (2001 г.). Последний вариант УСК, а именно, УСК-6, уже полностью опирается на УСК-алгебру с ее аксиомами порождения и преобразования.

Согласно В.В.Мартынову, создание на основе внутреннего семантического кода базового компьютерного языка, способного порождать знаковую систему с встроенным в нее универсальным решателем интеллектуальных задач, позволит перейти к конструированию интеллектуальных роботов [Мартынов, 2001]. Мы находимся на пороге эры семантических и семиотических интеллектуальных систем, т.е.

систем в которые смысл вложен в формализмы представления.

4. О вкладе В.В.Мартынова в становление когнитивной лингвистики, семиотики и информатики

В знаменитом отчете *Converging Technologies for Improving Human Performance*, подготовленного в 2002 г. М.Роко и У.Бейнбриджем в Всемирном центре оценки технологий (WTEC), при описании перспектив развития нашей цивилизации был введен термин NBIC-конвергенция, означающий формирование нового технологического уклада в результате схождения нано-, био-, инфо- и когно-технологий (NBIC по первым буквам областей: *N* - нано; *B* -био; *I* -инфо; *C* -когно). Это объясняет все возрастающий интерес к когнитивной науке, обеспечивающей теоретическую базу для широкой реализации пока еще наименее развитого компонента NBIC.

Сегодня когнитивная наука (когнитивистика) есть обширная междисциплинарная область, занимающаяся исследованием процессов познания и когнитивных систем. Она стремится ответить на вопросы, что такое познание и как оно работает, каковы общие принципы организации и механизмы функционирования мозга. Когнитивистика как синергетическое научное направление объединяет ряд гибридных микронаук, в том числе теорию познания, когнитивную психологию, когнитивную лингвистику и семантику, нейрофизиологию (и, вообще комплекс нейронаук), искусственный интеллект. На наш взгляд, вклад профессора В.В.Мартынова в современную когнитивную науку (см.рисунок 1) заслуживает отдельного обсуждения.

Само название последней книги выдающегося белорусского лингвиста «В центре сознания человека» [Мартынов, 2009] показывает, что автор стоит на позициях *когнитивной лингвистики*, т.е. рассматривает язык и как единый механизм познания, и как проявление общей способности человека познавать мир, других людей и самого себя. В целом, лейтмотивом всей книги является неразрывная связь между знанием и языком.

В когнитивной лингвистике отбрасывается традиционное разделение лингвистики на фонологию, синтаксис, морфологию и пр. [Маслова, 2004]. Здесь в центре внимания оказывается *когнитивная семантика*, связанная с конструированием смысла и представлением знаний. В результате когнитивной деятельности создается система смыслов, определяющая, что субъект знает и думает о мире («образ мира» у А.Н.Леонтьева). Часто используемая лингвистами категория пресуппозиции является показательным примером внеязыковой категории, напрямую связанной с моделью мира.

По сути, в когнитивной лингвистике язык понимается как средство доступа к познавательным процессам. При этом изучаются связи отдельных когнитивных процессов и способностей человека с языком и формы их взаимодействия.

Таким образом, одной из главных категорий когнитивной лингвистики является категория «знания»; при этом исследуются как виды знаний, так и способы их языкового представления. Специалисты по когнитивной лингвистике стремятся понять, как протекают процессы восприятия, узнавания, осмысления, классификации, категоризации, абстракции, как происходит формирование и накопление знаний. Одной из главных инструментов познания является языковая коммуникация, связанная с обменом знаниями.

В математической логике имеется независимо существующий синтаксис, также независимо существующие модельные структуры и принципы отображения синтаксиса на модельные структуры. Естественный язык устроен по-другому. Он возник вместе со значением, и когда мы мыслим при помощи ЕЯ, мы мыслим о вещах в терминах, которые имеют значение [Лакофф, 2004]. При этом рассматриваются гештальты и такие базовые схемы как «целое-часть», «центр-периферия», «источник-путь-цель», «вместилище» и др. (см.[Кузнецов, 2012]).

По В.В.Мартынову [Мартынов, 2001 и 2009], язык есть форма существования знания в виде системы знаков. Любые знания приходят к человеку через язык, а язык возникает тогда, когда исходная система сигналов превращается в знаковую (семиотическую) систему. Таким образом, когнитивная лингвистика у него оказывается неразрывно связанной с *когнитивной семиотикой*.

В [Валькман, 2013] указано, что появление когнитивной семиотики предполагает движение когнитивных наук и семиотики встречным курсом. С одной стороны, когнитивная семиотика есть наука о получении знаний из знаковых структур, а с другой стороны, речь идет о синтезе знаковых конструкций, представляющих соответствующие структуры знаний. Типичным способом получения процедурных знаний из знаковой структуры является анализ связей между именем 1, понятием 2, представлением 3 и денотатом 3' в треугольнике Фреге (рисунок 2): 1-2 – описание; 2-1 – обозначение; 1-3 – представление; 3-1 – именование; 3-3' – обобщение; 3'-3 – конкретизация.

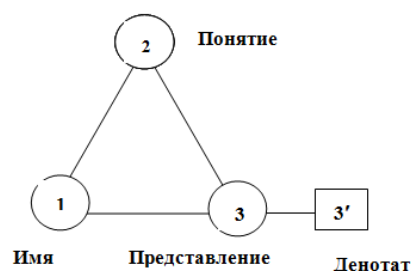


Рисунок 2 – Вариант треугольника Фреге

В русле когнитивной семиотики центральное место занимает определение соответствий между знаковыми структурами и структурами знаний. Здесь замечательным примером является описанное в [Поспелов, 1999] соотнесение знаков и фреймов и введение структуры знака-фрейма (рисунок 2).

Так, например, паре процедур 1 соответствуют поиск информации по адресу и ассоциативный поиск информации по содержанию, а паре процедур 2 отвечают приобретение знаний и построение конкретных представлений на базе понятия, т.е. порождение экзофреймов на основе имеющегося протофрейма.

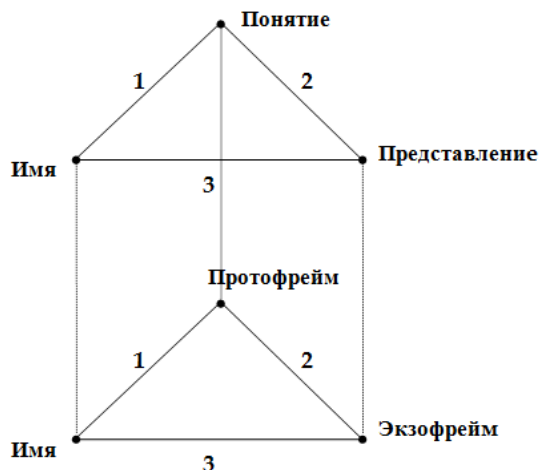


Рисунок 3 – К определению знака-фрейма

На наш взгляд, приведенное выше из статьи [Валькман, 2013] определение когнитивной семиотики следует расширить и не ограничиваться только уровнем знаний. С одной стороны, ее можно интерпретировать как науку о развитии моделей и методов познания и конструирования смысла на основе знаковых структур. С другой стороны, она занимается проблемами семиозиса, формирования семиотических отношений для когнитивных процессов и систем любого уровня (восприятие, представление, мышление). Здесь следует введенное Ч.С.Пирсом понятие семиозиса как постоянного движения знака. Семиозис у него включает две одинаково важные части: производство знаков и их интерпретацию. При этом, все, входящее в процесс семиозиса, становится знаком.

В данном контексте рассмотрим некоторые системы классификации знаков. В [Моррис, 1983] проводится различие между лингвистическими знаками. Выделены: а) знаки-идентификаторы (т.е. те знаки, которые отвечают на вопрос "где?"); б) знаки-десигнаторы (знаки, отвечающие на вопрос "что такое?"); в) оценочные знаки, связанные с предпочтением, которые отвечают на вопрос "почему?"; г) прескриптивные знаки (отвечающие на вопрос "как?"); д) формирующие, или знаки систематизации (направляющие поведение интерпретатора в отношении других знаков).

Следуя Ч.С.Пирсу [Пирс, 1999], профессор В.В.Мартынов в монографии [Мартынов, 2009]

приводит ряд других интересных классификаций знаков, представляющих интерес для когнитивной семиотики. В частности, им рассмотрен переход от симптома к знаку в плане возрастания семантического наполнения (таблица 2).

Таблица 2. Переход от симптома к знаку

	Информативность	Коммуникативность	Номинативность
Симптом	+	–	–
Сигнал	+	+	–
Знак	+	+	+
X - субъект	Y-инструмент	Z - объект	W-результат

Наконец, когнитивная информатика есть междисциплинарная область на стыке психологии, лингвистики, ИИ и нейронаук, изучающая механизмы и процессы переработки информации человеком и варианты их инженерной реализации на компьютерах. Первоначально, это направление ограничивалось разработкой методов и систем обработки информации с целью получения знаний и их использования при решении задач проблемной области. В его рамках сформировался подход к формированию интеллектуальных систем с автоматическим приобретением знаний путем обучения и самообучения [Станкевич, 2006]. Здесь пионерской разработкой стала система Eurisco Д.Лената (вариант развития методологии ее создания в контексте УСК хорошо описан в книгах В.В.Мартынова [Мартынов, 2001 и 2009]).

Современное представление когнитивной информатики (по В.В.Мартынову) определяет ее главную проблему как поддержку и усиление человеческих способностей познания, понимания, обучения с помощью информационных технологий, нового поколения (с программно-аппаратной реализацией искусственных когнитивных структур) обеспечивающих дружелюбный интерфейс для сотрудничества человека и компьютера. В этом плане создание компьютерных чипов с реализацией на одном кристалле цепочек УСК представляется прорывной задачей стратегической NBIC-конвергенции технологий будущего.

Заключение

В заключение, следует отметить, что работы В.В.Мартынова являются замечательным примером междисциплинарных исследований на стыке гуманитарных и математических наук. Открытые им связи между лингвистикой и абстрактной алгеброй, семиотикой и геометрией, орграфами и психологическими механизмами познания позволили разработать основы новой методологии создания информационных технологий и построения интеллектуальных систем следующего поколения. Сформулированная В.В.Мартыновым научная программа перехода от внутреннего семантического кодирования в кратковременной памяти к универсальному семантическому коду с последующим созданием УСК-компьютеров, снабженных семантическими платами, и есть одна

из «первых ласточек» в плане формирования когнотехнологий будущего, развития новой гибридной системы «наук-перекрестков», предполагающей симбиоз когнитивной лингвистики, когнитивной психологии, когнитивной семиотики в русле развития когнитивной информатики.

Библиографический список

- [Бойко, 1991] Бойко И.М. Семантическое кодирование и решение интеллектуальных задач/ И.М.Бойко, А.П. Гуминский, В.В.Мартынов // Журнал ТРИЗ, 1991, т.2, №1, с.43-47.
- [Валькман, 2013] Валькман, Ю.Р. Когнитивная семиотика: истоки и перспективы/ Ю.В.Валькман//Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов VII-й международной научно-практической конференции (ИММВ-2013, Коломна, 20-22 мая 2013 г.). – М.: Физматлит, 2013. – Т.1. – С.48-61.
- [Гааза-Рапопорт, 1991] Гааза-Рапопорт М.Г. Всесоюзные симпозиумы по кибернетике в Грузии/ М.Г. Гааза-Рапопорт// Новости искусственного интеллекта, 1991, №4, с.82-120.
- [Гуминский, 1985] Гуминский А.П. ПОСТРОЕНИЕ И РЕАЛИЗАЦИЯ АЛГОРИТМА ПЛАНИРОВАНИЯ, ОСНОВАННОГО НА УСК / А.П. Гуминский, В.В.МАРТЫНОВ/ ИЗВЕСТИЯ АН СССР: ТЕХНИЧЕСКАЯ КИБЕРНЕТИКА, 1985, №5.
- [КОТАРБИНСКИЙ, 1963] КОТАРБИНСКИЙ, Т. ЭЛЕМЕНТЫ ТЕОРИИ ПОЗНАНИЯ, ФОРМАЛЬНОЙ ЛОГИКИ И МЕТОДОЛОГИИ НАУК. – М.: Изд-во иностранной литературы, 1963.
- [Кузнецов, 2012] Кузнецов О.П. Когнитивная семантика и искусственный интеллект/ О.П.Кузнецов// Искусственный интеллект и принятие решений, 2012, №4, с.32-42.
- [Лакофф, 2004] Лакофф Дж. Женщины, огонь и опасные вещи: Что категории языка говорят нам о мышлении: Пер. с англ. – М.: Языки славянской культуры, 2004
- [Лотман, 2010] Лотман Ю.М. Семиосфера.–СПб.: Искусство-СПб, 2010.
- [Мартынов, 1966] Мартынов, В.В. Кибернетика, семиотика, лингвистика. – Мн.: Наука и техника, 1966.
- [Мартынов, 1974] Мартынов, В.В. Семиологические основы информатики. – Мн.: Наука и техника, 1974.
- [Мартынов, 1977] Мартынов, В.В. Универсальный семантический код. Грамматика, Словарь, Тексты. – Мн.: Наука и техника, 1977.
- [Мартынов, 1978] Мартынов, В.В. Семиологические проблемы искусственного интеллекта/ В.В.Мартынов// Известия АН СССР: Серия литературы и языка, 1978, том 37, №1, с.3-9.
- [Мартынов, 1979] Мартынов, В.В. Человеко-машинные методы решения задач в системе УСК/ В.В.Мартынов // Вопросы кибернетики, 1979, № 60.
- [Мартынов, 1983] Мартынов, В.В. УСК-3: новый вариант/ В.В.Мартынов// Материалы международного симпозиума по ИИ (Одесса, 1983).
- [Мартынов, 1984] Мартынов, В.В. Универсальный семантический код: УСК- 3. – Мн.: Наука и техника, 1984.
- [Мартынов, 1990] Мартынов, В.В. Семантическое кодирование для представления и преобразования знаний/ В.В.Мартынов// Лекции Всесоюзной школы по основным проблемам искусственного интеллекта и интеллектуальным системам. Ч.1 (29 октября-1 ноября 1990г.). – Мн.: САИИ, 1990. – С.46-63.
- [Мартынов, 1992] Мартынов, В.В. Семантические особенности представления и преобразования знаний в УСК-5/ В.В.Мартынов// Сборник научных трудов III-й конференции по искусственному интеллекту: в 2-х томах. Т.1. – Тверь: АИИ, 1992. – С.35-37.
- [Мартынов, 2001] Мартынов, В.В. Основы семантического кодирования. Опыт представления и преобразования знаний. – Мн.: Европейский гуманитарный университет, 2001.
- [Мартынов, 2009] Мартынов, В.В. В центре сознания человека. – Мн.: Изд-во БГУ, 2009.
- [Маслова, 2004] Маслова, В.А. Когнитивная лингвистика: учебное пособие. – Мн.: ТетраСистемс, 2004.
- [Моррис, 1983] Моррис, Ч.У. Основания теории знаков/ Ч.У.Моррис//Семиотика: Антология.–М.: Академический проект, 2001. – С.45-97.

[Налимов, 1979] Налимов, В.В. Вероятностная модель языка. – М.: Наука, 1979.

[Пирс, 1999] Пирс, Ч.С. Логика как семиотика: теория знаков /Ч.С.Пирс// Метафизические исследования. Вып.11. Язык. – СПб: Алетейя, 1999. – С.199-217.

[Поспелов, 1970] Поспелов, Д.А. Системный подход к моделированию мыслительной деятельности/ Д.А.Поспелов// Проблемы методологии системного исследования. – М.: Мысль, 1970. – С.333-358.

[Поспелов, 1976] Поспелов, Д.А. Семиотические модели: успехи и перспективы/ Д.А.Поспелов// Кибернетика, 1976, №6, с.114-123.

[Поспелов, 1986] Поспелов, Д.А. Ситуационное управление: теория и практика. – М.: Наука, 1986.

[Поспелов, 1999] Поспелов, Д.А. Прикладная семиотика/ Д.А.Поспелов, Г.С.Осипов// Новости искусственного интеллекта, 1999, №1, с.9-35.

[Соссюр, 2004] Соссюр де, Ф. Курс общей лингвистики. – М.: Едиториал УРСС, 2004.

[Станкевич, 2006] Станкевич, Л.А. Когнитивный подход к управлению гуманоидными роботами/ Л.А.Станкевич// От моделей поведения к искусственному интеллекту/ Под ред. В.Г.Редько. – М.: КомКнига, 2006. – С.386-443.

[Тарасов, 1997] Тарасов, В.Б. Эволюционная семиотика – новое синергетическое направление в искусственном интеллекте/ В.Б.Тарасов // Искусственный интеллект (Киев, Украина), 1997, №1-2, с.9-20.

[Тарасов, 2006] Тарасов, В.Б. Логико-лингвистические модели в искусственном интеллекте/ В.Б.Тарасов// Политехнические чтения. Сборник трудов. Вып.7. Искусственный интеллект – проблемы и перспективы. – М.: РАИИ, 2006. – С.48-68.

[Хофман, 1986] Хофман, И. Активная память: Пер. с нем. – М.: Прогресс, 1986.

FROM SEMANTIC CODE TO COGNITIVE LINGUISTICS, SEMIOTICS AND INFORMATICS: ON MARTYNOV'S SCIENTIFIC HERITAGE

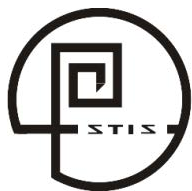
Tarassov V. B.

Bauman Moscow State Technical University

Vbulbov@yahoo.com

The paper is aimed to analyze main components of Martynov's scientific heritage. Victor V. Martynov is an outstanding Soviet and Byelorussian linguist, pioneer of semiologic approaches in Artificial Intelligence (AI), initiator of constructing formal linguistic models by using abstract algebras, author of a well-known knowledge representation language called «Universal Semantic Code» (USC). It is obvious that he can be viewed as a father of new pluridisciplinary sciences appeared at the crossroads of some classical areas such as cognitive linguistics, cognitive semiotics, cognitive informatics.

First of all, the author tries to describe some milestones of Martynov's life. Then he considers basic ideas of his semiologic approach to AI, related to linguistic knowledge models and friendly dialogic «man-computer» interface, specifies the fundamentals of Martynov's multidisciplinary AI methodology. Linguistic bases, multidisciplinary principles and recursive structure of USC are reviewed. A special emphasis is made on eliciting the role of professor Martynov in generating a clump of new cognitive sciences to enable cognitive informatics.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ТЕОРИЯ АВТОМАТИЧЕСКОГО ПОРОЖДЕНИЯ АРХИТЕКТУРЫ ЗНАНИЙ (ТАПАЗ-2) И ДАЛЬНЕЙШАЯ МИНИМИЗАЦИЯ СЕМАНТИЧЕСКИХ ИСЧИСЛЕНИЙ

Гордей А.Н.

*Белорусский государственный университет,
г. Минск, Республика Беларусь
alieks2001@yahoo.com*

При помощи геометрического метода объясняется приоритет модели мира над языковой картиной мира, излагается новая версия теории автоматического порождения архитектуры знаний (ТАПАЗ-2), предлагается алгебраический аппарат для исчисления семантики предметных областей и процедурального представления и преобразования знаний в искусственных интеллектуальных системах, приводятся примеры формального описания русской глагольной семантики в зависимости от суперпозиции процессов познания.

Ключевые слова: модель мира, семантический код, стереотип, геометрическая модель, семантический примитив, семантический классификатор, формализованная теория, набор аксиом.

Существуют такие понятия, как "виновник", "орудие", "продукт труда" <...> Мы находимся здесь в сфере различных категорий, по видимости онтологических, а по существу семантических.

Т. Котарбинский [Котарбинский, 1975]

Введение

На рубеже XX – XXI веков проблема построения больших баз знаний на основе универсальных семантических классификаторов, несмотря на все попытки ее обойти или свести к недостаткам аппаратного обеспечения, в исследованиях по компьютерному моделированию интеллектуальной деятельности стала центральной. Д. Ленат, автор программы EURISCO'83, по этому поводу выразился определенно: «Рецепт повышения способности программы к рассуждению по аналогии так же, как и общего прогресса в программах искусственного интеллекта, заключается в том, чтобы расширить базу знаний <...> записать целую энциклопедию в форме, доступной для машины, но не в виде текста, а в виде системы структурированных, многократно проиндексированных фрагментов» [Ленат, 1986]. Понятная и решаемая при соответствующем уровне развития вычислительной техники проблема оказалась связанной с другой давней и трудной, с которой до программистов уже столкнулись математики, логики, философы и лингвисты, и на которую неоднократно обращали внимание многие специалисты на заре становления и развития искусственного интеллекта как отдельной отрасли научного знания, а именно: проблемой вложения семантики в формализмы представления. «Чего

только не делалось, чтобы не принимать во внимание значение, избежать его и отделаться от него. Напрасные попытки – оно, как голова Медузы, всегда в центре языка, околдовывая тех, кто его созерцает» [Бенвенист, 1974]. Отметим лишь несколько драматических эпизодов. «Заслуга семантического анализа, – писал Г.-Г. Гадамер, – мне видится в фиксации всеобщих структур языка, повлекшей за собой отказ от ложного идеала однозначности знака (соответственно символа) и от возможности логической формализации языковых выражений» [Гадамер, 1991]. Однако в конце 60-х годов XX века С. Амарел пришел к выводу о том, что эффективность решения задачи зависит от формы ее представления [Amarel, 1968], т.е. в неявном виде высказал мысль о том, что правильное представление задачи есть, по сути дела, ее решение. Вывод хорошо согласовывался с общенаучным пониманием доказательства как сведения сложных высказываний (теорем) к простым (аксиомам)¹, т.е. переводу скрытых тавтологий в явные [Бурбаки, 1963], и имел резонанс. В частности, М. Минский идею замены эвристики алгоритмическим представлением знаний воспринял «как уход от традиционных подходов и физиологов-бихевиористов и ориентирующихся на математическую логику ученых в области

¹ «Доказательство *more geometrico* (по геометрическому методу) считалось философами всех веков высшим достижением науки» [Рейхенбах (2), 1985].

искусственного интеллекта, которые пытаются представить знания в виде совокупности отдельных простых фрагментов» [Минский, 1978]. Тем не менее, ученик М. Минского Т. Виноград, несмотря на все отчаянные усилия продвинуться в данном направлении, вынужден был констатировать: «Никто еще не оперирует системами, которые бы не сводились к изолированным примерам» [Winograd, 1980], а «существующие системы, даже лучшие из них, часто напоминают карточный домик <...> В качестве результата возникает крайне хрупкая структура, коллапсирующая при малейшей попытке поколебать область и даже отдельные примеры из области, для которой она построена» [Bobrow, et al., 1977]. Вхождение в парадокс Рассела, неизбежно возникающее из-за совпадения объекта и инструмента исследования¹ при отсутствии формализованной семантической метатеории², явно ощущается в следующем высказывании Т. Винограда: «Язык есть процесс коммуникации между людьми, и он сложнейшим образом переплетен со знаниями о мире, которыми эти люди располагают. Знания эти не являются простой сводкой определений и аксиом – полных, четких и последовательных. Скорее, они представляют собой совокупность понятий, выработанных для того, чтобы манипулировать мыслями. На самом деле они несовершенны, в сильной степени избыточны и часто непоследовательны. Не существует такого замкнутого множества «примитивов», исходя из которого можно было бы определить все, что угодно. Определения «циркулярны», циклически связаны между собой, значения каждого из них зависят от других понятий» [Виноград (1), 1976]. Т. Виноград осознавал, что в практическом плане от семантики требуется «преобразователь, который может работать с синтаксическим анализатором и выдавать данные, приемлемые для логической дедуктивной системы. Если имеются синтаксический анализатор для грамматики английского языка и дедуктивная система на базе знаний о конкретном предмете, роль семантики сводится к тому, чтобы заполнить пространство между ними» [Виноград (2), 1976]. Однако как добиться этого с помощью математики, где также совпадают объект и инструмент исследования, никто не знал. Логический парадокс, обнаруженный Б. Расселом в 1902 году в основаниях арифметики Г. Фреге – первой попытке теоретико-множественных построений³, чуть не довел автора логики предикатов и основателя логической семантики [Фреге, 2000] до самоубийства. Злополучный парадокс Г. Фреге безуспешно

пытался разрешить до конца жизни, и на протяжении всего XX века математика через формализацию метаязыка с трудом избавлялась от противоречий в ее основаниях. Ни исчисления предикатов [Фреге, 1997], ни булевы алгебры [Сикорский, 1969], ни псевдофизические и модальные логики [Кандрашина и др., 1989], ни теория множеств в версии Г. Кантора [Кантор, 1985] не смогли формализовать языковую семантику: математика по-прежнему не располагала собственными средствами преобразования выражений, а логика – собственными средствами их представления⁴. К чести белорусской науки, прогресс в данной сфере во многом обязан работам В.В. Мартынова по семантическому кодированию [Мартынов, 1966]. Создатель «Универсального семантического кода (УСК)» так охарактеризовал свое творение: «Это система, способная формировать новые понятия и строить гипотезы о причинах и следствиях ситуаций. И то и другое реализуется в системе в результате формальных преобразований цепочек. Таким образом, языки типа УСК представляют собой дедуктивные системы, семантика которых не задается, а исчисляется. В итоге УСК располагает собственными средствами представления и преобразования семантики» [Мартынов (1), 2009]. Первая версия УСК появилась в 1974 году [Мартынов, 1974], в 1977 – вторая [Мартынов, 1977] в 1984 – третья [Мартынов, 1984], в 1988 – четвертая [Мартынов, 1988], в 1995 – пятая [Martynov, 1995], в 2001 – шестая [Мартынов (3), 2001]. В версиях совершенствовался алгебраический аппарат, предлагался уточненный список «несуществующих», по мнению Т. Винограда, исходных семантических примитивов и, тем самым, сужался перечень задач по оснащению компьютера энциклопедическими базами знаний, который в окончательном виде насчитывал пять пунктов:

«1. Исчислить примитивы, т.е. семантически неразложимые ключевые слова и правила их комбинаторики.

2. Установить необходимый и достаточный набор формальных характеристик, составляющих «словарную статью».

3. Определить набор семантических операций, позволяющий исчислять предметные области любого вида (выделено нами – А.Г.).

4. Выработать эвристические правила обучения работе с такой системой.

5. Разработать систему взаимных отсылок на семантической основе» [Мартынов (1), 2001].

Достигнутые успехи уже в 1993 году позволили сотрудникам научно-исследовательского центра «Семантика» под руководством В.В. Мартынова приступить к интенсивному поиску способов расширения базового семантического классификатора до энциклопедических баз знаний. Оказалось, что мультипликация сложных цепочек

¹ В обоих случаях – естественного языка: на эту трудность лингвистического анализа неоднократно указывали Л. Ельмслев [Ельмслев, 1960] и У. Вейнрейх [Вейнрейх, 1970; 1981].

² «Для исключения из аксиоматической теории таких противоречий нужно точно описать ее язык, т.е. множество предложений теории и множество используемых при их построении символов. Так мы избежим противоречий, возникающих из коллизии теории и ее метатеории, т.е. из включения метатеоретических утверждений в теорию. Это побуждает нас ввести еще большую точность в построение математических теорий и ведет к понятию *формализованной теории*, в которой не только свойства элементарных понятий заданы точным аксиоматическим способом, но точно определен также язык теории» [Расёва и др.(2), 1972].

³ Содержание парадокса и наш комментарий относительно включения наблюдателя в созданную им модель мира см. в заключении настоящей статьи.

⁴ Более подробно об этом [Гордей (1), 1998].

УСК не обеспечивает вход в предметную область, поскольку превышает порог глубины Ингве [Ингве, 1965]: при числе умножений больше восьми формула не читалась и не воспринималась [Гордей (1), 1995]. В 1994 году нами впервые была предложена процедура исчисления предметных областей в виде особо ориентированного графа ранжирования сложных цепочек [Гордей (2), 1995]. Применение процедуры потребовало установления одно-однозначного (векторного) перехода между макропроцессами в базовом семантическом классификаторе и привело к созданию теории автоматического порождения архитектуры знания (ТАПАЗ), основу которой составили формализованная теория, семантический двойник, таблица семантических элементов (макропроцессов), алгоритм ролей индивидов и граф поиска гипонимов через гиперонимы [Гордей (3,5), 1998]. В.В. Мартынов по этому поводу писал: «Решая задачу "исчисления семантики", Гордей отказался от представления знаний на основе построения цепочек типа "если..., то..." (стимул – реакция) и вместо этого обратился к построению известным образом организованных кортежей компонентов (может быть использована и другая терминология, которая в любом случае относится к числу неопределяемых понятий). Фактически это означало ограничение семантических построений простыми расширенными цепочками (в смысле УСК) без преобразования их в сложные <...>. Кроме принятых в УСК операции совмещения (*) и операции приложения (дополнения) (–) он вводит операцию "взятия внутреннейности" (~) <...>. Исчисление простых расширенных цепочек построено у Гордея на основании так называемых транспозиционных преобразований цепочек по С.Н. Фурсу. При этом Гордей семантизировал эти преобразования на основе построенной им модели мира <...>. Главным достижением Гордея как автора нового варианта теории семантического кодирования является проект использования ее для исчисления семантики предметных областей. Это пока еще проект, но в случае его выполнения мы должны получить чрезвычайно полезный инструмент для работающих систем искусственного интеллекта» [Мартынов (1), 1998].

1. Теория автоматического порождения архитектуры знаний: ТАПАЗ-2

Новую версию ТАПАЗ отличает от предыдущей упрощение алгебраического аппарата, увеличение правил интерпретации типовых совмещений индивидов и минимизация семантических исчислений. Количество операций на цепочках семантического кода сокращено до двух. Теперь это алгебра, вида:

$$A = \langle M, *, \sim \rangle$$

В предыдущей версии операция взятия внутреннейности, обозначавшаяся символом ~ (волна), использовалась для разграничения физических и информационных процессов, поскольку на

предельно абстрактном семантическом уровне физический процесс рассматривался как воздействие одного индивида на другой посредством своей оболочки, а информационный – посредством своей среды [Гордей (2), 2005]. Вместе с тем, совмещение индивида со своей средой интерпретировалось как его аннигилирование [Гордей (1), 1998], поэтому пришлось разграничивать внешнюю и внутреннюю среду индивида, чтобы совмещение индивида со своей внешней средой трактовать как аннигилирование, а совмещение со своей внутренней средой – как переход физического процесса в информационный [Гордей (2), 1998]. Такой формальный прием устанавливал параллелизм и симметрию физических и информационных процессов. Действительно, для передачи информации необходим материальный носитель, который оказывает на объект физическое воздействие, а по изменениям, происходящим в предметах после их физического взаимодействия, можно судить о характере самого взаимодействия. Например, во время обучения (информационного процесса) учитель использует голосовые связки, напрягая их, он увеличивает амплитуду звуковых волн (материального носителя) и может оглушить учеников, т.е. оказать на них физическое воздействие; столкновение металлических шаров во время метания – физический процесс, но по вмятинам на шарах можно получить информацию о силе их столкновения. Иными словами, выдвигание в процессе на первый план физической или информационной составляющей производится наблюдателем и зависит от его точки зрения точно так же, как описание угла куба в трехмерном пространстве в виде \searrow (вилки) или \rightarrow (стрелки) [Уинстон, 1980]. Учитывая требование одно-однозначного соответствия между представлением и содержанием¹, выражения с оператором взятия внутреннейности у первого индивида передавали информационный процесс, без оператора взятия внутреннейности – физический. При этом все выражения с оператором взятия внутреннейности и без оператора взятия внутреннейности, имевшие одинаковые наборы аргументов, были алгебраически и геометрически равносильны: совмещение индивидов предполагало совмещение их границ и внутренностей, равно как и совмещение внутренностей индивидов предполагало совмещение их границ.

Дальнейшее исследование показало, что совмещение индивида со своей средой при условии продолжения протекания процесса может быть рассмотрено как переход физического процесса в информационный². Например, если бурлящая в

¹ «Каждой цепочке (комбинации элементарных символов) должен соответствовать один и только один смысл» [Мартынов (1), 1988].

² Попутно отметим, что подобный переход Г. фон Вригту представлялся непостижимым: «Допустим, меня спросили, как я повернул ручку, и я отвечаю, что ухватился за ручку правой рукой и повернул ее по часовой стрелке. В этом случае также верно утверждение, что совершая эти действия, я вызвал поворачивание ручки. Но если меня спросят, как я повернул руку, то сказать, что я вызвал это путем сокращения и расслабления особой группы мышц, не будет правильно. Ведь если я случайно не обладаю специальными знаниями по анатомии, я не знаю, ни какие это мышцы, ни как их сокращать, не поворачивая руки» [Вригт (1), 1986].

пробирке соляная кислота выплеснется и обожжет руку, то она окажет на руку физическое воздействие, однако если причиной бурления кислоты станет растворенный в ней цинк (совмещенный со своей средой и таким образом аннигилированный как свободный элемент), который по своим свойствам ни до, ни после реакции сам руку не обожжет, то по отношению к руке он осуществит информационное действие: цинк, так сказать, «заставит» кислоту обжечь руку¹. Серьезным изучением синергетических связей ученые занялись не так давно [Хакен, 2001; 2003], [Николис и др., 1990], [Пригожин и др., 1986], [Капра, 2002], хотя о наличии подобных связей неоднократно высказывались многие деятели искусств², а у древних людей наивная синергетика составляла основу мировоззрения³. В современной теоретической семантике принято говорить о наивной языковой картине мира и семантических примитивах [Вежбицка, 2001], однако определение значения слова *аннигиляция* в словаре иностранных слов в русском языке примитивным не назовешь, оно явно заимствовано из физики: «превращение в ничто, уничтожение – превращение электрона и позитрона при столкновении в 2 или 3 фотона; при этом происходит превращение материи из одной формы (электрон, позитрон) в другую форму – электромагнитное излучение (фотоны)» [Словарь иностранных слов, 1996]. Ср.: **АННИГИЛЯЦИЯ** п а р ы ч а с т и ц а - а н т и ч а с т и ц а (от позднелат. *annihilatio* – уничтожение, исчезновение) – один из видов взаимопревращения элементарных частиц. Термином «А.» первоначально наз. эл.-магн. процесс превращения электрона и его *античастицы* – позитрона при их столкновении в эл.-магн. излучение (в фотоны, или γ -кванты). Однако этот термин неудачен, т. к. в процессах А. материя не уничтожается, а лишь превращается из одной формы в другую» [Физическая энциклопедия, 1988]. В этой связи обратим внимание на то, как понимается соотношение физических и информационных процессов в оптоэлектронике: «В качестве материальных объектов информационных процессов в оптоэлектронике выступают электроны, свободные или входящие в состав атомов, молекул или твердых тел, а также фотоны, взаимодействующие с

соответствующей средой. **Взаимодействие между фотонами, атомами и электронами происходит путем поглощения одних и испускания других фотонов** (выделено нами – А.Г.)» [Карих, 2002]. Еще одним подтверждением того, что через аннигиляцию физический процесс переходит в информационный являются данные по нейрофизиологии мозга. В.В. Фролькис отмечает, что с возрастом уменьшается масса и объем головного мозга человека: в возрасте от 60 до 75 лет масса мозга снижается на 6% неравномерно в разных отделах, кора больших полушарий уменьшается на 4%, а в лобной доле на 12-15%, причем наблюдаются половые различия степени атрофии – между 40 и 90 годами у мужчин масса мозга уменьшается на 2,85 г в год, у женщин – на 2,92 г [Фролькис, 1988]. Есть и более радикальные утверждения на этот счет⁴, а также эпатазирующие научную общественность высказывания о «квантовой сцепленности мозга и внешнего мира» [Уилсон, 1998], которые, однако, оказываются не столь эпатазирующими с учетом данных Б.Б. Кажинского по биологической радиосвязи [Кажинский, 1962] и В.П. Морозова по биоакустике [Морозов, 1987]. В любом случае, как утверждает акад. В.В. Фролькис, большинство исследователей мозга человека указывают на преимущественную потерю нейронов в коре, гиппокампе и мозжечке. По обобщенным данным Института геронтологии АМН СССР, с детского по позднего старческого возраста плотность расположения нейронов в различных участках коры снижается на 10-60%, «таким образом, филогенетически "новые" структуры мозга, связанные с познавательной функцией, в большей степени подвержены возрастной потере нейронов, чем филогенетически "старые" образования (ствол)» [Старение мозга, 1991].

В результате обоснования аннигиляции в качестве семантического примитива⁵, устанавливающего переход физического процесса в информационный, удалось сократить операцию взятия внутренности и все информационные макропроцессы описать при помощи операции совмещения * (звездочка) и операции продления — (крышка), добившись не только параллельной и симметричной формализации информационных и физических макропроцессов, но и суперпозиции информационных процессов к физическим в базовом семантическом классификаторе.

¹ Ср. с рассуждением Г. фон Вригта: «Допустим, что некто может "подсмотреть", что происходит в моем мозгу, и выделить нервное событие или совокупность событий *N*, которые, как мы считаем, должны появляться при поднятии руки. Я говорю наблюдателю: "Я могу вызвать в своем мозгу событие *N*. Смотри". Затем я поднимаю руку, и наблюдатель следит за тем, что происходит в мозгу. Он видит событие *N*. Однако если он одновременно видит мое действие, он обнаруживает, что оно совершается долей секунды позже, чем появляется *N*. Строго говоря, он будет наблюдать результат моего действия, хотя моя рука поднимается чуть позже, чем происходит *N*» [Вригт (2), 1986].

² Особенно показателен в этом отношении роман Станислава Лема «Солярис» и снятый по его мотивам одноименный фильм Андрея Тарковского. См.: [Лем, 1988].

³ «Для первобытного человека безопасность мира заключается в упорядоченности обычных явлений. Любое исключение из этого представляется ему тающим угрозой актом произвола, который, стало быть, необходимо исклупить, ибо он является не просто синонимичным нарушением обычного порядка вещей, но вместе с тем и предзнаменованием нежелательных событий в дальнейшем <...> Для нас такой способ группирования фактов, конечно, бессмыслен, однако для первобытного человека он полон смысла и убедителен. *И здесь он неожиданно прав*. Его наблюдения внушают доверие. Опираясь на древнейший опыт, он знает, что такие связи существуют на самом деле <...> Внимательно наблюдая за всем выходящим за рамки обычного, он задолго до нас открыл закон образования групп и серий случаев» [Юнг, 1991].

⁴ «Начиная с юного возраста, мужчина к моменту достижения зрелости теряет 15 процентов объема фронтальной доли, которая осуществляет контроль за вниманием, абстрактным мышлением и подавлением импульсов, и 8,5 % височной доли, управляющей памятью» [Сузарев, 1997].

⁵ Напомним, что «ограничения должны соответствовать интуитивному представлению модели мира. Система аксиом в геометрии, принятая Гильбертом, также основана на интуитивном представлении модели мира, но, разумеется, легче воспринимается, чем соответствующая модель в семантике. Она более привычна, поскольку связана с повседневной физической деятельностью человека» [Мартынов (2), 1998]. Произвольные множества или «множества без объема», допускающие получение из одного шара в евклидовом пространстве путем разрезания и склейки методом Банаха – Тарского два других, равнообъемных первому, здесь явно не подходят, ибо мы интуитивно понимаем, что нельзя из одного апельсина сделать два таких же при помощи одного лишь ножа [Яценко, 2002].

1.1. Предварительные сведения из теории множеств, геометрии и алгебры

Множество – собрание элементов. Пишут $a \in A$, если a является элементом множества A , и $a \notin A$ в противоположном случае. Если каждый элемент множества A принадлежит множеству B и при этом $A \neq B$, то A является **собственным подмножеством** множества B , т.е. $A \subset B$ или $B \supset A$ – A **строго включено** в B или B **строго включает** A ; если каждый элемент множества A принадлежит множеству B и при этом $A=B$, то A является **несобственным подмножеством** множества B , т.е. $A \subseteq B$ или $B \supseteq A$ – A **включено** в B или B **включает** A . Некоторое фиксированное множество X , имеющее подмножества, называется **пространством**. Множество \emptyset , не имеющее элементов, называется **пустым**. Для любых множеств A, B символ $A \cup B$ ($A \cap B$) обозначает **объединение** (**пересечение**) множеств A, B , т.е. множество всех элементов, принадлежащих по крайней мере одному из множеств A, B (принадлежащих обоим множествам A, B). Если $A \cap B = \emptyset$, то множества A, B **не пересекаются**. **Разность** множеств A, B , т.е. множество таких элементов из A , которые не принадлежат B , обозначается $A \setminus B$. Если f – функция, то запись $f(x)$ представляет значение функции f в точке x . «Одной из важнейших операций анализа является предельный переход. В основе этой операции лежит тот факт, что на числовой прямой определено расстояние от одной точки до другой. Многие фундаментальные факты анализа не связаны с алгебраической природой действительных чисел $\langle \dots \rangle$, а опираются лишь на понятие расстояния» [Колмогоров и др., 1989], поэтому алгебраическая операция применима не только к числам, но и к фигурам.

Метрическое пространство – пара M, ρ , состоящая из некоторого множества M точек и расстояния, т.е. однозначной, неотрицательной, действительной функции $\rho(x, y)$, определенной для любых x и y из M и подчиненной следующим трем аксиомам:

- 1) $\rho(x, y) = 0$, тогда и только тогда, когда $x = y$,
- 2) $\rho(x, y) = \rho(y, x)$ (аксиома симметрии),
- 3) $\rho(x, z) \leq \rho(x, y) + \rho(y, z)$ (аксиома треугольника).

Метрическое пространство, т.е. пару (M, ρ) , обычно обозначают буквой R .

Трехмерное арифметическое евклидово пространство R^3 есть множество упорядоченных троек действительных чисел $x = (x_1, x_2, x_3)$ с расстоянием $\rho(x, y) = \sqrt{(y_1 - x_1)^2 + (y_2 - x_2)^2 + (y_3 - x_3)^2}$. В изоморфном трехмерному арифметическому евклидову пространству **трехмерном геометрическом евклидовом пространстве R^3** точки задаются непосредственно, а не представлены наборами своих координат.

Открытый шар $\alpha(x_0, r)$ в пространстве R^3 есть совокупность точек $x \in R^3$, удовлетворяющих условию

$$\rho(x, x_0) < r,$$

где x_0 – центр шара, а r – его радиус;

\mathcal{E} -окрестностью $O_{\mathcal{E}}(x)$ точки x называется открытый шар радиуса \mathcal{E} с центром x .

Замкнутый шар $\alpha[x_0, r]$ – совокупность точек $x \in R^3$, удовлетворяющих условию

$$\rho(x, x_0) \leq r.$$

Граничная сфера $\alpha'[x_0, r]$ – совокупность точек $x \in \alpha[x_0, r]$, удовлетворяющих условию

$$\rho(x, x_0) = r.$$

Замкнутый граничный шаровой слой $\alpha''[x_0, r_0, r]$ – совокупность точек $x \in \alpha[x_0, r]$, удовлетворяющих условию

$$r_0 \leq \rho(x, x_0) \leq r,$$

где r_0 – радиус $\alpha[x_0, r_0]$, r – радиус $\alpha[x_0, r]$, $\alpha[x_0, r_0] \subset \alpha[x_0, r]$, причем для достаточно малой окрестности каждой точки $x_1 \in \alpha'[x_0, r_0]$ и $x_2 \in \alpha'[x_0, r]$ существует такая точка x , что $x \in O_{\mathcal{E}}(x_1)$ и $x \in O_{\mathcal{E}}(x_2)$.

Оператор взятия границы шара обозначается буквой B (от англ. *boundary*).

1.2. Геометрическая модель¹

Пусть R^3 – трехмерное евклидово пространство, в котором находятся восемь замкнутых и четыре открытых шара. Открытые шары не пересекаются. Каждый открытый шар содержит в себе два замкнутых, причем один замкнутый шар вложен в другой. Вложенный замкнутый шар меньше заключающего его замкнутого шара. Замкнутый шар, заключающий вложенный замкнутый шар, меньше открытого. Каждый вложенный замкнутый шар имеет замкнутый граничный шаровой слой. Каждый замкнутый шар, заключающий вложенный замкнутый шар, имеет граничную сферу. Открытые шары границы не имеют (рисунок 1):

$$\alpha_1 \subset \alpha_2 \subset \alpha_3 \subset R^3, \alpha = X, Y, Z, W;$$

$$X \cap Y \cap Z \cap W = \emptyset;$$

α_1 и α_2 – замкнутые шары, α_3 – открытый шар;

¹ Унаследована из предыдущей версии и предназначена для формализации, визуализации и верификации ТАПА3-2. Под **формализованной теорией** как основания и математического обоснования новой версии ТАПА3 понимается «множество некоторых конечных последовательностей символов, называемых формулами и терминами, и множество некоторых простых операций, производимых над этими последовательностями» [Расёва и др.(1), 1972]. Принятые в ней правила соответствуют требованиям, предъявляемым к аксиоматическим системам в отношении непротиворечивости, независимости и полноты, и призваны, с одной стороны, преодолеть недостатки интуитивных семантических теорий, в которых нет четкой границы между тем, что очевидно, и тем, что нуждается в доказательстве, с другой стороны, исключить в иерархических построениях противоречия, возникающие при смешении уровня и метауровня – чрезвычайно острой проблемы для математики и лингвистики, в которых совпадают объект и инструмент исследования. Достоверность формализованной теории подтверждается ее интерпретацией методом Клейна через модель метрического пространства, непротиворечивость которой доказана посредством арифметической модели [Клейн, 1956], [Гильберт, 1948].

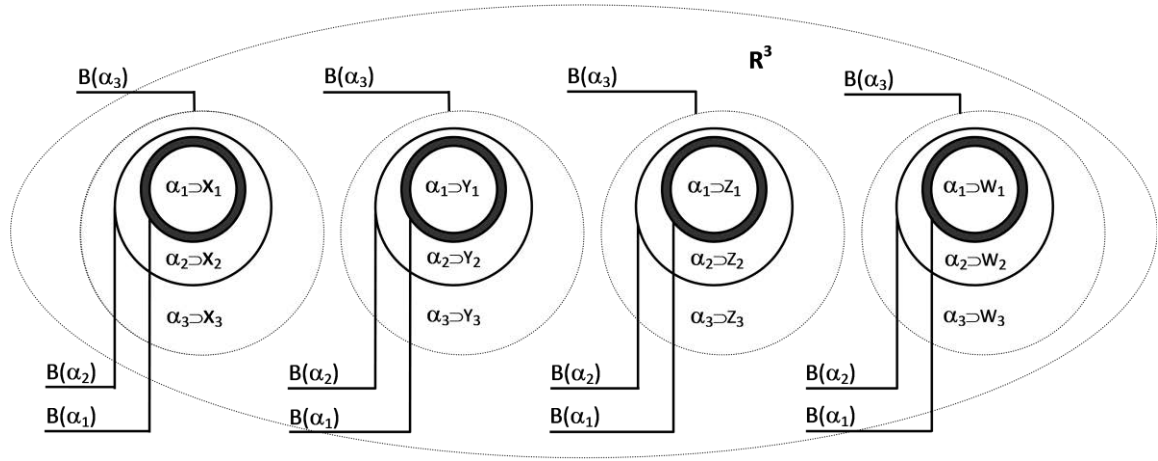


Рисунок 1.

$B(\alpha_1)$ – замкнутый граничный шаровой слой;
 $B(\alpha_2)$ – граничная сфера;
 $B(\alpha_3) = \emptyset$.

1.3. Операция продления

Введем операцию продления α_1 , которую обозначим символом $\overline{\alpha_1}$ (крышка): $\overline{\alpha_1}$ – продление α_1 до α_2 ($\overline{\alpha_1} = \alpha_2 \setminus \alpha_1$), $\overline{\overline{\alpha_1}}$ – продление продления α_1 до α_3 ($\overline{\overline{\alpha_1}} = \alpha_3 \setminus \alpha_2$)¹⁾ (рисунок 2).

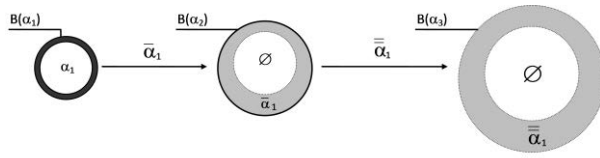


Рисунок 2.

¹⁾ Операция продления применяется к разнородным объектам (шарам с граничным шаровым слоем и шарам с граничной сферой), поэтому требуется ее переопределение. Здесь и далее (рис. 2 – 6) результаты применения операций изображены заштрихованными фигурами. Поскольку $\overline{\alpha_1}$ и $\overline{\overline{\alpha_1}}$ содержат внутри себя пустые объекты, а из алгебры множеств известно, что пустое множество есть подмножество любого множества, т.е. для любого множества A верно $A \cup \emptyset = A$, то графическими репрезентантами объектов $\overline{\alpha_1}$ и $\overline{\overline{\alpha_1}}$ вполне можно считать изображения рис. 3:

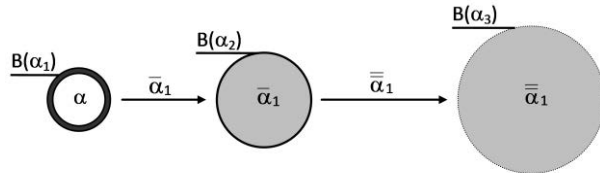


Рисунок 3.

Подобное допущение было сделано в предыдущей версии ТАПАЗ. Действительно, наличие пустых сферических полостей без границ внутри шаров или неоднородность (разреженность) их внутренностей никак не отражается на формализмах ТАПАЗ и их интерпретациях. В физическом мире открытые шары и внутренности замкнутых шаров ассоциированы с однородными нетвердыми телами, граничные сферы и замкнутые граничные шаровые слои – с разнородными твердыми. В ТАПАЗ-2, как и в ТАПАЗ-1, установлен приоритет физических эффектов над геометрическими и геометрических эффектов над алгебраическими (о приоритете «наивной физики» над геометрией см. [Whitehead, 1919]), однако сокращение операции взятия внутренности позволяет сохранить геометрический эффект пустых открытых шаров в графических представлениях, так сказать, «для чистоты метода».

Таким образом, системы объектов α_2 и α_3 являются результатом объединения шаров со своими продлениями (рисунок 4).

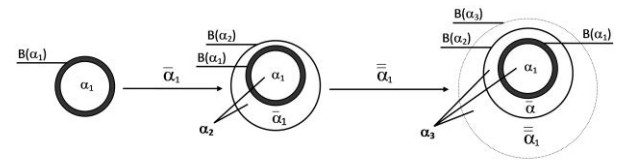


Рисунок 4.

1.4. Операция совмещения

Введем операцию совмещения α_1 и продлений α_1 , которую обозначим символом $*$ (звездочка). Операция совмещения *некоммутативна*¹⁾:

$\alpha_1 * \overline{\alpha_1} = \overline{\alpha_1}$ – преобразование α_1 в продление α_1 (рис. 5(a));

$\overline{\alpha_1} * \alpha_1 = \alpha_1$ – преобразование продления α_1 в α_1 (рис. 5(б)).

$(\alpha_1 * \overline{\alpha_1}) * \overline{\alpha_1}$ – преобразование α_1 в продление α_1 и продления α_1 в продление продления α_1 (рис.6(a)),
 $(\overline{\alpha_1} * \overline{\alpha_1}) * \alpha_1$ – преобразование продления продления α_1 в продление α_1 и продления α_1 в α_1 (рис.6(б)).

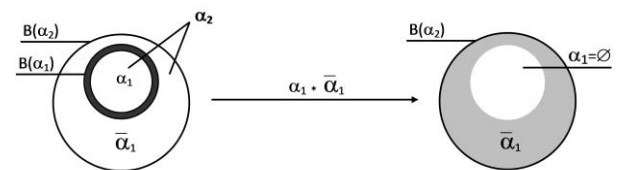


Рисунок 5a.

¹⁾ «Пусть дано некоторое множество M . Мы говорим, что в M определена *бинарная алгебраическая операция*, если всяким двум (различным или одинаковым) элементам множества M , взятым в определенном порядке, по некоторому закону ставится в соответствие вполне определенный третий элемент, принадлежащий к этому же множеству <...> В этом определении содержится указание на порядок, в котором берутся элементы множества M при выполнении операции. Иными словами, не исключается возможность того, что паре элементов a, b из M и паре b, a будут поставлены в соответствие различные элементы из M , т.е. что рассматриваемая операция будет *некоммутативной*» [Курш (1), 1967].

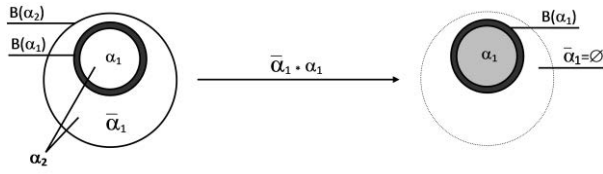


Рисунок 5б.

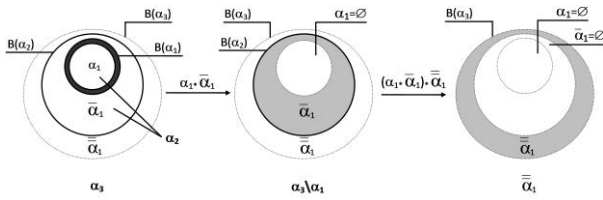


Рисунок 6а.

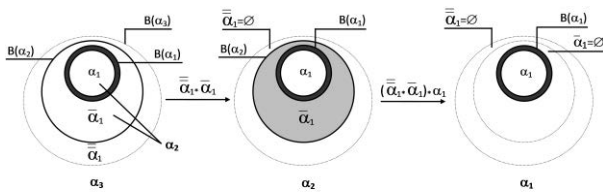


Рисунок 6б.

Операция совмещения ассоциативна¹:
 $(\alpha_1 * \bar{\alpha}_1) * \bar{\alpha}_1 \approx \alpha_1 * (\bar{\alpha}_1 * \bar{\alpha}_1)$ – преобразование α_1 в
 продление α_1 и продления α_1 в продление продления
 α_1 равносильно преобразованию продления α_1 в
 продление продления α_1 и α_1 в продление продления
 α_1 . Действительно, $(\alpha_1 * \bar{\alpha}_1) * \bar{\alpha}_1 = \bar{\alpha}_1 * \bar{\alpha}_1 = \bar{\alpha}_1$,
 $\alpha_1 * (\bar{\alpha}_1 * \bar{\alpha}_1) = \alpha_1 * \bar{\alpha}_1 = \bar{\alpha}_1$. Совмещение α_1 с продлением
 продления α_1 возможно только при преобразовании
 выражений, построенных по перечисленным ниже
 допустимым правилам.

1.5. Правила построения

Пусть $\alpha_1 = X, Y, Z, W$. Разрешены построения
 такого вида²:

- $X * \bar{X}$; (1)
- $(X * \bar{X}) * \bar{Y}$; (2)
- $((X * \bar{X}) * \bar{X}) * \bar{Y}$; (3)
- $((X * \bar{X}) * \bar{Y}) * \bar{Y}$; (4)
- $((X * \bar{X}) * \bar{X}) * \bar{Y} * \bar{Y}$; (5)
- $((X * \bar{X}) * \bar{Y}) * \bar{Y} * Y$; (6)
- $((X * \bar{X}) * \bar{X}) * \bar{Y} * \bar{Y} * Y$; (7)
- $((X * \bar{X}) * \bar{Y}) * \bar{Y} * Y * \bar{Y}$; (8)
- $((X * \bar{X}) * \bar{X}) * \bar{Y} * \bar{Y} * Y * \bar{Y}$; (9)
- $((X * \bar{X}) * \bar{Y}) * \bar{Y} * Y * \bar{Y} * \bar{Y}$; (10)
- $((X * \bar{X}) * \bar{X}) * \bar{Y} * \bar{Y} * Y * \bar{Y} * \bar{Y}$; (11)
- $((X * \bar{X}) * \bar{Y}) * \bar{Y} * Y * \bar{Y} * \bar{Z}$; (12)
- $((X * \bar{X}) * \bar{X}) * \bar{Y} * \bar{Y} * Y * \bar{Y} * \bar{Z}$; (13)
- $((X * \bar{X}) * \bar{Y}) * \bar{Y} * Y * \bar{Y} * \bar{Z} * \bar{Z}$; (14)
- $((X * \bar{X}) * \bar{X}) * \bar{Y} * \bar{Y} * Y * \bar{Y} * \bar{Z} * \bar{Z}$; (15)
- $((X * \bar{X}) * \bar{Y}) * \bar{Y} * Y * \bar{Y} * \bar{Z} * \bar{Z} * \bar{Z}$; (16)
- $((X * \bar{X}) * \bar{X}) * \bar{Y} * \bar{Y} * Y * \bar{Y} * \bar{Z} * \bar{Z} * \bar{Z}$; (17)

¹ «Пусть дано некоторое множество S , конечное или бесконечное. Рассмотрим всевозможные однозначные отображения множества S в себя, т.е. отображения, каждое из которых ставит в соответствие всякому элементу из S вполне определенный элемент этого же множества, хотя, быть может, разные элементы из S отображаются в один и тот же элемент и, с другой стороны, в S могут существовать элементы, в которые ничто не отображается. Если умножением (здесь Курош употребляет мультипликативную терминологию – А.Г.) таких отображений мы назовем их последовательное выполнение, то получим в множестве отображений ассоциативную алгебраическую операцию» [Курош (2), 1967].

² Ограничения, накладываемые на комбинаторику, связаны с правилами интерпретации типовых совмещений индивидов (см. ниже).

- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (18)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (19)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (20)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (21)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (22)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (23)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (24)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (25)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (26)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (27)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (28)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (29)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (30)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (31)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (32)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (33)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (34)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (35)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (36)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (37)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (38)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (39)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (40)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (41)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (42)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (43)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (44)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (45)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (46)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (47)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (48)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (49)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (50)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (51)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (52)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (53)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (54)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (55)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (56)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (57)
- $(((((X * \bar{X}) * \bar{Y}) * Y) * \bar{Y}) * \bar{Z}) * \bar{Z}) * Z$; (58)

1.6. Правила ограничения

За недостаточностью количества индивидуальных
 переменных для семантической интерпретации не
 рассматриваются:

– выражения, состоящие из менее 4-х
 множителей (индивидуальных переменных и их
 продлений); (59)

– выражения, заканчивающиеся совмещением
 продлений различных индивидуальных переменных.
 (60)

Таким образом, исключаются выражения (1) –
 (3), (12) – (14), (27) – (30).

1.7. Правила сокращения

$$(((X * \bar{X}) * \bar{X}) * \bar{Y}) * \bar{Y} \approx ((X * \bar{X}) * \bar{Y}) * \bar{Y} \quad (61)$$

– во всех случаях совмещения продления и
 продления продления одной и той же индивидуальной
 переменной допускается выбрасывание из
 выражения продления этой индивидуальной переменной
 с целью обозначения перехода физического
 процесса в информационный;

$$(((X * \bar{X}) * \bar{Y}) * \bar{Y}) * Y \approx ((X * \bar{X}) * \bar{Y}) * Y,$$

$$(((X * \bar{X}) * \bar{Y}) * \bar{Y}) * Y \approx ((X * \bar{X}) * \bar{Y}) * Y,$$

$$(((X * \bar{X}) * \bar{Y}) * \bar{Y}) * Y * \bar{Y} \approx ((X * \bar{X}) * Y) * \bar{Y},$$

$$(((X * \bar{X}) * \bar{Y}) * \bar{Y}) * Y * \bar{Y} \approx ((X * \bar{X}) * Y) * \bar{Y},$$

$$(((X * \bar{X}) * \bar{Y}) * \bar{Y}) * Y * \bar{Y} * \bar{Y} \approx$$

$$((X * \bar{X}) * \bar{Y}) * \bar{Y},$$

(62)

окружение. Ядро, его оболочка и среда образуют статическую микросистему α_3 , ядро и его оболочка – подсистему α_2 микросистемы α_3 . Статическая микросистема α_3 переводится в динамическое состояние при слиянии (совмещении) ядра с его оболочкой¹. Динамическая система оказывает воздействие на ближайшие к ней статические системы. В рассматриваемой модели мира процесс развивается слева направо от микросистемы X_3 к микросистеме W_3 . Непосредственным инициатором процесса в микросистеме X_3 и перевода ее в динамическое состояние является ядро X , которое выступает в роли *субъекта* действия. Оболочка ядра Y , на которую направлено действие, выполняет роль *объекта*, оболочка или среда ядра X , при помощи которой производится действие, – *инструмента*², среда ядра Y , через которую осуществляется действие, выполняет роль *медиатора*. В процесс последовательно вовлекаются от двух до восемнадцати *предельных индивидов*, от двух до четырех микросистем, причем распределение ролей между микросистемами аналогично распределению ролей между предельными индивидами: микросистема X_3 выполняет роль *составного субъекта*, микросистема W_3 – *составного объекта*, ближайшая к X_3 микросистема Y_3 выполняет роль *составного инструмента*, ближайшая к W_3 микросистема Z_3 – *составного медиатора*. При совмещении предельных индивидов и микросистем происходит передача импульса от активного предельного индивида к пассивному, от динамической микросистемы к статической³. Поглощая активный предельный индивид и воспринимая его импульс, пассивный предельный индивид активизируется, статическая микросистема динамизируется⁴. Очередность совмещения предельных индивидов, как и последовательность перетекания импульса от активного предельного индивида к пассивному, в выражениях формализованной теории отмечена круглыми скобками. Например, в выражении $((X * \bar{X}) * \bar{Y}) * \bar{Y}$ скобки показывают, что, во-первых, вначале ядро X совмещается со своей оболочкой, затем оболочка X -а – со средой ядра Y , затем среда Y -а – с оболочкой Y -а, во-вторых, вначале импульс X -а накапливается в оболочке X -а, а затем передается среде Y -а, в-третьих, вначале происходит совмещение предельных индивидов, а затем передача импульса от активного предельного индивида к пассивному – X совмещается со своей оболочкой и начинает передавать ей импульс, оболочка X -а активизируется, накапливает импульс, совмещается со средой Y -а и начинает

передавать ей импульс, среда Y -а активизируется и совмещается с оболочкой Y -а. Совмещению среды Y -а с его оболочкой предшествует накопление импульса в среде Y -а, что отображается переписыванием скобок при преобразовании выражения: $((X * \bar{X}) * \bar{Y}) * \bar{Y} \rightarrow (X * (\bar{X} * \bar{Y})) * \bar{Y}$ – среда Y -а накапливает импульс и совмещается с оболочкой Y -а, $(X * (\bar{X} * \bar{Y})) * \bar{Y} \rightarrow X * ((\bar{X} * \bar{Y}) * \bar{Y})$ – среда Y -а накапливает импульс, совмещается с оболочкой Y -а и начинает передавать ей импульс, $X * ((\bar{X} * \bar{Y}) * \bar{Y}) \rightarrow X * (\bar{X} * (\bar{Y} * \bar{Y}))$ – среда Y -а совмещается с оболочкой Y -а и начинает передавать ей импульс, оболочка Y -а активизируется и накапливает импульс. Переписывание скобок изменяет порядок совмещения индивидов: $(X * (\bar{X} * \bar{Y})) * \bar{Y}$ – оболочка X -а совмещается со средой Y -а, одновременно с ней и через нее X также совмещается со средой Y -а, затем среда Y -а – с оболочкой Y -а, $X * ((\bar{X} * \bar{Y}) * \bar{Y})$ – оболочка X -а совмещается со средой Y -а, затем среда Y -а – с оболочкой Y -а, одновременно со средой Y -а и через среду Y -а с оболочкой Y -а совмещается X , $X * (\bar{X} * (\bar{Y} * \bar{Y}))$ – среда Y -а совмещается с оболочкой Y -а, одновременно со средой Y -а и через среду Y -а с оболочкой Y -а совмещается оболочка X -а, одновременно с оболочкой X -а и через оболочку X -а с оболочкой Y -а совмещается X ⁵.

1.11. Неконвенциональность семантического двойника

Семантический двойник не приписывается выражению, а выводится из его структуры. Исчисление семантики ведется относительно подсистемы Y_2 микросистемы Y_3 . Первые два предельных индивида сокращенного выражения задают исходную ситуацию, последние два – конечную. Семантический двойник вычисляется в пять этапов. На первом этапе в выражении анализируется очередность совмещения индивидов, например: $((X * \bar{X}) * \bar{Y}) * \bar{Y}$ – вначале ядро X -а через свою оболочку⁶ совмещается со средой X -а, затем среда X -а – со средой Y -а⁷, затем среда Y -а – с оболочкой Y -а. На втором этапе рассматривается последовательность перетекания информации от активного предельного индивида к пассивному: ядро X -а передает информацию среде X -а, среда X -а накапливает ее и передает среде Y -а. На третьем этапе определяются начальная и конечная ситуации взаимодействия предельных индивидов: ядро X -а при помощи среды X -а воздействует на среду Y -а

¹ Примером перевода статической системы в динамическую может послужить активизация земной поверхности вследствие вулканической деятельности (совмещения глубинных слоев Земли с ее поверхностью).

² Если роль инструмента выполняет среда ядра X -а, то в этом случае роль субъекта играет ядро X -а с его оболочкой.

³ «Существенное изменение, внесенное Айдукевичем <...>, заключалось в том, что он предложил рассматривать комбинацию составных частей *в целом* не как равноправное соединение, а скорее как результат действия одной из этих частей на другие» [Бар-Хиллел, 1965].

⁴ «Целесообразно классифицировать термодинамические системы соответственно характеру обмена энергии (теплоты и работы) и массы через их границы. Мы будем различать *изолированные системы*, которые не обмениваются с внешней средой ни энергией, ни массой, *закрытые системы*, которые обмениваются энергией, но не массой, и *открытые системы*, которые обмениваются и энергией, и массой» [Пригожин, 1960].

⁵ Идея рассматривать совмещение индивидов в выражении $(a \cdot b) \cdot c$ как темпоральную конъюнкцию «и затем», а в преобразовании выражения $a \cdot (b \cdot c)$ – как темпоральную конъюнкцию «и одновременно» принадлежит С.Н. Фурсу. В качестве иллюстрации идеи С.Н. Фурса приводит следующий пример: «Сначала шахтер совмещается с отбойным молотком, затем система шахтер-отбойный молоток – со стеной. После совмещения системы со стеной непосредственное воздействие на стену оказывает отбойный молоток – он долбит стену, и одновременно с ним и через него стену долбит шахтер».

⁶ Обозначающее оболочку X -а пролонгированное \bar{X} выброшено из выражения в соответствии с правилом сокращения (61).

⁷ Такой процесс характеризуется как информационный (см. определения исходных семантических понятий).

(исходная ситуация), в результате чего преодолевается сопротивление среды Y -а и оболочка X -а начинает передавать, а среда Y -а – потреблять информацию от ядра X -а (конечная ситуация). На четвертом этапе процесс описывается относительно подсистемы Y_2 : если среда Y -а потребляет информацию ядра X -а, то подсистема Y_2 *воспринимает* ее. Итак, выражение $((X * \bar{X}) * \bar{Y}) * \bar{Y}$ обозначает восприятие Y_2 информации от X_2 . Параллельное выражение $((X * \bar{X}) * \bar{Y}) * \bar{Y}$ в области физических процессов обозначает *притягивание*¹ подсистемой Y_2 импульса² от подсистемы X_2 ³.

1.12. Интерпретация типовых совмещений индивидов

$\alpha_1 * \bar{\alpha}_1$ после $\alpha_1 * \bar{\alpha}_1$ при $\alpha_1 = \alpha$, β (совмещение α_1 со своей оболочкой как результат физического воздействия β) – *выхолащивание* α_1 ;

$\alpha_1 * \bar{\alpha}_1$ после $(\alpha_1 * \bar{\alpha}_1) * \bar{\alpha}_1 \approx \alpha_1 * \bar{\alpha}_1$ при $\alpha_1 = \alpha$, β (совмещение α_1 со своей оболочкой как результат информационного воздействия β) – *перевоплощение* α_1 ;

$\bar{\alpha}_1 * \bar{\alpha}_1$ после $\alpha_1 * \bar{\alpha}_1$ при:

$\alpha_1 = \alpha$ (совмещение α со своей средой) – *аннигилирование* α ;

$\alpha_1 = \alpha$, β (совмещение α со средой β) – *перемещение* α ;

$\bar{\alpha}_1 * \bar{\alpha}_1$ после $(\alpha_1 * \bar{\alpha}_1) * \bar{\alpha}_1 \approx \alpha_1 * \bar{\alpha}_1$ при:

$\alpha_1 = \alpha$ (совмещение α со средой α) – *умаление* α ;

$\alpha_1 = \alpha$, β (совмещение α со средой β) – *транслирование* α ;

$\bar{\alpha}_1 * \bar{\alpha}_1$ после $\alpha_1 * \bar{\alpha}_1$ (совмещение среды α_1 с оболочкой α_1 как результат физического воздействия на среду α_1) – *формование* α_1 ;

$\bar{\alpha}_1 * \bar{\alpha}_1$ после $(\alpha_1 * \bar{\alpha}_1) * \bar{\alpha}_1 \approx \alpha_1 * \bar{\alpha}_1$ (совмещение среды α_1 с оболочкой α_1 как результат информационного воздействия на среду α_1) – *предрасположение* α_1 ;

$\bar{\alpha}_1 * \alpha_1$ после $\alpha_1 * \bar{\alpha}_1$ (совмещение оболочки α_1 с α_1 как результат физического воздействия) – *формирование* α_1 ;

$\bar{\alpha}_1 * \alpha_1$ после $(\alpha_1 * \bar{\alpha}_1) * \bar{\alpha}_1 \approx \alpha_1 * \bar{\alpha}_1$ при $\alpha_1 = \alpha$, β (совмещение оболочки α_1 с α_1 как результат информационного воздействия β) – *воспитание* α_1 ;

¹ Попадание метеорита, к примеру, в атмосферу Земли является следствием земного притяжения: атмосфера Земли потребляет (расщепляет) метеорит, Земля притягивает его.

² Всякий посредник, передавая импульс от одного индивида к другому, часть энергии потребляет сам (так называемое неблагоприятное влияние посредника на процесс), поэтому чем больше посредников между субъектом и объектом, тем выше потери энергии в системе и ниже коэффициент ее полезного действия.

³ Ср.: «Притянуть – ...тягой приблизить, таща придвинуть. *Притянуть лодку к берегу*» [Ожегов (2), 1984]. «Воспринять – ... понять и усвоить. *Хорошо воспринять содержание книги*» [Ожегов (1), 1984]. «Восприятие – ... способность воспринимать и усваивать явления внешнего мира... *Изучать восприятия ребенка*» [Ожегов (1), 1984]. «Притягивать см. привлечь. Привлекать 1. притягивать, влечь, увлекать, тянуть, манить, звать, прельщать, соблазнять... см. вовлекать» [Александрова (5-4), 1986]. «Вовлекать, привлечь..., завлекать, затягивать, втягивать..., заманивать, вmeshивать, впутывать...» [Александрова (1), 1986]. «Воспринимать см. понимать (2). Понимать 2. расценивать, воспринимать, рассматривать, считать за что, усматривать что в чем» [Александрова (2-3), 1986].

$\bar{\alpha}_1 * \bar{\alpha}_1$ после $\alpha_1 * \bar{\alpha}_1$ при $\alpha_1 = \alpha$ (совмещение среды α_1 с оболочкой α_1 как результат физического восстановления) – *реформование* α_1 ;

$\alpha_1 * \bar{\alpha}_1$ после $\alpha_1 * \bar{\alpha}_1$ при $\alpha_1 = \alpha$ (совмещение α_1 с оболочкой α_1 как результат физического восстановления) – *реанимирование* α_1 .

$\alpha_1 * \bar{\alpha}_1$ после $(\alpha_1 * \bar{\alpha}_1) * \bar{\alpha}_1 \approx \alpha_1 * \bar{\alpha}_1$ при $\alpha_1 = \alpha$ (совмещение α_1 с оболочкой α_1 как результат информационного восстановления) – *воскрешение* α_1

$\bar{\alpha}_1 * \alpha_1$ после $(\alpha_1 * \bar{\alpha}_1) * \bar{\alpha}_1 \approx \alpha_1 * \bar{\alpha}_1$ при $\alpha_1 = \alpha$ (совмещение оболочки α_1 с α_1 как результат информационного восстановления) – *воспроизведение* α_1

1.13. Архитектура физических макропроцессов

Принимание подсистемой Y_2 импульса⁴ от подсистемы X_2 :

$$((X * \bar{X}) * \bar{Y}) * \bar{Y} \rightarrow (X * (\bar{X} * \bar{Y})) * \bar{Y} \rightarrow$$

притягивание $Y_2 \text{ imp } X_2$ скапливание $Y_2 \text{ imp } X_2$

$$\rightarrow X * ((\bar{X} * \bar{Y}) * \bar{Y}) \rightarrow X * (\bar{X} * (\bar{Y} * \bar{Y})) \rightarrow$$

ужимание $Y_2 \text{ imp } X_2$ присоединение $Y_2 \text{ imp } X_2$

$$\rightarrow ((X * \bar{X}) * \bar{Y}) * Y \rightarrow (X * (\bar{X} * \bar{Y})) * Y \rightarrow$$

вбирание $Y_2 \text{ imp } X_2$ накапливание $Y_2 \text{ imp } X_2$

$$\rightarrow X * ((\bar{X} * \bar{Y}) * Y) \rightarrow X * (\bar{X} * (\bar{Y} * Y)) \rightarrow$$

центрирование $Y_2 \text{ imp } X_2$ ассимилирование $Y_2 \text{ imp } X_2$

$$\rightarrow ((X * \bar{X}) * Y) * \bar{Y} \rightarrow (X * (\bar{X} * Y)) * \bar{Y} \rightarrow$$

перевбирание $Y_2 \text{ imp } X_2$ концентрирование $Y_2 \text{ imp } X_2$

$$\rightarrow X * ((\bar{X} * Y) * \bar{Y}) \rightarrow X * (\bar{X} * (Y * \bar{Y})) \rightarrow$$

центрифугирование $Y_2 \text{ imp } X_2$ диссимилирование⁵ $Y_2 \text{ imp } X_2$

Проведение подсистемой Y_2 импульса от подсистемы X_2 через подсистему Z_2 :

$$\rightarrow ((Y * \bar{Y}) * \bar{Z}) * \bar{Z} \rightarrow (Y * (\bar{Y} * \bar{Z})) * \bar{Z} \rightarrow$$

подведение $Y_2 \text{ imp } X_2 \text{ к } Z_2$ наращивание $Y_2 \text{ imp } X_2 \text{ на } Z_2$

$$\rightarrow Y * ((\bar{Y} * \bar{Z}) * \bar{Z}) \rightarrow Y * (\bar{Y} * (\bar{Z} * \bar{Z})) \rightarrow$$

прижимание $Y_2 \text{ imp } X_2 \text{ к } Z_2$ подсоединение $Y_2 \text{ imp } X_2 \text{ к } Z_2$

$$\rightarrow ((Y * \bar{Y}) * \bar{Z}) * Z \rightarrow (Y * (\bar{Y} * \bar{Z})) * Z \rightarrow$$

введение $Y_2 \text{ imp } X_2$ нагнетание $Y_2 \text{ imp } X_2$

⁴ Сокр.: *imp*.

⁵ Перемещение стрелки макропроцессов от *диссимилирования* → *подведения* к *диссимилированию* → *выделению* при катастрофическом воздействии подсистемы X_2 на подсистему Y_2 с возможным дальнейшим переходом физического процесса в информационный:

$$\begin{array}{ccc} \rightarrow ((X * \bar{X}) * \bar{Y}) * \bar{Y} & \rightarrow & (X * (\bar{X} * \bar{Y})) * \bar{Y} \rightarrow \\ \text{выделение} & & \text{разуплотнение} \\ Y_2 \text{ imp } X_2 & & Y_2 \text{ imp } X_2 \\ \rightarrow X * ((\bar{X} * \bar{Y}) * \bar{Y}) & \rightarrow & X * (\bar{X} * (\bar{Y} * \bar{Y})) \rightarrow \\ \text{отжигание} & & \text{отъединение} \\ Y_2 \text{ imp } X_2 & & Y_2 \text{ imp } X_2 \end{array}$$

$Y_2 \text{ imp } X_2 \text{ в } Z_2$	$Y_2 \text{ imp } X_2 \text{ в } Z_2$
$\rightarrow Y * ((\bar{Y} * \bar{Z}) * Z) \rightarrow Y * (\bar{Y} * (\bar{Z} * Z)) \rightarrow$	
вжимание	соединение
$Y_2 \text{ imp } X_2 \text{ в } Z_2$	$Y_2 \text{ imp } X_2 \text{ с } Z_2$
$\rightarrow ((Y * \bar{Y}) * Z) * \bar{Z} \rightarrow (Y * (\bar{Y} * Z)) * \bar{Z} \rightarrow$	
проведение	распространение
$Y_2 \text{ imp } X_2 \text{ через } Z_2$	$Y_2 \text{ imp } X_2 \text{ в } Z_2$
$\rightarrow Y * ((\bar{Y} * Z) * \bar{Z}) \rightarrow Y * (\bar{Y} * (Z * \bar{Z})) \rightarrow$	
выжимание	разъединение ¹
$Y_2 \text{ imp } X_2 \text{ из } Z_2$	$Y_2 \text{ imp } X_2 \text{ и } Z_2$

Применение подсистемой Y_2 импульса от подсистемы X_2 по подсистеме W_2 через подсистему Z_2 :

$\rightarrow ((Z * \bar{Z}) * \bar{W}) * \bar{W} \rightarrow (Z * (\bar{Z} * \bar{W})) * \bar{W} \rightarrow$	
затрагивание	обволакивание
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow Z * ((\bar{Z} * \bar{W}) * \bar{W}) \rightarrow Z * (\bar{Z} * (\bar{W} * \bar{W})) \rightarrow$	
обжимание	формование
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow ((Z * \bar{Z}) * \bar{W}) * W \rightarrow (Z * (\bar{Z} * \bar{W})) * W \rightarrow$	
вскрытие	наполнение
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow Z * ((\bar{Z} * \bar{W}) * W) \rightarrow Z * (\bar{Z} * (\bar{W} * W)) \rightarrow$	
сжатие	формирование
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow ((Z * \bar{Z}) * W) * \bar{W} \rightarrow (Z * (\bar{Z} * W)) * \bar{W} \rightarrow$	
пронизывание	переполнение
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow Z * ((\bar{Z} * W) * \bar{W}) \rightarrow Z * (\bar{Z} * (W * \bar{W})) \rightarrow$	
разжимание	выхолащивание
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow ((Z * \bar{Z}) * \bar{W}) * \bar{W} \rightarrow (Z * (\bar{Z} * \bar{W})) * \bar{W} \rightarrow$	
пробивание ²	вздымание
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow Z * ((\bar{Z} * \bar{W}) * \bar{W}) \rightarrow Z * (\bar{Z} * (\bar{W} * \bar{W})) \rightarrow$	
распускание	аннигилирование
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2

Восстановление подсистемой Y_2 импульсом от подсистемы X_2 через подсистему Z_2 подсистемы W_2 :

¹ Перемещение стрелки макропроцессов от *разъединения* \rightarrow *затрагивания* к *выведению* \rightarrow *выведению* при катастрофическом воздействии подсистемы Y_2 на подсистему Z_2 с возможным дальнейшим переходом физического процесса в информационный:

$\rightarrow ((Y * \bar{Y}) * \bar{Z}) * \bar{Z} \rightarrow$	$(Y * (\bar{Y} * \bar{Z})) * \bar{Z} \rightarrow$
выведение	осаживание
$Y_2 \text{ imp } X_2 \text{ из } Z_2$	$Y_2 \text{ imp } X_2 \text{ на } Z_2$
$\rightarrow Y * ((\bar{Y} * \bar{Z}) * \bar{Z}) \rightarrow$	$Y * (\bar{Y} * (\bar{Z} * \bar{Z})) \rightarrow$
оттеснение	отсоединение
$Y_2 \text{ imp } X_2 \text{ от } Z_2$	$Y_2 \text{ imp } X_2 \text{ от } Z_2$

² Начало катастрофического воздействия подсистемы Z_2 на подсистему W_2 .

$\rightarrow ((W * \bar{W}) * \bar{W}) * \bar{W} \rightarrow (W * (\bar{W} * \bar{W})) * \bar{W} \rightarrow$	
рекристаллизование	реинтегрирование
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow (W * ((\bar{W} * \bar{W}) * \bar{W}) \rightarrow W * (\bar{W} * (\bar{W} * \bar{W})) \rightarrow$	
регенерирование	реформование
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow ((W * \bar{W}) * W) * \bar{W} \rightarrow (W * (\bar{W} * W)) * \bar{W} \rightarrow$	
рекуперирование	реабилитирование
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2
$\rightarrow (W * ((\bar{W} * W) * \bar{W}) \rightarrow W * (\bar{W} * (W * \bar{W})) \rightarrow$	
реактивирование	реанимирование
$Y_2 \text{ imp } X_2 W_2$	$Y_2 \text{ imp } X_2 W_2$
через Z_2	через Z_2

1.14. Архитектура информационных макропроцессов

Принимание подсистемой Y_2 информации³ от подсистемы X_2 :

$((X * \bar{X}) * \bar{Y}) * \bar{Y} \rightarrow (X * (\bar{X} * \bar{Y})) * \bar{Y} \rightarrow$	
воспринимание	запоминание
$Y_2 \text{ inf } X_2$	$Y_2 \text{ inf } X_2$
$\rightarrow X * ((\bar{X} * \bar{Y}) * \bar{Y}) \rightarrow X * (\bar{X} * (\bar{Y} * \bar{Y})) \rightarrow$	
осмысливание	понимание
$Y_2 \text{ inf } X_2$	$Y_2 \text{ inf } X_2$
$\rightarrow ((X * \bar{X}) * \bar{Y}) * Y \rightarrow (X * (\bar{X} * \bar{Y})) * Y \rightarrow$	
перенимание	заучивание
$Y_2 \text{ inf } X_2$	$Y_2 \text{ inf } X_2$
$\rightarrow X * ((\bar{X} * \bar{Y}) * Y) \rightarrow X * (\bar{X} * (\bar{Y} * Y)) \rightarrow$	
обдумывание	усваивание
$Y_2 \text{ inf } X_2$	$Y_2 \text{ inf } X_2$
$\rightarrow ((X * \bar{X}) * Y) * \bar{Y} \rightarrow (X * (\bar{X} * Y)) * \bar{Y} \rightarrow$	
прочувствование	созерцание
$Y_2 \text{ inf } X_2$	$Y_2 \text{ inf } X_2$
$\rightarrow X * ((\bar{X} * Y) * \bar{Y}) \rightarrow X * (\bar{X} * (Y * \bar{Y})) \rightarrow$	
переживание	изведывание ⁴
$Y_2 \text{ inf } X_2$	$Y_2 \text{ inf } X_2$

Проведение подсистемой Y_2 информации от подсистемы X_2 через подсистему Z_2 :

$\rightarrow ((Y * \bar{Y}) * \bar{Z}) * \bar{Z} \rightarrow (Y * (\bar{Y} * \bar{Z})) * \bar{Z} \rightarrow$	
сообщение	рекламирование
$Y_2 \text{ inf } X_2 Z_2$	$Y_2 \text{ inf } X_2 Z_2$
$\rightarrow Y * ((\bar{Y} * \bar{Z}) * Z) \rightarrow Y * (\bar{Y} * (\bar{Z} * Z)) \rightarrow$	
внушение	констатирование
$Y_2 \text{ inf } X_2 Z_2$	$Y_2 \text{ inf } X_2 Z_2$

³ Сокр.: inf.

⁴ Перемещение стрелки макропроцессов от *изведывания* \rightarrow *сообщения* к *изведыванию* \rightarrow *отверганию* при катастрофическом воздействии подсистемы X_2 на подсистему Y_2 :

$\rightarrow ((X * \bar{X}) * \bar{Y}) * \bar{Y} \rightarrow (X * (\bar{X} * \bar{Y})) * \bar{Y} \rightarrow$	
отвергание	изглаживание
$Y_2 \text{ inf } X_2$	$Y_2 \text{ inf } X_2$
$\rightarrow X * ((\bar{X} * \bar{Y}) * \bar{Y}) \rightarrow X * (\bar{X} * (\bar{Y} * \bar{Y})) \rightarrow$	
переосмысливание	изживание
$Y_2 \text{ inf } X_2$	$Y_2 \text{ inf } X_2$

$\rightarrow ((Y * \bar{Y}) * \bar{Z}) * Z \rightarrow (Y * (\bar{Y} * \bar{Z})) * Z \rightarrow$
 объяснение пропагандирование
 $Y_2 \inf X_2 Z_2 Y_2 \inf X_2 Z_2$
 $\rightarrow Y * ((\bar{Y} * \bar{Z}) * Z) \rightarrow Y * (\bar{Y} * (\bar{Z} * Z)) \rightarrow$
 доказывание удостоверение
 $Y_2 \inf X_2 Z_2 Y_2 \inf X_2 Z_2$
 $\rightarrow ((Y * \bar{Y}) * Z) * \bar{Z} \rightarrow (Y * (\bar{Y} * Z)) * \bar{Z} \rightarrow$
 ниспослание вещание
 $Y_2 \inf X_2 Z_2 Y_2 \inf X_2 Z_2$
 $\rightarrow Y * ((\bar{Y} * Z) * \bar{Z}) \rightarrow Y * (\bar{Y} * (Z * \bar{Z})) \rightarrow$
 просветление явление¹
 $Y_2 \inf X_2 Z_2 Y_2 \inf X_2 Z_2$

Применение подсистемой Y_2 информации от подсистемы X_2 по подсистеме W_2 через подсистему Z_2 :

$\rightarrow ((Z * \bar{Z}) * \bar{W}) * \bar{W} \rightarrow (Z * (\bar{Z} * \bar{W})) * \bar{W} \rightarrow$
 информирование заинтересовывание
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow Z * ((\bar{Z} * \bar{W}) * \bar{W}) \rightarrow Z * (\bar{Z} * (\bar{W} * \bar{W})) \rightarrow$
 уверение предрасположение
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow ((Z * \bar{Z}) * \bar{W}) * W \rightarrow (Z * (\bar{Z} * \bar{W})) * W \rightarrow$
 наставление обучение
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow Z * ((\bar{Z} * \bar{W}) * W) \rightarrow Z * (\bar{Z} * (\bar{W} * W)) \rightarrow$
 убеждение воспитание
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow ((Z * \bar{Z}) * W) * \bar{W} \rightarrow (Z * (\bar{Z} * W)) * \bar{W} \rightarrow$
 прониимание преисполнение
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow Z * ((\bar{Z} * W) * \bar{W}) \rightarrow Z * (\bar{Z} * (W * \bar{W})) \rightarrow$
 преобразование перевоплощение
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow ((Z * \bar{Z}) * \bar{W}) * \bar{W} \rightarrow (Z * (\bar{Z} * \bar{W})) * \bar{W} \rightarrow$
 донимание² зомбирование
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow Z * ((\bar{Z} * \bar{W}) * \bar{W}) \rightarrow Z * (\bar{Z} * (\bar{W} * \bar{W})) \rightarrow$
 умопомрачение умалишение
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2

¹ Перемещение стрелки макропроцессов от явления \rightarrow информирования к явлению \rightarrow затемнению при катастрофическом воздействии подсистемы Y_2 на подсистему Z_2 :

$\rightarrow ((Y * \bar{Y}) * \bar{Z}) * \bar{Z} \rightarrow (Y * (\bar{Y} * \bar{Z})) * \bar{Z} \rightarrow$
 затемнение шифрование
 $Y_2 \inf X_2 Z_2 Y_2 \inf X_2 Z_2$
 $\rightarrow Y * ((\bar{Y} * \bar{Z}) * \bar{Z}) \rightarrow Y * (\bar{Y} * (\bar{Z} * \bar{Z})) \rightarrow$
 дискредитирование дезавуирование
 $Y_2 \inf X_2 Z_2 Y_2 \inf X_2 Z_2$

² Начало катастрофического воздействия подсистемы Z_2 на подсистему W_2 .

Восстановление подсистемой Y_2 информацией от подсистемы X_2 через подсистему Z_2 подсистемы W_2 :

$\rightarrow ((W * \bar{W}) * \bar{W}) * W \rightarrow (W * (\bar{W} * \bar{W})) * W \rightarrow$
 вспоминание воссоздание
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow W * ((\bar{W} * \bar{W}) * W) \rightarrow W * (\bar{W} * (\bar{W} * W)) \rightarrow$
 возобновление воспроизведение
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow ((W * \bar{W}) * W) * \bar{W} \rightarrow (W * (\bar{W} * W)) * \bar{W} \rightarrow$
 репродуцирование рекультивирование
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2
 $\rightarrow W * ((\bar{W} * W) * \bar{W}) \rightarrow W * (\bar{W} * (W * \bar{W})) \rightarrow$
 возрождение воскрешение
 $Y_2 \inf X_2 W_2 Y_2 \inf X_2 W_2$
 через Z_2 через Z_2

1.15. Список ролей индивидов³

а) субъект: б) инструмент:
 $((\alpha * \alpha) * \alpha) * \alpha$; – активатор
 $(\alpha * (\alpha * \alpha)) * \alpha$; – суппрессор
 $\alpha * ((\alpha * \alpha) * \alpha)$; – усилитель
 $\alpha * (\alpha * (\alpha * \alpha))$; – преобразователь $\alpha * (\alpha * (\alpha * \alpha))$;
 в) медиатор:
 – ориентир $((\alpha * \alpha) * \alpha) * \alpha$;
 – локус $(\alpha * (\alpha * \alpha)) * \alpha$;
 – транспортер $\alpha * ((\alpha * \alpha) * \alpha)$;
 – адаптер $\alpha * (\alpha * (\alpha * \alpha))$;
 – материал $\alpha * (\alpha * (\alpha * W))$;
 – прототип $((\alpha * W) * \alpha) * \alpha$;
 $(\alpha * (W * \alpha)) * \alpha$;
 $\alpha * ((W * \alpha) * \alpha)$;
 $\alpha * (W * (\alpha * \alpha))$;
 – источник $(\alpha * (W * \alpha)) * \alpha$;
 $\alpha * ((W * \alpha) * \alpha)$;
 $(\alpha * (W * \alpha)) * \alpha$;
 $\alpha * ((W * \alpha) * \alpha)$;
 $(\alpha * (W * \alpha)) * \alpha$;
 $\alpha * ((W * \alpha) * \alpha)$;
 г) объект: д) продукт:
 $((\alpha * \alpha) * \alpha) * \alpha$; $((\alpha * \alpha) * \alpha) * W$;
 $(\alpha * (\alpha * \alpha)) * \alpha$; $(\alpha * (\alpha * \alpha)) * W$;
 $\alpha * ((\alpha * \alpha) * \alpha)$; $\alpha * ((\alpha * \alpha) * W)$;
 $\alpha * (\alpha * (\alpha * \alpha))$; $\alpha * (\alpha * (\alpha * W))$.

³ Предельный индивид, играющий соответствующую роль, заключен в квадрат.

1.16. Таблица семантических элементов (макропроцессов)*

		I	II	III	IV
A	a	1 воспринимание	2 запоминание	3 осмысливание	4 понимание
		притягивание 57	скапливание 58	ужимание 59	присоединение 60
	b	5 перенимание	6 заучивание	7 обдумывание	8 усваивание
		вбирание 61	накапливание 62	центрирование 63	ассимилирование 64
	c	9 прочувствование	10 созерцание	11 переживание	12 изведывание
		перевбирание 65	концентрирование 66	центрифугирование 67	диссимилирование 68
	d	13 отвергание	14 изглаживание	15 переосмысливание	16 изживание
		выделение 69	разуплотнение 70	отжимание 71	отъединение 72
B	a	17 сообщение	18 рекламирование	19 внушение	20 констатирование
		подведение 73	наращивание 74	прижимание 75	подсоединение 76
	b	21 объяснение	22 пропагандирование	23 доказывание	24 удостоверивание
		введение 77	нагнетание 78	вжимание 79	соединение 80
	c	25 ниспослание	26 вещание	27 просветление	28 явление
		проведение 81	распространение 82	выжимание 83	разъединение 84
	d	29 затемнение	30 шифрование	31 дискредитирование	32 дезавуирование
		выведение 85	осаживание 86	оттеснение 87	отсоединение 88
C	a	33 информирование	34 заинтересовывание	35 уверение	36 предрасположение
		затрагивание 89	обволакивание 90	обжимание 91	формование 92
	b	37 наставление	38 обучение	39 убеждение	40 воспитание
		вскрывание 93	наполнение 94	сжимание 95	формирование 96
	c	41 прониимание	42 преисполнение	43 преобразование	44 перевоплощение
		пронизывание 97	переполнение 98	разжимание 99	выхолащивание 100
	d	45 донимание	46 зомбирование	47 умопомрачение	48 умалишение
		пробивание 101	вздымание 102	распускание 103	аннигилирование 104
D	a	49 воспоминание	50 воссоздание	51 возобновление	52 воспроизведение
		рекристаллизование 105	реинтегрирование 106	регенерирование 107	реформование 108
	b	53 репродуцирование	54 рекультивирование	55 возрождение	56 воскрешение
		рекуперирование 109	реабилитирование 110	реактивирование 111	реанимирование 112

* Заштрихованы физические макропроцессы.

A – группа активизации;

B – группа эксплуатации;

C – группа трансформации;

D – группа нормализации;

a – подгруппа *среда-оболочка*;

b – подгруппа *оболочка-ядро*;

c – подгруппа *ядро-оболочка*;

d – подгруппа *оболочка-среда*;

I – ряд инициации;

II – ряд аккумуляции;

III – ряд амплификации;

IV – ряд генерации.

1.17. Процесс как индикатор предметной области

Знаки макропроцессов *лечить, ремонтировать, настраивать* соотносятся со знаком макропроцесса *реформовать* как гипонимы с гиперонимом. Гиперонимы процессов и производные от них гиперонимы имен устанавливают изоморфизм предметных областей и создают структуру знания, гипонимы процессов с производными от них гипонимами имен заполняют ячейки структуры знания конкретным содержанием.

Предлагается следующая процедура поиска гипонимов через гиперонимы:

а) из закрытого списка (1) гиперонимов структуры знания поочередно выбирается один,

который определяется как активный (например, гипероним *реформовать*);

б) с активным гиперонимом в порядке следования макропроцессов таблицы семантических элементов сопоставляются остальные гиперонимы списка (1), которые определяются как уточняющие (активный гипероним обозначает процесс, уточняющие – способ его реализации), например:

реформовать – наращивать → реформовать при помощи наращивания;

реформовать – соединять → реформовать при помощи соединения;

реформовать – обволакивать → реформовать при помощи обволакивания;

в) составляется список (2) производных гипонимов:

реформовать при помощи наращивания → согреть¹;

реформовать при помощи соединения → воссоединять;

реформовать при помощи обволакивания → подкрашивать²;

г) из списка (2) производных гипонимов поочередно выбирается один, который определяется как активный производный;

д) с активным производным гипонимом списка (2) в порядке следования макропроцессов таблицы семантических элементов сопоставляются уточняющие гиперонимы списка (1):

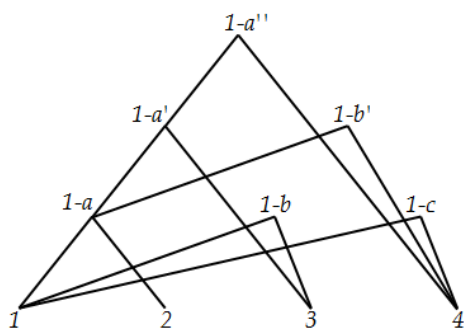
согреть – соединять → согреть при помощи соединения;

согреть – обволакивать → согреть при помощи обволакивания;

е) составляется список (3) производных гипонимов второй степени и т.д.

После того, как найден представляющий ту или иную предметную область гипоним процесса, исчисление семантики самой предметной области осуществляется в соответствии с закрытым списком производных от гипонима процесса гипонимов имен субъекта → активатора → суппрессора → усилителя → преобразователя → ориентира → локуса → транспортера → адаптера → материала → прототипа → источника → объекта → продукта.

Процедура поиска гипонима через гипероним может быть представлена в виде графа, матрицы или системы векторов. С помощью графа отношения активного гиперонима и уточняющих гиперонимов представляются следующим образом:



где 1 – активный гипероним; 2, 3, 4 – уточняющие гиперонимы; 1-a, 1-b, 1-c – производные гипонимы, причем 1-a – активный производный гипоним; 1-a' и 1-b' – производные гипонимы второй степени, причем 1-a' – активный производный гипоним второй степени; 1-a'' – активный производный гипоним третьей степени.

Заключение

ТАРАЗ-2, как и ее предыдущая версия, представляет собой динамическую модель пульсирующей Вселенной³ и обладает необходимым

и достаточным набором формальных средств для исчисления семантики предметных областей и построения энциклопедических баз знаний в искусственных интеллектуальных системах. Все описанные в семантическом классификаторе макропроцессы являются центробежными и центростремительными и как фрагменты события не зависят от точки зрения наблюдателя⁴. Векторный (одно-однозначный) переход между макропроцессами, когда один процесс становится условием протекания второго, был достигнут благодаря формализмам с неконвенциональной семантикой. Главное, что отличает ТАПАЗ-2 от ее прототипа – минимизация исчислений при повышении семантической мощности классификатора, комбинаторика которого покрывает ядро процессуальной семантики⁵. По сравнению с предыдущей версией количество формул сокращено почти в 2 раза (58 вместо 113), упрощена алгебра и введены правила ограничения, но количество семантических элементов в таблице не только не уменьшилось, наоборот, в ней оказались заполненными все пустые ячейки. Был дополнен список ролей индивидов и ряд интерпретаций их типовых совмещений. Это открыло новые перспективы для семантических исчислений. Исследование показало, что один информационный процесс переходит в другой лишь через физический, что остановить переход способна лишь двойная аннигиляция, что полного стирания информации при аннигиляции не происходит и что, наконец, процесс восстановления информации всегда активизирует индивид, который, превращаясь в субъекта, инициирует цепи других директивных событий. В обязательно четном числе посредников между субъектом и объектом всегда выделялись четверки, в которых один индивид играл роль субъекта, другой – объекта, третий – инструмента и четвертый – медиатора⁶. Четырех участников было необходимо и достаточно для оптимального хода события, остальные лишь понижали коэффициент полезного действия – закон сохранения четности в акциональной семантике действовал неукоснительно. В этой связи обращаем внимание на недопустимость произвольного аппликативного наращивания множителей в формализмах ТАПАЗ. Отношения *часть-целое* в ТАПАЗ сведены к отношениям *частей в пределах целого*. Иными словами, во избежание парадокса Рассела один и тот же индивид не может одновременно квалифицироваться как субъект и объект. Вместо этого

уравнений Эйнштейна предполагают Вселенную не вечно равную самой себе, а находящуюся то в сжатии, то в расширении» [Пригожин, 1989]. См. также: [Вайнберг, 1981], [Девис, 1985].

⁴ Наблюдатель, при всем желании, не может включить себя в созданную им модель мира, поскольку такая попытка приводит к парадоксу Рассела. Напоминим его содержание: «Большинство множеств не являются элементами самих себя. Например, множество всех котов не является элементом самого себя, потому что оно само не кот. Возможны, однако, и такие множества, которые принадлежат сами себе как элементы, – например, множество всех множеств. Рассмотрим теперь множество А всех таких множеств Х, что Х не есть элемент Х. Согласно определению, если А есть элемент А, то А также и не есть элемент А, и если А не есть элемент А, то А есть элемент А. В любом случае А есть элемент А и А не есть элемент А» [Менделсон, 1971]. Однако наблюдатель может выразить свое отношение к миру, и в этом отношении – суть его веры.

⁵ Примеры сведения содержания частотных словарей к базовым семантическим исчислениям см.: [Мартынов (2), 2009].

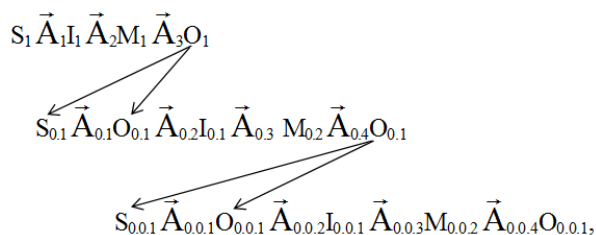
⁶ Ср. с законом сохранения СРТ-четности в физике [Пескин и др., 2001].

¹ Наращивание импульса предполагает *нагревание* индивида.

² *Подкрашивать* означает восстанавливать покрытие при помощи нового аналогичного покрытия.

³ «С 1922 года для математиков стало ясно, что естественные следствия из

он **рекурсивно** разлагается на две части, одна из которых играет роль субъекта, а другая – объекта, т.е. наращивание множителей происходит **ступенчато**¹:



где S – субъект, \bar{A} – акция, I – инструмент, M – медиатор, O – объект.

В.В. Мартынов неоднократно подчеркивал, что «**множество примитивов должно не постулироваться, а рекурсивно исчисляться**, для чего примитивы должны быть распределены в семантическом поле по степени близости» [Мартынов (2), 2001]. Этому требованию ТАПАЗ-2 отвечает в полном объеме, конструктивный дизайн которой позволяет давать несколько решений даже обычных житейских задач, например, по очистке яйца от скорлупы: в подгруппе *среда-оболочка* – раздробить; в подгруппе *оболочка-ядро* – ввести расщепляющий реагент; в подгруппе *ядро-оболочка* – разжать изнутри расширением объема ядра. Последний способ чаще всего используют птицы, когда высасывают птенцов.

ТАПАЗ-2 – один из возможных проектов исчисления семантики. Несмотря на то, что этот проект по ряду показателей превосходит аналоги, а в исчислении предметных областей их не имеет, он не претендует на исключительность. Языковая семантика многогранна и допускает различные способы формализации. Однако все способы, подобно евклидовым и неевклидовым геометриям, должны быть непротиворечивы и эффективны в решении стоящих перед ними задач, и те, кто спорят с этим, по меткому выражению Г. фон Рейхенбаха, лишь «путают строгость метода с ограниченностью цели» [Рейхенбах (1), 1985].

Библиографический список

- [Александрова (1), 1986] З.Е. Александрова, З.Е. Словарь синонимов русского языка / З.Е. Александрова. – М.: Русский язык, 1986. – С.64.
- [Александрова (2), 1986] Александрова, З.Е. Словарь синонимов русского языка / З.Е. Александрова. – М.: Русский язык, 1986. – С.77.
- [Александрова (3), 1986] Александрова, З.Е. Словарь синонимов русского языка / З.Е. Александрова. – М.: Русский язык, 1986. – С.386.
- [Александрова (4), 1986] Александрова, З.Е. Словарь синонимов русского языка / З.Е. Александрова. – М.: Русский язык, 1986. – С.412.
- [Александрова (5), 1986] Александрова, З.Е. Словарь синонимов русского языка / З.Е. Александрова. – М.: Русский язык, 1986. – С.422.
- [Бар-Хиллел, 1965] Бар-Хиллел, И. Новые результаты в теоретической лингвистике / И. Бар-Хиллел // Математическая логика и ее применения. – М.: Мир, 1965. – С.275.
- [Бенвенист, 1974] Бенвенист, Э. Общая лингвистика / Э. Бенвенист. – М.: Прогресс, 1974. – С.136.
- [Бурбаки, 1963] Бурбаки, Н. Очерки по истории математики

- / Н. Бурбаки. – М.: Изд-во иностр. лит, 1963. – 292 с.
- [Вайнберг, 1981] Вайнберг, С. Первые три минуты. Современный взгляд на происхождение Вселенной / С. Вайнберг. – М.: Энергоиздат, 1981. – 209 с.
- [Вейнрейх, 1970] Вейнрейх, У. О семантической структуре языка / У. Вейнрейх // Новое в лингвистике. – М.: Прогресс, 1970. – Вып. V. – С.163-249.
- [Вейнрейх, 1981] Вейнрейх, У. Опыт семантической теории / У. Вейнрейх // Новое в зарубежной лингвистике. – М.: Прогресс, 1981. – Вып. X. – С.50-176.
- [Вежицка, 2001] Вежицка, А. Из книги "Семантические примитивы" / А. Вежицка // Семиотика: Антология / Сост. Ю.С. Степанов. – М.: Академический проект; Екатеринбург: Деловая книга, 2001. – С.242-270.
- [Виноград (1), 1976] Т. Виноград, Т. Программа, понимающая естественный язык / Т. Виноград. – М.: Мир, 1976. – С.48.
- [Виноград (2), 1976] Виноград, Т. Программа, понимающая естественный язык / Т. Виноград. – М.: Мир, 1976. – С.51.
- [Вригт (1), 1986] Вригт, Г.Х. Логико-философские исследования: Избр. тр. / Г.Х. фон Вригт. – М.: Прогресс, 1986. – С.100-101.
- [Вригт (2), 1986] Вригт, Г.Х. Логико-философские исследования: Избр. тр. / Г.Х. фон Вригт. – М.: Прогресс, 1986. – С.109-110.
- [Гадамер, 1991] Гадамер, Г.-Г. Актуальность прекрасного / Г.-Г. Гадамер. – М.: Искусство, 1991. – С.61.
- [Гильберт, 1948] Гильберт, Д. Основания геометрии / Д. Гильберт. – М.-Л.: Гостехиздат, 1948. – 492 с.
- [Гордей (1), 1998] Гордей, А.Н. Дедуктивная теория языка / А.Н. Гордей. – Мн.: Белорусская наука, 1998. – С.26.
- [Гордей (2), 1998] Гордей, А.Н. Дедуктивная теория языка / А.Н. Гордей. – Мн.: Белорусская наука, 1998. – С.33.
- [Гордей (3), 1998] Гордей, А.Н. Дедуктивная теория языка / А.Н. Гордей. – Мн.: Белорусская наука, 1998. – 46 с.
- [Гордей (1), 1995] Гордей, А.Н. Когнитивный подход к обучению знакам / А.Н. Гордей // Компьютерные программы в обучении белорусскому и иностранному языкам: материалы II Респ. конф., Минск, 23-24 февр. 1995 г. / МГЛУ; редкол.: А.А. Габис [и др.]. – Мн., 1995. – С.18-20.
- [Гордей (1), 2005] Гордей, А.Н. Основания комбинаторной семантики / А.Н. Гордей // Слово и словарь = Vocabulum et vocabularium. – Гродно: ГрГУ, 2005. – С.32-35.
- [Гордей (2), 2005] Гордей, А.Н. Основания комбинаторной семантики / А.Н. Гордей // Слово и словарь = Vocabulum et vocabularium. – Гродно: ГрГУ, 2005. – С.35.
- [Гордей (4), 1998] Гордей, А.Н. Принципы исчисления семантики предметных областей / А.Н. Гордей. – Мн.: БГУ, 1998. – С.3-19.
- [Гордей (5), 1998] Гордей, А.Н. Принципы исчисления семантики предметных областей / А.Н. Гордей. – Мн.: БГУ, 1998. – 156 с.
- [Гордей (2), 1995] Гордей, А.Н. Процедуральная семантика и исчисление предметных областей / А.Н. Гордей // Язык: семантика, синтактика, прагматика: материалы I Междунар. науч. конф., Минск, 21-22 июня. 1994 г. / МГЛУ; ред.: Д.Г. Богусевич [и др.]. – Мн., 1995. – Ч.1. – С.16-17.
- [Девис, 1985] Дэвис, П. Случайная Вселенная / П. Дэвис. – М.: Мир, 1985. – 160 с.
- [Ельмслев, 1960] Ельмслев, Л. Прологомены к теории языка / Л. Ельмслев // Новое в лингвистике. – М.: ИЛ, 1960. – Вып. 1. – С.264-389;
- [Ингве, 1965] Ингве, В. Гипотеза глубины / В. Ингве // Новое в лингвистике. – М.: Прогресс, 1965. – Вып. IV. – С.126-138.
- [Кажинский, 1962] Кажинский, Б.Б. Биологическая радиосвязь / Б.Б. Кажинский. – Киев: Изд-во АН УССР, 1962. – 168 с.
- [Кандрашина и др., 1989] Кандрашина, Е.Ю. Представление знаний о времени и пространстве в интеллектуальных системах / Е.Ю. Кандрашина, Л.В. Литвинцева, Д.А. Поспелов. – М.: Наука, 1989. – 328 с.
- [Кантор, 1985] Кантор, Г. Труды по теории множеств / Г. Кантор. – М.: Наука, 1985. – 430 с.
- [Капра, 2002] Капра, Ф. Дао физики / Ф. Капра. – К.: «София», М.: ИД «Гелиос», 2002. – 352 с.
- [Карих, 2002] Карих, Е.Д. Оптоэлектроника / Е.Д. Карих. – Мн.: БГУ, 2002. – С.5-6.
- [Клейн, 1956] Клейн, Ф. О так называемой неевклидовой геометрии / Ф. Клейн // Об основаниях геометрии. – М.: ГИТТЛ, 1956. – С.253-303.
- [Колмогоров и др., 1989] Колмогоров, А.Н. Элементы теории функций и функционального анализа / А.Н. Колмогоров, С.В. Фомин. – М.: Наука, 1989. – С.57.
- [Котарбинский, 1975] Котарбинский, Т. Трактат о хорошей работе / Т. Котарбинский. – М.: Экономика, 1975. – С.31.
- [Курош (1), 1967] Курош, А.Г. Теория групп / А.Г. Курош. – М.: Наука, 1967. – С.15.
- [Курош (2), 1967] Курош, А.Г. Теория групп / А.Г. Курош. – М.: Наука, 1967. – С.17.
- [Лем, 1988] Лем, С. Солярис. Непобедимый. Звездные дневники Ийона Тихого / С. Лем; пер. с польск.: Д. Брускина,

¹ Ср. с идеей «ступенчатого кодирования» [Скороходько, 1978].

- А. Громовай; послесл. Е. Парнова. – М.: Правда, 1988. – 477 с.
- [Ленат, 1986] Ленат, Д.Б. Искусственный интеллект / Д.Б. Ленат // Современный компьютер: сб. науч.-популяр. статей / под ред. В.М. Курочкина. – М.: Мир, 1986. – С.182.
- [Мартинов (1), 2009] Мартинов, В.В. В центре сознания человека / В.В. Мартинов. – Мн.: БГУ, 2009. – С.62.
- [Мартинов (2), 2009] Мартинов, В.В. В центре сознания человека / В.В. Мартинов. – Мн.: БГУ, 2009. – С.92-113.
- [Мартинов (3), 2009] Мартинов, В.В. В центре сознания человека / В.В. Мартинов. – Минск: БГУ, 2009. – 272 с.
- [Мартинов, 1966] Мартинов, В.В. Кибернетика. Семиотика. Лингвистика / В.В. Мартинов. – Мн.: Наука и техника, 1966. – 148 с.
- [Мартинов (1), 1998] Мартинов, В.В. О книге А.Н. Гордея "Дедуктивная теория языка" / В.В. Мартинов // Дедуктивная теория языка / А.Н. Гордей. – Мн.: Беларуская навука, 1998. – С.3-5.
- [Мартинов (2), 2009] Мартинов, В.В. О книге А.Н. Гордея "Дедуктивная теория языка" / В.В. Мартинов // Дедуктивная теория языка / А.Н. Гордей. – Мн.: Беларуская навука, 1998. – С.4.
- [Мартинов (1), 2001] Мартинов, В.В. Основы семантического кодирования. Опыт представления и преобразования знаний / В.В. Мартинов. – Мн.: ЕГУ, 2001. – С.42.
- [Мартинов (2), 2001] Мартинов, В.В. Основы семантического кодирования. Опыт представления и преобразования знаний / В.В. Мартинов. – Мн.: ЕГУ, 2001. – С.116.
- [Мартинов (3), 2001] Мартинов, В.В. Основы семантического кодирования. Опыт представления и преобразования знаний / В.В. Мартинов. – Мн.: ЕГУ, 2001. – 140 с.
- [Мартинов, 1974] Мартинов, В.В. Семиологические основы информатики / В.В. Мартинов. – Мн.: Наука и техника, 1974. – 192 с.
- [Мартинов, 1977] Мартинов, В.В. Универсальный семантический код. Грамматика. Словарь. Тексты / В.В. Мартинов. – Мн.: Наука и техника, 1977. – 191 с.
- [Мартинов, 1984] Мартинов, В.В. Универсальный семантический код: УСК-3 / В.В. Мартинов. – Мн.: Наука и техника, 1984. – 134 с.
- [Мартинов (1), 1988] Мартинов, В.В. Универсальный семантический код: УСК-4 / В.В. Мартинов. – Мн.: Наука и техника, 1988. – С.4.
- [Мартинов (2), 1988] Мартинов, В.В. Универсальный семантический код: УСК-4 / В.В. Мартинов. – Мн.: Наука и техника, 1988. – 30 с. – (Препринт / Ин-т языкознан. АН БССР, №2).
- [Мендельсон, 1971] Мендельсон, Э. Введение в математическую логику / Э. Мендельсон. – М.: Наука, 1971. – С.8.
- [Минский, 1978] Минский, М. Структура для представления знаний / М. Минский // Психология машинного зрения. – М.: Мир, 1978. – С.249.
- [Морозов, 1987] Морозов, В.П. Занимательная биоакустика: Язык эмоций в мире животных и человека / В.П. Морозов. – М.: Знание, 1987. – 208 с.
- [Николис и др., 1990] Николис, Г. Познание сложного / Г. Николис, И. Пригожин. – М.: Мир, 1990. – 342 с.
- [Ожегов (1), 1984] Ожегов, С.И. Словарь русского языка / С.И. Ожегов. – М.: Русский язык, 1984. – С.85.
- [Ожегов (2), 1984] Ожегов, С.И. Словарь русского языка / С.И. Ожегов. – М.: Русский язык, 1984. – С.520.
- [Пескин и др., 2001] Пескин, М.Е. Введение в квантовую теорию поля / М.Е. Пескин, Д.В. Шрёдер. Ижевск: Изд-во РХД, 2001. – 783 с.
- [Пригожин, 1960] Пригожин, И. Введение в термодинамику необратимых процессов / И. Пригожин. – М.: Иностр. лит., 1960. – С.19.
- [Пригожин, 1989] Пригожин, И. Переоткрытие времени / И. Пригожин // Вопросы философии. – 1989. – № 8. – С.18.
- [Пригожин и др., 1986] Пригожин, И. Порядок из хаоса: Новый диалог человека с природой / И. Пригожин, И. Стенгерс. – М.: Прогресс, 1986. – 431 с.
- [Расёва и др.(1), 1972] Расёва, Е. Математика метаматематики / Е. Расёва, Р. Сикорский. – М.: Наука, 1972. – С.9.
- [Расёва и др.(2), 1972] Расёва, Е. Математика метаматематики / Е. Расёва, Р. Сикорский. – М.: Наука, 1972. – С.174.
- [Рейхенбах (1), 1985] Рейхенбах, Г. Философия пространства и времени / Г. Рейхенбах. – М.: Прогресс, 1985. – С.16.
- [Рейхенбах (2), 1985] Рейхенбах, Г. Философия пространства и времени / Г. Рейхенбах. – М.: Прогресс, 1985. – С.18-19.
- [Сикорский, 1969] Сикорский, Р. Булевы алгебры / Р. Сикорский. – М.: Мир, 1969. – 376 с.
- [Скороходько, 1978] Скороходько, Э.Ф. Лингвистические проблемы проектирования информационных систем / Э.Ф. Скороходько. – Киев, 1978. – 263 с.
- [Словарь иностранных слов, 1996] Словарь иностранных слов в русском языке. – М.: Ювенс, 1996. – С.54.
- [Старение мозга, 1991] Старение мозга / Под общей редакцией академика АН УССР В.В. Фролькиса. – Л.: Наука, 1991. – С.14.
- [Сухарев, 1997] Сухарев, В.А. Психология интеллекта / В.А. Сухарев. – Донецк: Сталкер, 1997. – С.4.
- [Уилсон, 1998] Уилсон, Р. Квантовая психология / Р. Уилсон. – Киев: София, 1998. – 224 с.
- [Уинстон, 1980] Уинстон, П. Искусственный интеллект / П. Уинстон. – М.: Мир, 1980. – 520.
- [Физическая энциклопедия, 1988] Физическая энциклопедия: [в 5 т.] / Гл. ред. А.М. Прохоров, редкол.: Д.М. Алексеев [и др.]. – М.: Советская энциклопедия. – 1988. – Т.1. – С.85.
- [Фреге, 1997] Фреге, Г. Избранные работы / Г. Фреге. – М.: Дом интеллектуальной книги: Русск. феноменол. об-во, 1997. – 160 с.
- [Фреге, 2000] Фреге, Г. Логика и логическая семантика: Сборник трудов / Под ред. З.А. Кузичевой: Учебное пособие для студентов вузов. – М.: Аспект Пресс, 2000. – 512 с.
- [Фролькис, 1988] Фролькис, В.В. Старение и увеличение продолжительности жизни / В.В. Фролькис. – Л.: Наука, 1988. – 239 с.
- [Фурс, 1999] Фурс, С.Н. Введение в аксиоаналитику / С.Н. Фурс. – Мн.: Беларуская навука, 1999. – 67 с.
- [Хакен, 2001] Хакен, Г. Принципы работы головного мозга: Синергетический подход к активности мозга, поведению и когнитивной деятельности / Г. Хакен. – М.: Изд-во Per Se, 2001. – 353 с.
- [Хакен, 2003] Хакен, Г. Тайны природы. Синергетика: учение о взаимодействии / Г. Хакен. – Москва-Ижевск: Ин-т компьютерных исследований, 2003. – 320 с.
- [Юнг, 1991] Юнг, К.-Г. Архаический человек / К.-Г. Юнг // Архетип и символ. – М.: Ренессанс, 1991. – С.167-168.
- [Ященко, 2002] Ященко, И.В. Парадоксы теории множеств / И.В. Ященко // Библиотека «Математическое просвещение». – М.: МЦНМО, 2002. – Вып. 20. – 40 с.
- [Amarel, 1968] Amarel, S. On the Representation of Problem of Reasoning about Action / S. Amarel // Machine Intelligence. – 1968. – №3.
- [Bobrow et al., 1977] Bobrow, D.G. An Overview of KRL, a Knowledge Representation Language / D.G. Bobrow, T. Winograd // Cognitive Science. – 1977. – Vol. 1. – №1. – P.11.
- [Martynov, 1995] Martynov, V.V. Universal Semantic Code: USC-5 / V.V. Martynov. – Minsk: MSLU, 1995. – 21 p. – (Pre-print / MSLU; №4).
- [Whitehead, 1919] Whitehead, A.N. An Enquiry concerning the principles of natural knowledge / A.N. Whitehead. – Cambridge: University Press, 1919. – 200 p.
- [Winograd, 1980] Winograd, T. Extended Inference Models in Reasoning by Computers Systems / T. Winograd // Artificial Intelligence. – 1980. – XIII. – № 1. – P.5-26.
- [Wolniewicz, 1982] Wolniewicz, B. A Formal Ontology of Situations / B. Wolniewicz // Studia Logica. – 1982. – T.41. – №4. – P.381-413.

THEORY OF THE KNOWLEDGE ARCHITECTURE AUTOMATIC FORMATION (TKAAF-2) AND FURTHER MINIMIZATION OF SEMANTIC CALCULUS

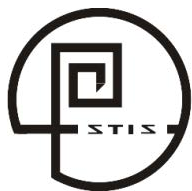
Aliaksandr Hardzei

*Belarusian State University,
Minsk, Republic of Belarus*

aliaks2001@yahoo.com

To clarify the priority of the «world model» to the «language picture of the world», geometric methods are applied. The new version of the Theory of the Knowledge Architecture Automatic Formation (TKAAF-2) are stated, and algebraic tools for the calculus of the semantic in different fields of knowledge and procedural representing and transformation knowledge for artificial intelligence systems are determined. The examples of formal description of the Russian verbal semantics are given according to their dependence on the superposition of the cognitive processes.

Key words: world model, semantic code, stereotype, geometric model, semantic primitive, semantic classifier, formalized theory, scope of axioms.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

СТРУКТУРИЗАЦИЯ СМЫСЛОВОГО ПРОСТРАНСТВА

Голенков В. В., Гулякина Н.А.

** Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

golen@bsuir.by
guliakina@bsuir.by

В работе рассматривается подход к представлению информации в интеллектуальных системах, основанный на семантических сетях. Основное внимание уделяется рассмотрению таких понятий, как предметная область, онтология предметной области, рассматривается их классификация и особенности наиболее характерных классов. Отдельный раздел описывает подход к согласованию семантических сетей и Универсального семантического кода, предложенного В.В. Мартыновым.

Ключевые слова: интеллектуальные системы, семантические сети, универсальный семантический код, формализация смысла

Введение

К числу основных современных тенденций развития искусственного интеллекта следует отнести

- (1) переход от частных теорий (моделей) различных компонентов интеллектуальных систем к общей (единой, комплексной, интегрированной) теории интеллектуальных систем;
- (2) переход от теории интеллектуальных систем к практике создания широко используемых интеллектуальных систем, к существенному расширению сфер реального практического использования интеллектуальных систем;
- (3) переход от создания прикладных интеллектуальных систем к созданию технологий, обеспечивающих быструю их разработку (как минимум, быстрое прототипирование) широким контингентом разработчиков и реализованных в виде интеллектуальных метасистем, осуществляющих поддержку такой разработки;
- (4) переход от разработки интеллектуальных систем к постоянному их совершенствованию (в том числе, в ходе эксплуатации) в целях существенного продления их жизненного цикла;
- (5) переход от технологий разработки интеллектуальных систем к технологиям их постоянного совершенствования (эволюции) в ходе эксплуатации;
- (6) переход от технологий разработки и совершенствования интеллектуальных систем к

технологиям совершенствования (эволюции) самих этих технологий (в том числе и соответствующих им интеллектуальных метасистем);

- (7) переход от автономных интеллектуальных систем к коллективным интеллектуальным системам и к коллективам, состоящим из интеллектуальных и традиционных компьютерных систем, а также к технологиям, обеспечивающим разработку таких коллективов взаимодействующих систем.

Для обеспечения перечисленных тенденций необходимо создание не только общей теории интеллектуальных систем, но и общей семантической теории любых компьютерных систем, обеспечивающей их унификацию и совместимость и четко разделяющей многообразие форм реализации систем, от многообразия их смысловых (семантических) моделей.

Такую общую теорию можно строить только на основе формализации смысла обрабатываемой информации. Проблема формализации смысла в современной информатике является ключевой, поскольку без ее решения невозможно решить целый ряд проблем, таких, как проблема семантической совместимости компьютерных систем, проблема дублирования технических решений, например, при разработке программных систем из-за многообразия форм их реализации.

Нужно четко отличать

- (1) многообразие языков (форм) представления информации от многообразия смысла представляемой информации;
- (2) многообразие форм организации обработки информации (машин обработки информации) от многообразия сути обработки информации на смысловом уровне.

Необходимо четко отделить суть дела, суть представления и обработки информации от формы, от многообразия вариантов реализации (воплощения) одной и той же сути представления и обработки информации.

В данной статье будут затронуты следующие вопросы:

- Что такое смысл.
- Что такое смысловое представление знаний, каким требованиям оно должно удовлетворять.
- Какова семантическая типология знаний и принципы смыслового представления каждого вида знаний.
- Какие существуют связи между знаниями (метасвязи) – отношения, заданные на множестве всевозможных знаний.
- Что такое смысловое пространство и какова его структура.
- Какими особенностями обладает обработка знаний (в т.ч. решение задач), осуществляемая на уровне смыслового представления обрабатываемых знаний – какова особенность семантических моделей обработки знаний

1. Понятие смысла и понятие семантической сети как способа внутреннего смыслового представления информации в памяти интеллектуальных систем

Без уточнения понятия смысла невозможно:

- решить задачу понимания информации, поступающей на вход интеллектуальной системы по разным каналам;
- разработать такой способ внутреннего смыслового представления (кодирования) знаний в памяти интеллектуальной системы, который был бы лишен каких бы то ни было особенностей, обусловленных не смыслом (сутью) хранимых знаний, а предлагаемой формой их представления.

Следуя Л. Вингенштейну можно сказать, что язык – это форма представления изображения различных видов знаний, это "одежда", в которую эти знания "одеваются". Смысл же представляемого знания есть инвариант многообразия форм его выражения, многообразия "одежд", в которые это знание может быть "одето".

Смысл знания (знаковой конструкции) – это такое абстрактное представление этого знания, которое является инвариантом всего многообразия

семантически эквивалентных форм (вариантов) представления этого знания в самых различных языках.

Смысл – это абстрактная знаковая конструкция, являющаяся инвариантом максимального класса семантически эквивалентных знаковых конструкций (текстов), принадлежащих самым разным языкам. Аналогами данной трактовки понятия смысла являются такие понятия, как:

- инвариант максимального класса изоморфных графовых структур;
- инвариант максимального класса изоморфных алгебраических систем;
- инвариант максимального класса конгруэнтных геометрических фигур.

Смысл (смысловое представление информации) является абстрактной знаковой конструкцией, обладающей следующими свойствами:

- (1) Среди знаков, входящих в состав указанной знаковой конструкции не должно существовать пар синонимичных знаков, т.е. знаков имеющих один и тот же денотат знаков, обозначающих одну и ту же сущность. Другими словами, каждый знак, входящий в состав указанной знаковой конструкции, входит в нее однократно. Если в процессе обработки смыслового представления информации появляются синонимичные знаки, они должны быть "склеены".
- (2) Ни в каком виде не должно быть дублирования информации – не только в виде многократного вхождения знаков, обозначающих одни и те же сущности (т.е. синонимичные знаки), но и в виде многократного вхождения семантически эквивалентных текстов (знаний), что, собственно, является следствием синонимии знаков. При этом необходимо четко отличать семантическую эквивалентность знаний от их логической эквивалентности, т.е. от их взаимной логической выводимости.
- (3) Среди знаков, входящих в состав указанной знаковой конструкции, не должно существовать омонимичных знаков, т.е. знаков, которые в разных контекстах могут обозначать разные сущности, т.е. иметь разные денотаты. Из этого, в частности, следует, что в состав рассматриваемой знаковой конструкции не могут входить местоимения. Если в процессе анализа смыслового представления информации будут выявлены омонимичные знаки, то они должны быть "расклеены" ("расщеплены") на два или более неомонимичных знака.
- (4) Все знаки, входящие в состав указанной знаковой конструкции, не должны иметь внутренней структуры, анализ которой необходим для понимания (прочтения) этой знаковой конструкции. Внутренняя структура знаков необходима только в тех знаковых конструкциях, в которых разрешена множественность вхождения знаков, имеющих

один и тот же денотат, и анализ совпадения внутренних структур таких знаков является средством установления их синонимии или предполагаемой синонимии, уточняемой на этапе последующего семантического анализа знаковой конструкции. Следовательно, все знаки, входящие в состав смыслового представления информации, можно считать абстрактными, поскольку они (а) абстрагируются от своего внутреннего устройства (в т.ч. и от вариантов своей материализации кодирования в памяти интеллектуальной системы), (б) обладают только одним свойством – обозначать взаимно однозначно соответствующие им денотаты, (в) являются инвариантами соответствующих максимальных классов синонимичных знаков. В смысловом представлении информации абсолютно несущественно то, как выглядят знаки, входящие в эту информационную конструкцию, как они строятся из незначимых атомарных фрагментов информационной конструкции (из графических примитивов, букв, пикселей,...). Во всех же других языках знаки, входящие в состав текстов, не являются атомарными (элементарными) фрагментами текстов – они строятся из "кирпичей" (символов, букв, пикселей, ...), которые знаками не являются. Знаки из таких текстов надо еще уметь выделить. Так, например, надо уметь читать машиностроительные чертежи (надо уметь выделять компоненты деталей, связи между ними, ...), фотографии местности, карты местности, рукописи. В смысловом представлении информации все знаки, входящие в состав информационно конструкции, являются ее атомарными (элементарными, примитивными) фрагментами.

- (5) В состав знаковой конструкции, являющейся смысловым представлением информации, не должно входить ничего, кроме рассмотренных выше абстрактных знаков. Это означает, что в состав рассматриваемой знаковой конструкции не могут входить не только символы (примитивы), из которых строятся знаки, но и такие элементы знаковой конструкции, как разделители, ограничители, предлоги. А это, в свою очередь, означает, что все атомарные (элементарные) фрагменты рассматриваемой знаковой конструкции являются знаками, т.е. семантически значимыми фрагментами.
- (6) В рамках построения смыслового представления информации не должны использоваться не только такие конструкции, как слова, термины (словосочетания), но и такие языковые приемы, как склонение, спряжение.
- (7) С синтаксической точки зрения все знаки, входящие в состав смыслового представления информации должны четко делиться на два вида:
 - *знаки связей* между описываемыми сущностями (подчеркнем при этом, что связь

считается одним из видов описываемых сущностей);

- *знаки сущностей*, которые связями не являются. При этом между множеством *знаков связей* и множеством знаков иных сущностей задается несколько соответствий инцидентности, которые определяют то, какие сущности связываются каждой связью (являются ее компонентами), и то, какие роли выполняют указанные сущности в рамках соответствующих им связей.

Важным достоинством смыслового представления информации является то, что в нем явно и четко задаются (представляются) связи между описываемыми сущностями (в т.ч. и связи между связями) в виде связей между знаками этих сущностей: четко указывается семантический тип связи, т.е. отношение, которому связь принадлежит, а также указываются компоненты связи (связываемые знаки) и роли этих компонентов в рамках этой связи. Подчеркнем также, что все связи каждой описываемой сущности в смысловом представлении информации локализованы (представлены) только множеством знаков связей, инцидентных знаку указанной сущности, т.е. множеством связей, одним из компонентов которых является знак рассматриваемой сущности. Таким образом, любую знаковую конструкцию можно представить (в семантически эквивалентном виде) как множество знаков описываемых сущностей и множество знаков связей, связывающих эти описываемые сущности с другими сущностями. При этом на описываемые сущности и на связи между ними не накладывается никаких ограничений. Описываемые сущности могут быть:

- *материальными* (физическими) и *абстрактными* (виртуальными) – числами, множествами, знаками каких-либо сущностей;
- *действительно* существующими и *вымышленными*;
- *фиксированными* (константными) и *произвольными* (переменными);
- *связями* между сущностями.

Нетрудно заметить, что знаковая конструкция, являющаяся смысловым представлением информации, в общем случае не может быть линейной, поскольку каждая описываемая сущность, являющаяся денотатом соответствующего знака, может быть связана неограниченным числом связей с другими сущностями, описываемыми в этой же знаковой конструкции. Таким образом, смысловое представление информации есть графовая структура, имеющая различную теоретико-графовую конфигурацию. Такие графовые структуры, обладающие указанными выше семантическими свойствами, называют также семантическими сетями [Болотова, 2012], [Скорородько, 1983]. Приведенная трактовка семантических сетей дает возможность в полной мере использовать теорию графов для исследований

синтаксических свойств семантических сетей и для построения алгоритмов обработки семантических сетей.

Важнейшим достоинством семантических сетей является то, что в них достаточно просто и наглядно выглядит процедура навигационного поиска знаков описываемых сущностей на основе априорного знания семейства отношений, заданных на этих сущностях.

Следует при этом четко отличать саму семантическую сеть, являющуюся внутренним абстрактным представлением информации в памяти интеллектуальной системы, от ее кодирования в рамках выбранного варианта технической реализации указанной памяти, а также от различных вариантов ее визуализации для пользователей. Сама семантическая сеть и ее графическое или текстовое изображение – не одно и то же. Аналогично этому следует четко отличать графовую структуру как абстрактный объект от различных вариантов ее представления.

2. Унификация семантических сетей

Рассмотрим следующие вопросы:

- Зачем нужна унификация семантических сетей.
- В каких направлениях такая унификация может осуществляться.
- Как выглядит синтаксис и семантика языка унифицированных семантических сетей.
- Какими достоинствами обладает предлагаемый язык унифицированных семантических сетей.
- Какие предлагаются варианты визуализации унифицированных семантических сетей.

Унификация семантических сетей необходима, прежде всего, для конструктивного решения проблемы интеграции знаний.

Предложенный нами язык унифицированных семантических сетей назван *SC-кодом* (Semantic Code). Тексты этого языка будем называть *sc-текстами*. Знаки, входящие в состав *sc-текстов*, будем называть *sc-элементами*. Переход от введенного выше общего понятия семантической сети к унифицированным семантическим сетям (*sc-текстам*) рассмотрим как задание целого ряда ограничений на *семантические сети* общего вида, но таких ограничений, которые не снижают семантической мощности языка семантических сетей, претендующего на универсальность.

Ограничение 1. Семантическая нормализация описываемых множеств. Если элемент семантической сети (*sc-элемент*) является знаком некоторого *множества*, то каждый элемент этого *множества*, представляющий собой сущность, не являющуюся знаком (*sc-элементом*), заменяется на знак указанной сущности (на *sc-элемент*, обозначающий эту сущность). Таким образом, все описываемые *sc-текстом* множества становятся множеством знаков, а точнее, множеством *sc-элементов* (*sc-множествами*). Такие множества

условно будем называть семантически нормализованными. Очевидно, любое множество может быть представлено в семантически нормализованном виде. Для описания связей между *sc-множествами* и элементами этих множеств вводятся *связи принадлежности*. Каждая из этих связей связывает знак некоторого *sc-множества* с одним из элементов этого множества, который всегда является знаком (*sc-элементом*) благодаря семантической нормализации *sc-множества*.

Ограничение 2. Каждая *связь* описываемых сущностей трактуется как множество, элементами которого являются описываемые сущности, связываемые этой связью. Семантическая нормализация каждого такого множества означает то, что все связи, входящие в состав *семантической сети*, будут связывать не сами описываемые сущности, а *знаки* этих сущностей за исключением случаев, когда указанные сущности уже являются знаками. Напомним при этом, что знаки описываемых сущностей, связи между знаками описываемых сущностей и знаки этих связей включаются в число сущностей, описываемых семантической сетью. Подчеркнем также, что *связь-*i** между *sc-элементом*, который является знаком некоторой *связи-*j**, и *sc-элементом*, который является компонентом этой *связи-*j** (т.е. является знаком, связываемым *связью-*j**), в *sc-тексте* может быть представлена двумя способами:

- с помощью явно вводимого *sc-элемента*, обозначающего *связь принадлежности*;
- с помощью синтаксически задаваемой *пары инцидентности*, для которой соответствующий ей *sc-элемент* (знак) не вводится.

Ограничение 3. Если в состав *семантической сети* входит знак небинарной связи, то связи между этим знаком и компонентами обозначаемой им связи задаются не с помощью пар инцидентности (на синтаксическом уровне), а явно с помощью явно вводимых связей принадлежности. Это означает, что в текстах *SC-кода* все *небинарные связи* представляются с помощью бинарных (сводятся к бинарным).

Ограничение 4. Каждая *семантическая сеть* и, в частности, каждая унифицированная семантическая сеть (*sc-текст*), которая описывается и, соответственно, обозначается в другой семантической сети (*метасети*), трактуется как семантически нормализованное множество, элементами которого являются все те и только те знаки, которые входят в состав описываемой семантической сети.

Ограничение 5. В рамках *SC-кода* четко задается и минимизируется алфавит элементов семантических сетей (Алфавит знаков, входящих в состав унифицированных семантических сетей, *Алфавит sc-элементов*). Указанный алфавит представляет собой семейство тех классов элементов унифицированных семантических сетей (*sc-элементов*), принадлежность *sc-элементов*

которым задается не с помощью явно вводимых связей принадлежности, а с помощью синтаксически задаваемых меток *sc-элементов*. При этом каждой такой метке взаимно однозначно соответствует свой синтаксически задаваемый класс *sc-элементов*. Семейство таких классов, т.е. *Алфавит sc-элементов*, включает в себя:

- Класс *sc-узлов*
- Класс *sc-ссылок на внешние информационные ресурсы*
- Класс *sc-ребер*
- Класс *sc-дуг общего вида*
- Класс *sc-дуг принадлежности или непринадлежности*
- Класс *стационарных константных sc-дуг принадлежности*

Ограничение 6. В рамках *SC-кода* четко задаются правила перехода *sc-элементов* из одного синтаксически задаваемого класса *sc-элементов* в другой при полном сохранении его семантики (т.е. при сохранении обозначаемого им денотата). Речь идет об изменении синтаксического типа (метки) *sc-элемента* при появлении определенного вида новой (дополнительной) информации об этом *sc-элементе*. Так, например, *sc-элемент* может перейти:

- из *Класса sc-узлов* в *Класс sc-ссылки*;
- из *Класса sc-узлов* в *Класс sc-ребер*;
- из *Класса sc-ребер* в *Класс sc-дуг общего вида*;
- из *Класса sc-дуг общего вида* в *Класс sc-дуг принадлежности или непринадлежности*;
- из *Класса sc-дуг принадлежности или непринадлежности* в *Класс стационарных константных sc-дуг принадлежности*.

Ограничение 7. Важнейшая особенность любого языка, текстами которого являются семантические сети, (в том числе и *SC-кода*) заключается в том, что все элементы (примитивы, атомарные фрагменты) семантических сетей (в т.ч. и *sc-элементы*) являются знаками. При этом в *SC-коде* сущности, обозначаемые *sc-элементами*, делятся на два вида:

- семантически нормализованные множества, элементами которых являются *sc-элементы*;
- "внешние" описываемые сущности, не являющиеся множествами.

Таким образом, каждый *sc-элемент* есть либо знак множества (но не просто множества, а множества, элементами которого являются *sc-элементы*), либо знак "внешней" описываемой сущности (сущности, которая множеством не является). Особо подчеркнем то, что *sc-элементов*, являющихся знаками множеств *sc-элементов* большинство:

- каждая связь трактуется как множество *sc-элементов*, обозначающих связываемые сущности;
- каждая структура трактуется как множество всех *sc-элементов*, входящих в эту структуру;
- каждое понятие трактуется как множество *sc-*

элементов, обозначающих сущности, являющиеся экземплярами этого понятия.

Следовательно, имеет место четкая теоретико-множественная трактовка таких сущностей, как связи, структуры и понятия. Это означает то, что *SC-код* имеет четкую базовую теоретико-множественную семантическую интерпретацию всех его *sc-элементов* – одни *sc-элементы* могут быть только элементами каких-либо множеств, а другие могут быть как элементами одних множеств, так и знаками других множеств.

Все *sc-элементы*, кроме знаков связей принадлежности или непринадлежности, можно разбить на следующие уровни иерархии:

- *первичные sc-элементы* – знаки "внешних" сущностей;
- *sc-элементы второго уровня* – знаки множеств, элементами которых являются только первичные *sc-элементы*;
- *sc-элементы третьего уровня* – знаки множеств, среди элементов которых есть по крайней мере, один *sc-элемент второго уровня*, но нет ни одного *sc-элемента* более высокого уровня;
- и так – до бесконечности.

Ограничение 8. Все *sc-элементы* имеют описанную ниже четкую базовую семантическую типологию:

sc-элемент

<= разбиение:*

- {
- *sc-константа*
= *sc-знак фиксированной (конкретной) сущности*
- *sc-переменная*
= *sc-знак нефиксированной (произвольной) сущности*
- }

<= разбиение:*

- {
- *первичный sc-элемент*
= *sc-знак внешней описываемой сущности*
- *sc-множество*
= *вторичный sc-элемент*
= *знак множества sc-элементов*
= *sc-метазнак*
= *sc-текст*
= *текст SC-кода*
= *знак sc-текста*
- }

sc-множество

<= разбиение:*

- {
- *sc-связка*
= *sc-элемент, обозначающий класс sc-элементов*
= *sc-знак множества sc-элементов, эквивалентных в том или ином смысле*
- *sc-структура*

= *sc-знак множества sc-элементов, в состав которого входят sc-связки или sc-структуры, связывающие эти sc-элементы*

- *sc-класс*

sc-связка

<= разбиение*:

- *бинарная sc-связка*
- *небинарная sc-связка*

<= разбиение*:

- *ориентированная sc-связка*
- *неориентированная sc-связка*

бинарная sc-связка

⊃ *квазибинарная sc-связка*
 ⊃ *sc-связка принадлежности или непринадлежности*

<= разбиение*:

- *бинарная неатомарная sc-связка*
- *sc-коннектор*
- *бинарная атомарная sc-связка*

<= разбиение*:

- *sc-ребро*
- *sc-дуга*

sc-связка принадлежности или непринадлежности

<= разбиение*:

- *sc-связка принадлежности*
- *sc-связка непринадлежности*
- *sc-связка нечеткой принадлежности*

<= разбиение*:

- *стационарная sc-связка принадлежности или непринадлежности*
- *нестационарная sc-связка принадлежности или непринадлежности*

sc-структура

<= разбиение*:

- *тривиальная sc-структура*
 = *sc-структура, не содержащая sc-связок между своими элементами*
- *двухуровневая sc-структура*
 = *sc-структура, содержащая первичные элементы (вершины) и связи, связывающие только первичные элементы*

- *иерархическая sc-структура*
 = *sc-структура, среди элементов которой имеются связи или структуры, связывающие другие связи или структуры этой же структуры*

<= разбиение*:

- *связная sc-структура*
- *несвязная sc-структура*

sc-класс

<= разбиение*:

- *класс первичных sc-элементов*
- *sc-отношение*
 = *класс sc-связок*
- *класс sc-структур*
- *класс sc-классов*
 = *sc-метакласс*
- *класс sc-элементов разного структурного типа*

⊃ *sc-понятие*
 = *sc-концепт*
 = *ключевой класс sc-элементов*
 = *sc-знак ключевого класса sc-элементов*

Достоинствами SC-кода являются следующие его свойства:

- Все основные семантические связи между текстами (семантическая эквивалентность, семантическое включение, семантическое пересечение) в SC-коде становятся теоретико-множественными (равенство, включение, пересечение множеств).
- Неограниченная возможность перехода от sc-текстов к sc-метатекстам, содержащим знаки описываемых sc-текстов.
- Тексты SC-кода (в том числе и те тексты, sc-знаки которых явно вводятся в рамках соответствующих им метатекстов) могут быть иерархическими структурами, имеющими любое число уровней иерархии, поскольку sc-элемент может обозначать множество, состоящее из любых sc-элементов (в т.ч. и из sc-элементов, обозначающих любые множества других sc-элементов). В отличие от этого, например, такие традиционные структуры, как алгебраические системы, являются трехуровневыми:
- На первом уровне – элементы носителя (основного множества) алгебраической системы;
- На втором уровне – кортежи, элементами которых являются элементы носителя;
- На третьем уровне – отношения, элементами которых являются указанные кортежи.

Формализация знаний на основе SC-кода предполагает теоретико-множественную интерпретацию всех sc-элементов, не являющихся

знаками "внешних" сущностей. Все такие *sc-элементы* являются знаками множеств *sc-элементов* и необходимо, прежде всего, уточнять то, какие *sc-элементы* являются знаками этих множеств. При этом совсем не обязательно, чтобы все эти *sc-элементы* были представлены в текущем состоянии *sc-памяти*.

3. Смысловое пространство

Многие проблемы искусственного интеллекта будут решены, если мы поймем, что главными объектами формализации в искусственном интеллекте являются не базы знаний интеллектуальных систем и не машины обработки знаний этих интеллектуальных систем, а *смысловое пространство* и глобальная интегрированная машина обработки знаний, находящихся в указанном смысловом пространстве.

Понятие *смыслового пространства* и представление этого пространства в виде унифицированной *семантической сети* имеют ключевое значение для решения таких задач, как:

- обеспечение семантической совместимости *баз знаний* различных интеллектуальных систем;
- интеграция различного вида знаний;
- обеспечение компонентного (модульного) проектирования *баз знаний* интеллектуальных систем путем использования ранее разработанных и апробированных знаний, заимствованных из *смыслового пространства*.

смысловое пространство

- = *семантическое пространство*
- = *бесконечная семантическая сеть*, включающая в себя все известные понятия (концепты) и все элементы (экземпляры) этих понятий, а также знаки всех известных структур и связей вместе с их элементами (компонентами)
- = *смысловой универсум*
- = *смысловое пространство знаний*

Смысловое пространство – это результат интеграции всевозможных знаний, представленных в смысловой форме (т.е. в виде *семантических сетей*). Таким образом, смысловое пространство представляет собой бесконечную семантическую сеть, содержащую всевозможные (и, как минимум, все известные, добытые к текущему моменту) знания. Очевидно, что смысловое пространство должно быть структурировано. Это означает, что в нем должны быть явно выделены и обозначены самые различные знания, а также должны содержаться *метазнания*, описывающие (специфицирующие) эти выделенные знания.

Очевидно также, что важнейшим направлением повышения эффективности современной научно-технической деятельности является преобразование современного *глобального информационного пространства* в структурированное *смысловое пространство*, в основе которого лежит универсальная смысловая формализация научно-

технических знаний, носящая интернациональный характер. Существенно подчеркнуть, что предложенный нами *SC-код* дает возможность не просто представлять смысл любых научно-технических знаний, но и делать это поэтапно, сочетая в рамках одной и той же унифицированной семантической сети как формализованные знания, так и пока неформализованные знания, представленные, в том числе, в виде естественно-языковых текстов, являющихся содержимым файлов, хранимых в файловой памяти и имеющих соответствующие *sc-узлы* (*sc-ссылки*), которые обозначают указанные файлы и хранятся в соответствующей смысловой (семантической) памяти. Так, например, таким способом можно поэтапно перейти от современной Википедии к семантически структурированной Википедии, постепенно переводя накопленные в ней естественно-языковые тексты на формальный смысловой (семантический) язык – *SC-код*.

Подчеркнем, что разным языкам *семантических сетей* (т.е. разным вариантам уточнения синтаксиса и семантики *семантических сетей*) с формальной точки зрения соответствуют разные *смысловые пространства*, которые являются *семантически эквивалентными* друг другу *бесконечными семантическими сетями*.

Введем еще несколько понятий, связанных с понятием *смыслового пространства*.

SC-модель смыслового пространства

- = *Смысловое пространство, представленное в SC-коде*
- = *SC-текст смыслового пространства*
- = *Глобальный sc-текст*
- = *Унифицированная модель смыслового универсума*
- = *SC-универсум*
- = *Унифицированная модель смыслового пространства*
- = *Глобальная унифицированная семантическая сеть*
- = *Результат интеграции всевозможных текстов SC-кода*
- <= объединение*:
- SC-код*
- ∈ *бесконечное sc-множество*
- = *бесконечный sc-текст*
- ∈ *смысловое пространство*

глобальная база знаний

- = *конечная информационная конструкция, являющаяся формальным представлением всех знаний, накопленных человечеством к текущему моменту*
- ∈ *максимальный класс семантически эквивалентных информационных конструкций*
- ∈ *конечное sc-множество*
- = *конечный sc-текст*

смысловая модель глобальной базы знаний

- = *глобальная база знаний, представленная в виде семантической сети*

= семантическая модель глобальной базы знаний
⊆ глобальная база знаний

SC-модель глобальной база знаний

= Глобальная база знаний, представленная в SC-коде
= Унифицированная смысловая модель глобальной базы знаний
= Унифицированная семантическая модель глобальной базы знаний
∈ глобальная база знаний
∈ смысловая модель глобальной база знаний

база знаний

= конечная информационная конструкция, являющаяся формальным представлением всех знаний, достаточных для функционирования некоторой интеллектуальной системы и хранимых в памяти этой системы
= локальная база знаний
= формальная модель, интегрирующая все знания, хранимые в памяти интеллектуальной системы

Для обеспечения интегрируемости баз знаний различных интеллектуальных систем необходимо эти частные базы знаний разрабатывать в контексте интегрированной глобальной базы знаний.

смысловая модель базы знаний

= база знаний, представленная в виде семантической сети
⊆ база знаний

sc-модель база знаний

= база знаний, представленная в SC-коде
= Унифицированная смысловая модель базы знаний
⊆ смысловая модель база знаний

4. Типология знаний и отношения, заданные на множестве знаний

Структуризация *смыслового пространства* задаётся:

- Типологией знаний, входящих в состав *смыслового пространства*, путём явного введения понятий, которые обозначают различные классы (виды) знаний, представленных семантическими сетями, являющимися фрагментами *смыслового пространства*;
- Семейством *отношений*, заданных на множестве выделенных знаний, являющихся фрагментами *смыслового пространства*.

Подчеркнём, что не существует ни одного знания и, соответственно, ни одного вида знаний, которые бы не входили в состав *смыслового пространства*. Это значит, что рассматриваемое нами смысловое пространство носит универсальный характер. Подчеркнём также, что структуризацию *смыслового пространства* удобнее всего проводить в рамках Унифицированной модели (SC-модели) *смыслового*

пространства, поскольку SC-код является весьма удобным средством перехода от представления знаний к представлению *метазнаний*, то есть знаний, описывающих свойства других знаний.

Для структуризации *смыслового пространства* ключевую роль играют такие виды знаний, как

- предметные области
- онтологии (спецификации предметных областей)
- семантические окрестности

Предметная область – это важнейший вид знаний, входящих в состав *смыслового пространства*. Каждая предметная область фокусирует внимание на описание связей соответствующего класса объектов исследования. Каждый знак, входящий в состав базы знаний, должен принадлежать (входить в состав) хотя бы одной предметной области, выполняя в ней ту или иную роль. Каждой предметной области можно поставить в соответствие:

- множество семантических окрестностей, описывающих объекты исследования этой предметной области;
- семейство различного вида онтологий, описывающих свойства понятий этой предметной области.

Предметные области являются основой структуризации *смыслового пространства*, средством локализации, фокусирования внимания на свойствах наиболее важных классов описываемых сущностей, которые становятся классами объектов исследования в *предметных областях*.

Предметная область – структура, фокусирующая (локализирующая, концентрирующая) внимание только на заданном классе объектов исследования и только в заданном "ракурсе", который формально задаётся набором отношений, заданных на исследуемых объектах.

При накоплении больших объёмов знаний появляется огромное количество самых различных понятий (концептов) и возникает проблема структуризации и систематизации этого многообразия понятий. Для этого все накапливаемые знания делятся на связанные между собой *предметные области*, каждой из которых соответствует свой набор *понятий*, каждое из которых в рамках этой предметной области выполняет определенную роль (либо является классом объектов исследования, либо является отношением, заданным на множестве объектов исследования, и так далее).

Предметная область – структура, в состав которой входят:

- (1) основные исследуемые (описываемые) объекты – первичные и вторичные;
- (2) различные классы исследуемых объектов;
- (3) различные связи, компонентами которых

являются исследуемые объекты (как первичные, так и вторичные), а также, возможно, другие такие связи – то есть связи (как и объекты исследования) могут иметь различный структурный уровень;

- (4) различные классы указанных выше связей (то есть отношений);
- (5) различные классы объектов, не являющихся ни объектами исследования, ни указанными выше связками, но являющихся компонентами этих связей.

Предметная область – это результат интеграции (объединения) частичных семантических окрестностей, описывающих все исследуемые сущности заданного класса и имеющих одинаковый (общий) предмет исследования (то есть один и тот же набор отношений, которым должны принадлежать связи, входящие в состав интегрируемых семантических окрестностей).

Необходимо чётко отличать:

- (1) саму предметную область, которая практически всегда бесконечна;
- (2) тот фрагмент предметной области, который в текущий момент хранится в памяти и элементы которого явно связаны парами принадлежности со знаком соответствующей предметной области. Подчеркнём, что указанный фрагмент предметной области всегда должен быть конечным множеством;
- (3) различного вида онтологии указанной предметной области.

Кроме общего понятия *предметной области*, введём также уточнение этого понятия на основе *SC-кода*, то есть уточнение того, как предметные области будут представлены в рамках *SC-модели смыслового пространства*.

sc-модель предметной области

= *sc-текст предметной области*

= *предметная область, представленная в SC-коде*

= *sc-граф предметной области*

= *уточнение понятия предметной области на основе SC-кода*

= *предметная область, представленная в рамках SC-модели смыслового пространства*

⊂ *бесконечное sc-множество*

⊂ *предметная область*

⊂ *sc-знание*

Если *предметную область* рассматривать как часть *смыслового пространства*, то её ролевая структура будет выглядеть существенно сложнее, чем просто класс объектов исследования с семейством заданных на нём отношений.

Нам важно учитывать не только роли элементов в рамках самой *предметной области*, но и их соотношение со *смысловым пространством*. Так, например, важно знать, все ли элементы (экземпляры) каждого понятия (концепта) предметной области входят в состав этой

предметной области, то есть являются её элементами.

Предметная область – это бесконечная знаковая структура, в которой явно указываются роли тех знаков, обозначающие понятия (концепты), экземпляры (элементы) которых входят в состав этой же *предметной области*. Роль каждого понятия в рамках предметной области определяется по двум признакам:

- по уровню исследовательского внимания;
- по структурному уровню.

По уровню исследовательского внимания понятия в рамках предметной области могут выполнять следующие роли:

- быть *максимальным классом исследуемых объектов* в рамках данной предметной области;
- быть *классом исследуемых объектов* в рамках данной предметной области (не обязательно максимальным классом);
- быть *исследуемым классом неисследуемых объектов* (например, связей, структур или даже классов, состоящих из исследуемых объектов);
- быть *неосновным исследуемым классом* (неосновным понятием, вводимым в рамках данной предметной области);
- быть *неисследуемым классом* (классом дополнительных, вспомогательных сущностей, классом, который исследуется в другой предметной области).

По структурному уровню понятия, рассматриваемые в рамках предметной области, могут выполнять следующие роли:

- быть *максимальным классом первичных элементов* данной предметной области – этот класс также должен быть либо максимальным классом исследуемых объектов, либо подклассом этого класса;
- быть *отношением данной предметной области*, заданным над первичными или вторичными элементами этой предметной области, то есть являющимся классом связей, входящих в состав указанной предметной области;
- быть *классом структур*, целиком входящих в состав данной предметной области;
- быть *классом классов*, целиком (полностью) входящих в состав предметной области;
- быть *классом сущностей, принадлежащих разным структурным уровням* данной предметной области – в каждый такой класс могут одновременно входить и первичные элементы, и связи, и структуры, и классы.

Кроме того, для каждого *неисследуемого класса* элементов предметной области должно быть дополнительно уточнено теоретико-множественное соотношение этого класса с множеством элементов, входящих в состав предметной области:

- быть *неисследуемым классом*, который является подмножеством данной предметной области;

- быть *неисследуемым классом*, который строго пересекается с данной предметной областью.

Подчеркнём, что все исследуемые классы предметной области (в том числе и классы непосредственно исследуемых объектов) являются подмножествами этой предметной области.

Множество предметных областей развивается на два класса:

- статические (стационарные) предметные области;
- динамические (нестационарные) предметные области.

Кроме того, можно говорить о первичных и вторичных предметных областях (по аналогии с первичными и вторичными sc-элементами). Можно также говорить о классе *рефлексивных предметных областей*, в каждой из которых хотя бы некоторые её ключевые элементы (исследуемые понятия) либо она сама являются объектами исследования (экземплярами исследуемых классов) в этой же предметной области.

Но во всём бесконечном множестве предметных областей особо следует выделить конечное число предметных областей, которые назовём общими *предметными областями* и которые являются основной для построения различного вида спецификаций всевозможных предметных областей, а также основой для построения самих предметных областей.

К числу *общих предметных областей* относятся:

- *Общая предметная область предметных областей*
- *Общая предметная область знаков* (в том числе sc-элементов)
- *Общая предметная область множеств* (в том числе sc-множеств)
- *Общая предметная область отношений*
- *Общая предметная область структур* (в том числе sc-структур)
- *Общая предметная область чисел, числовых отношений и структур*
- *Общая предметная область параметрических моделей* (как количественных, так и качественных)
- *Общая предметная область онтологий*
- *Общая предметная область семантических окрестностей*
- *Общая предметная область знаний* (sc-знаний)
- и так далее

Онтология – вид знаний, каждое из которых является спецификацией (описанием свойств) соответствующей предметной области, ориентированной на описание свойств и взаимосвязей понятий, входящих в состав указанной предметной области.

Из общей интегрированной онтологии предметной области предметной области можно выделить следующие частные онтологии:

- *структурные спецификации предметных областей*, в которых описываются роли понятий, входящих в состав предметных областей, а также связи специфицируемых предметных областей с другими предметными областями;
- *теоретико-множественные онтологии* предметных областей, в которых описываются теоретико-множественные связи между понятиями специфицируемых предметных областей, включая связи между классами объектов исследования и доменами или областями определения отношений, входящих в состав специфицируемых предметных областей;
- *терминологические онтологии*, описывающие всевозможные термины, соответствующие понятиям специфицируемых предметных областей, а также правила построения терминов, которые соответствуют экземплярам указанных понятий;
- *логические онтологии*, представляющие собой формальные теории, высказывания которых интерпретируются на специфицируемых предметных областях и описывают свойства понятий, входящих в состав этих предметных областей;
- *онтологии информационных программ и задач*, решаемых в рамках специфицируемых предметных областей.

Каждому виду онтологий соответствует своя общая предметная область, интегрирующая все онтологии этого вида:

- структурным спецификациям (онтологиям) предметных областей соответствует *Общая предметная область предметных областей*
- теоретико-множественным онтологиям предметных областей соответствует *Общая предметная область множеств*
- логическим онтологиям предметных областей соответствует *Общая предметная область формальных теорий и высказываний*
- и так далее.

Очевидно, что всем *общим предметным областям* (в том числе и тем, которые соответствуют различным видам *онтологий*) соответствуют свои онтологии, специфицирующие эти общие предметные области. Онтологии общих предметных областей будем называть онтологиями верхнего уровня. К числу таких онтологий относятся:

- *Онтология Общей предметной области знаков* (в частности, sc-элементов)
- *Онтология Общей предметной области терминов* (в частности, идентификаторов sc-элементов)

- *Онтология Общей предметной области информационных конструкций* (в т.ч. файлов, сохраняющих эти конструкции)
- *Онтология Общей предметной области множеств* (в частности, sc-множеств)
- *Онтология Общей предметной области структур* (графовых структур, sc-структур)
- *Онтология Общей предметной области предметных областей*
- *Онтология Общей предметной области онтологий*
- *Онтология Общей предметной области формальных теорий и высказываний*
- *Онтология Общей предметной области семантических окрестностей*
- *Онтология Общей предметной области знаний* (в частности, sc-знаний)
- *и так далее.*

Семантическая окрестность заданного знака (который будем называть ключевым элементом этой семантической окрестности) – это текст, описывающий некоторые свойства той сущности, которая обозначается указанным ключевым знаком.

Другими словами, *семантическая окрестность* – это некоторого вида спецификация указанной описываемой сущности.

Точно так же, как и для онтологий, можно говорить о многообразии видов *семантических окрестностей*, которые определенным образом коррелируются с типологией специфицируемых сущностей.

В частности, можно выделить следующие виды семантических окрестностей:

- *определение* понятия
- *пояснение* понятия
- *примечание* (описание различных свойств специфицируемой сущности)
- *описание преимуществ* (достоинств)
- *описание недостатков*
- *сравнительный анализ*
- *типичная семантическая окрестность* (пример типичного использования описываемого понятия)
- *описание декомпозиции* заданной сущности
- *теоретико-множественная семантическая окрестность*
- *логическая семантическая окрестность* (семейство высказываний, описывающих свойства данного понятия)

Расширение *семантической окрестности* (т.е. повышение уровня детализации заданного знака) может осуществляться (1) как путем расширения числа указываемых свойств (характеристик) и

связей заданного знака – назовем это увеличением валентности специфицируемого знака, (2) так и путем включения в состав расширяемой семантической окрестности семантических окрестностей тех знаков, которые смежны специфицируемому (центральному) знаку заданной (расширяемой) семантической окрестности – назовем это увеличением радиуса заданной семантической окрестности в заданном направлении (т.е. для заданного смежного знака).

Теперь перейдем к рассмотрению отношений, заданных на множестве знаний.

Прежде всего, рассмотрим отношения, связывающие между собой *предметные области*. Важнейшим из них является отношение *частная предметная область**, с помощью которого задается иерархия предметных областей путем перехода от менее детального к более детальному рассмотрению соответствующих классов объектов исследования.

частная предметная область*

- = *быть частной предметной областью**
- = *предметная область, детализирующая описание одного из классов объектов исследования другой (более общей) предметной области**
- = *предметная область, максимальный класс объектов исследования которой является подмножеством максимального класса объектов исследования другой (более общей) предметной области**

Кроме этого, *предметные области* могут быть связаны между собой следующим образом:

- предметные области могут иметь совпадающие максимальные классы объектов исследования, но разные предметы исследования, в частности, в одной предметной области могут рассматриваться внешние связи исследуемых объектов, а в другой – внутренняя структура этих объектов;
- максимальный класс объектов исследования одной предметной области может быть булеаном (семейством всевозможных подмножеств) максимального класса объектов исследования другой предметной области. Например, Предметная область геометрических фигур и Предметная область геометрических точек;
- максимальный класс объектов исследования одной предметной области может быть семейством всевозможных структур, являющихся фрагментами другой предметной области. Например, Предметная область конфигураций геометрических фигур и Предметная область геометрических фигур;
- понятия одной предметной области могут быть объектами исследования в другой предметной области;

- каждое неисследуемое понятие одной предметной области должно быть исследуемым понятием в некоторой другой предметной области;
- предметные области могут быть изоморфными и гомоморфными;
- и так далее.

Связь между предметной областью и ее онтологией задается отношением *быть онтологией**.

5. Особенность представления динамических предметных областей и их онтологий

В базе знаний интеллектуальных систем динамика представлена в следующих аспектах:

- динамика самого внешнего описываемого мира;
- динамика процесса автоматической обработки базы знаний (в том числе процесса решения пользовательских задач);
- динамика взаимодействия с пользователями и другими компьютерными системами;
- динамика совершенствования базы знаний и интеллектуальной системы в целом усилиями разработчиков;
- динамика проекта, направляемого на совершенствование базы знаний и интеллектуальной системы в целом (то есть динамика организации управления указанной деятельностью).

Для того, чтобы в памяти интеллектуальной системы можно было хранить информацию не только о текущем состоянии некоторой динамической предметной области, но также и о прошлых и будущих (прогнозируемых) её состояниях, необходимо от *исходной sc-модели динамической предметной области*, отражающей только меняющееся текущее состояние этой предметной области, перейти к *ситуационно-событийной sc-модели динамической предметной области*, в которой описываются (с любой требуемой степенью детализации) не только ситуации и события, происходящие в настоящее время (в текущий момент), но и ситуации и события, происходившие в прошлом времени, а также ситуации и события, прогнозируемые или планируемые в будущем времени.

Если ввести *Общую предметную область ситуационно-событийных sc-моделей динамических предметных областей*, то очевидно, что объектами исследования в этой *предметной области* будут всевозможные *ситуации* и *события*.

Каждой *предметной области* (не только динамической) можно поставить в соответствие в общем случае несколько предметных областей, описывающих разные виды *деятельности* в рамках заданной предметной области. Так, например,

каждой предметной области можно поставить в соответствие

- предметную область, описывающую *деятельность* субъектов по совершенствованию (эволюции) исходной предметной области;
- предметную область, описывающую *деятельность* субъектов (агентов) по решению различных информационных задач в рамках исходной предметной области (в том числе и по решению задач планирования поведения, если исходная предметная область является динамической).

Если мы имеем динамическую предметную область, объектами исследования которой являются, например, автомобили, то ей можно поставить в соответствие следующие предметные области, описывающие:

- *деятельность* по проектированию и изготовлению автомобилей;
- *деятельность* по управлению автомобилями (вождению автомобилей);
- *деятельность* по поддержанию основных эксплуатационных характеристик и ремонту автомобилей (в том числе их мониторингу и профилактике);
- *штурманскую деятельность* по организации индивидуального дорожного движения (по выбору маршрута, времени, скорости, остановок);
- *деятельность* по организации коллективного дорожного движения (на трассе, в населённом пункте);
- *деятельность*, связанную с дорожно-транспортными происшествиями и нарушениями правил дорожного движения.

Таким образом, предметных областей, описывающих самые различные виды деятельности, существует достаточно много. Но существует также и *Общая предметная область деятельности*, описывающая общие свойства любой деятельности в самых различных *предметных областях* (прежде всего, в динамических).

К числу исследуемых (ключевых) понятий (концептов) *Общей предметной области деятельности* относятся такие понятия, как *действие** (акция*, воздействие*), *субъект* (агент) и другие.

6. Универсальный семантический код В.В. Мартынова как онтология общей предметной области деятельности

Введение *Общей предметной области деятельности*, которая описывает субъектно-объектные отношения в различных предметных областях, предполагает выявление общих принципов организации субъектно-объектных связей в ходе выполнения различной деятельности, абстрагируясь от несущественных деталей

(несущественных для организации деятельности), обусловленных спецификой различных предметных областей, в рамках которых деятельность осуществляется. Именно это и было основной задачей, которую решал В. В. Мартынов, разрабатывая Универсальный семантический код (УСК). Рассматривая соотношение УСК с проблемой смыслового представления знаний в памяти интеллектуальных систем, сделаем ряд примечаний.

Примечание 1. Идеи, лежащие в основе УСК, существенно приблизили практическое решение проблемы смыслового представления знаний и, самое главное, четко сформулировали то, какими свойствами должно обладать смысловое представление знаний, и то, почему именно смысловое представление знаний является важнейшим фактором повышения эффективности интеллектуальных систем.

Примечание 2. В разработанных В. В. Мартыновым версиях УСК акцентируется внимание на рассмотрении субъектно-объектных прагматических свойств описываемых объектов.

Любой объект *X* может быть описан в системе УСК, если его описание будет ответом на вопрос «Для чего *X*?», а не «Что такое *X*?». Так, например, в тексте УСК термин «винт» определяется как «средство становления совокупности частей предметов» (то есть как ответ на вопрос «Для чего *X*?»). Определение винта как «металлического стержня с резьбой на одном конце и граненной головкой на другом» (т.е. как ответ на вопрос «Что такое *X*?») в пределах УСК осуществлено быть не может. При необходимости «реального» определения семантические множители УСК дополняются дескрипторами информационного языка данной предметной области (в нашем случае машиноведения). [Мартынов, 1974, с.186-187]

Таким образом, основой УСК является *Онтология Общей предметной области деятельности*, которая обеспечивает унификацию описания деятельности в любых *предметных областях* и которая занимает важнейшее место среди *онтологий верхнего уровня*.

Примечание 3. В. В. Мартынов не считал цепочки УСК единственно возможной формой представления текстов УСК и рассматривал варианты их нелинейного (графового) представления в виде семантических сетей [Мартынов, 1980]. Но для этого необходимо было перейти от классических графовых структур к структурам, в которых компонентами ребер и дуг могут быть не только вершины графовой структуры, но также и дуги или ребра этой же графовой структуры.

Таким образом, обладая рассмотренными выше достоинствами и обеспечивая конструктивное представление знаний как фрагментов (подмножеств) смыслового пространства,

семантические сети также являются весьма комфортной формой представления и текстов УСК.

Заключение

В настоящее время не только искусственный интеллект, но информатика в целом представляют собой эклектическое собрание, вавилонское столпотворение языков самого различного назначения, самых различных моделей обработки информации, самых различных платформ и средств технической реализации этих моделей. При этом серьезному фундаментальному анализу не подвергается ни исследование совместимости этих языков моделей и средств, ни разработка общей целостной картины, обеспечивающей интеграцию указанных языков, моделей и средств и четкое разграничение их смысловой (семантической) трактовки от многообразия форм их воплощения. Фактически это говорит не столько о кризисе современного состояния компьютерных наук, сколько о востребованности превращения информатики в целостную фундаментальную науку. И основой такого переосмысления информатики является формализация смысла, к чему и призывал В.В. Мартынов в своих работах.

Библиографический список

- [Gomez-Perez и др., 2004] Gomez-Perez, A. Ontological Engineering/ A. Gomez-Perez, M. Fernando-Lopez, O. Corcho – Springer – Ferlag, 2004.
- [Jurafsky, 2009] Jurafsky, D. Speech and Language Processing/ D. Jurafsky, J. Martin // An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 2nd Edition – Pearson Prentice Hall, 2009, 988 pp. (Stanford University and University of Colorado at Boulder).
- [Nirenburg и др., 2004] Nirenburg, R. Ontological Semantics/ S. Nirenburg, V. Raskin – Cambridge, MA: MIT Press, 2004.
- [Аджиев, 2003] Аджиев, А.С. Всероссийский Web-портал математических ресурсов / А.С. Аджиев, А.Н. Бездушный // Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды пятой всероссийской науч. конф. RCDL'2003 (С.-Петербург, 29-31 октября 2003 г.) – СПб: НИИ Химии СПбГУ, 2003. – С. 283-291.
- [Анно, 1981] Анно, Е.И. Описание грамматики русского языка с помощью окрестностей /Е.И. Анно // НТИ. Сер. 2. 1981. № 10. – С. 17-23.
- [Арутюнова, 2003] Арутюнова, Н.Д. Предложение и его смысл /Н.Д. Арутюнова – М., 2003.
- [Башмаков и др., 2005] Башмаков, А.И. Интеллектуальные информационные технологии: Учебное пособие / А.И. Башмаков, И.А. Башмаков. – М.: МГТУ им. Н.Э. Баумана, 2005. – 304 с.
- [Белнап и др., 1981] Белнап, Н. Логика вопросов и ответов /Н. Белнап, Т. Стил – М.: Прогресс, 1981.
- [Болотова, 2012] Болотова, Л.С. Системы искусственного интеллекта: модели и технологии, основанные на знаниях / Л.С. Болотова – М.: Финансы и статистика, 2012. – 664 с.
- [Большаков, 2011] Большаков, И.А. Большой электронный словарь как политематический справочник и формирователь запросов к Интернету/И.А. Большаков, А.Ф. Гельбух // Труды международной конференции «Диалог 2011» – М.: Изд-во РГТУ, 2011.
- [Варосян и др., 1982] Варосян, С.О. Неметрическая пространственная логика /С.О. Варосян, Д.А. Пospelov // Техническая кибернетика. - 1982. № 5. – С. 86-99.
- [Виленкин и др., 1974] Виленкин, Н.Я. Понятия математики и объекты науки / Н.Я. Виленкин, Ю.А. Шрейдер // Вопросы философии. – 1974. № 2.
- [Виноград, 1976] Виноград, Т. Программа, понимающая естественный язык / Т. Виноград - М.: Мир, 1976. – 294 с.

- [Гаврилова и др., 2000] Гаврилова, Т.А. Базы знаний интеллектуальных систем / Т. А. Гаврилова, В.Ф. Хорошевский. – СПб.: Питер, 2000. – 384 с.
- [Гаврилова, 2008] Гаврилова, Т. А. Визуальные методы работы со знаниями: попытка обзора / Т. А. Гаврилова, Н. А. Гулякина // Искусственный интеллект и принятие решений, 2008, № 1, С. 15-21
- [Гастев, 1975] Гастев, Ю.А. Гомоморфизмы и модели. (Логико-алгебраические аспекты моделирования) / Ю.А. Гастев – Изд-во «Наука», М. 1975. – 152 с.
- [Добров и др., 2004] Добров, Б.В. Методы и средства автоматизированного проектирования практической онтологии / Б.В. Добров, Н.В. Лукашевич, О.А. Невзорова, Б.Е. Федунцов // Известия РАН. Теория и системы управления. – 2004. №2. – С. 58-68.
- [Добров и др., 2009] Добров, Б.В. Онтологии и тезаурусы: модели, инструменты, приложения. Программа, понимающая естественный язык / Б.Н. Добров, В.В. Иванов, Н.В. Лукашевич, В.Д. Соловьев – Изд-во: Интернет-университет информационных технологий – ИНТУИТ.ру, БИНОМ. -2009. – 294 с.
- [Загорюлько, 1988] Загорюлько, Ю.А. Технология конструирования средств обработки знаний на основе семантических сетей. Средства спецификации и настройки / Ю. А. Загорюлько. - Новосибирск, 1988.
- [Загорюлько и др., 2009] Загорюлько Ю.А. Технология разработки порталов научных знаний // Программные продукты и системы. – 2009. – № 4. – С.25-29.
- [Кандрашина и др., 1989] Кандрашина, Е.Ю. Представление знаний о времени и пространстве в интеллектуальных системах / Е. Ю. Кандрашина, Л. В. Литвинцева, Д. А. Поспелов - М.: Наука, 1989.
- [Карпов, 1992] Карпов, В.А. Язык как система. / В.А. Карпов – Минск: Вышэйшая школа, 1992. – 302 с.
- [Клещев и др., 2001а] Клещев А.С., Артемьева И.Л. Математические модели онтологий предметных областей. Часть 1. Существующие подходы к определению понятия "онтология" // НТИ. Серия 2 "Информационные процессы и системы", 2001, № 2
- [Клещев и др., 2001б] Клещев А.С. Артемьева И.Л. Математические модели онтологий предметных областей. Часть 2. Компоненты модели // НТИ. Серия 2 "Информационные процессы и системы", 2001, № 3
- [Клещев и др., 2001с] Клещев А.С. Артемьева И.Л. Математические модели онтологий предметных областей. Часть 3. Сравнение разных классов моделей онтологий // НТИ. Серия 2 "Информационные процессы и системы", 2001, № 4
- [Лапшин, 2010] Лапшин, В.А. Онтологии в компьютерных системах / В.А. Лапшин - М.: Научный мир, 2010. – 224 с.
- [Лукашевич, 2011] Лукашевич, Н.В. Тезаурусы в задачах информационного поиска / Н.В. Лукашевич. – М.: Изд-во Московского университета, 2011. – 512 с.
- [Мартынов и др., 1980] Мартынов В.В., Принципы построения нелинейного графового кода / В.В. Мартынов, В.П. Сидоренко // Семантический код в линейном и нелинейном представлении Мн. 1980.
- [Мартынов, 1974] Мартынов, В.В. Семиологические основы информатики / В.В. Мартынов - Мн.: Наука и техника, 1974. – 192 с.
- [Мартынов, 1977] Мартынов, В.В. Универсальный семантический код (Грамматика. Словарь. Тексты) / В.В. Мартынов – Мн.: Наука и техника, 1977. – 191 с.
- [Мартынов, 1984] Мартынов, В.В. Универсальный семантический код: УСК-3 / В.В. Мартынов – Мн.: Наука и техника, 1984. – 132 с.
- [Мельчук, 1974] Мельчук И.А., Опыт теории лингвистических моделей «смысл->текст» / И.А. Мельчук - М. 1974.
- [Молокова, 1992] Молокова, О.С. Методология анализа предметных знаний / О. С. Молокова. // Новости искусственного интеллекта. - 1992. – № 3. - С.11-60.
- [Онтологическое моделирование, 2011] Онтологическое моделирование // Труды Второго Симпозиума «Онтологическое моделирование», г. Казань, 11-12 октября 2010 г., Ред. Калининченко Л.А. - М.: ИПИ РАН, 2011.
- [Осипов, 1990] Осипов, Г.С. Построение моделей предметных областей. Неординарные семантические сети / Г. С. Осипов. // Известия АН СССР. Техническая кибернетика. – 1990. - № 5.
- [Плесневич, 2008] Плесневич, Г.С. Бинарные модели знаний / Г. С. Плесневич // Труды Международных научно-технических конференций «Интеллектуальные системы» (AIS'08) и «Интеллектуальные САПР» (CAD-2008). Научное издание в 4-х томах. – М : Физматлит, 2008, Т.2. – С. 424 – 135-146.
- [Поспелов, 1986а] Поспелов, Д.А. Представление знаний. Опыт системного анализа / Д. А. Поспелов. // Системные исследования. Методологические проблемы. Ежегодник. - М.: Наука, 1986. - с. 83-102.
- [Поспелов, 1986б] Поспелов, Д.А. Ситуационное управление. Теория и практика / Д. А. Поспелов. – М : Наука, 1986.
- [Рубашкин и др., 2005] Рубашкин, В.Ш. Онтология: от натурфилософии к научному мировоззрению и инженерии знаний в компьютерных системах / В.Ш. Рубашкин, Д.Г. Лахути // Вопросы философии. – 2005. № 1. - С. 64-81.
- [Рубашкин, 2012] Рубашкин, В.Ш. Онтологическая семантика. Знания. Онтологически ориентированные методы информационного анализа текстов/В.Ш. Рубашкин – М.: ФИЗМАТЛИТ, 2012. – 348 с.
- [Скороходько, 1983] Скороходько, Э.Ф. Семантические сети и автоматическая обработка текста. / Э. Ф. Скороходько. – Киев: Наук. думка, 1983. - 220 с.
- [Смирнов и др., 2002] Смирнов, А.В. Онтологии в системах искусственного интеллекта: способы построения и организации / А.В. Смирнов, М.П. Пашкин, Н.Г. Шилов, Т.В. Левашова // Новости искусственного интеллекта. – 2002. №1(49). – С. 3-13.
- [Смирнов, 2012] Смирнов, С.В. Онтологическое моделирование в ситуационном управлении/ С.В.Смирнов // Онтология проектирования, 2012, №2. – С16-24
- [Соловьев и др., 2006] Соловьев, В.Д. Онтологии и тезаурусы: Учебное пособие / В.Д. Соловьев, Б.В. Добров, В.В. Иванов, Н.В. Лукашевич – Казань, Москва: Казанский государственный университет, МГУ им. М.В. Ломоносова, - 2006. – 157 с.
- [Тондл, 1975] Тондл, Л. Проблемы семантики. / Л. Тондл. – Изд-во «Прогресс», М., 1975. – 486 с.
- [Цейтин, 1985] Цейтин, Г.С. Программирование на ассоциативных сетях / Г.С. Цейтин // ЭВМ в проектировании и производстве. Вып. 2. - Л.: Машиностроение, 1985.
- [Шенк, 1980] Шенк, Р. Обработка концептуальной информации / Р. Шенк. – Москва: Энергия, 1980.

STRUCTURIZATION OF SENSE SPACE

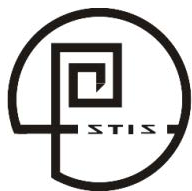
Golenkov V.V., Guliakina N.A.

*Belarusian State University of Informatics and
Radioelectronics, Minsk, Republic of Belarus*

golenkov@bsuir.by

guliakina@bsuir.by

The article is devoted to an approach to information representation in the intelligent systems, based on semantic networks in the aspect of appealing to the sense of represented information. Main attention is paid to the considering of such notions, as subject domain and ontology of the subject domain, their classification and traits of the most typical classes. One of the sections is devoted to the integration of semantic networks with Universal semantic code, proposed by V.V. Martynov.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС ИНТЕЛЛЕКТУАЛЬНОЙ МЕТАСИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Корончик Д. Н.

* Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь

denis.koronchik@gmail.com

В работе описывается реализованный пользовательский интерфейс для интеллектуальной метасистемы поддержки проектирования интеллектуальных систем, которая располагается по адресу <http://ims.ostis.net>. В рамках статьи рассмотрены команды доступные пользователю в рамках ПИ, компоненты, которые реализованы для его функционирования и основные идеи, которые лежат в его основе.

Ключевые слова: пользовательский интерфейс; компонент пользовательского интерфейса; команда пользовательского интерфейса.

Введение

В рамках *Проекта OSTIS* [Голенков, 2013] ведется разработка **SC-технологии проектирования пользовательских интерфейсов** (семантическая технология проектирования пользовательских интерфейсов интеллектуальных систем). Более подробно с её описанием можно ознакомиться по ссылке [Корончик, 2012]. Основные положения, которые лежат в основе указанной технологии следующие:

- **пользовательский интерфейс (ПИ)** рассматривается как специализированная интеллектуальная система, которая направлена на получение сообщений от пользователя и вывода ему ответов системы. Следовательно, пользовательский интерфейс, как и любая другая система, разрабатывается по *Технологии OSTIS* является многоагентной системой, основанной на знаниях и, прежде всего, на онтологиях. Основной задачей пользовательского интерфейса является перевод сообщения от пользователя, полученного на некотором внешнем языке, на внутренний язык системы (SC-код), а также перевод ответа системы на некоторый внешний язык, понятный пользователю и отображение этого ответа;
- в основе графических интерфейсов лежит SCg-код (Semantic Code graphical – который является одним из возможных способов визуального представления текстов SC-кода)

[Голенков и др, 2001]. Объекты, отображаемые на экране с помощью SCg-кода, будем называть sc.g-элементами. Основным принципом, положенным в его основу, является то, что все изображенные на экране объекты (sc.g-элементы), в том числе и элементы управления, являются изображением узлов семантической сети. Другими словами каждому изображенному на экране объекту соответствует узел в семантической сети (базе знаний) синонимичный ему;

- формализация семантики пользовательских действий, с последующим анализом, а также их унификация и четкая типология.

В данной статье рассматривается пользовательский интерфейс интеллектуальной метасистемы поддержки проектирования интеллектуальных систем (*Метасистемы IMS.OSTIS*) реализованной по *Технологии OSTIS* [OSTIS, 2013].

1. Внешний вид пользовательского интерфейса

Внешний вид главного окна пользовательского интерфейса *Метасистемы IMS.OSTIS* представлен на рисунке 1. В основе этого ПИ лежит *SCg-код*. Все объекты (в том числе и элементы управления) изображаемые в рамках главного окна, являются знаками некоторых sc-элементов в базе знаний. Это позволяет пользователю указывать их в качестве аргументов для различных команд.



Рисунок 1 - Внешний вид пользовательского интерфейса метасистемы

2. Взаимодействие с пользователем

Взаимодействие пользователя с системой осуществляется через обмен сообщениями. Сообщения пользователь формирует с использованием пользовательских команд. Команды пользователя инициируются с помощью заранее определенной последовательности элементарных пользовательских действий. *Элементарное пользовательское действие* - это действие, которое осуществляется с использованием некоторого устройства ввода и не может быть декомпозировано (разбито на другие, более мелкие действия). Примерами таких действий могут быть, нажатия (отпускания) клавиш мыши или клавиатуры, перемещение мыши и т. д. Выделены следующие классы команд пользователя по типу инициируемого действия:

- *команда-вопрос* - класс команд, которые инициируют вопрос системе;
- *команды редактирования* - класс команд, которые инициируют действия связанные с редактированием внешних *sc-текстов*;
- *команды просмотра* - класс команд, которые инициируют действия связанные с просмотром внешних *sc-текстов*.

По способу определения аргументов выделены следующие классы команд пользователя:

- класс команд с заранее определенными аргументами, т. е., объекты, над которыми инициируется действие, заранее известны. Например, команда раскрытия пункта меню - это команда вывода декомпозиции объекта, где этим объектом является она сама;
- класс команд с дополнительно указываемыми аргументами. В этих командах, объекты, над которыми инициируется действие, необходимо указать.

Существует базовый способ инициирования команд с дополнительно указываемыми аргументами. Сначала указываются аргументы команды. Они просто перетягиваются на панель

аргументов. После чего инициируется команда, щелчком левой клавиши мыши на знаке соответствующего класса команд.

Команды с заранее определенными аргументами инициируются щелчком левой клавиши мыши на знаке, обозначающем эту команду.

В системе, к настоящему моменту, реализованы следующие *пользовательские команды*:

- *запрос полной семантической окрестности*. Данная команда может быть инициирована двумя способами. Первый способ: указать как аргумент *sc-элемент* к которому задается вопрос; инициировать команду левым щелчком мыши на её знаке в главном меню (*Просмотр БЗ > Семантическая окрестность > Полная семантическая окрестность*). Второй способ: инициировать команду с заранее определенными аргументами - это ссылки в рамках страницы. Инициирование также происходит по щелчку левой клавиши мыши по знаку команды (курсивный текст синего цвета, который подчеркивается при наведении курсора мыши);
- *команда переключения активного окна*. Эта команда инициируется с помощью щелчка левой клавиши мыши на знаке окна (в левой части главного окна). После того, как окно становится активным, знак его обозначающий становится синего цвета, а его содержимое отображается в центральной области главного окна;
- *команда изменения режима идентификации*. Чтобы инициировать эту команду необходимо сделать щелчок левой клавиши мыши на кнопку с треугольником, которая расположена справа от текущего режима идентификации. Затем, в появившемся списке, сделать щелчок левой клавиши мыши по одному из доступных режимов идентификации;
- *команда подготовки содержимого активного окна к печати*. Она инициируется нажатием левой клавиши мыши на команде с изображением принтера, которая расположена над списком окон, в левой части главного окна. После инициирования команды в браузере создается вкладка, содержимое которой можно выводить на печать с помощью браузера.

3. Логико-семантическая модель пользовательского интерфейса

Как и любая система, построенная с использованием *Технологии OSTIS*, пользовательский интерфейс метасистемы строится с использованием компонентного подхода. Выделены следующие классы компонентов ПИ:

- *компоненты трансляции*. Компоненты данного класса обеспечивают трансляцию из *SC-кода* на внешний язык и обратно;
- *компоненты визуализации*. Компоненты данного класса обеспечивают вывод информации представленной на внешнем языке;

- *компоненты редактирования.* Компоненты данного класса обеспечивают ввод информации пользователем на внешнем языке.

Каждый компонент состоит из некоторого фрагмента базы знаний и набора *sc*-агентов. Из компонентов трансляции в ПИ метасистемы реализованы следующие компоненты:

- транслятор из *SC*-кода в *формат SCs Json* (формат схожий по структуре с *SCs*-кодом *1-go* уровня, адаптированный для Web);
- транслятор из *формата SCs Json* в *SC-код*;
- транслятор *SCs*-кода в *SC-код*;
- транслятор *SCg*-кода в *SC-код*;
- транслятор из *SC*-кода в *SCg-код*.

Кроме компонентов трансляции в ПИ метасистемы реализованы следующие компоненты визуализации:

- визуализатор *sc.s-текстов* и *sc.n-текстов* [Голенков, 2012];
- визуализатор *sc.g-текстов*;
- визуализатор гипермедийных текстов (html);
- визуализатор карт (Google Maps);
- визуализатор видео (youtube).

Вся *база знаний пользовательского интерфейса* метасистемы делится на следующие разделы:

- описание пользователей – это часть базы знаний, в которой хранится информация обо всех пользователях системы (предпочитаемый естественный язык, различные настройки, протоколы действия пользователей и т. д.);
- описание синтаксиса и семантики всех используемых внешних языков – это часть базы знаний, в которой хранится формальное описание всех внешних языков представления знаний, которые используются при диалоге с пользователем. Примерами таких описаний могут быть: описание *SCg*-кода, *SCs*-кода и *SCn*-кода;
- описание используемых компонентов – это часть базы знаний, в которой описаны все компоненты пользовательского интерфейса, используемые в системе;
- описание *пользовательских команд* – это часть базы знаний, в которой хранятся описания команд, с помощью которых пользователь может взаимодействовать с системой;
- описание принципов работы самого пользовательского интерфейса – часть базы знаний, в которой хранится вся справочная информация по пользовательскому интерфейсу.

Абстрактная машина обработки знаний пользовательского интерфейса состоит из следующих *sc*-агентов (помимо *sc*-агентов, которые являются частью компонентов трансляции и отображения):

- *sc*-агент генерации экземпляра команды, по её обобщенному описанию. Все команды

пользовательского интерфейса представлены в базе знаний и имеют некоторое обобщенное описание (шаблон), на основании которого данный *sc*-агент генерирует экземпляр этой команды с подставленными в неё аргументами;

- *sc*-агент инициирования трансляции ответов на пользовательские запросы. Данный *sc*-агент ожидает ответ на запрос пользователя и генерирует экземпляр команды, которая иницирует трансляцию ответа на внешний язык;
- *sc*-агент сборки мусора. Данный *sc*-агент осуществляет поиск и удаление информации, которая уже не актуальна. Такой информацией может быть уже завершённая команда трансляции и т. д.;
- *sc*-агент сбора *sc*-идентификаторов. Данный *sc*-агент занимается сбором всех идентифицируемых *sc*-элементов, для того, чтобы обеспечить поиск элементов по идентификатору.

4. Архитектура технической реализации

На рисунке 2 представлена архитектура реализованного пользовательского интерфейса. Машина обработки знаний ПИ располагается на серверной стороне. Клиентская часть лишь делает запросы с помощью *http*, *ajax* и получает на них ответы. При получении ответа изменяется состояние на клиентской части.

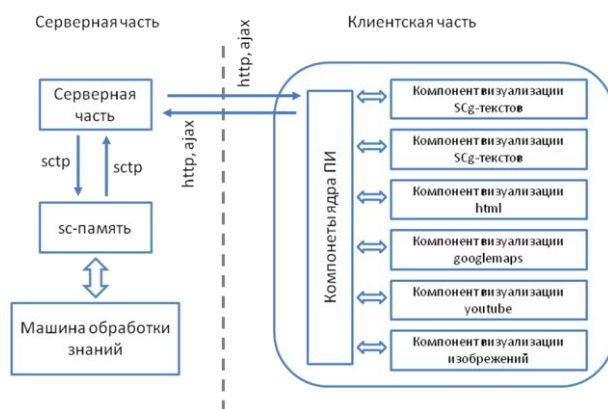


Рисунок 2 - Архитектура реализованного пользовательского интерфейса

Серверная часть ПИ реализована на языке *python*, с использованием *django* [django, 2013] она взаимодействует с *sc*-памятью с использованием протокола *sctp* [sctp, 2013]. *SC*-агенты пользовательского интерфейса (включая *sc*-агенты трансляции) реализованы на языке программирования *Си*. Программная реализация *SC*-памяти также реализована на языке программирования *Си*. Описание её архитектуры можно найти по ссылке [Корончик, 2013].

Клиентская часть ПИ состоит из компонентов ядра, которые обеспечивают взаимодействие с сервером, и компонентами, которые решают задачу визуализации и редактирования информации на внешних языках.

Заключение

Описанный в данной статье пользовательский интерфейс реализован и используется в рамках интеллектуальной метасистемы поддержки проектирования интеллектуальных систем. Посмотреть её можно на сайте <http://ims.ostis.net>. Можно выделить следующие преимущества пользовательского интерфейса этой системы:

- к элементам управления можно задавать любые вопросы. Это значительно сокращает стартовую инструкцию по использованию системы, так как достаточно лишь научить пользователя задавать вопросы базовым способом (указывая аргументы), а дальше он может узнать всю необходимую информацию по различным командам и элементам управления, задавая к ним вопросы;
- документация (help) по пользовательскому интерфейсу является частью самого ПИ, что сокращает накладные расходы по его разработке. В современных подходах необходимо реализовать ПИ, а затем еще и написать help по нему, который, при изменении самого ПИ, необходимо актуализировать. Здесь же описание команд используется самим пользовательским интерфейсом для функционирования;
- интернационализация. Простая возможность перехода между различными естественными языками. Используемый подход идентификации значительно позволяет сократить проблемы с переводимой информацией. К примеру, в современной wikipedia, каждой статье в соответствие ставятся статьи на других языках. При этом они могут сильно отличаться друг от друга по содержанию и стилю, а также многие из них являются более актуальными, чем другие. В данном ПИ меняются лишь идентификаторы sc-узлов в БЗ, а сама информация остается неизменной, что сводит проблему интернационализации к добавлению новых идентификаторов и трансляция SC-текстов на необходимом языке;
- компонентный подход. Благодаря компонентному подходу можно легко расширять функционал ПИ системы, за счет добавления новых компонентов;
- многообразие отображаемых видов знаний. С помощью набора компонентов для отображения информации на различных языках, система может отображать любые знания. При этом имеется ряд компонентов, которые позволяют визуализировать любые знания с помощью унифицированных языков представления знаний: *SCg-код*, *SCn-код*, *SCs-код*;

Библиографический список

- [Голенков и др., 2001] Представление и обработка знаний в графодинамических ассоциативных машинах / В. В. Голенков, [и др.]; – Мн. : БГУИР, 2001
- [Голенков, 2012] Графодинамические модели параллельной обработки знаний / В. В. Голенков, Н. А. Гулякина // Материалы

международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» - Минск, 2012

[Голенков, 2013] Открытый проект, направленный на создание технологии компонентного проектирования интеллектуальных систем / В. В. Голенков, Н. А. Гулякина // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» - Минск, 2013

[Корончик, 2012] Семантические модели мультимодальных пользовательских интерфейсов и семантическая технология их проектирования / Д. Н. Корончик // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» - Минск, 2012

[Корончик, 2013] Реализация хранилища унифицированных семантических сетей / Д. Н. Корончик // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» - Минск, 2013

[django, 2013] The Web framework for perfectionists with deadlines [Электронный ресурс]. – 2013 – Режим доступа: <https://www.djangoproject.com> – Дата доступа: 02.10.2013

[OSTIS, 2013] Открытая семантическая технология проектирования интеллектуальных систем [Электронный ресурс]. – 2013. - Режим доступа: <http://ostis.net>. – Дата доступа: 11.10.2013

[sctp, 2013] Протокол sctp [Электронный ресурс]. – 2013 – Режим доступа: <https://github.com/deniskoronchik/sc-machine/wiki/sctp> – Дата доступа: 02.10.2013

THE USER INTERFACE OF THE INTELLIGENT DESIGN SUPPORT METASYSTEM FOR INTELLIGENT SYSTEMS

Koronchik D. N.

**Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus*

denis.koronchik@gmail.com

This article describes implementation of semantic user interface for intelligent metasystem that provides support of intelligent system development.

Introduction

User interface is an intelligent system that works to provide dialog between user and system. Like another intelligent systems developed with OSTIS technology, it consist of knowledge base and knowledge processing machine.

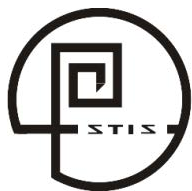
Main Part

Knowledge base of UI for metasystem consist of next sections: user profiles; description of syntax and semantic of used external languages; description of used UI components; description of UI commands; description of principles that used in UI.

Knowledge process machine consist of next sc-agents: sc-agent to generate instance of command based on template; sc-agent to initiate translation of information to SC-code and backward; sc-agent to collect garbage; sc-agent to collect sc-identifiers.

Conclusion

Implemented UI has many advantages. It allows used to ask any questions about UI controls. It's simplify internationalization problems. Also it can be simply extended by development of new components.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

БАЗА ЗНАНИЙ ИНТЕЛЛЕКТУАЛЬНОЙ МЕТАСИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Гракова Н.В., Давыденко И.Т., Русецкий К.В.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

grakova.nv@gmail.com

ir.davydenko@gmail.com

rusetski.k@gmail.com

В работе рассматривается структура базы знаний интеллектуальной Метасистемы IMS.OSTIS, предназначенной для поддержки проектирования интеллектуальных систем.

Ключевые слова: база знаний, интеллектуальная система, метасистема.

Введение

На сегодняшний день актуальной является задача быстрой и качественной разработки прикладных интеллектуальных систем. Для решения данной задачи необходимо создать комплекс моделей, методов и средств, предлагаемых конкретной технологией в виде интеллектуальной метасистемы поддержки проектирования интеллектуальных систем.

Указанная интеллектуальная система должна включать в себя:

- теорию (принципы построения) проектируемых интеллектуальных систем, которая входит в состав базы знаний метасистемы;
- библиотеку типовых многократно используемых компонентов интеллектуальных систем, которая входит в состав базы знаний рассматриваемой метасистемы;
- средства автоматизации синтеза, анализа и имитационного моделирования проектируемых интеллектуальных систем и их компонентов (это подсистема интеллектуальной метасистемы, ориентированная на решение задач проектирования интеллектуальных систем);
- интеллектуальную help-систему, являющуюся подсистемой рассматриваемой интеллектуальной метасистемы ориентированной на информационное обслуживание и обучение разработчиков интеллектуальных систем;
- методику проектирования интеллектуальных

систем, которая оформляется как часть базы знаний метасистемы;

- методику обучения проектированию интеллектуальной системы, которая также является частью базы знаний метасистемы;
- интеллектуальную подсистему управления проектированием самой метасистемы;
- интеллектуальную подсистему управления информационной безопасностью метасистемы;
- семейство различных платформ интерпретации унифицированных абстрактных логико-семантических моделей интеллектуальных систем. [Голенков, 2012]

В качестве решения указанной выше задачи для **Технологии OSTIS** [OSTIS,2013] предлагается **Метасистема IMS.OSTIS** (Intelligent Metasystem of Open Semantic Technology for Intelligent Systems) [IMS.OSTIS, 2013], которая представляет собой комплекс моделей, средств и методов, предназначенных для постоянного обновления и совершенствования предлагаемой технологии. Данная метасистема может быть использована не только для развития и сопровождения самой себя (что является ее отличительной особенностью), но и любых других интеллектуальных систем, построенных на основе **Технологии OSTIS**.

Пользователями **Метасистемы IMS.OSTIS** являются:

- читатели, желающие познакомиться с **Технологией OSTIS** (для них нужна навигация по **Базе знаний IMS.OSTIS** и средства отображения);

- разработчики прикладных интеллектуальных систем, проектируемых на основе **Технологии OSTIS** (им нужны методика и средства проектирования, а также библиотеки многократно используемых компонентов);
- разработчики самой **Метасистемы IMS.OSTIS**
 - разработчики *sc-модели базы знаний IMS.OSTIS* (им нужны средства интеграции, редактирования, верификации знаний);
 - разработчики *sc-модели машины обработки знаний IMS.OSTIS*;
 - разработчики *sc-модели пользовательского интерфейса IMS.OSTIS*;
 - разработчики средств технической реализации *Метасистемы IMS.OSTIS*.

1. Спецификация Базы знаний IMS.OSTIS

Базу знаний Метасистемы IMS.OSTIS рассмотрим как результат интеграции текстов **SC-кода** [Голенков, 2012]:

- 1) являющихся формальными спецификациями всевозможных продуктов (результатов) **Проекта OSTIS**, который направлен на создание **Технологии OSTIS**, а также подпроектов и проектных задач, входящих в состав этого проекта;
- 2) являющихся полной документацией указанных продуктов и обеспечивающих их воспроизведение (если продукт является базой знаний, то его документация совпадает с представлением указанной базы знаний на некотором внешнем языке);
- 3) являющихся формальными спецификациями проектов или проектных задач, результатами выполнения которых указанные продукты являются.

1.1. Спецификация Технологии OSTIS

Указанную технологию можно трактовать как специальным образом организованную деятельность по проектированию интеллектуальных систем. С другой стороны, **Технологию OSTIS** можно трактовать как комплекс моделей, методов и средств, предназначенных для разработки интеллектуальных систем, а также для постоянного обновления и совершенствования самой этой технологии. Актуальность и принципы, лежащие в основе технологии компонентного проектирования интеллектуальных систем, рассматривались в [Голенков, 2011], [Голенков, 2012].

Спецификация **Технологии OSTIS** в **Базе знаний IMS.OSTIS** представлена следующим образом:

Раздел. Базовая спецификация Технологии OSTIS

= **Раздел. Обоснование, принципы построения и состав Технологии OSTIS**

=> **базовый порядок разделов***:

Раздел. Унифицированные семантические сети и различные варианты их представления

=> **аннотация***:

o

<= **трансляция sc-текста***:

o

Э **пример**':

[

Данный раздел базы знаний представляет собой обоснование проекта, направленного на создание открытой массовой семантической технологии компонентного (модульного) проектирования семантически совместимых интеллектуальных систем – Технологии OSTIS (Open Semantic Technology for Intelligent Systems).

В тексте рассматриваются:

- **Недостатки** современных технологий проектирования интеллектуальных систем
- **Проблемы** – почему указанные недостатки не устранены, в чем заключаются трудности
- Анализ попыток решения указанных проблем – **причины неудач**
- Предлагаемый подход – **принципы**, лежащие в основе предлагаемой Технологии OSTIS
- **Предпосылки** – что уже сделано другими авторами в данном направлении
- **Уточнение** того, что мы называем технологией вообще, и Технологией OSTIS в частности
- **Состав** Технологии OSTIS

]

<= **декомпозиция раздела***:

{

- **Раздел. Тип и назначение Технологии OSTIS**
- **Раздел. Аналоги Технологии OSTIS и их недостатки**
- **Раздел. Проблемы, препятствующие устранению недостатков у аналогов Технологии OSTIS и анализ попыток решения этих проблем**
- **Раздел. Принципы, лежащие в основе Технологии OSTIS**
- **Раздел. Предпосылки создания Технологии OSTIS**
- **Раздел. Характеристики, используемые для сравнительного анализа Технологии OSTIS, и набор тестовых задач**
- **Раздел. Сравнительный анализ и тестирование Технологии OSTIS**
- **Раздел. Конкурентные преимущества и достоинства Технологии OSTIS**
- **Раздел. Новизна и оригинальность решений, реализуемых в Технологии OSTIS**
- **Раздел. Состав Технологии OSTIS**
- **Раздел. Проект OSTIS, направленный на создание Технологии OSTIS**

}

В базе знаний могут быть представлены документации и спецификации различного вида – документации различных проектов, документации различных технических систем, спецификации (онтологии) различных предметных областей. В качестве примера рассмотрим текущую версию *Документации Технологии OSTIS*.

Документация. Технология OSTIS

<= декомпозиция раздела*:

- {
- Раздел. Базовая спецификация Технологии OSTIS
- Раздел. Унифицированные семантические сети и различные варианты их представления
- Раздел. Унифицированные логико-семантические модели интеллектуальных систем
- Раздел. Семейство платформ интерпретации sc-моделей интеллектуальных систем
- Раздел. Метасистема IMS.OSTIS
- Раздел. Прикладные интеллектуальные системы, построенные по Технологии OSTIS
- }

Если говорить в целом о *Проекте OSTIS*, направленном на создание и развитие *Технологии OSTIS*, то, кроме разработки самой этой технологии в виде *Метасистемы IMS.OSTIS*, в состав такого комплексного проекта должен входить целый спектр подпроектов, направленных на расширение контингента разработчиков интеллектуальных систем, использующих предлагаемую технологию, на расширение спектра разрабатываемых приложений, на расширение контингента разработчиков самой технологии (точнее, *Метасистемы IMS.OSTIS*) и, в первую очередь, на расширение библиотеки многократно используемых компонентов интеллектуальных систем.

1.2. Спецификация Метасистемы IMS.OSTIS

Метасистема IMS.OSTIS представляет собой интеллектуальную систему, которая построена на основе *Технологии OSTIS* и, соответственно этому, состоит из следующих основных компонентов:

- унифицированной семантической модели (sc-модели) базы знаний *Метасистемы IMS.OSTIS* [Давыденко, 2013];
- унифицированной семантической модели (sc-модели) машины обработки знаний *Метасистемы IMS.OSTIS* [Шункевич, 2013];
- унифицированной семантической модели (sc-модели) пользовательского интерфейса *Метасистемы IMS.OSTIS* [Корончик, 2013a];
- технической реализации интерпретатора указанных унифицированных семантических моделей. Сюда входит техническая реализация sc-памяти для хранения унифицированных

семантических сетей (sc-текстов) и техническая реализация интерпретатора программ, описывающих обработку информации в указанной sc-памяти [Корончик, 2013b].

Раздел *базы знаний IMS.OSTIS*, посвященный описанию *Метасистемы IMS.OSTIS*, имеет следующую структуру:

Раздел. Метасистема IMS.OSTIS

= Раздел. IMS.OSTIS

= Раздел. Полное описание Метасистемы IMS.OSTIS

= Документация. Интеллектуальная метасистема, построенная по Технологии OSTIS и ориентированная на комплексную поддержку проектирования интеллектуальных систем на основе Технологии OSTIS

<= базовый порядок разделов*:

Раздел. Семейство платформ интерпретации sc-моделей интеллектуальных систем

=> базовый порядок разделов*:

Раздел. Прикладные интеллектуальные системы, построенные по технологии OSTIS

<= декомпозиция раздела*:

- {
- Раздел. Базовая спецификация IMS.OSTIS
- Раздел. Семейство подсистем IMS.OSTIS для поддержки проектирования различных компонентов и различных классов интеллектуальных систем
- Раздел. Семейство подсистем IMS.OSTIS для поддержки проектирования подсистем интеллектуальных систем, ориентированных на их эффективную эксплуатацию, сопровождение и обновление
- Раздел. Общая архитектура унифицированной семантической модели IMS.OSTIS
- }

Проект IMS.OSTIS - это метапроект, направленный на разработку, сопровождение и обновление интеллектуальной *Метасистемы IMS.OSTIS*. Указанная метасистема рассматривается нами как конструктивная форма воплощения предлагаемой нами технологии компонентного проектирования интеллектуальных систем.

Проект IMS.OSTIS

∈ проект

=> основной русскоязычный идентификатор*:

Проект IMS.OSTIS

=> основной англоязычный идентификатор*:

Project IMS.OSTIS

=> основной продукт*:

Метасистема IMS.OSTIS

= Технология OSTIS как определенный комплекс моделей, методов и средств разработки интеллектуальных систем на основе библиотеки многократно используемых компонентов

=> документация*:

Раздел. *Метасистема IMS.OSTIS*

<= основной подпроект*:

Проект *OSTIS*

=> исполнитель*:

- {
- руководитель проекта': Голенков В.В.
- администратор базы знаний': Давыденко И.Т.
- администратор машины обработки знаний': Шункевич Д.В.
- системный администратор': Корончик Д.Н.
- }

<= декомпозиция проекта*:

- {
- Проект. SC-модель базы знаний *IMS.OSTIS*
- Проект. SC-модель машины обработки знаний *IMS.OSTIS*
- Проект. SC-модель пользовательского интерфейса *IMS.OSTIS*
- Проект. Платформа интерпретации SC-модели *IMS.OSTIS*
- }

<= декомпозиция проекта*:

- {
- Проект. SC-модель Подсистемы *IMS.OSTIS* для консультационного обслуживания и обсуждения конечных пользователей *IMS.OSTIS* (разработчиков прикладных интеллектуальных систем)
- Проект. SC-модель Подсистемы *IMS.OSTIS* для информационной поддержки и автоматизации обновления *IMS.OSTIS* в ходе её эксплуатации, а также для повышения квалификации её разработчиков
- Проект. SC-модель Подсистемы *IMS.OSTIS* для информационной поддержки и автоматизации управления проектированием *IMS.OSTIS* в ходе её эксплуатации
- Проект. SC-модель Подсистемы *IMS.OSTIS* для управления информационной безопасностью *IMS.OSTIS*
- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей баз знаний
- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей программ
- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей машин обработки знаний
- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей пользовательских интерфейсов
- }

- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей интеллектуальных обучающих систем
- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей подсистем консультационного обслуживания конечных пользователей интеллектуальных систем
- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей подсистем информационной поддержки и автоматизации обновления интеллектуальных систем в ходе их эксплуатации, а также для повышения квалификации их разработчиков
- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей подсистем информационной поддержки и автоматизации управления проектированием интеллектуальных систем
- Проект. SC-модель Подсистемы *IMS.OSTIS* для поддержки проектирования sc-моделей подсистем управления информационной безопасностью интеллектуальных систем

1.3. Спецификация Машины обработки знаний *IMS.OSTIS*

База знаний *Метасистемы IMS.OSTIS* также содержит спецификацию *Машины обработки знаний Метасистемы IMS.OSTIS*, включающую перечень sc-агентов, обеспечивающих функционирование этой системы, их спецификации [Шункевич, 2013].

Машина обработки знаний Метасистемы IMS.OSTIS

<= декомпозиция*:

- {
- абстрактный sc-агент информационного поиска
- абстрактный sc-агент погружения интегрируемого знания в базу знаний
- абстрактный sc-агент выравнивания онтологии интегрируемого знания с основной онтологией базы знаний
- абстрактный sc-агент поиска решения явно сформулированных задач
- абстрактный sc-агент логического вывода
- абстрактный sc-агент обнаружения и удаления информационного мусора
- }

абстрактный sc-агент информационного поиска

<= декомпозиция абстрактного sc-агента*:

- {
- *sc-агент поиска всех выходящих константных позитивных стационарных sc-дуг принадлежности*
- *sc-агент поиска всех конструкций, полностью изоморфных заданному образцу*
- *sc-агент поиска всех идентификаторов заданного sc-элемента*
- *sc-агент поиска всех выходящих константных позитивных стационарных sc-дуг принадлежности с отношениями*
- *sc-агент поиска всех идентифицированных sc-элементов в базе знаний*
- *sc-агент поиска полной семантической окрестности*
- *sc-агент поиска всех подклассов в квазибинарном отношении*
- *sc-агент поиска всех выходящих константных позитивных стационарных sc-дуг принадлежности с отношениями*
- *sc-агент поиска всех входящих константных позитивных стационарных sc-дуг принадлежности*
- *sc-агент поиска связок заданного отношения, компонентом которых является заданный sc-элемент*
- *sc-агент поиска декомпозиции заданного sc-элемента*
- *sc-агент поиска всех надклассов в квазибинарном отношении*
- }

абстрактный sc-агент погружения интегрируемого знания в базу знаний

- <= декомпозиция абстрактного sc-агента*:*
- {
 - *sc-агент канторизации множества*
 - }

Рассмотрим описание проекта, направленного на разработку и развитие **Машины обработки знаний Метасистемы IMS.OSTIS**.

Проект. SC-модель машины обработки знаний IMS.OSTIS

∈ проект

=> основной продукт*:

Метасистема IMS.OSTIS

=> документация*:

Раздел. Состав унифицированной семантической модели машины обработки баз знаний IMS.OSTIS

=> исполнитель*:

- {
- *ответственный исполнитель': Шункевич Д.В.*
- }

=> дата создания*:

2013.сентябрь

=> планируемые сроки завершения*:

2014.май

=> проектная задача*:

- {
- *Реализовать sc-агент поиска полной семантической окрестности заданного sc-элемента на платформенно-независимом уровне*
 ∈ горячая проектная задача
 => тип проектной задачи*: улучшение
 => приоритет проектной задачи*: важный
- *Реализовать sc-агент поиска декомпозиции заданного sc-элемента на платформенно-независимом уровне*
 ∈ горячая проектная задача
 => тип проектной задачи*: улучшение
 => приоритет проектной задачи*: важный
- *Реализовать sc-агент поиска всех подклассов в квазибинарном отношении на платформенно-независимом уровне*
 ∈ горячая проектная задача
 => тип проектной задачи*: улучшение
 => приоритет проектной задачи*: важный
- *Реализовать sc-агент поиска всех надклассов в квазибинарном отношении на платформенно-независимом уровне*
 ∈ горячая проектная задача
 => тип проектной задачи*: улучшение
 => приоритет проектной задачи*: важный
- *Реализовать sc-агент поиска всех sc-элементов, имеющих основной идентификатор на платформенно-независимом уровне*
 ∈ горячая проектная задача
 => тип проектной задачи*: улучшение
 => приоритет проектной задачи*: важный
- }

2. Спецификация подсистемы поддержки проектирования баз знаний интеллектуальных систем

Основной задачей *Подсистемы поддержки проектирования баз знаний интеллектуальных систем* является предоставление разработчику баз знаний информационной поддержки, необходимых методов и средств проектирования, таких как документация, набор агентов верификации и отладки базы знаний, средства редактирования базы знаний и т.д.

Данная подсистема также имеет базу знаний, машину обработки знаний и пользовательский интерфейс.

Рассмотрим некоторые фрагменты базы знаний указанной подсистемы.

sc-знание

= Множество всевозможных знаний, представленных в SC-коде
∈ ключевой sc-элемент*:

o

∈ sc-пояснение

<= трансляция sc-текста*:

o

Э[*sc-знание* – это текст SC-кода, обладающий некоторой семантической целостностью. Можно говорить о предметной области, в которой максимальным классом исследуемых объектов является Множество всевозможных знаний, представленных в SC-коде. Концептами такой предметной области являются различные классы знаний (типы, виды знаний), а отношениями указанной предметной области являются различные отношения, заданные на Множестве всевозможных *sc-знаний*. Соответственно этому можно говорить об онтологии указанной предметной области]
∈ Русский язык

класс sc-знаний

= вид sc-знаний

= тип sc-знаний

= Семейство подмножеств Множества всевозможных sc-знаний

Э sc-модель предметной области

⊃ sc-модель стационарной предметной области

⊃ sc-модель нестационарной предметной области

Э sc-онтология

Э sc-окрестность

Э sc-сравнение

Э sc-текст определения

Э sc-текст задачи

Э sc-текст обобщенной формулировки класса задач

Э sc-текст программы

Э sc-текст формальной теории

Э sc-текст высказывания

Э sc-текст логической формулы

Помимо этого в базе знаний могут храниться:

- sc-описание конкретной интеллектуальной системы (ее sc-модели и интерпретатора этой модели);
- sc-модель конкретной интеллектуальной системы;
- sc-модель частной машины обработки знаний;
- sc-описание атомарного sc-агента, реализованного на языке SCP;
- sc-описание конкретного программно реализованного интерпретатора sc-моделей интеллектуальных систем;
- sc-описание платформенно зависимой реализации атомарного sc-агента (в частности, sc-агента пользовательского интерфейса);
- sc-описание любого программного продукта, реализованного на современных технологиях;
- sc-описание типового (многократно используемого) компонента баз знаний интеллектуальных систем;
- sc-описание типовой (многократно используемой) подсистемы интеллектуальных систем (типовой подсистемы пользовательских интерфейсов, типового решателя задач);
- sc-описание внешнего языка (SCg, SCs, Русского языка) – его синтаксиса и семантики;
- sc-описание синтаксической структуры внешнего текста;
- sc-описание семантики внешнего текста;
- sc-описание стиля отображения внешнего текста;
- sc-описание протокола изменения состояния базы знаний;
- sc-описание проекта;
- sc-описание разработчика (участника проектов) – его статуса в рамках разных проектов;
- sc-описание предложения (результата деятельности) разработчика базы знаний – как аналога научно-технической статьи, подлежащей рецензированию и согласованию;
- sc-описание выявленной ошибки в *Метасистеме IMS.OSTIS* (в том числе в ее базе знаний).

База знаний представляется в виде иерархической системы разделов.

В зависимости от наличия подразделов разделы базы знаний могут быть *атомарными* (то есть не декомпозируемыми на подразделы) и *неатомарными* (которые декомпозируются на подразделы).

Атомарный раздел трактуется как последовательность *sc-окрестностей*, ключевые *sc-элементы* которых объявляются также и *ключевыми sc-элементами* соответствующего *атомарного раздела*.

sc-окрестность

= sc-текст, являющийся семантической окрестностью заданного sc-элемента

⊃ *sc-знание*
 ⊃ *sc-пояснение*
 ⊃ *sc-текст определения*
 ⊃ *sc-примечание*
 ⊃ *sc-комментарий*
 <=> *семантически близкий sc-элемент**:
 sc-окрестность
 ∈ *ключевой элемент*':
 Раздел. Предметная область знаний

Одним из классов знаний, хранящихся в базе знаний могут быть спецификации (онтологии) самых различных предметных областей (и, в первую очередь, метаобластей) или так называемые онтологии верхнего уровня.

Предметная метаобласть

∃ *онтология Предметной области sc-элементов*
 ∃ *Предметная область sc-множеств*
 ∃ *Предметная область предметных областей*
 ∃ *Предметная область формальных теорий*
 ∃ *Предметная область информационных задач, классов информационных задач и способов их решения*

В базе знаний помимо формальных текстов могут храниться *естественно-языковые тексты*, семантически им эквивалентные. Любой *sc-текст* может быть связан с внешним представлением этого *sc-текста* связкой отношения *трансляция sc-текста**.

трансляция sc-текста*

∈ *ключевой sc-элемент*':
 0
 ∈ *sc-примечание*
 <=> *трансляция sc-текста**:
 0
 ∃ [Дуга отношения *трансляция sc-текста** направлена в знак транслируемого *sc-текста* и выходит из знака класса семантически и синтаксически эквивалентных внешних информационных конструкций, которому мог бы принадлежать и транслируемый *sc-текст*, если бы он был внешним.]
 ∈ *Русский язык*
 ∈ *ключевой элемент*':
 Раздел. Предметная область sc-ссылок

В естественно-языковых текстах существует необходимость применять некоторые средства форматирования (выделения, акцентирования внимания) к отдельным частям текста. При этом из естественно-языковых текстов существует возможность делать ссылки на *sc-элементы* базы знаний и использовать их в качестве аргументов любых поисковых команд.

Рассмотрим описание проекта, направленного на разработку и развитие *SC-модели Подсистемы IMS.OSTIS для поддержки проектирования sc-моделей баз знаний*.

Проект. SC-модель Подсистемы IMS.OSTIS для поддержки проектирования sc-моделей баз знаний

∈ *проект*
 => *основной продукт**:
 Метасистема IMS.OSTIS
 => *документация**:
 Раздел. SC-модель базы знаний Подсистемы IMS.OSTIS для поддержки проектирования sc-моделей баз знаний
 <=> *декомпозиция раздела**:
 {
 • *Раздел. Унифицированные семантические сети и различные варианты их представления*
 • *Раздел. Структура унифицированной семантической модели базы знаний*
 }
 => *исполнитель**:
 {
 • *ответственный исполнитель*':
 Давыденко И.Т.
 }
 => *дата создания**:
 2013.май
 => *планируемые сроки завершения**:
 2014.июнь

Неотъемлемой частью *Подсистемы IMS.OSTIS для поддержки проектирования sc-моделей баз знаний* является *Библиотека многократно используемых компонентов баз знаний* [Давыденко, 2013].

К основным типам компонентов баз знаний, хранящихся в библиотеке относятся:

- онтологии различных предметных областей, которые могут быть самыми различными по содержанию, однако должны быть семантически совместимыми;
- базовые фрагменты теорий, соответствующие различным уровням знания пользователя, начиная от базового школьного до профессионального;
- семантические окрестности различных объектов;
- спецификации формальных языков описания различных предметных областей.

Для включения компонента базы знаний в библиотеку необходимо его специфицировать по следующим критериям:

- предметная область, описание которой содержится в компоненте;
- класс (тип) компонента базы знаний;
- состав базы знаний;
- количественные характеристики ключевых узлов базы знаний;
- информация о разработчиках базы знаний;
- дата создания базы знаний;
- информация о верификации базы знаний;

- версия компонента базы знаний;
- условия распространения компонента базы знаний;
- сопровождающая информация.

3. Спецификация Подсистемы поддержки управления проектированием Метасистемы IMS.OSTIS

В основе современных методов управления проектированием лежат методики структуризации работ по проекту и целеориентированность процесса управления. Целью управления проектирования является достижение заранее определенных целей проекта при заранее известных исполнителях, ограничениях по срокам и целесообразном использовании возможностей, реагирования на риски [Гракова, 2013а].

Традиционно проектом является масштабная, технически сложная деятельность, направленная на достижение определённых целей. Для более эффективного и качественного получения результата проекта необходимо организовывать его управление. Управление проектами — это совокупность знаний, опыта, методов и средств, прилагаемая к процессам проекта для удовлетворения требований, предъявляемых к проекту, и ожиданий участников проекта. Для удовлетворения этим требованиям и ожиданиям, необходимо найти оптимальное сочетание между целями, сроками, затратами, качеством и другими характеристиками проекта.

Для того, чтобы найти оптимальное сочетание между этими критериями проекта необходимо чётко понимать, что каждый критерий из себя представляет и как осуществляется взаимодействие внутри проекта в соответствии с этими критериями. Для того чтобы организовать такое взаимодействие и необходима подсистема управления проектированием.

Для формального описания знаний, касающихся **Подсистемы управления проектированием интеллектуальных систем IMS.OSTIS** была выделена и описана **Предметная область управления проектами**. К основным ключевым понятиям данной предметной области относятся понятия *проекта*, *продукта**, *проектной задачи**, *приоритета проектной задачи** [Гракова, 2013б].

проект

∈ *ключевой sc-элемент*':

o

∈ *sc-пояснение*

<= *трансляция sc-текста**:

o

Э [*проект* — это целенаправленная деятельность, результатом которой является определенный ранее не существовавший продукт.]

∈ *Русский язык*

∈ *ключевой sc-элемент*':

o

∈ *sc-пояснение*

<= *трансляция sc-текста**:

o

Э [Описание (спецификация) каждого проекта включает в себя: указание продукта, являющегося результатом выполнения этого проекта; указание проектной задачи (проектной цели, технического задания) – требований, которым должен удовлетворять разрабатываемый продукт; срок выполнения проекта; команда исполнителей проекта (с указанием роли каждого исполнителя); план выполнения проекта (для неатомарного проекта это его декомпозиция на подпроекты, а для атомарного проекта – декомпозиция на проектные задания)]

продукт*

= *быть продуктом соответствующей деятельности (в том числе и явно обозначенного, явно заявленного проекта)*

⊃ *основной продукт**

= *быть основным продуктом данной деятельности (в том числе проекта)*

проектная задача*

= *быть проектной задачей соответствующего проекта**

⊃ *основная проектная задача**

= *быть основной проектной задачей заданного проекта**

⊃ *предлагаемая проектная задача**

= *быть предлагаемой проектной задачей заданного проекта**

= *проектная задача-черновик**

⊃ *горячая проектная задача**

= *быть срочной проектной задачей заданного проекта**

⊃ *утверждаемая проектная задача**

= *быть утверждаемой проектной задачей заданного проекта**

⊃ *исполняемая проектная задача**

= *быть исполняемой проектной задачей заданного проекта**

⊃ *проверяемая проектная задача**

= *быть проверяемой проектной задачей заданного проекта**

⊃ *завершённая проектная задача**

= *быть завершённой проектной задачей заданного проекта**

тип проектной задачи*

= *быть типом проектной задачи соответствующего проекта**

∈ *ключевой sc-элемент*':

o

∈ *sc-пояснение*

<= трансляция *sc*-текста*:

о

Э [тип проектной задачи – классификация проектных задач в зависимости от того с какой целью создавалась проектная задача.]

∈ Русский язык

∈ ключевой *sc*-элемент*:

о

∈ *sc*-пояснение

<= трансляция *sc*-текста*:

о

Э [Проектная задача может иметь один из следующих типов:

основная задача – это проектная задача, связанная с написанием некоторой спецификации для Проекта *IMS.OSTIS*;

задача верификации – это проектная задача, созданная с целью выявления возможных ошибок в проекте;

дефект – это ошибка, обнаруженная в проекте в процессе разработки, тестирования или эксплуатации;

улучшение – это тип проектной задачи, подразумевающий внесение изменений в уже существующий проект, направленное на усовершенствование этого проекта;

инновация – это тип проектной задачи, результатом которой является какое-либо нововведение в некотором проекте.

]

∈ Русский язык

приоритет проектной задачи*

= быть приоритетом проектной задачи соответствующего проекта*

∈ ключевой *sc*-элемент*:

о

∈ *sc*-пояснение

<= трансляция *sc*-текста*:

о

Э [приоритет проектной задачи* – это отношение, показывающее важность, первенство или значимость определённого процесса, объекта или явления.]

∈ Русский язык

∈ ключевой *sc*-элемент*:

о

∈ *sc*-пояснение

<= трансляция *sc*-текста*:

о

Э [Приоритет проектной задачи определяет порядок её выполнения. Соответственно проектная задача может иметь один из следующих приоритетов:

блокирующий приоритет – это приоритет проектной задачи, предполагающий её немедленное выполнение, так как в ином случае дальнейшая реализация проекта будет приостановлена;

критичный приоритет – это приоритет

проектной задачи, означающий, что в проекте обнаружены серьёзные ошибки, связанные с потерей данных, неработоспособностью *Метасистемы IMS.OSTIS* в целом;

важный приоритет – это приоритет проектной задачи, означающий, что результат выполнения данной проектной задачи принесет значимые результаты в развитие *Метасистемы IMS.OSTIS*;

неважный приоритет – это приоритет проектной задачи, предполагающий, что текущая проектная задача может быть выполнена в последнюю очередь.

]

∈ Русский язык

Рассмотрим описание проекта, направленного на разработку и развитие *Подсистемы поддержки управления проектированием Метасистемы IMS.OSTIS*.

Проект. SC-модель Подсистемы поддержки управления проектированием Метасистемы IMS.OSTIS

∈ проект

=> основной продукт*:

Метасистема IMS.OSTIS

=> документация*:

Раздел. SC-модель Подсистемы поддержки управления проектированием Метасистемы IMS.OSTIS

=> исполнитель*:

{

• ответственный исполнитель*:
Гракова Н.В.

• исполнитель*:
Жуков И.И.

}

=> дата создания*:

2013.май

=> планируемые сроки завершения*:

2015.июнь

Заключение

В данной статье рассмотрена структура базы знаний интеллектуальной *Метасистемы поддержки проектирования интеллектуальных систем IMS.OSTIS*. Использование данной *Метасистемы* позволит эффективно использовать *Технологию OSTIS* для проектирования прикладных систем различного назначения.

Результаты, приведенные в работе, апробируются в рамках открытого *Проекта OSTIS* [OSTIS, 2013].

Библиографический список

[Голенков, 2011] Голенков, В.В., Гулякина Н.А. Принципы построения массовой семантической технологии компонентного проектирования интеллектуальных систем. – В кн. Междунар. научн.-техн. конф. «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2011).

Материалы конф. [Минск, 10-12 февр. 2011 г.]. – Минск: БГУИР, 2011, с. 21-59.

[Голенков, 2012] Голенков, В.В., Гулякина Н.А. Графодинамические модели параллельной обработки знаний: принципы построения, реализации и проектирования. – В кн Междунар. научн.-техн. конф. . «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2012). Материалы конф. [Минск, 16-18 февр. 2012 г.]. – Минск: БГУИР, 2012.

[Гракова, 2013а] Гракова, Н.В. Управление проектированием интеллектуальных систем / Н.В. Гракова, И.И. Жуков // Интеллектуальный анализ информации: сб. тр. междунар. науч. конф. им. Т.А.Таран ИАИ-2013, Киев, 15–17 мая 2013г. / Министерство образования и науки Украины, Российская ассоциация искусственного интеллекта, Национальный технический университет Украины "КПИ", Факультет прикладной математики, Институт прикладного системного анализа, Издательство "Прогресс"; редкол.: Ю.Р. Валькман [и др.]. – Киев, 2013. – С. 137-141

[Гракова, 2013б] Гракова, Н.В. Проектные задачи в интеллектуальной системе управления проектированием / Н.В. Гракова, И.И. Жуков // Информационные технологии и системы 2013 (ИТС 2013): материалы международной научной конференции, БГУИР, Минск, Беларусь, 23 октября 2013 г. = Information Technologies and Systems 2013 (ITS 2013) : Proceeding of The International Conference, BSUIR, Minsk, 24th October 2013 / редкол.: Л. Ю. Шилин [и др.]. – Минск : БГУИР, 2013. – С. 122-123

[Давыденко, 2013] Давыденко, И.Т. Технология компонентного проектирования баз знаний на основе унифицированных семантических сетей. – В кн Междунар. научн.-техн. конф. . «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2013). Материалы конф. [Минск, 2013 г.]. – Минск: БГУИР, 2013.

[Корончик, 2013а] Корончик Д.Н. Унифицированные семантические модели пользовательских интерфейсов интеллектуальных систем и технология их компонентного проектирования. – В кн Междунар. научн.-техн. конф. . «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2013). Материалы конф. [Минск, 2013 г.]. – Минск: БГУИР, 2013.

[Корончик, 2013б] Корончик Д.Н. Реализация хранилища унифицированных семантических сетей. – В кн Междунар. научн.-техн. конф. . «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2013). Материалы конф. [Минск, 2013 г.]. – Минск: БГУИР, 2013.

[Шункевич, 2013] Шункевич, Д.В. Модели и средства компонентного проектирования машин обработки знаний на основе семантических сетей. – В кн Междунар. научн.-техн. конф. . «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2013). Материалы конф. [Минск, 2013 г.]. – Минск: БГУИР, 2013.

[IMS.OSTIS, 2013] Метасистема IMS.OSTIS [Электронный ресурс]. Минск, 2013. – Режим доступа: <http://ims.ostis.net/>. – Дата доступа: 07.12.2013.

[OSTIS, 2013] Проект OSTIS [Электронный ресурс]. Минск, 2013. – Режим доступа: <http://ostis.net/>. – Дата доступа: 11.12.2013.

THE KNOWLEDGE BASE OF THE INTELLIGENT DESIGN SUPPORT METASYSTEM FOR INTELLIGENT SYSTEMS

Grakova N.V., Davydenko I.T., Rusetski K.V.

*Belarusian State University of Informatics and
Radioelectronics, Minsk, Republic of Belarus*

grakova.nv@gmail.com

ir.davydenko@gmail.com

rusetski.k@gmail.com

This work presents the structure of the *Knowledge base of the IMS.OSTIS* intelligent Metasystem designed to

support development of applied intelligent systems for different purposes.

Introduction

Rapid development of high-quality applied intelligent systems is a very important problem nowadays. To solve this problem we have to wrap a set of models, methods and tools offered by a given technology as an intelligent design support metasystem. Such system has to include the theory of intelligent systems, IP component libraries, help and support subsystem for developers and end users, intelligent system synthesis, analysis and modeling aids, project management subsystem, security subsystem and various implementations of unified semantic models interpreters.

Main Part

Knowledge base built with *OSTIS Technology* may contain various kinds of documents and specifications, such as project documentation, technical documentation, various subject domain specifications (ontologies). This article discusses the current version of the *OSTIS project documentation*. The knowledge base is built using Semantic Code, which allows to represent a wide variety of knowledge types within the knowledge base.

IMS.OSTIS documentation is represented as a hierarchical structure of sections within the knowledge base. Sections can be atomic and non-atomic, depending on whether they decompose any further. Atomic section of the documentation includes key sc-elements with their semantic neighborhoods (other sc-elements related to them in some way). Any formal text within a section can have its corresponding natural language representation.

The *IMS.OSTIS Metasystem* also has a project management subsystem and a corresponding subject domain of project management in its knowledge base. For example, it contains such terms as *project*, *product** and *project task**. Having such terms in a knowledge base allows us to apply a semantic approach to a project management (i.e., describe projects and subprojects in a formal way), which looks appropriate for building an intelligent system.

Conclusion

This article has shown the structure of the *Knowledge base of the IMS.OSTIS* intelligent design support metasystem for intelligent systems. Its use will lead to more efficient application of the *OSTIS Technology*.

Results mentioned in this work are being tested within an open-source *OSTIS project* [OSTIS, 2013].



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

МАШИНА ОБРАБОТКИ ЗНАНИЙ ИНТЕЛЛЕКТУАЛЬНОЙ МЕТАСИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Шункевич Д.В.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

shu.dv@tut.by

Данная статья посвящена принципам организации и функционирования машин обработки знаний интеллектуальных систем в целом, лежащим в основе технологии проектирования машин обработки знаний в рамках технологии OSTIS. В качестве примера применения такой технологии рассматривается машина обработки знаний интеллектуальной метасистемы IMS.OSTIS направленной на поддержку проектирования интеллектуальных систем.

Ключевые слова: машина обработки знаний; интеллектуальная система; многоагентная система; технология проектирования.

Введение

Данная статья посвящена принципам организации и функционирования машин обработки знаний, построенных на базе технологии проектирования машин обработки знаний в рамках технологии OSTIS [OSTIS, 2013]. Более подробное формальное описание понятий и принципов, рассматриваемых в данной статье, можно найти на сайте метасистемы поддержки проектирования интеллектуальных систем IMS.OSTIS [IMS.OSTIS, 2013]

1. Унифицированная модель машин обработки знаний, разрабатываемых на основе технологии OSTIS

В предлагаемом подходе к построению машин обработки знаний сама машина представляет собой графодинамическую sc-машину (память в качестве модели представления знаний использует семантическую сеть с теоретико-множественной интерпретацией [Голенков и др., 2001]), состоящую из двух частей:

- графодинамическая sc-память;
- система sc-агентов над общей памятью.

Система sc-агентов над общей памятью представляет собой коллектив sc-агентов, условием инициирования которых является появление в памяти системы некоторой определенной конструкции. При этом операции взаимодействуют

между собой через память системы посредством генерации конструкций, являющихся условиями инициирования для другой операции. При таком подходе становится возможным обеспечить гибкость и расширяемость функционала системы путем добавления или удаления из ее состава некоторого набора агентов, не внося при этом изменений, затрагивающих других агентов.

Отличительной особенностью машины обработки знаний как многоагентной системы в рамках данного подхода является принцип взаимодействия агентов. По сути, предлагаемый подход реализует принцип «доски объявления», рассматриваемый в теории многоагентных систем [Тарасов, 2002]. Агенты обмениваются сообщениями исключительно через общую память путем использования соответствующего языка взаимодействия, в отличие от большинства классических МАС, в которых агенты обмениваются сообщениями непосредственно друг с другом. В рассматриваемом подходе каждый агент, формулируя вопросную конструкцию в памяти, априори не знает, какой из агентов будет обрабатывать указанную конструкцию, а лишь дожидается появления в памяти факта окончания обработки вопроса. При этом в решении поставленной таким образом задачи может принимать участие целый коллектив агентов, некоторые из которых могут выполнять определенные действия параллельно.

2. Предметная область *sc*-агентов

Под *абстрактным sc-агентом* понимается некоторый класс функционально эквивалентных *sc-агентов*, разные экземпляры (т.е. представители) которого могут быть реализованы по-разному.

Каждый *абстрактный sc-агент* имеет соответствующую ему спецификацию. В спецификацию каждого *абстрактного sc-агента* входит:

- указание ключевых *sc-элементов* этого *sc-агента*, т.е. тех *sc-элементов*, хранимых в *sc-памяти*, которые для данного *sc-агента* являются «точками опоры»
- формальное описание условий инициирования данного *sc-агента*, т.е. тех ситуаций в *sc-памяти*, которые инициируют деятельность данного *sc-агента*.
- формальное описание первичного условия инициирования данного *sc-агента*, т.е. такой ситуации в *sc-памяти*, которая побуждает *sc-агента* перейти в активное состояние и начать проверку наличия своего полного условия инициирования.
- строгое, полное, однозначно понимаемое описание деятельности данного *sc-агента*, оформленное при помощи каких-либо понятных, общепринятых средств, не требующих специального изучения, например на естественном языке.
- описание результатов выполнения данного *sc-агента*.

Понятие *абстрактного sc-агента* разбивается на два подкласса:

- *неатомарный абстрактный sc-агент*
- *атомарный абстрактный sc-агент*

Под *неатомарным абстрактным sc-агентом* понимается *абстрактный sc-агент*, который декомпозируется на коллектив более простых *абстрактных sc-агентов*, каждый из которых в свою очередь может быть как *атомарным абстрактным sc-агентом*, так и *неатомарным абстрактным sc-агентом*. При этом в каком либо варианте декомпозиции *абстрактного sc-агента* дочерний *неатомарный абстрактный sc-агент* может стать *атомарным абстрактным sc-агентом*, и реализовываться соответствующим образом.

Отношение *декомпозиции абстрактного sc-агента** трактует *неатомарные абстрактные sc-агенты* как коллективы более простых *абстрактных sc-агентов*, взаимодействующих через *sc-память*.

Другими словами, *декомпозиция абстрактного sc-агента** на *абстрактные sc-агенты* более низкого уровня уточняет один из возможных подходов к реализации этого *абстрактного sc-агента* путем построения коллектива более простых *абстрактных sc-агентов*.

Под *атомарным абстрактным sc-агентом* понимается *абстрактный sc-агент*, для которого уточняется платформа его реализации, т.е. существует соответствующая связка отношения *программа sc-агента**.

Понятие *абстрактного sc-агента* разбивается на два подкласса:

- *платформенно-независимый абстрактный sc-агент*
- *платформенно-зависимый абстрактный sc-агент*

К *платформенно-независимым абстрактным sc-агентам* относят *атомарные абстрактные sc-агенты*, реализованные на базовом языке программирования Технологии OSTIS, т.е. на *Языке SCP*. [Голенков и др., 2001]

При описании *платформенно-независимых абстрактных sc-агентов* под платформенной независимостью понимается платформенная независимость с точки зрения Технологии OSTIS, т.е. реализация на специализированном языке программирования, ориентированном на обработку семантических сетей (*Языке SCP*), поскольку *атомарные sc-агенты*, реализованные на указанном языке могут свободно переноситься с одной платформы интерпретации *sc-моделей* на другую. При этом языки программирования, традиционно считающиеся платформенно-независимыми в данном случае не могут считаться таковыми [Петцольд, 2002], [Шилдт, 2004], [Хорстманн, 2004].

К *платформенно-зависимым абстрактным sc-агентам* относят *атомарные абстрактные sc-агенты*, реализованные ниже уровня *sc-моделей*, т.е. не на *Языке SCP*, а на каком-либо другом языке описания программ.

Существуют *sc-агенты*, которые принципиально должны быть реализованы на платформенно-зависимом уровне, например, собственно *sc-агенты* интерпретации *sc-моделей* или рецепторные и эффекторные *sc-агенты*, обеспечивающие взаимодействие с внешней средой.

Под *sc-агентом* понимается конкретный экземпляр (с теоретико-множественной точки зрения - элемент) некоторого *атомарного абстрактного sc-агента*, работающий в какой-либо конкретной интеллектуальной системе.

При этом каждый *sc-агент* инициируется при появлении в *sc-памяти* конкретной ситуации, соответствующей условию инициирования *атомарного абстрактного sc-агента*, экземпляром которого является заданный *sc-агент*. В данном случае можно провести аналогию между принципами объектно-ориентированного программирования, рассматривая *атомарный абстрактный sc-агент* как класс, а конкретный *sc-агент* – как экземпляр, конкретную имплементацию этого класса [Хорстманн, 2004], [Гамма, 2007].

Взаимодействие *sc-агентов* осуществляется только через *sc-память*. Как следствие, результатом

работы любого *sc-агента* является некоторое изменение состояния *sc-памяти*, т.е. удаление либо генерация каких-либо *sc-элементов*.

В общем случае один *sc-агент* может явно передать управление другому *sc-агенту*, если этот *sc-агент* априори известен. Для этого каждый *sc-агент* в *sc-памяти* имеет обозначающий его *sc-узел*, с которым можно связать конкретную ситуацию в текущем состоянии базы знаний, которую иницилируемый *sc-агент* должен обработать.

Однако далеко не всегда легко определить того *sc-агента*, который должен принять управление от заданного *sc-агента*, в связи с чем описанная выше ситуация возникает крайне редко. Более того, иногда условие инициирования *sc-агента* является результатом деятельности непредсказуемой группы *sc-агентов*, равно как и одна и та же конструкция может являться условием инициирования целой группы *sc-агентов*.

При этом общаются через *sc-память* не программы *sc-агентов**, а сами описываемые данными программами *sc-агенты*.

Под **активным *sc-агентом*** понимается *sc-агент* интеллектуальной системы, который реагирует на события соответствующие его условию инициирования, и, как следствие, его *первичному условию инициирования**. Не входящие во множество **активных *sc-агентов*** *sc-агенты* не реагируют ни на какие события в *sc-памяти*.

Связки отношения **ключевые *sc-элементы sc-агента**** связывают между собой *sc-узел*, обозначающий *абстрактный sc-агент* и *sc-узел*, обозначающий множество *sc-элементов*, которые являются ключевыми для данного *абстрактного sc-агента*, то данные *sc-элементы* явно упоминаются в рамках программ, реализующих данный *абстрактный sc-агент*.

Связки отношения **программа *sc-агента**** связывают между собой *sc-узел*, обозначающий *атомарный абстрактный sc-агент* и *sc-узел*, обозначающий множество программ, реализующих указанный *атомарный абстрактный sc-агент*.

В случае *платформенно-независимого абстрактного sc-агента* каждая связка отношения *программа sc-агента** связывает *sc-узел*, обозначающий указанный *абстрактный sc-агент* с множеством *scp-программ*, описывающих деятельность данного *абстрактного sc-агента*. Данное множество содержит одну *агентную scp-программу*, и произвольное количество (может быть, и ни одной) *scp-программ*, которые необходимы для выполнения указанной *агентной scp-программы*.

В случае *платформенно-зависимого абстрактного sc-агента* каждая связка отношения *программа sc-агента** связывает *sc-узел*, обозначающий указанный *абстрактный sc-агент* с множеством файлов, содержащих исходные тексты

программы на некотором внешнем языке программирования, реализующей деятельность данного *абстрактного sc-агента*.

Связки отношения **первичное условие инициирования*** связывают между собой *sc-узел*, обозначающий *абстрактный sc-агент* и бинарную ориентированную пару, описывающую первичное условие инициирования данного *абстрактного sc-агента*, т.е. такой ситуации в *sc-памяти*, которая побуждает *sc-агента* перейти в активное состояние и начать проверку наличия своего полного условия инициирования.

Первым компонентом данной ориентированной пары является знак некоторого подмножества понятия *sc-событие*, например *sc-событие добавления выходящей sc-дуги*, т.е. по сути конкретный тип события в *sc-памяти*.

Вторым компонентом данной ориентированной пары является произвольный в общем случае *sc-элемент*, с которым непосредственно связан указанный тип события в *sc-памяти*, т.е., например, *sc-элемент*, из которого выходит либо в который входит генерируемая либо удаляемая *sc-дуга*, либо *sc-ссылка*, содержимое которой было изменено.

После того, как в *sc-памяти* происходит некоторое *sc-событие*, активизируются все **активные *sc-агенты***, **первичное условие инициирования*** которых соответствует произошедшему *sc-событию*.

Связки отношения **условие инициирования и результат*** связывают между собой *sc-узел*, обозначающий *абстрактный sc-агент* и бинарную ориентированную пару, связывающую условие инициирования данного *абстрактного sc-агента* и результаты выполнения данного экземпляров данного *sc-агента* в какой-либо конкретной системе.

Указанную ориентированную пару можно рассматривать как логическую связку импликации, при этом на *sc-переменные*, присутствующие в обеих частях связки, неявно накладывается квантор всеобщности, на *sc-переменные*, присутствующие либо только в посылке, либо только в заключении неявно накладывается квантор существования.

Первым компонентом указанной ориентированной пары является логическая формула, описывающая условие инициирования описываемого *абстрактного sc-агента*, то есть конструкции, наличие которой в *sc-памяти* побуждает *sc-агента* начать работу по изменению состояния *sc-памяти*. Данная логическая формула может быть как атомарной, так и неатомарной, в которой допускается использование любых связок логического языка.

Вторым компонентом указанной ориентированной пары является логическая формула, описывающая возможные результаты выполнения описываемого *абстрактного sc-агента*,

то есть описание произведенных им изменений состояния *sc-памяти*. Данная логическая формула может быть как атомарной, так и неатомарной, в которой допускается использование любых связей логического языка.

sc-описание поведения sc-агента представляет собой *sc-окрестность*, описывающую деятельность *sc-агента* до какой-либо степени детализации, однако такое описание должно быть строгим, полным и однозначно понимаемым. Как любая другая *sc-окрестность*, **sc-описание поведения sc-агента** может быть протранслировано на какие-либо понятные, общепринятые средства, не требующие специального изучения, например на естественный язык.

Описываемый *абстрактный sc-агент* входит в **sc-описание поведения sc-агента** под атрибутом *ключевой sc-элемент*'.

Под **sc-событием** понимается элементарное изменение состояния *sc-памяти*, которое может стать *первичным условием инициирования** какого-либо *sc-агента*.

Класс *sc-событий* в целом разбивается на следующие подклассы:

- *sc-событие добавления выходящей sc-дуги*
- *sc-событие добавления входящей sc-дуги*
- *sc-событие добавления инцидентного sc-ребра*
- *sc-событие удаления выходящей sc-дуги*
- *sc-событие удаления входящей sc-дуги*
- *sc-событие удаления инцидентного sc-ребра*
- *sc-событие удаления sc-элемента*
- *sc-событие изменения содержимого sc-ссылки*

3. Библиотека компонентов машин обработки знаний

Одним из важнейших компонентов интеллектуальной метасистемы поддержки проектирования интеллектуальных систем является библиотека совместимых компонентов таких систем, в частности, библиотека многократно используемых компонентов машин обработки знаний. Для машин обработки знаний роль совместимых компонентов, легко интегрируемых в различные интеллектуальные системы, выполняют *абстрактные sc-агенты*, а также их реализации, в особенности, платформенно-независимые.

Библиотека многократно используемых компонентов машин обработки знаний может быть декомпозирована на следующие частные библиотеки:

- Библиотека агентов информационного поиска
- Библиотека агентов погружения интегрируемого знания в базу знаний
- Библиотека агентов выравнивания онтологии интегрируемого знания с основной онтологией базы знаний
- Библиотека агентов поиска решения явно сформулированных задач

- Библиотека агентов логического вывода
- Библиотека агентов обнаружения и удаления информационного мусора
- Библиотека агентов-координаторов

В свою очередь, каждая из рассмотренных библиотек может быть декомпозирована по различным признакам.

Заключение

В данной работе рассмотрены понятия предметной области агентов машин обработки знаний, лежащие в основе технологии проектирования машин обработки знаний интеллектуальных систем в рамках Технологии OSTIS.

Библиографический список

- [IMS.OSTIS, 2013] Метасистема IMS.OSTIS [Электронный ресурс]. Минск, 2013. – Режим доступа: <http://ims.ostis.net/>. – Дата доступа: 30.12.2013.
- [OSTIS, 2013] Проект OSTIS [Электронный ресурс]. Минск, 2013. – Режим доступа: <http://ostis.net/>. – Дата доступа: 30.12.2013.
- [Голенков и др., 2001] Голенков, В.В. Представление и обработка знаний в графодинамических ассоциативных машинах / Голенков В.В. [и др.]; под ред. В.В. Голенкова – Минск, 2001.
- [Голенков и др., 2001] Программирование в ассоциативных машинах / Голенков В. В. [и др.]; под ред. В. В. Голенкова – Минск, 2001.
- [Петцольд, 2002] Петцольд, Ч. Программирование для Microsoft Windows на C#. В 2х томах / Ч. Петцольд; – М. : Издательско-торговый дом «Русская Редакция», 2002.
- [Тарасов, 2002] Тарасов, В.Б. От многоагентных систем к интеллектуальным организациям / В.Б. Тарасов; – М. :Изд-во УРСС, 2002.
- [Хорстманн, 2004] Хорстманн, К.С. Библиотека профессионала. Java 2. В 2х томах / К.С. Хорстманн, Корнелл Г.; – М. : Издательский дом «Вильмс», 2004.
- [Шилдт, 2004] Шилдт, Г. Полный справочник по C# / Г. Шилдт; – М. : Издательский дом «Вильмс», 2004.
- [Гамма, 2007] Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес; – СПб. : «Питер», 2007.

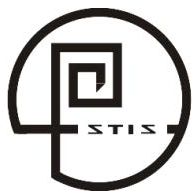
THE KNOWLEDGE PROCESSING MACHINE OF THE INTELLIGENT DESIGN SUPPORT METASYSTEM FOR INTELLIGENT SYSTEMS

Shunkevich D.V.

*Belarusian State University of Informatics and
Radioelectronics, Minsk, Republic of Belarus*

shu.dv@tut.by

This article is devoted to principles of organization and operating of knowledge processing machines of intelligent systems as whole, which underly the technology of knowledge processing machine design is borders of OSTIS technology. As an example of such a technology application the author considers knowledge processing machine of the intelligent metasytem IMS.OSTIS, which is oriented to the intelligent systems design support.



УДК 004.822:514

ПОСТРОЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ, ОСНОВАННЫХ НА ЗНАНИЯХ, С ПОВТОРНЫМ ИСПОЛЬЗОВАНИЕМ КОМПОНЕНТОВ

Борисов А.Н.

Рижский технический университет, г. Рига, Латвия

Arkadijs.Borisovs@cs.rtu.lv

В работе рассмотрены теоретические основы и средства построения интеллектуальных систем с многократным использованием компонентов на основе методов инженерной онтологии. Приведены два основных семейства инструментальных средств: модельно-ориентированный подход и подход, использующий декомпозицию решаемой проблемы. Описаны основные языки представления онтологий, а также программные средства построения онтологий. Рассмотрены средства представления декларативных знаний и механизмы обработки процедурных знаний, а также средства взаимодействия онтологических баз знаний и декларативных правил. Проанализированы дискуссионные вопросы развития реляционных языков и онтологической семантики.

Ключевые слова: инженерная онтология; повторное использование компонентов; системы, основанные на знаниях.

Введение

Традиционные системы, основанные на знаниях, разработанные в 70-80гг. при их реальном использовании встретили много проблем, показывающих их несовершенство.

Введение технологии «пустых» систем, основанных на знаниях, лишь частично сняло возникшие проблемы. Появилась идея повторного использования компонент при проектировании систем. Этот подход требует декомпозиции задачи на подзадачи, а именно - декомпозиции предметной области, декомпозиции проблемы. Каждая из частных полученных после декомпозиции подзадач требует своей порции знаний, которые позволят решать эти частные подзадачи. В этом случае понятие онтологии используется для описания таких частных подзадач.

1. Ранние системы, основанные на знаниях (70-80гг.)

Создатели первых систем, основанных на знаниях [Jackson, 1999], ожидали, что их системы будут идеально решать задачу повторно используемых компонентов [Болотова, 2012]. Эти системы состояли из двух частей:

- базы знаний, содержавшей продукционные правила, то есть декларативную часть;
- блока логического вывода, обеспечивавшего процедурную функцию.

Предполагалось, что данный подход может быть применён к различным решаемым задачам. Однако эти надежды не оправдались по следующим причинам:

- создание базы продукционных правил для новой предметной области оказалось весьма дорогостоящим;
- проблема «узкого горлышка» не может быть решена, так как существуют трудности описания предметной области в терминах понятий низкого уровня абстракции, необходимых для функционирования блока логического вывода.

2. Современные системы, основанные на знаниях

Стало очевидным, что экспертные знания, представленные на уровне продукционных правил или фреймов не обеспечивают достаточно абстрактную модель для создания интеллектуальных систем [Russel, 2010].

С целью абстрагирования от конкретной задачи и перехода к мета-описанию поведения системы предложены четыре основных подхода:

- понятие родовой, общей (generic) задачи, например, использование предметно независимого метода решения задачи;
- понятие ограниченной роли метода решения задачи, которое позволяет рассматривать, например, различные роли составляющих метода решения и, соответственно, определять порцию знаний для конкретной прикладной задачи;

- предположение об «открытости мира»;
- библиотека онтологического описания знаний.

3. Теоретические основы методов многократного использования компонентов

3.1. Предположение «открытого мира»

Онтология, реализованная на языке OWL (Web Ontology Language), представляет собой множество декларативных утверждений о сущностях словаря предметной области [Лапшин, 2010; Нерр, 2007]. OWL предполагает концепцию «открытого мира», в соответствии с которой применимость описаний предметной области, помещённых в конкретном физическом документе, не ограничивается лишь рамками этого документа – содержание онтологии может быть использовано и дополнено другими документами, добавляющими новые факты о тех же сущностях или описывающими другую предметную область в терминах данной. «Открытость мира» достигается путём добавления URI каждому элементу онтологии, что позволяет воспринимать описанную на OWL онтологию как часть всеобщего объединённого знания.

3.2. Библиотеки методов решения проблем

Библиотека Бенджамина [Gómez-Pérez, 1999] содержит набор методов решения задач (Problem Solving Methods-PSMs). Каждый PSM декомпозирует задачу на подзадачи и/или примитивные логические заключения и располагает структурой для контролирования выполнения подзадач и примитивных заключений, а также – для получения общего решения задачи. Подзадача – это задача, для решения которой может быть применен другой подходящий метод. Каждая задача имеет описание информации на входе и на выходе. Каждый PSM имеет множество критериев пригодности, которые используются для выбора метода в приложении. Эти критерии выражаются в терминах свойств среды, предметной области и имеющихся знаний; в методологии CommonKADS (Knowledge Acquisition and Documentation Structuring) все это называется свойствами задачи.

3.3. Общая задача

Считается, что онтология задачи является общей (generic), если она независима от специфики предметной области, или приложения, или метода вывода [Chandrasekaran, 1998]. Например, в проблеме планирования действий онтология задачи формализуется без указания любой специфики задачи планирования, но в то же время извлекает и объединяет важные концептуальные теоретические отличия, которые существуют в большинстве парадигм планирования. Понятие общей задачи может быть распространено на метод решения, на онтологию предметной области и является ключевым подходом для построения библиотеки

повторно используемых компонентов в процессе проектирования систем, основанных на знаниях.

3.4. Методы ограничения роли (Role-limiting methods – RLM)

Подход, использующий RLM, можно охарактеризовать как оболочный. Такая оболочка получается вместе с реализацией конкретного PSM, таким образом, её можно использовать только для решения такого типа задач, для которого PSM подходит. Данный PSM также определяет роли, которые знания могут играть в ходе процесса решения задач; он фиксирует представление знаний для ролей, так что эксперту необходимо подтвердить общие понятия и отношения, которые определяются с помощью этих ролей [Mizoguchi, 1995]. RLM, основанные на эвристической классификации, например, предлагают роль, точно определяемую экспертом. Используя эту роль, эксперт:

- должен указать, какое характерное для предметной области понятие соответствует этой роли, например, «данные о пациенте»;
- должен обеспечить экземпляры предметной области для этого понятия, например, конкретные факты о пациентах. Важно подчеркнуть, что тип знаний, который использует RLM, определен заранее. Следовательно, получение требуемых экземпляров, характерных для предметной области, может поддерживаться графическими интерфейсами, которые выполнены специально для данного PSM.

4. Инструментальные средства для построения систем, основанных на знаниях

4.1. Модельно-ориентированный подход

Исходной мотивацией для модели экспертизы KADS послужила поддержка вербальной информации от человека-эксперта, в частности, рассуждения вслух. Модель экспертизы это реализация независимой модели экспертных знаний о решении проблемы [Gašević, 2009]. Её можно разделить на четыре уровня по различным видам знаний с ограниченным взаимодействием между уровнями:

- *уровень предметной области*, который описывает концепты и отношения между ними;
- *уровень вывода*, который описывает элементарные выводы, которые можно применить к знаниям о предметной области, называемые источниками знаний;
- *уровень задачи*, со структурой управления, называемой структурой задачи, которая показывает, как можно комбинировать источники знаний, чтобы достичь цели;
- *стратегический уровень*, содержит знания о том, как создавать структуры задачи и находить решения в тупиковых случаях. Стратегический уровень обычно явно не

моделируется, в частности потому, что суждения о стратегии не кажутся важными при решении рутинной задачи, которая является объектом при разработке системы. В системе, основанной на знаниях, стратегический уровень отражает способность пользователя вникнуть в процесс решения задачи.

Так же как и технологии Protégé, KADS со временем развивается. Методология CommonKADS создавалась с целью использования более точного, формализованного языка для описания моделей экспертизы, а также - путём включения новых идей относительно компонентов экспертизы. CommonKADS также ввела идею библиотеки повторного использования методов решения проблем, построенной на базе моделей уровня вывода и уровня задач KADS.

4.2. Подход, основанный на декомпозиции задачи

С момента представления системы Protégé были введены ещё три версии (Protégé -I, Protégé -II, and Protégé/Win), пока не появилась текущая версия Protégé -2000 [Gómez-Pérez, 2004].

Современные версии Protégé написаны на языке Java, который позволяет создавать не привязанные к определённой операционной системе приложения. Т.е. если в целевой операционной системе установлена Java Machine – JDK нужной версии, то в ней можно использовать и Protégé версии выше 3.0.

Модель знаний Protégé -2000 основана на модели Open Knowledge Base Connectivity (OKBC), которая обеспечивает единообразную модель Knowledge Representation Systems (KRS) и является более гибкой, чем другие модели знаний, использованные в более ранних версиях Protégé. Знания сохраняются в специализированном формате. Это позволяет читать и писать в PDF файлы и использовать формат реляционных баз данных. Разработка инструментальной среды Protégé/Win изначально вызвана прагматическими соображениями: Protégé-II могла работать только под операционной системой NeXTStep, и чтобы расширить базу пользователей, необходимо было так переработать систему, чтобы она могла работать под операционной системой Windows. Однако самым главным вкладом Protégé/Win является расширение коллектива внешних пользователей. Protégé/Win свободно доступна как легко устанавливаемое приложение для любого пользователя. Впервые пользователь начал получать обратную связь от реального мира о проблемах и особенностях прикладной системы. Эта связь повлияла на развитие системы. Protégé-2000 позволяет пользователям вносить изменения в онтологию и предоставляет undo/redu функциональность. Более того, даётся возможность архивировать различные версии онтологии и вернуться к предыдущим версиям. Это один из немногих инструментов, которые поддерживают работу с версиями онтологии.

4.3. Подход, основанный на WWW. Проект IBROW3

Основная цель проекта IBROW3 (An Intelligent Brokering Service for Knowledge Component Reuse on the World-Wide Web) заключалась в разработке архитектуры, которая облегчала «интеллектуальный брокерский сервис» для создания системы, основанной на знаниях (Knowledge-based System – KBS), с помощью многократного использования компонентов знаний третьей стороны через WWW. Мета-онтология UPML (United Problem-Solving Method Development Language) была разработана для поддержки определения компонентов знаний. UPML вводит четырехкомпонентную архитектуру для систем знаний посредством расширения модели экспертизы CommonKADS. Идентифицируются следующие компоненты:

- *Задача.* Определяет проблему, которая будет решаться.
- *Метод решения задачи.* Описывает процесс вывода заключений.
- *Модель предметной области.* Определяет знания о предметной области.
- *Адаптер.* Облегчает соединение трех других компонент, определенных с помощью независимых и, следовательно, многократно используемых спецификаций.

5. Средства построения KBS на основе многократно используемых компонентов

5.1. Языки для описания онтологий

Существует семейство языков представления онтологий RuleML, включающее DAML+OIL, RDF & RDF Schema, OWL и пр. Рассмотрим некоторые из них.

5.1.1. RDF (Resource Description Framework)

RDF (Resource Description Framework) представляет собой технологическое решение для представления информации в WWW. Любое выражение RDF основано на коллекции так называемых триплетов, каждый из которых состоит из субъекта (subject), предиката (predicate) и объекта (object). Триплет представляет собой утверждение о связи между понятиями «субъект» и «объект», обозначенными как узлы, которые эта связь соединяет. Направление связи важно: оно всегда указывает на объект. Таким образом, RDF представляет модель (framework) для аннотации ресурсов Сети при помощи XML-синтаксиса и триплетной модели (Subject, Predicate, Object), которая имеет наглядное графическое представление (ориентированный размеченный граф).

5.1.2. RDF Schema

RDF Schema является языком описания пользовательских словарей (наборов понятий) при помощи стандартизованного набора тегов в формате RDF-триплетов, который и составляет основу RDF Schema. Для простоты можно считать, что RDF

Schema предоставляет систему типов для RDF. Сам по себе RDF не включает никаких механизмов ни для описания свойств, ни для описания отношений между этими свойствами и другими ресурсами. Эту роль играет RDF Schema, который определяет понятия «класс», «свойство» и некоторые другие, позволяющие описывать классы, свойства и другие ресурсы.

5.1.3. OWL – язык описания онтологий.

OWL является развитием DAML + OIL, и был разработан для использования приложениями, которым требуется обрабатывать информацию, а не просто предоставлять её агентам [Kamel, 2007]. OWL предлагает большую способность к взаимодействию, чем RDF(S), благодаря наличию дополнительного словаря вместе с поддержкой формальной семантики. У языка OWL есть три (по возрастанию выразительной мощности) подмножества: OWL Lite, OWL DL и OWL Full. OWL Lite содержит основные конструкции для описания классов и задания простых ограничений, OWL DL предлагает максимум возможностей, оставаясь при этом в рамках дескриптивной логики. OWL Full не гарантирует возможности существования вычислительной поддержки и рассчитан на пользователей, которые хотят получить максимальную выразительную мощность и свободу от RDF. В современных версиях стандарта OWL Lite отсутствует.

6. Среды для построения онтологий

Существует большой набор сред для поддержания онтологий: OilEd, Apollo, RDFedt, OntoLingua, OntoEdit, WebODE. Рассмотрим некоторые из них.

6.1. RDFedt

RDFedt даёт возможность пользователю строить сложные и структурированные RDF and RSS документы. Он обеспечивает обзор сложных структур данных с элементами дерева и позволяет пользователю тестировать данные, делать комментарии и выдавать сообщения об ошибках с помощью дополнительных функций. RDFedt поддерживает основные элементы RDF, RDFS, и Dublin. RSS 1.0 обеспечивает такие модули как агрегирование, нотация, организация, изменение страницы, поточность и т.д. RSS 0.91 поддерживает декларирование и уровни стилей в XML, множества импортируемых элементов и автоматическое генерирование связанного списка на базе RDF из HTML (hypertext markup language) документа. RDFedt представляет собой текстовый редактор языков. Это не программа Java, этот редактор не является платформно-независимым, он работает только под Windows.

6.2. OntoLingua

OntoLingua разработана в лаборатории систем, основанных на знаниях, Стэнфордского университета. Эта библиотека обеспечивает среду

сотрудничества для распределённых пользователей, набор инструментов для авторинга онтологий и библиотеку модульных, многократно используемых онтологий. Она также поддерживает (WWW) интерфейс и перевод в различные форматы. OntoLingua представляет собой библиотеку онтологий и сервер, доступ к которым можно получить через традиционный Web браузер. С использованием модуля Chimaera реорганизуется таксономия и решаются конфликты имён в базе знаний. Если пользователей несколько, они могут использовать библиотеку OntoLingua через блокировку опции *только для записи* и назначение уровня доступа пользователя.

7. Механизмы обработки процедурных правил

Существует набор средств обработки процедурных знаний: JENA, Pellet, JESS, RACER, OO jDREW.

7.1. Jess

Jess является аббревиатурой названия Java Expert System Shell. Jess представляет собой механизм правил и среду сценариев, написанные целиком на языке Java. Разработка Jess первоначально вдохновлялась оболочкой экспертной системы CLIPS, но затем переросла в законченную самостоятельную среду, на которую повлияла Java. Используя Jess, можно создавать Java-апплеты и приложения, которые обладают способностью "делать заключения", используя знания, которые представлены в форме декларативных правил. Так же как и CLIPS, Jess имеет подобный Lisp синтаксис, который обозначает или «Обработка списков» или «Множества избыточных скобок» в зависимости от того, кому адресуется этот вопрос.

7.2. Jena

Jena является фреймворком для семантических Web приложений, который включает API (Application Programming Interface) для RDF. Он поддерживает постоянное хранение, механизм запросов SPARQL и основанный на правилах механизм вывода для RDFS и OWL. Jena обеспечивает прямой и обратный вывод и гибридную модель выполнения вывода, а также - способы комбинирования вывода для RDFS/OWL с выводами на созданных правилах.

7.3. Pellet

Pellet это общедоступная машина вывода на базе Java, которая поддерживает вывод на уровне OWL-DL. Она представляет собой удобное средство для реализации тщательного и полного OWL DL вывода. Pellet обеспечивает такие функции вывода, как проверка на непротиворечивость, выполнимость, классификация и реализация понятий. Pellet включает также оптимизированный

механизм запросов, способный отвечать на запросы Abox.

8. Языки логического вывода на основе информации, представленной онтологиями

8.1. RuleML Initiative

RuleML Initiative разрабатывает открытый язык, основанный на правилах, для обработки RDF онтологий, что обеспечивает обмен правилами между различными системами включая распределенные программные компоненты на Web, неоднородные клиент-сервер системы, которые встречаются в больших корпорациях, и т.д. Язык RuleML использует XML-синтаксис для представления знаний в виде правил, а также - для организации взаимодействия между основными коммерческими и некоммерческими системами правил. RuleML строит иерархию подязыков правил на базе XML, RDF, и OWL, например, SWRL.

8.2. SWRL (Semantic Web Rule Language)

SWRL (Semantic Web Rule Language) был предложен для того, чтобы увеличить возможности онтологии на основе OWL. SWRL правила обеспечивают процедурные знания, которые компенсируют некоторые ограничения логического вывода на базе онтологий, в частности, в плане идентификации семантических отношений между экземплярами. SWRL использует типичное логическое выражение антецедент \rightarrow консеквент для представления семантических правил. Как антецедент, так и консеквент могут быть конъюнкцией одного или более атомов.

9. Средства, поддерживающие взаимодействие онтологий и правил

9.1. Protégé OWL плагин

Машина вывода Jess функционирует в рамках среды Protégé и является основой для интеграционной модели плагина JessTab. Так как Protégé и Jess выполнены при помощи языка Java, их можно запускать вместе в единой виртуальной машине Java. Этот подход даёт возможность использовать Jess как интерактивное средство для действий с онтологиями и базами данных Protégé. Более того, становится возможным распространить процедуру изменений в Protégé на оболочку Jess.

9.2. SWRL JESS Tab

Выполнение правил SWRL требует наличия машины обработки правил, которая осуществляет вывод, используя в качестве входной информации множество правил и фактов. Любые новые выводимые факты используются как входные данные с тем, чтобы активировать по возможности больше правил (т.е., прямой вывод). В большинстве реализаций используется машина обработки правил

Jess для того, чтобы сделать возможным SWRL вывод на множестве данных Protégé [Eriksson, 1995]. SWRL-правила хранятся как OWL экземпляры вместе с ассоциируемой с ними базой знаний. Прежде всего, SWRL-правила переводятся в Jess правила, используя SWRL JessBridge, который встроен в Protégé и добавлен в машину обработки правил Jess; затем онтологии и база знаний транслируются в Jess факты а также – вводятся в машину вывода; в-третьих, машина обработки правил Jess делает возможным вывод и производит результаты в формате Jess и, наконец, эти результаты переводятся обратно в формат OWL. SWRL-правила можно легко моделировать, используя механизм SWRL Jess Tab, который включен в Protégé и обеспечивает графический интерфейс для взаимодействия с механизмом SWRL JessBridge. Механизм SWRL Jess Tab позволяет вводить, удалять и редактировать SWRL-правила.

10. Вопросы для обсуждения

10.1. Реляционные языки в онтологиях

Ситуационные модели управления одними из первых начали осуществлять формализацию объекта на языке, близком к естественному [Поспелов, 1986]. Основными сохраняемыми знаками являлись знаки двух видов: *понятия* и *отношения*. Главные идеи языка ситуационного управления основаны на работах Мартынова В.В. об универсальном семантическом коде (УСК) [Мартынов, 1997].

Простой ядерной конструкцией в универсальном семантическом коде является тройка вида (SAO). В этой тройке S соответствует субъекту, совершающему акцию, а О – объекту, на который направлена данная акция А. Такой тройке может, например, соответствовать предложение «Кран разгружает судно». В УСК вводится замкнутая система операций, позволяющих из простых ядерных конструкций строить более сложные цепочки.

Дальнейшее развитие реляционных языков связано с разработкой RDF представления, которое выражает семантические отношения на уровне примеров (экземпляров), выраженные в терминах тройки. В то же время, RDF Schema выражает отношения на уровне классов, описывая приемлемые отношения на уровне примеров (экземпляров), выраженные в терминах тройки. RDFS является мета-уровнем или более абстрактной моделью, описывающей уровень объектов в RDF. На сегодняшний день RDFS и OWL полностью удовлетворяют запросы онтологических систем как в концептуальном плане, так и в плане поддерживающих средств.

Совершенно естественно возникает вопрос о дальнейших путях развития реляционных языков.

10.2. Онтологическая семантика

В логике высказываний и в логике предикатов семантика операций определяется с помощью интерпретаций смысла. В существующей интерпретации смысла в данном случае семантика полностью удовлетворяет потребностям практики [Рубашкин, 2013]. В системах представления знаний, использующих семантические сети, можно также воспользоваться интерпретациями, принятыми в логике предикатов. Как известно, графическая интерпретация логики предикатов может быть осуществлена на основе объединения концептуальных графов и семантической сети. Таким образом, при смысловой интерпретации семантических сетей можно воспользоваться опытом, полученным при решении подобной задачи в логике предикатов.

Язык RDF, обладая высокой степенью общности, не позволяет варьировать глубиной и точностью семантических описаний. На решение этой задачи нацелен другой язык системы, OWL. В составе OWL продублированы средства формирования словаря семантических примитивов, разработанные для RDF. Тем самым обеспечена системная совместимость языков. Наряду с этим OWL позволяет уточнить смысл того или иного семантического примитива прежде всего за счет внесения ограничений на кардинальность отдельных классов и на использование отношений между классами. В этом случае при внесении ограничений на кардинальность в дескриптивной логике нарушается предположение об открытости мира. Так как в замкнутом мире действуют законы логики предикатов, здесь также можно воспользоваться методами интерпретации, принятыми в логике предикатов.

В общей системе RDF/OWL язык OWL играет роль инструмента для задания аксиоматики словаря семантических примитивов, в то время как RDF ориентирован, прежде всего, на семантическое представление конкретных фактов в рамках заданной на языке OWL аксиоматики. Системой RDF/OWL оформлена новая технология описания и представления семантики данных – технология Semantic Web.

Каковы альтернативные пути семантической интерпретации онтологии для открытых миров?

Заключение

Привлечение методов и средств инженерной онтологии стимулировало интенсивное развитие систем, основанных на знаниях, нового поколения. Одним из новых направлений является проектирование систем с повторным использованием компонентов. Разработаны фундаментальные теоретические основы этого подхода, однако для их успешной реализации на практике предстоит еще проделать большую работу.

Библиографический список

- [Мартынов, 1997] Мартынов В.В. Универсальный семантический код / В.В. Мартынов; – Минск: Наука и техника, 1997.
- [Поспелов, 1986] Поспелов Д.А. Ситуационное управление. Теория и практика / Д.А. Поспелов; – М.: Наука, 1986.
- [Болотова, 2012] Болотова Л.С. Системы искусственного интеллекта: модели и технологии, основанные на знаниях / Л.С. Болотова; – М.: Финансы и статистика, 2012.
- [Лапшин, 2010] Лапшин В.А. Онтологии в компьютерных системах / В.А. Лапшин; – М.: Научный мир, 2010.
- [Рубашкин, 2013] Рубашкин В.Ш. Онтологическая семантика / В.Ш. Рубашкин; – М.: Физматлит, 2013.
- [Chandrasekaran, 1998] Chandrasekaran B. Ontology of Tasks and Methods / B.Chandrasekaran, J.R.Josephson, V.R. Benjamins // IEEE Intelligent Systems, 14(1), 1998, P. 20-26.
- [Eriksson, 1995] Eriksson H. Task modeling with reusable problem-solving methods / H. Eriksson // Artificial Intelligence. - Vol. 79, Issue 2. - 1995, P. 293–326.
- [Gašević, 2009] Gašević D. Model Driven Engineering and Ontology Development / D. Gašević; – Springer-Verlag, 2013.
- [Gómez-Pérez, 2004] Gómez-Pérez A. Ontological Engineering / A. Gómez-Pérez, M. Fernández-López, O. Corcho; – Springer-Verlag, 2004.
- [Hepp, 2007] Hepp M. Ontologies: State of the Art, Business Potential, and Grand Challenges, Chapter 1 / M. Hepp // Ontology Management: Semantic WEB, Semantic WEB Services, and Business Application, 2007, Springer, P. 3-22.
- [Kamel, 2007] Kamel M.N. A methodology for developing ontologies using the Ontology Web Language (OWL) / M.N. Kamel // ICEIS 2007 - Proceedings of the Ninth International Conference on Enterprise Information Systems, June 12-16, Funchal, Madeira, Portugal, 2007, Springer, P. 261-268.
- [Mizoguchi, 1995] Mizoguchi R., Vanwelkenhuysen J, Ikeda M. Task ontology for reuse of problem solving knowledge / R. Mizoguchi, J. Vanwelkenhuysen, M. Ikeda // Proceedings of the Second International Conference on Building and Sharing of Very Large-Scale Knowledge Bases (KB & KS'95), 1995, P. 45-59.
- [Gómez-Pérez, 1999] Gómez-Pérez A., Benjamins V.R. Applications of ontologies and problem-solving methods / A. Gómez-Pérez, V.R. Benjamins // AI Magazine. - Vol. 20, N 1. - 1999, P. 119–122.
- [Russel, 2010] Russel S., Norvig P. Artificial Intelligence: A Modern Approach. 3rd Edition / S. Russel, P. Norvig; – Prentice Hall, 2010.
- [Jackson, 1999] Jackson P. Introduction to Expert Systems. 3rd Edition / P. Jackson; – Addison-Wesley, 1999.

ONTOLOGY-BASED INTELLIGENT SYSTEM CONSTRUCTION THROUGH COMPONENT REUSE

Borisov A.N. *

**Riga Technical University, Riga, Latvia*

Arkadijs.Borisovs@cs.rtu.lv

This work considers the theoretical foundations and tools for ontology-based intelligent system construction through component reuse. Two major tool families are discussed: the model-driven approach and the technique using the decomposition of the problem to be solved. Main ontology representation languages as well as ontology construction software are described. Declarative knowledge representation tools and mechanisms for processing procedural knowledge are examined as well as ontology knowledge base and declarative rule cooperation tools. Discussion questions addressing the development of relational languages and ontology semantics are analysed.



УДК 004.522: 912

О РЕШЕНИИ ЗАДАЧ СНЯТИЯ ОМОНИМИИ ПРИ РАСПОЗНАВАНИИ И СИНТЕЗЕ РЕЧИ

Лобанов Б.М. *, Житко В.А. **

* *Объединённый институт проблем информатики НАН Беларуси, Минск, Беларусь*

Lobanov@newman.bas-net.by

** *Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

zhitko.vladimir@gmail.com

В статье рассматриваются вопросы снятия омонимии при распознавании и синтезе речи, используя результаты синтаксического и семантического анализа текста. В качестве инструментальных средств анализа текстов русского языка используется программный комплекс автоматической обработки текстов – АОТ. Приводятся примеры снятия омонимии при распознавании речи (омофоны и паронимы) и синтезе речи (омографы).

Ключевые слова: распознавание речи; синтез речи; синтаксический анализ; семантический анализ; омонимия; омографы; омофоны.

Введение

При распознавании и синтезе речи возникает ряд проблем связанных с синтаксической, лексической, грамматической, словообразовательной и фонетической неоднозначностью естественного языка (см. словарь лингвистических терминов [Розенталь, 1976]). Один из источников неоднозначности является омонимия, явление, при котором некоторые слова естественного языка могут иметь одинаковое написание (графическая омонимия) или произношение (фонетическая омонимия).

Если слова имеют одинаковое написание, но различное произношение, такие слова называются омографы, например «за́мок» — «замóк», «ду́хи» — «духи́», «вёсти — вестí», «бо́чка — бочка́», «бе́рег — бере́г», «сте́кла» — «стекла́», «все — всё». Такие слова чаще всего имеют различие в позиции ударения или в написании буквы «е» без точек. Задача снятия омографии особо остро стоит при синтезе речи по тексту, так как неправильная интерпретация омографов может значительно ухудшить, а в ряде случаев и исказить смысл синтезируемого текста.

Если слова, напротив, имеют одинаковое произношение (одинаковую фонетическую транскрипцию), но различное написание — их называют омофоны, например: «порог-порок-парок»

— «прок», «луг-лук» — «лк», «муш-мушь» — «тш», «бал-балл» — «bl».

Если же слова имеют как одинаковое написание, так и одинаковое произношение их называют полными омонимами, например слово «бор» может обозначать химических элемент или сосновый лес, слово «эфир» может быть как в смысле органического вещества, так и в смысле «радиовещание и телевидение».

Кроме того существует ещё класс слов называемых паронимами — слова, сходные по звучанию (близкие по фонемному составу и акцентной структуре), но разные по значению и строению, часто ошибочное употребление одного из них вместо другого. Например: «адресат» — «адресант», «экскаватор» — «эскалатор», «абонемент» — «абонент», «экономический» — «экономичный» — «экономный», «ординарный» — «одинарный».

Такого рода неоднозначности может «запутать» систему распознавания, что приведет к тому, что реакция системы может оказаться нелепой или ошибочной. Например, пользователь хочет узнать список лекарств от головной боли: «Что поможет если голова болит?», однако система может не выдать правильного ответа, так как будет пытаться обработать запрос «Что поможет, если голова — болид?».

Более сложный пример ошибки при паронимии:

пользователь, задавая различные вопросы о театре и кино, на очередной вопрос «*Кто такая Сара Бернар?*» получить неожиданный ответ «*Сенбернар — порода собак. Происходит от азиатских догообразных*», т.к. система восприняла «*Сара Бернар*» как «*Сенбернар*» и «решила», что пользователь спрашивает о сенбернарах.

Решение такого рода задач, называемых задачами разрешения (снятия) омонимии, во многом удаётся достичь методами статистического анализа текстов. Однако, окончательное её решение возможно только с использованием методов синтаксического и семантического анализа, являющихся составной частью систем автоматической обработки текстов.

Системы лингвистического анализа текстов

Для решения задач анализа текстов существует множество различных систем, ориентированные под разные задачи и языки. Наиболее известной системой лингвистического анализа текста для английского языка является Stanford Parser [NLP Stanford, 2013]. Для нашей задачи по ряду причин наиболее подходящим выглядит программный комплекс АОТ для русского языка [АОТ, 2013], который обеспечивает:

- поддержку русского языка;
- поверхностно-семантический анализ текста;
- открытость исходного кода.

Программный комплекс автоматической обработки текстов АОТ включает в себя следующие компоненты [Sokirko, 2001]:

- графематический анализ – выделение слов, цифровых комплексов, дат, формул и пр.;
- морфологический анализ – морфологическая интерпретация слов;
- синтаксический анализ – построение дерева синтаксических зависимостей текста;
- семантический анализ – построение поверхностно-семантического графа.

Для каждого компонента комплекса существует свой язык представления данных, состоящий из понятий текущего уровня абстракции структуры текста и правил их комбинирования.

Для решения поставленной задачи разрешения омонимии будут использованы результаты синтаксического и семантического уровней.

Синтаксический анализ – один из этапов обработки текста, задачей которого является построение синтаксических групп на одном морфологическом варианте простого предложения с использованием синтаксических правил. Понятия, используемые в языке представления на синтаксическом уровне разбора текста, включают в себя типы синтаксических фрагментов и синтаксических групп [Панкратов, 2000].

Типы синтаксических фрагментов включают в себя:

- деепричастный оборот (ДПР),
- причастный оборот (ПРЧ),
- вводный оборот (ВВОД),
- необособленное согласованное определение в препозиции (НСО),
- фрагмент с личной формой глагола (ГЛ_ЛИЧН),
- фрагмент с кратким причастием (КР_ПРЧ),
- фрагмент с кратким прилагательным (КР_ПРИЛ),
- фрагмент с предикативом (ПРЕДК), с инфинитивом (ИНФ),
- фрагмент с тире (ТИРЕ),
- фрагмент со сравнительным прилагательным (СРАВН).

К типам синтаксических групп относятся:

- количественная группа (КОЛИЧ);
- последовательность чисел вперемешку со знаками препинания (КОЛИЧ);
- существительное из заданного перечня и числовой идентификатор (СУЩ-ЧИСЛ);
- слова степени (типа «очень») с группой прилагательного или причастия (НАР_ПРИЛ);
- однородные прилагательные (ОДНОР_ПРИЛ);
- однородные наречия (ОДНОР_НАР);
- однородные инфинитивы (ОДНОР_ИНФ);
- однородные прилагательные сравнительной степени (ОДНОР_ПРИЛ);
- группы даты (ДАТА);
- группа временных отрезков (СЛОЖ_ПГ);
- аналитическая форма сравнительной степени прил. Или наречия (СРАВН-СТЕПЕНЬ);
- наречие и глагол (НАРЕЧ-ГЛАГОЛ);
- одно или несколько прилагательных, согласованных по роду, числу и падежу со стоящим сразу после них существительным (ПРИЛ-СУЩ);
- наречное числительное и именная группа (НАР-ЧИСЛ-СУЩ);
- числительное и именная группа (ЧИСЛ-СУЩ);
- генитивная пара (ГЕНИТ_ИГ);
- предложная группа (ПГ);
- однородные именные группы (ОДНОР_ИГ);
- отрицание и глагольная форма (ОТР_ФОРМА);
- глагольная форма и контактное прямое дополнение (ПРЯМ_ДОП);
- группа электронного адреса (ЭЛ_АДРЕС);
- глагольная форма и контактный инфинитив (ГЛАГ_ИНФ);
- подлежащее (ПОДЛ);
- сказуемое. (вершина клаузы).

Семантический анализ – этап обработки текста задачей, которого является построение

семантической структуры предложения, состоящей из семантических узлов (понятий) и семантических отношений на этих понятиях.

Каждому семантическому узлу приписан ряд атрибутов:

- набор графематических слов, из которых состоит данный узел;
- номер семантически главного слова в узле;
- грамматическая интерпретация узла (внешняя синтаксическая характеристика);
- номер фрагмента (клаузы), которому принадлежит узел;
- предлог, который в синтаксисе управляет этим узлом;
- ссылка на словарную статью в семантических словарях, которая является интерпретацией этого узла (может быть не определена);
- ссылка на словарную статью открытого словосочетания и номер элемента в поле СОСТАВ этого словосочетания (может быть не определена) и др.

Параметрами семантического отношения могут быть:

- имя отношения;
- ссылка на словарную статью, откуда было взято это отношение, и номер валентности в этой статье;
- перечень русских слов, которые являются лексическими реализациями этого отношения во входном тексте (предлоги, союзы и т.д.);
- русское синтаксическое отношение, которое является реализацией этого семантического отношения.

В языке представления используются следующие понятия [Леонтьева, 1997]:

- Автор(AUTHOR)
- Агент (AGENT)
- Адресат (ADR)
- В направлении (IN-DIRECT)
- Время (TIME)
- Значение (VALUE)
- Идентификатор (IDENT)
- Имя (NAME)
- Инструмент (INSTR)
- Исходная точка (SRC-PNT)
- Контрагент (C-AGENT)
- Количество (QUANTIT)
- Конечная точка (TRG-PNT)
- Локация (LOC)
- Масштаб (SCALE)
- Материал (MATER)
- Назначение (PURP)
- Объект (OBJ)
- Ограничение (RESTR)
- Оценка (ESTIM)
- Параметр (PARAM)

- Пациент (PACIEN)
- Посредник (MEDIATOR)
- Признак (PROPERT)
- Принадлежность (BELNG)
- Причина (CAUSE)
- Результат (RESLT)
- Содержание (CONTEN)
- Способ (METHOD)
- Средство (MEANS)
- Степень (DEGREE)
- Субъект (SUB)
- Тема (THEME) Цель (AIM)
- Часть (PART)

Для каждого синтаксического варианта фрагмента строится множество семантических представлений, лучшее из которых и является результатом работы семантического анализа.

Решение задач снятия фонетической омонимии при распознавании речи

Решение задач автоматического распознавание речи (Automatic Speech Recognition (ASR)) может сильно варьироваться от формы устной речи: дикторская речь, спонтанная речь, разговоры (диалоги, обсуждения) и голосовые команды. Мы ограничимся здесь рассмотрением случаев, когда на вход модуля распознавания речи поступают отдельные речевые команды в виде достаточно простых фраз, состоящих из небольшого количества слов. Обработка начинается с получения нескольких гипотез от модуля распознавания речи. Так как омофоны фонетически неразличимы то и их вероятности в гипотезах будут равными. В дальнейшем остаётся решить задачу снятия омонимии, т.е. решить какой из гипотез (распознанных фраз) отдать предпочтение.

Процесс снятия омонимии можно представить как последовательность следующих шагов:

- отсеивание гипотез на основе синтаксического анализа;
- отсеивание гипотез на основе семантического анализа;
- отсеивание гипотез на основе анализа семантической (и-или статистической) близости словосочетаний.

На первом этапе для каждой из гипотез строится синтаксическое дерево разбора. Далее отсеиваются гипотезы, для которых дерево разбора не было построено, что указывает на синтаксическую некорректность распознанных предложений. Если на этом шаге осталось только одна гипотеза, то дальнейшего анализа не происходит, а оставшаяся гипотеза возвращается как результат. На данном этапе есть возможность снятия омонимии, если омонимы имеют различные морфологические характеристики.

Например: «лес» — «лез», «старожил» — «сторожил», «течь» (протекать) и «течь» (протекание).

На втором этапе для всех гипотез строится поверхностно-семантический граф. Далее отсеиваем гипотезы, для которых дерево разбора не было построено, что указывает на то, что не все понятия в предложении имеют общие семантические отношения с остальными понятиями в предложении. Если на этом шаге осталось только одна гипотеза, то дальнейшего анализа не происходит, а оставшаяся гипотеза возвращается как результат. На данном этапе есть возможность снятия омонимии, если омонимы имеют различные семантические характеристики, то есть участвуют в различных семантических отношениях, что значит, имеют различный смысл или домен в иерархии понятий.

Третий этап анализа необходим в тех случаях, когда на предыдущих этапах не удалось снять омонимию. Это может случиться, если более чем для одной гипотезы были построены корректные как синтаксическое, так и семантическое деревья, или противоположный случай, когда ни для одной из гипотез не было построено корректного синтаксического или семантического дерева разбора. В таком случае нельзя однозначно сказать какая из гипотез верная, но можно сделать предположение, основываясь на следующих признаках:

- близость одного из омонимов к контексту диалога;
- большая статистическая вероятность встречаемости словосочетания в одной из гипотез (N-граммы).

Для учета контекста необходимо сохранять список значащих понятий последних обработанных предложений. Такой список отражает тематическую [Валгина, 1998] последовательность связи нескольких последних предложений диалога. Список понятий контекста диалога должен быть ограничен по длине и построен по принципу очереди: «первым добавили – последним изъяли».

Расчет близости понятий рассчитывается как кратчайший путь в семантическом графе связи понятий: по ребрам отношений гипоним/гипероним и синоним. Для этого удобно использовать различные тезаурусы, такие как Wiktionary (wiktionary.org), RussNet (project.phil.spbu.ru/) или UNLWEB (http://unlweb.net). Среди гипотез выбирается та, среднее расстояние которой ближе к текущему контексту диалога.

Приведём далее несколько примеров снятия омонимии рассмотренными выше способами.

В том случае если в ряде омофонов есть морфологические различия, задача разделения омофонов, как уже было сказано, может быть решена уже на первом этапе методами синтаксического анализа. Рассмотрим омофоны

«старожил — сторожил» во фразе «Где деревенский старожил лет сорок с ключницей бранился» (Пушкин). На выходе модуля распознавания речи появилось две гипотезы: «Где деревенский старожил лет сорок с ключницей бранился», «Где деревенский сторожил лет сорок с ключницей бранился». Используя синтаксический анализатор АОТ на этих фразах мы получим результаты отображенные на рисунке 1.



Рисунок 1 – Дерево синтаксического разбора для фразы: «где деревенский старожил/сторожил лет сорок с ключницей бранился»

Из рисунка 1 видно, что полное синтаксическое дерево удалось построить для первого примера и проводить дальнейший семантический анализ нет необходимости.

Рассмотрим более сложный пример омофонов «балл — бал» во фразе «Мы решили поехать на осенний бал». На выходе модуля распознавания речи две гипотезы: «Мы решили поехать на осенний бал», «Мы решили поехать на осенний балл». Используя синтаксический анализатор АОТ, для этих фраз мы получим результаты, представленные на рисунке 2.



Рисунок 2 – Деревья синтаксического разбора для гипотез «Мы решили поехать на осенний бал/балл»

Так как омофоны «балл — бал» имеют сходные морфологические характеристики то и деревья разбора предложений получились сходными и полными. В этом случае исходя из синтаксического анализа нельзя сделать вывод о достоверности гипотез, и мы переходим к семантическому анализу данных фраз.

Для проведения анализ на втором этапе методами семантического анализа нам необходимо знать семантические характеристики каждого омофона, для этого в программном комплексе АОТ используется Русский ОбщеСемантический Словарь (РОСС). В статьях семантического словаря указаны различные семантические отношения, в которых

может участвовать описываемое слово (понятие). Используя поверхностно-семантический анализатор АОТ, мы получим граф, представленный на рисунке 3.

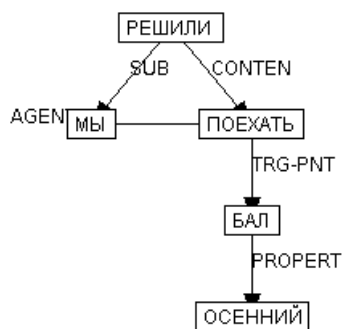


Рисунок 3 – Дерево семантического разбора для фразы «Мы решили поехать на осенний бал»

Слова «*поехать*» и «*бал*» связаны отношением «конечный пункт». Для гипотезы «*Мы решили поехать на осенний бал*» построить связный граф нельзя, т.к. слово «балл» не может участвовать в отношении «конечный пункт». Таким образом, у нас осталась только одна гипотеза и дальнейшего анализа не требуется.

Рассмотрим пример решения задачи снятия омонимии на третьем этапе анализа распознанной фразы. Если омонимы обладают сходными морфологическими характеристиками их сложно разделить на этапе синтаксического анализа. В том случае если омонимы обладают, к тому же и сходными семантическими характеристиками либо о некоторых омонимах нет информации, то разделить их на этапе семантического анализа так же нет возможности. Примерами омонимов со сходными синтаксическими и семантическими характеристиками могут служить: «*пруд*» – «*прут*», «*предать*» – «*придать*» и др. Семантическая информация о словах профессиональной лексики, жаргонов и других, редко используемых слов, может отсутствовать в семантическом словаре. Пример такого рода омонимов: «*осветить*» – «*освятить*», «*лук*» – «*луг*» и др.

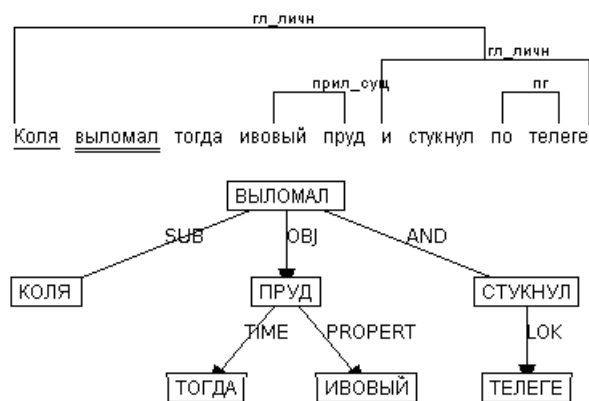


Рисунок 4 – Деревья синтаксического и семантического разбора для гипотез «Коля выломал тогда ивовый пруд/прут и стукнул по телеге»

Рассмотрим произнесённую фразу: «*Коля выломал тогда ивовый пруд и стукнул по телеге*»

(В. Белов) и гипотезы её распознавания: «*Коля выломал тогда ивовый пруд и стукнул по телеге*» и «*Коля выломал тогда ивовый пруд и стукнул по телеге*». Синтаксические и семантические деревья разбора, построенные программным комплексом АОТ для этих гипотез выглядят одинаково и представлены на рисунке 4.

Так как синтаксический и семантический этапы не дали результата, переходим на третий этап анализа. Рассчитаем минимальное расстояние для понятий «*пруд*» и «*прут*» с другими понятиями из их отношений (анализ на базе семантического словаря Wiktionary). Для понятия «*пруд*»: «*пруд*» – «*водоем*» – «*природные географические объекты*» – «*география*» – «*естественные науки*» – «*ботаника*» – «*растения*» – «*ива*» – «*ивовый*» итого 8 переходов между понятиями «*пруд*» и «*ивовый*». Для понятия «*прут*»: «*прут/ветка/побег*» – «*ботанические термины*» – «*ботаника*» – «*растения*» – «*ива*» – «*ивовый*» итого 5 переходов и 2 перехода по синонимам между понятиями «*прут*» и «*ивовый*». Таким образом, находим, что использование понятия «*прут*» более вероятно в данном контексте фразы.

При рассмотрении гипотез «*Иван осветил фонариком*» и «*Иван освятил фонариком*» обе гипотезы являются грамматически и семантически корректными. Для того чтобы выбрать более правдоподобную из них, необходимо: рассчитать расстояние между понятиями «*осветить*» и «*фонарик*» («*осветить*» – «*свет*» – «*фонарик*») и понятиями «*освятить*» и «*фонарик*». Очевидно, что расстояние в первом случае будет значительно меньше, что говорит о большей вероятности первой гипотезы.

При практическом использовании систем распознавания речи наиболее часто приходится иметь дело не со словами-омофонами (полностью совпадающими по звучанию, т.е. по фонемному составу), а со словами-паронимами, близкими по фонемному составу и по акцентной структуре. Рассмотрим пример распознавания устной речи системой «СТЕНОГРАФ», построенной на основе Интернет приложения *Google Speech Recognition* [Житко, 2012].

Произнесённая фраза: **Они собрали все кости.**
Варианты гипотез системы «СТЕНОГРАФ»:

- они собрались все гости
- они собрали все гости
- они собрали все кости
- они собрали всех кости
- они собрались все кости
- они собрание всех кости
- они собрать и все кости
- они собрать всех кости
- они собрали все кости

В данном примере паронимами являются слова: «*собрали-собрать-собрание-собрались*» и «*гости-кости*». Как видно из полученного результата,

истинную гипотезу «СТЕНОГРАФ» поместил на последнее место. Это произошло оттого, что в основе стратегии распознавания *Google* лежат статистические методы и фраза «они собрались все гости» оказалась значительно более вероятной чем: «они собрали все кости».

В описанном эксперименте для произнесённой фразы «Они собрали все кости» после синтаксического анализа остались следующие гипотезы: «они собрали все кости», «они собрали всех кости», «они собрали все гости», «они собрались все кости», «они собрались все гости».

После семантического анализа осталась только одна верная гипотеза: «они собрали все кости». Синтаксическое и семантическое деревья разбора представлены на рисунке 5.



Рисунок 4 – Деревья синтаксического и семантического разбора
для гипотезы
«они собрали все кости»

Решение задач снятия графической омонимии при синтезе речи по тексту

В русском языке существуют два источника графической омонимии: вариативность словесного ударения, местоположение которого в письменной речи не указывается (СУ- омографы), и письменная традиция не обязательного проставления необходимых точек на букве «Ё» («Ё»-омографы) [Лобанов, 2009]. Там же показано, что при обследовании достаточно представительного корпуса текстов, средний процент вхождения омографов составил 3,15%. Если считать, что среднее число слов на странице равно 450, то около 15-ти слов на странице могут оказаться омографами. В случае их неадекватного раскрытия это приводит к весьма негативному впечатлению при прослушивании синтезированной речи.

В работе [Цирульник, 2009] описаны некоторые контекстуальные правила разрешения графической омонимии при синтезе речи по тексту. В работе [Иомдин, 2011] для этой же цели использован синтаксический анализатор ЭТАП-3. Было показано, что применение синтаксического анализатора ЭТАП-3 в интегрированной системе речевого синтеза во многих случаях снимает проблему правильной передачи омографичных

словоформ текста. Но это происходит не всегда, и в некоторых случаях всё ещё остаётся потребность в использовании дополнительного семантического анализа омографической неоднозначности.

Рассмотрим и проанализируем некоторые примеры фраз, включающих по 2 омографа, допускающих их альтернативное прочтение синтезатором речи:

Но в другом случае он был ее другом.

Но в другом случае он был ее другом.

На самом деле причина была в самом человеке.

На самом деле причина была в самом человеке.

Скоро минут несколько минут.

Скоро минут несколько минут.

Они подождали пока минут несколько минут.

Они подождали пока минут несколько минут.

На горе соседям он оказался на горе.

На горе соседям он оказался на горе.

Для черта это была запретная черта.

Для черта это была запретная черта.

Этот берег реки он берег от пожаров.

Этот берег реки он берег от пожаров.

Проведём синтаксический и, при необходимости, семантический анализ этих фраз. Так, при рассмотрении предложения *«Но в другом случае он был ее другом»* были получены синтаксическое и семантическое деревья, представленные на рисунке 5.

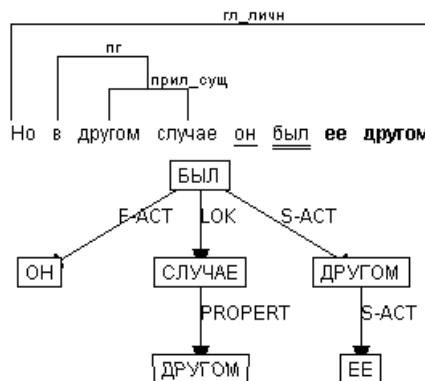


Рисунок 5 – Деревья синтаксического и семантического разбора
«Но в другом случае он был ее другом»

После семантического анализа мы получаем, что в первом случае используется понятие «другой», а во втором «друг», что соответствует предложению *«Но в друго́м случае он был ее дру́гом»*.

При рассмотрении примера предложения «*На самом деле причина была в самом человеке*» мы получаем синтаксическое и семантическое деревья разбора, представленные на рисунке 6.

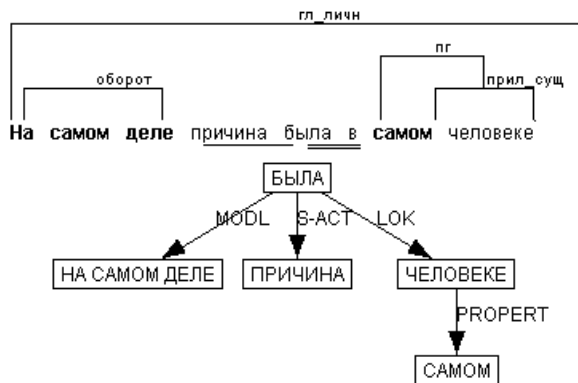


Рисунок 6 – Деревья синтаксического и семантического разбора «На самом деле причина была в самом человеке»

Из примера видно, что система выделила устойчивое вводное словосочетание, а для второго случая использовала понятие «самый» что соответствует предложению «На *самом* деле причина была в *самом* человеке».

Однако не всегда можно корректно снять омографию, используя синтаксический и семантический анализ. Причина этого в неполноте используемых морфологических словарей, синтаксических и семантических грамматик. В таком случае используются классические подходы снятия омографии, основанные на статистике [Цирульник, 2009].

Рассмотрим пример снятия омографии слова «минут», которое может быть как глаголом «минуть», так и существительным «минута».

При синтаксическом и семантическом анализе предложения «Скоро минут несколько минут» были получены деревья разбора, представленные на рисунке 7.



Рисунок 7 – Деревья синтаксического и семантического разбора «Скоро минут несколько минут»

Система верно сняла омографию слова «минут», где для первого случая был выбран глагол «минуть», а для второго – существительное «минута».

При анализе предложения «Они подождали пока минут несколько минут» не удалось построить корректного дерева разбора, результаты представлены на рисунке 8.

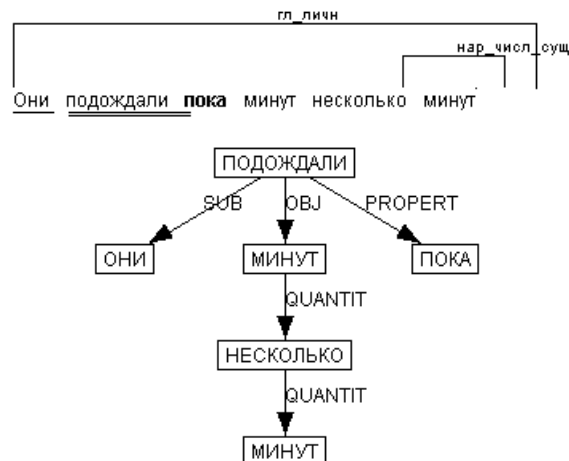


Рисунок 8 – Деревья синтаксического и семантического разбора «Они подождали пока минут несколько минут»

Из рисунка 8 видно, что система посчитала, что в обоих случаях используется существительное «минута».

Рассмотрим пример снятия омонимии в случае ё-омографов. Для предложения «Этот берег реки он берег от пожаров» получены деревья разбора, представленные на рисунке 9.

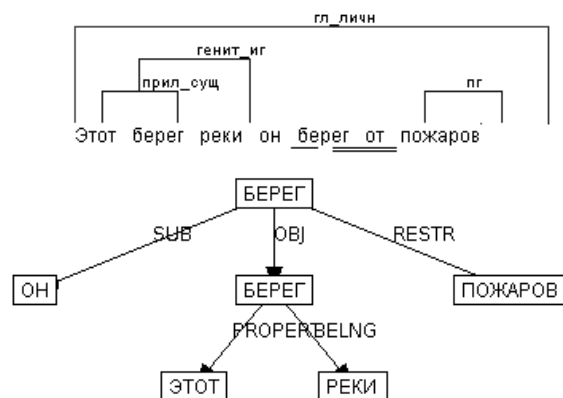


Рисунок 9 – Деревья синтаксического и семантического разбора «Этот берег реки он берег от пожаров»

После семантического анализа мы получаем, что в первом случае используется понятие «берег», а во втором «беречь», что соответствует предложению «Этот *бѣрег* реки он *берѣг* от пожаров».

Как видно из приведенных примеров использование синтаксического и семантического анализа является достаточно эффективной процедурой разрешения графической омонимии. Использование такого подхода в дополнении к классическим методам, основанным на статистике, позволяет существенно повысить качество синтеза речи по тексту.

Заключение

В статье были рассмотрены основные сценарии решения задачи снятия омонимии при распознавании и синтезе речи, описаны существующие средства и ресурсы позволяющие провести синтаксический и семантический анализы

текста, показана практическая применимость описываемого метода.

Использование синтаксического анализа гипотез при распознавании и синтезе речи позволяет отсеять большинство некорректных гипотез. Однако, так как большое количество фраз с омонимиями обладают сходными синтаксическими свойствами, обусловленными, в большинстве своем, их языковым родством, то после обработки синтаксической структуры неоднозначность не снимается полностью. Показано, что в таких случаях для снятия омонимии произнесённых или синтезируемых фраз применение дальнейшего анализа их семантической структуры оказывается достаточно эффективным приёмом.

Библиографический список

- [Валгина, 1998] Теория текста: Учебное пособие. / Валгина Н. С. // М.: Изд-во МГУП «Мир книги», 1998
- [Житко, 2012] Применение облачных интернет-технологий при распознавании речи / Житко В.А., Лобанов Б.М. // Информатика. - № 4. - Минск. 2012
- [Иомдин, 2011] Говорящий «ЭТАП». Опыт использования синтаксического анализатора системы ЭТАП в русском речевом синтезе / Л.Л. Иомдин, Б.М. Лобанов, Ю.С. Гецевич // Труды Международной конференции «Компьютерная лингвистика и интеллектуальные технологии» (Диалог'2011), Бекасово 25 – 29 мая 2011, Вып. 10 (17). – М.: РГТУ, 2011. – С. 269-279.
- [Леонтьева, 1997] Русский общесемантический словарь (РОСС): структура, наполнение. / Леонтьева Н.Н. // НТИ. Сер. 2. - 1997. - N 12. - С.5-20.
- [Лобанов, 2009] Проблема разрешения «ё»-омографов при синтезе речи по тексту / Б.М. Лобанов // Труды Международной конференции «Компьютерная лингвистика и интеллектуальные технологии» (Диалог'2009), 1 – 4 июня 2009. – М.: Наука, 2009. – С. 330-338.
- [Панкратов, 2000] Описание фрагментации и синтаксического анализа в системе Диалинг. / Панкратов Д. В., Гершензон Л. М., Ножов И. М. // Техническая документация [Electronic resource]. – 2000. – Mode of access: <http://www.aot.ru/docs/synan.html>. – Date of access : 21.11.2013.
- [Розенталь, 1976] Словарь-справочник лингвистических терминов / Д.Э. Розенталь, М.А. Теленкова. // Изд. «Просвещение», М. 1976, 543 с.
- [Цирульник, 2009] Статистический анализ и контекстуальные правила разрешения графической омонимии при синтезе речи по тексту / Л.И. Цирульник, С.Г. Барбук, Б.М. Лобанов // Труды Международной конференции «Компьютерная лингвистика и интеллектуальные технологии» (Диалог'2009) / Москва, 27 - 31 мая 2009г. – С. 530-536.
- [AOT, 2013] AOT, 2013. – [Electronic resource] – Mode of access : <http://www.aot.ru>. – Date of access : 21.11.2013
- [NLP Stanford, 2013] NLP Stanford, 2013. – [Electronic resource] – Mode of access : <http://nlp.stanford.edu>. – Date of access : 21.11.2013..
- [Sokirko, 2001] Short description of Dialing Project / Sokirko A.A. // Technical documentation [Electronic resource]. – 2013. – Mode of access: <http://www.aot.ru/docs/sokirko/sokirko-candid-eng.html>. – Date of access : 21.11.2013.

SOLVING DISAMBIGUATION IN SPEECH RECOGNITION

B.M. Lobanov*, V.A. Zhitko**

** United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Minsk, Republic of Belarus*

lobanov@newman.bas-net.by

*** Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus*

zhitko.vladimir@gmail.com

In work the method for disambiguation in speech recognition and synthesis using the results of the syntactic and semantic analysis of the text is described. As a tool of analysis of texts taken Russian language software system of automatic text processing AOT is used.

Introduction

Synthesis, recognition and understanding of speech have a number of problems related to the ambiguity of natural language. One of the sources of ambiguity is homonyms, a phenomenon in which some of the words of a natural language can have the same spelling and pronunciation. Other source of ambiguity are paronyms, a words which are a derivative of another and has a related meaning: 'wisdom' is a paronym of 'wise'.

Such ambiguity can be "confusing" soft system, which reaction may be a surprise to the user. For example user asks system for last data but get current date. For solve this we can use semantics approach to choose right variant of user request.

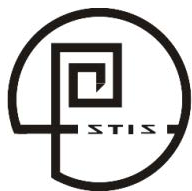
Semantic analysis can be used for improvement a real speech recognition system such as Google Voice recognition system. The basic of this system is statistical analysis so results can be meaningless, but using semantic analysis can range result by meaningful.

Main Part

Solving some of speech disambiguation we need only syntax phase, but to most of them we need to make semantic analysis of all variants and with information of context we can make decision which of them is right. To make syntax and semantic analysis we use Russian language software system of automatic text processing AOT.

Conclusion

Using this method we can approve speech recognition using context information and semantic analysis.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8+620

СИТУАЦИОННОЕ УПРАВЛЕНИЕ И СЕМАНТИЧЕСКОЕ МОДЕЛИРОВАНИЕ В ЭНЕРГЕТИКЕ

Массель Л.В., Массель А.Г.

*Институт систем энергетики им. Л.А. Мелентьева СО РАН,
г. Иркутск, Россия*

massel@isem.sei.irk.ru

amassel@gmail.com

В статье описаны этапы развития научного направления «ситуационное управление» и роль в его развитии Д.А. Поспелова и В.В. Мартынова. Уделяется внимание современному состоянию и тенденциям развития ситуационного управления, рассматривается предложенный авторами подход к ситуационному управлению в энергетике, основанный на ситуационном анализе и ситуационном моделировании, а также роль семантического моделирования в этом подходе. Прослеживается связь между ситуационной осведомленностью и получившей распространение в последнее время петлей Бойда (цикл OODA), делается вывод о перспективности их применения в ситуационном управлении совместно с технологиями семантического моделирования. Предлагается использовать этот подход при создании интеллектуальных энергетических систем.

Ключевые слова: ситуационное управление, семантические технологии; онтологическое; когнитивное и событийное моделирование, ситуационная осведомленность.

Введение

Научное направление «ситуационное управление» сформировалось и получило широкое распространение в 70-80-е гг. 20 века. В 90-х годах произошел спад интереса к этому направлению, объяснявшийся рядом объективных причин, но, начиная с середины 2000-х годов 21 века, наблюдается повышение интереса к ситуационному управлению.

В статье кратко рассматривается история развития «ситуационного управления», роль в его развитии белорусского ученого В.В. Мартынова, а также современное состояние и тенденции развития этого направления в наше время. Высказанная в свое время Д.А. Поспеловым идея о необходимости применения в этой области идей искусственного интеллекта получила в наше время новую трактовку (intelligent computing) и новое наполнение, связанные, в том числе, с применением семантического моделирования.

На примере исследований проблем энергетической безопасности рассматриваются вопросы применения ситуационного анализа и ситуационного моделирования, как предпосылки ситуационного управления в энергетике, а также роль и место семантического моделирования. Отмечается актуальность нового направления,

связанного с 3D-визуализацией, получившего название «ситуационная осведомленность». Предлагается совместное применение для целей ситуационного управления цикла OODA и инструментальных средств семантического моделирования и ситуационной осведомленности.

1. История возникновения термина «ситуационное управление»

Термин «ситуационное управление» сформировался в 60-е годы прошлого века. Основоположником этого направления по праву считается Д.А. Поспелов. Первоначально использовался термин «модельное управление», эта концепция активно развивалась В.Н. Пушкиным и Д.А. Поспеловым, итоги ее развития были подведены в книге [Поспелов, 1972]. Затем возник термин «ситуационная модель». С появлением статьи Д.А. Поспелова [Поспелов, 1971] термин «ситуационное управление» вытесняет все остальные. Вклад в формирование нового направления внесли ученики Д.А. Поспелова: Железов Ж.И., развивавший теорию дискретных ситуационных сетей и Клыков Ю.И., разработавший язык синтагматических цепей – специальный язык моделирования для описания ситуаций и принятия решений в ситуационном управлении, опирающийся на предложенный Д.А. Поспеловым и сейчас

незаслуженно забытый язык RX-кодов, который, по сути, является прообразом онтологий. Существенный вклад в развитие ситуационного направления внесли монографии Ю.И. Клыкова [Клыков, 1974] и Д.А. Поспелова [Поспелов, 1975]. Главным достижением этого направления в то время стала идея Д.А. Поспелова о необходимости применения в этой области методов искусственного интеллекта [Поспелов, 1981]. Наиболее полно итоги этого направления и исторический очерк его развития в 1960-х -1980-х гг. приведены в книге [Поспелов, 1986].

2. Вклад В.В. Мартынова в развитие направления «Ситуационное управление».

В 60-е гг. прошлого столетия научное направление «Искусственный интеллект» в нашей стране еще только формировалось, но уже возникло четкое понимание того, что без концептуальной структуризации знаний, без их формального представления и преобразования создать искусственный интеллект невозможно. Добиться этого интуитивными эвристическими приемами не удалось. Опираясь на взаимосвязи лингвистики с семиологией, праксиологией и теорией информации, белорусский ученый – лингвист В.В. Мартынов предложил способ «исчисления языковых смыслов». В монографии «Кибернетика. Семиотика. Лингвистика» [Мартынов, 1966] им излагаются основы и перспективы дедуктивной семиологии, а его следующая большая работа «Семиологические основы информатики» [Мартынов, 1974] уже содержит прототип универсального семантического кода (УСК) как средства снятия неопределенности и многозначности естественного языка. В.В. Мартынову удалось на основе своего кода построить исчисление смыслов и закодировать около 700 наиболее часто употребляемых глаголов. Первую версию УСК, описанную в одноименной монографии «Универсальный семантический код. Грамматика. Словарь. Тексты» [Мартынов, 1977], Виктор Владимирович совершенствует в ряде последующих работ в 1982 -1995 гг. (УСК-2, УСК-3, УСК-4, УСК-5), наиболее полное, завершающее описание его работ представлено в монографиях «Основы семантического кодирования. Опыт представления и преобразования знаний» [Мартынов, 2001] и «В центре сознания человека» [Мартынов, 2009]. Международный биографический центр в Кембридже справедливо признал, что вклад В.В. Мартынова в науку может быть оценен как «Достижение XX века».

3. Модели и языки ситуационного управления

Ситуационное управление, согласно [Поспелов, 1986], основано на понятиях ситуации, классификации ситуаций и их преобразования. Справедливо отмечалось, что ситуационное

управление требует больших затрат на предварительное создание базы знаний об объекте управления, его функционировании и способах управления им, причем эти затраты оправданы, если невозможно формализовать описания объекта и способов управления им. Потребовались новые модели представления объектов управления и разработка специальных языков для описания ситуаций, складывающихся на объекте управления и в системе управления им.

Для описания объектов управления предлагалось использовать *семиотические, или знаковые модели*, которые включали как необходимые формализмы, так и правила их изменения. От классической формальной модели $M = \langle T, P, A, \Pi \rangle$, где T – множество базовых элементов, P – синтаксические правила, A – система аксиом, Π – семантические правила, предлагалось перейти к семиотической модели, которая включала как формальную модель M , так и правила изменения множеств T, P, A, Π [Поспелов, 1986]. Предлагалось использовать также формальную интерпретируемую модель

$$L = \langle Z, D, H, V \rangle,$$

где Z – множество интерпретируемых значений, D – правила отображений $T \Leftrightarrow Z$, которое является многозначным в обе стороны, H – правила отображения, определяющие реализацию отображения, V – правила интерпретации.

Кроме того, для описания модели объекта управления использовались также дискретные ситуационные сети (ДСС). ДСС определялась как $\langle I, C, P, O, X \rangle$, где I – множество истоков, C – множество стоков, P – множество решателей, O – множество объектов, X – множество характеристик этих объектов. Две вершины ДСС соединялись связью, если объекты могли попасть из одной вершины в другую, минуя остальные вершины.

В качестве языков ситуационного управления рассматривались: язык исчисления предикатов первого порядка, язык RX-кодов, универсальный семантический код; предлагалось создать универсальный ЯСУ на основе ДСС. Рассмотрим подробнее язык RX-кодов и УСК.

В языке RX-кодов рассматриваются понятия и отношения. Каждое понятие, кроме базовых, задается соотношением:

$$X_N = X_1 R_1, X_2 R_2, \dots X_{N-1} R_{N-1},$$

где X_i – понятия, R_i – отношения. Таким образом, понятие X_N определяется через понятия $X_1, X_2, \dots X_{N-1}$ с помощью отношений $R_1, R_2, \dots R_{N-1}$.

Для описания ситуаций предлагалось использовать также универсальный семантический код (УСК) В.В. Мартынова. Простой ядерной конструкцией в УСК является тройка вида (SAO) , где S соответствует субъекту, совершающему акцию A , а O – объект, на который направлена данная акция. Вводится замкнутая система операций, позволяющая из простых ядерных конструкций строить более сложные цепочки.

Описанные выше подходы получили достаточно

широкое распространение к концу 1980-х гг. прошлого века, учитывая, что их развитие совпало с «пиком завышенных ожиданий» в области искусственного интеллекта.

4. Современное состояние и тенденции развития ситуационного управления

Спад интереса к ситуационному управлению в России, наступивший в 90-х гг., помимо объективного изменения внешних экономико-политических условий, можно объяснить как наступившей «зимой искусственного интеллекта», так и трудностями, с которыми столкнулись разработчики, пытаясь построить модели сложных объектов управления с помощью предлагаемых подходов.

Тем не менее, в настоящее время можно констатировать новый виток интереса к этому направлению, который подкрепляется как наличием более совершенной техники, так и появлением новых методов и подходов, в том числе семантического моделирования. За это время претерпела изменения сама парадигма искусственного интеллекта: если на ранних этапах его развития предполагалось, что системы ИИ могут заменять, в ряде случаев, естественный интеллект, сейчас получают распространение *интеллектуальные вычисления (Intelligent Computing)*, под которыми понимаются методы и системы искусственного интеллекта, направленные на усиление и поддержку естественного интеллекта (поддержку принятия решений экспертами).

Получают распространение более прагматические трактовки ситуационного управления. В словаре терминов МЧС (2010) ситуационное управление определяется как деятельность органов управления, при которой решения и управляющие воздействия субъекта управления основываются на анализе вариантов принятия решения с учетом: текущего состояния объекта управления, располагаемых вариантов действий и прогноза последствий принимаемых управленческих воздействий. В [Кураков, 2004] ситуационное управление определяется как оперативное управление, осуществляемое в дополнение к стратегическому, перспективному и заключается в принятии управленческих решений по мере возникновения проблем в соответствии со складывающейся экономической ситуацией.

В [Васильев, 2012] используется идея ситуационного управления, суть которой заключается в выборе управленческих решений с учетом сложившейся ситуации из некоторого набора допустимых (типовых, стандартных) управляющих воздействий. Под текущей ситуацией C при этом понимается совокупность текущего состояния объекта (вектор состояния X) и его внешней среды (вектор возмущений F). Тогда $C = \langle X, F \rangle$.

Вводится также понятие полной ситуации:

$S = \langle C, G \rangle$, где C – текущая ситуация, G – цель управления. В свою очередь, цель управления G

может быть представлена в виде целевой ситуации G_g , к которой должна быть приведена имеющаяся текущая ситуация. Тогда $S = \langle C, G_g \rangle$.

Полагая, что текущая ситуация C принадлежит некоторому классу Q' , а целевая (заданная) ситуация G_g – классу Q'' , ищется такое управление (вектор управляющих воздействий U), которое принадлежит множеству допустимых управлений Ω_u и обеспечивает требуемое преобразование одного класса ситуаций в другой:

$$C \in Q' \xrightarrow{U \in \Omega_u} G_g \in Q''$$

Таким образом, ситуационное управление выступает как отображение:

$$(Q', Q'') \rightarrow U \in \Omega_u,$$

сопоставляющее паре «текущая ситуация - целевая ситуация» требуемый результат – управление U . Другими словами, при ситуационном управлении проблема выбора управляющих воздействий сводится к адекватной оценке состояния объекта и среды (что усложняется при наличии факторов неопределенности), отнесению соответствующей текущей ситуации к одному из типовых классов и выбору такого управления (из определенного набора альтернатив), которое приводит к достижению поставленной цели управления (целевой ситуации) [Васильев, 2012].

5. Ситуационный анализ и семантическое моделирование в энергетике.

Изложенный выше подход можно проиллюстрировать на примере энергетики. На рис. 1 представлена общая схема исследований проблем энергетической безопасности (ЭБ) с точки зрения ситуационного управления, или, иначе, оценки состояния ТЭК в условиях возможных сценариев угроз ЭБ с учетом мероприятий, направленных на повышение уровня ЭБ. Сопоставим эту схему с подходом, описанным выше.

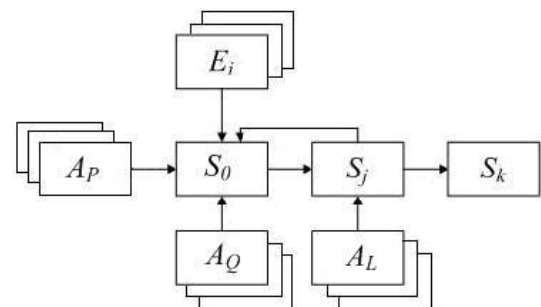


Рисунок 1 - Общая схема исследований по оценке состояния ТЭК с точки зрения ситуационного управления

Здесь S_0 – начальное состояние ТЭК, может рассматриваться как текущая ситуация C ; E_i – сценарии возможных чрезвычайных ситуаций, возникающих в случае реализации угроз ЭБ (аналогичны влиянию внешней среды F); $A = A_p \cup A_q \cup A_L$ – набор превентивных,

оперативных и ликвидационных мероприятий, предотвращающих, нейтрализующих или смягчающих последствия чрезвычайной ситуации (может рассматриваться как набор соответствующих управлений U); S_j – состояние ТЭК после чрезвычайной ситуации E_i (реализации угроз) с учетом выполнения набора мероприятий A_p и/или A_Q ; S_k – состояние ТЭК после проведения ликвидационных мер A_L (S_j и S_k могут рассматриваться как аналоги соответствующих целевых ситуаций G_g).

Авторами предложено использовать в этих исследованиях ситуационный анализ и ситуационное моделирование, как предпосылки ситуационного управления. Задачей ситуационного анализа является выявление параметров и существенных факторов, или «обстоятельств», определяющих ситуацию, взаимосвязей между факторами и степени их взаимовлияния. Под ситуацией понимается совокупность обстоятельств, определяющих внутреннее состояние объекта или системы, и обстоятельств, определяющих состояние окружающей среды по отношению к данному объекту или системе. Первые описываются параметрами, характеризующими состояние системы (X), вторые – условиями окружающей среды или существенными факторами, влияющими на развитие системы (F). Ситуационное моделирование заключается в моделировании ситуаций и перехода из одной ситуации в другую. Ситуационный анализ включает: анализ проблемных ситуаций (например, ЧС в энергетике); выявление путей разрешения проблемных ситуаций (альтернатив) или управляющих воздействий (U) (в нашем случае – выбор из описанного выше множества $A = A_p \cup A_Q \cup A_L$); определение критериев оценки альтернатив (например, экономических); анализ альтернатив; выбор и реализацию наилучшей альтернативы.

Учитывая, что наличие факторов неопределенности усложняет адекватную оценку состояния объекта и среды, авторами предложено использовать семантические технологии ситуационного анализа, к которым отнесены онтологическое, когнитивное и событийное моделирование [Массель Л., 2010], [Массель А., 2010] (рис. 2).

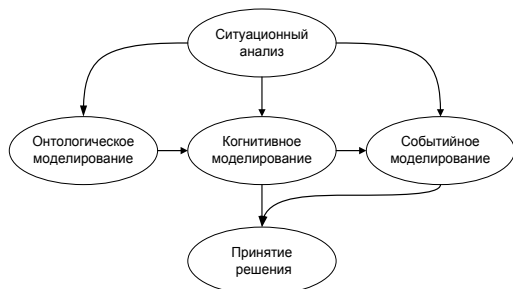


Рисунок 2 - Взаимосвязь семантических технологий ситуационного анализа.

Под *когнитивным моделированием* понимается построение когнитивных моделей, или, иначе, когнитивных карт (ориентированных графов), в которых вершины соответствуют факторам (концептам), а дуги – связям между факторами (положительным или отрицательным), в зависимости от характера причинно-следственного отношения [Трахтенгерц, 1988]. Математическим аппаратом для построения когнитивных моделей является теория графов. Следует отметить, что подход, основанный на использовании когнитивных карт для описания слабоструктурированных ситуаций, на протяжении ряда лет развивается в Институте проблем управления РАН, проводятся регулярные конференции, посвященные этим проблемам, на одной из которых были представлены также результаты авторов [Массель А., 2011].

Сущность *событийного метода моделирования* заключается в отслеживании на модели последовательности событий в том же порядке, в каком они происходили бы в реальной системе. Задаваемые моделью последовательности реализаций событий – цепочки событий – описывают сценарии реакции системы на возникновение инициирующего события, стоящего в начале цепочки.

В качестве инструмента событийного моделирования используется аппарат Joiner-сетей (JN) – одной из разновидностей алгебраических сетей, предложенной в [Столяров, 2004], [Столяров, 2010]. Joiner-сети можно рассматривать как расширение сетей Петри, ориентированное на построение поведенческих моделей. В основе теории JN лежит описание логики взаимодействия асинхронных процессов в виде набора пусковых и флаговых функций, состоящих из булевых функций. Особенностью JN является то, что они предусматривают как графическое представление, так и описание в виде логических формул, обработку которых можно автоматизировать.

Подробнее технологии семантического моделирования и инструментальные средства их поддержки (интеллектуальная ИТ-среда) рассматривались авторами в [Массель Л., А., 2012] а также на конференции OSTIS -2013 [Массель Л., А., 2013]. Рассмотрим еще один актуальный аспект современного подхода к ситуационному управлению – ситуационную осведомленность.

6. Ситуационная осведомленность и ситуационное управление

Понятие «*ситуационная осведомленность*», или Situational Awareness, сформировалось на рубеже 1990-х годов и связано в первую очередь с пионерными работами Мика Эндсли (Mica R. Endsley). Согласно классическому определению, принцип Situational Awareness представляет собой «чувственное восприятие элементов обстановки в (едином) пространственно-временном континууме, осознанное восприятие их значения, а также проецирование их в ближайшее будущее».

Ситуационная осведомленность базируется на новом подходе к визуализации геопространственных данных, первоначально названном в нашей стране «неогеографией», авторы используют термин 3D - визуализация.

Реализованный геокомпонент 3D - визуализации [Иванов, 2013] включен в состав интеллектуальной ИТ-среды и позволяет использовать ситуационную осведомленность при решении прикладных задач [Массель Л., А., 2012], [Массель Л. и др., 2013].

В этой связи следует упомянуть предложенную полковником Джоном Бойдом «петлю Бойда» или OODA [Черняк, 2013]. Цикл Бойда, описанный в 1995 году, состоит из четырех составляющих: Observe («наблюдай»), Orient («ориентируйся»), Decide («решай») и Act («действуй»). На рис. 3 приведена его простейшая версия.

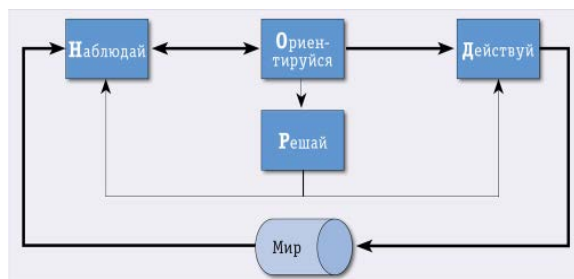


Рис. 3. Простейшая версия петли Бойда (OODA)

Составляющие цикла OODA: 1) *наблюдение (observation)* – преобразование изменений во внешней среде в форму сигналов-данных, которые могут быть использованы в последующем; 2) *ориентация (orientation)* – когнитивный процесс оценки данных в соответствии с контекстом, наделение данных смыслом (sensemaking), преобразование данных в информацию; 3) *решение (decision)* – выбор альтернатив из ряда возможных; 4) *действие (action)* – воздействие на окружающую среду. Очевидно, что ситуационная осведомленность и семантическое моделирование могут использоваться на первом и втором этапах цикла OODA («наблюдай» и «ориентируйся»).

Несмотря на то, что обычно ссылки на OODA обнаруживаются в текстах, посвященных попыткам практического использования технологий Big Data (Больших Данных) и связываются с переходом к экономике знаний в эпоху Big Data или когнитивных компьютерных систем, авторы считают, что здесь прослеживается прямая связь с ситуационным управлением, в котором на современном этапе играют большую роль как семантическое моделирование, так и ситуационная осведомленность.

7. Интеллектуальные системы ситуационного управления

В настоящее время все большее внимание привлекают задачи управления сложными динамическими объектами. К ним относятся современные летательные аппараты, силовые и энергетические установки, мобильные роботы и др.

Как справедливо отмечается в [Васильев, 2009], для них характерны отсутствие точных математических моделей либо их чрезмерная сложность, высокая размерность пространства состояний и принимаемых решений по управлению, иерархичность, многообразие критериев качества, высокий уровень шумов и внешних возмущений.

Очевидно, что для систем с неполной информацией и высокой сложностью объекта управления становится все более актуальным применение методов ситуационного управления, основанных на интеллектуальных технологиях.

В энергетике авторы развивают этот подход в рамках направления, связанного с созданием интеллектуальных энергетических систем (ИЭС), получившего за рубежом название Smart Grid [Массель Л., А., 2012], [Массель Л. и др., 2013].

Наряду с применением в ИЭС уже ставших традиционными методов искусственного интеллекта (нейронные сети, генетические алгоритмы, нечеткая логика), все большее внимание энергетиков привлекают так называемые «системы с целеполаганием», то есть интеллектуальные системы управления, которые имеют несколько целей функционирования (или умеют генерировать эти цели), выбирая самую подходящую цель в зависимости от окружающей среды, умеют прогнозировать поведение окружающей среды и свое собственное состояние. Пока такие системы в энергетике отсутствуют, но представляется, что совместные усилия ИТ-специалистов и энергетиков позволят приблизить решение этой проблемы.

Заключение

Основателем научного направления «ситуационное управление» заслуженно считается Д.А. Пospelov, в связи с чем в статье приведена библиография его основных работ по этой тематике. Отмечен вклад в это направление белорусского ученого-лингвиста В.В. Мартынова – один из языков ситуационного управления (универсальный семантический код – УСК). Анализируются современное состояние и тенденции развития ситуационного управления, интерес к которому повышается в последнее время. Авторами предлагается подход к ситуационному управлению в энергетике, основанный на ситуационном анализе и ситуационном моделировании, акцентируется внимание на роли семантического моделирования в этом подходе. Рассматривается ситуационная осведомленность, как актуальный аспект современного подхода к ситуационному управлению. Прослеживается связь между ситуационной осведомленностью и получившей распространение в последнее время петлей Бойда (цикл OODA), делается вывод о перспективности их применения в ситуационном управлении совместно с технологиями семантического моделирования.

Результаты, представленные в статье, получены при частичной финансовой поддержке грантов РФФИ № 13-07-00140, № 12-07-00359, гранта

Программы Президиума РАН №229 и гранта на выполнение интеграционного проекта СО РАН и НАН Беларуси №18 (2012-2014).

Библиографический список

- [Поспелов, 1972] Поспелов Д.А., Пушкин В.Н. Мышление и автоматы. – М.: Советское радио, 1972. – 22 с.
- [Поспелов, 1971] Поспелов Д.А. Принципы ситуационного управления. – Известия РАН СССР, Техническая кибернетика. – 1971. – №2. – С. 10-17.
- [Клыков, 1974] Клыков Ю.И. Ситуационное управление большими системами. – М.: Энергия. – 1974. – 134 с.
- [Поспелов, 1975] Поспелов Д.А. Большие системы. Ситуационное управление. – М.: Знание, 1975. – 62 с.
- [Поспелов, 1981] Поспелов Д.А. Логико-лингвистические модели в системах управления. – М.: Энергия, 1981. – 231 с.
- [Поспелов, 1986] Поспелов Д.А. Ситуационное управление. Теория и практика. – М.: Наука, 1986. – 284 с.
- [Мартынов, 1966] Мартынов В.В. Кибернетика. Семиотика. Лингвистика. – Минск: Наука и техника, 1966. – 148 с.
- [Мартынов, 1974] Мартынов В.В. Семиологические основы информатики. – Минск: Наука и техника, 1974. – 192 с.
- [Мартынов, 1977] Мартынов В.В. Универсальный семантический код. Грамматика. Словарь. Тексты. – Минск: Наука и техника, 1977. – 191 с.
- [Мартынов, 2001] Мартынов В.В. Основы семантического кодирования. Опыт представления и преобразования знаний. – Минск: Европейский государственный университет, 2001. – 140 с.
- [Мартынов, 2009] Мартынов В.В. В центре сознания человека. – Минск, БГУ, 2009. – 271 с.
- [Кураков, 2004] Экономика и управление, финансы и право: словарь-справочник / авт.-сост.: Л. П. Кураков, В. Л. Кураков, А. Л. Кураков. – М.: Вуз и школа, 2004. – 1288 с.
- [Васильев, 2012] Васильев В.И. Интеллектуальные системы защиты информации – М.: Машиностроение, 2012. – 171 с.
- [Массель Л., 2010] Массель Л.В. Применение онтологического, когнитивного и событийного моделирования для анализа развития и последствий чрезвычайных ситуаций в энергетике / Проблемы безопасности и чрезвычайных ситуаций. – №2. – 2010. – С. 34-43.
- [Массель А., 2010] Массель А.Г. Методологический подход к организации интеллектуальной поддержки исследований проблемы энергетической безопасности / «Информационные технологии». – №9. – 2010. – С. 32-36.
- [Трахтенгерц, 1988] Трахтенгерц Э.А. Компьютерная поддержка принятия решений. – М.: СИНТЕГ, 1998. – 376 с.
- [Массель А., 2011] Массель А.Г. Когнитивный подход в исследованиях проблем энергетической безопасности России / Когнитивный анализ и управление развитием ситуации (CASC'2011): Труды IX Международной конференции (14-16 ноября 2011 г., Москва). – М.: ИПУ РАН, 2011. – С. 224-228.
- [Столяров, 2004] Столяров Л.Н., Новик К.В. Реализация параллельных процессов с помощью сетей Joiner-net // Информационные и математические технологии / Труды Байкальской Всероссийской конференции. – Иркутск: ИСЭМ СО РАН, 2004. – С. 11-14.
- [Столяров, 2010] Столяров Л.Н. Философия событийного моделирования на примере сценария энергетической катастрофы // Труды Международной конференции «Информационные технологии в науке, образовании, телекоммуникации и бизнесе». – Украина, Гурзуф, 2010. – С. 197-200.
- [Массель Л., А., 2012] Массель Л.В., Массель А.Г. Интеллектуальные вычисления в исследованиях направлений развития энергетики // Известия Томского политехнического университета. – 2012. – Т. 321. – № 5. Управление, вычислительная техника и информатика. – С. 135-141.
- [Массель Л., А., 2013] Массель Л.В., Массель А.Г. Семантические технологии на основе интеграции онтологического, когнитивного и событийного моделирования / Материалы III международной научно-технической конференции «OSTIS-2013» – Беларусь, Минск: БГУИР, 2013. – С. 247-250.
- [Иванов, 2013] Иванов Р.А. Методика 3D-визуализации для поддержки принятия решений в энергетических исследованиях // Современные технологии. Системный анализ. Моделирование. – 2013. №1(37). – С. 116-121.
- [Массель Л. и др., 2013] Массель Л.В., Иванов Р.А., Массель А.Г. Моделирование этапов принятия решений на

основе сетецентрического подхода / Вестник ИргТУ. – №10 (81). – 2013. – С. 16-22.

[Черняк, 2013] Черняк Л. Петля Бойда и кибернетика второго порядка. – Открытые системы. – №7. – 2013. – С. 23-26.

[Васильев, 2009] Васильев В.И., Ильясов Б.Г. Интеллектуальные системы управления: теория и практика. – М.: Радиотехника, 2009. – 392 с.

[Массель Л., 2012] Массель Л.В. Интеллектуализация поддержки принятия решений при моделировании и управлении режимами в Smart Grid // Интеллектуализация обработки информации: Труды 9-й Международной конференции. – Черногория, Будва, 2012. – С. 692-695.

CONTINGENCY MANAGEMENT AND SEMANTIC MODELING IN ENERGY SECTOR

Massel L.V., Massel A.G.

*Melentiev Energy Systems Institute of Siberian Branch
of the Russian Academy of Sciences
Irkutsk, Russia*

massel@isem.sei.irk.ru

amassel@gmail.com

This article describes the stages of development of scientific direction "contingency management" and the role in its formation of D.A. Pospelov and V.V. Martynov. Attention is paid to the current status and trends of contingency management. The authors propose approach to contingency management in the energy sector, based on the situational analysis and situational modeling. It's offered to use ontological, cognitive and event modeling as semantic modeling techniques.

Introduction

The article briefly reviews the history of the "contingency management" development and role in this process of the Belarusian scientist V.V. Martynov. The current status and development trend in this direction in our time are considered.

Main Part

The authors propose approach to contingency management in the energy sector, based on the situational analysis and situational modeling and consider using of semantic modeling in this approach. Correlation between situational awareness and widely spread recently Boyd loop (cycle OODA) is traced, the prospects of their use for contingency management in conjunction with semantic modeling techniques is discussed. Intelligent systems of contingency management are considered. It is proposed to use the techniques of situational management and semantic modeling to create intelligent energy systems (Smart Grid).

Conclusion

The results presented in this paper were obtained with the partial financial support by RFBR grants № 13-07-00140, № 12-07-00359, by grant of RAS Presidium Program № 229 and by grant for the implementation of the integration project of SB RAS and the National Academy of Sciences of Belarus № 18 (2012-2014).



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

О ФОРМАЛИЗАЦИИ СЕМАНТИКИ ОБЛАСТЕЙ ЗНАНИЙ В ИНФОРМАЦИОННЫХ И ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ НА ОСНОВЕ ОНТОЛОГИЙ

Загорулько Ю.А. *, Загорулько Г.Б. *

** Институт систем информатики им. А.П. Ершова Сибирского отделения
Российской академии наук, г. Новосибирск, Россия*

zagor@iis.nsk.su

gal@iis.nsk.su

В докладе рассматриваются подходы и средства формализации семантики областей знаний на основе онтологий, обсуждается роль онтологий в интеллектуальных и информационных системах, обсуждаются примеры практического использования онтологий в системах такого класса. В заключении делается вывод, что онтологий могут успешно использоваться как при проектировании и разработке информационной и интеллектуальной системы, так и в качестве полноправного компонента во время ее функционирования.

Ключевые слова: формализация семантики; онтология; интеллектуальная система; база знаний.

Введение

Для того чтобы компьютер мог стать интеллектуальным помощником человека в его повседневной деятельности, необходимо формализовать и семантику тех аспектов этой деятельности, которые мы хотим автоматизировать, и семантику предметной области (ПО), в рамках которой эта деятельность осуществляется. Например, чтобы построить систему информационной поддержки научной и производственной деятельности в определенной области знаний (ОЗ), необходима формализация не только функциональности такой системы, но и семантики назначенной ей области знаний.

Наиболее адекватным и эффективным средством формализации областей знаний являются онтологий [Guarino, 1998]. В настоящее время формализм онтологий является своего рода универсальным семантическим кодом (УСК) [Мартынов, 1974], так как он повсеместно используется для построения формальных моделей областей знаний с целью фиксации общего разделяемого всеми экспертами (или, по крайней мере, их большинством) знания об этих областях [Gruber, 1995].

Формализация семантики области знаний в виде онтологий служит не только целям компактного и непротиворечивого представления моделируемой области знаний, она также формирует понятийный базис для представления тех знаний о данной ОЗ,

которые невозможно, неудобно или неэффективно представлять средствами онтологий. В частности, в ее терминах описывается семантика данных и информационных ресурсов, используемых в информационной системе, а также экспертные правила и другие компоненты базы знаний интеллектуальной системы. Онтология также может служить основой для других моделей представления знаний, как например, для интегрированной модели представления знаний [Загорулько, 2013], объединяющей различные взаимодополняющие друг друга методы и средства представления и обработки знаний и служащей базисом для оригинальной технологии разработки интеллектуальных систем [Загорулько, 2013a].

Таким образом, онтология занимает центральное место в системе знаний любой информационной системы, в том числе интеллектуальной. Можно также сказать, что система знаний любой интеллектуальной системы символического (семиотического) типа базируется на онтологий, только она не всегда явно задана.

В докладе обсуждается роль онтологий в интеллектуальных и информационных системах, рассматриваются средства формализации семантики области знаний на основе онтологий, приводятся и обсуждаются примеры практического использования онтологий в интеллектуальных и информационных системах, в разработке которых принимали участие авторы доклада.

1. Роль онтологии в интеллектуальных и информационных системах

Онтология выступает в качестве базового средства формализации областей знаний для широкого класса интеллектуальных и информационных систем, в частности, для экспертных систем, систем поддержки принятия решений и различных систем информационной поддержки научной и производственной деятельности.

Экспертные системы [Попов, 1987] относятся к классу программных систем, аккумулирующих знания специалистов в конкретных предметных областях и тиражирующих этот эмпирический опыт для консультаций менее квалифицированных пользователей. В связи с этим основным и наиболее важным компонентом ЭС является база знаний. Именно полнота и непротиворечивость представленных в ней знаний, определяют мощность экспертной системы и качество получаемых ею решений.

Онтология может составлять каркас базы знаний ЭС, создавать базис для описания основных понятий предметной области (ПО) и служить основой для интеграции баз данных, содержащих фактические знания, необходимые для полноценного функционирования экспертной системы. Кроме того, в терминах онтологии могут описываться экспертные правила и прецеденты [Варшавский и др., 2006], что значительно повышает их уровень описания и «понимаемость» пользователями-экспертами.

Интеллектуальные системы поддержки принятия решений (ИСППР) [Петровский, 2009] помогают лицу, принимающему решения (ЛПР), использовать данные и модели для решения его профессиональных слабо формализуемых задач. ИСППР и ЭС относятся практически к одному классу интеллектуальных систем. Очень часто в ИСППР включается несколько различных ЭС, поэтому все, что было сказано выше о роли онтологии в ЭС, относится и к ИСППР. В то же время имеются аспекты использования онтологий, специфичные для СППР.

Так, из-за того, что ИСППР в основном предназначена для решения плохо формализуемых задач, очень важно иметь детальное непротиворечивое описание проблемной области, в рамках которой система осуществляет поддержку решения задач ЛПР. И онтология является незаменимым инструментом для создания такого описания.

В большинстве ИСППР используются большие массивы разнородных данных и знаний. Благодаря тому, что онтология позволяет явно описывать семантику данных и знаний, она обеспечивает базис для их интеграции и совместного использования при решении задач.

Система информационной поддержки научной и

производственной деятельности [Ануреев и др., 2009], как правило, должна обеспечивать своим пользователям представление всей информации об определенной области знаний и/или производства, о ее составляющих (разделы/подразделы науки и/или отрасли/подотрасли производства, объекты, методы и техники исследования и т.п.), а также о субъектах (участниках) научной и/или производственной деятельности (персоналий, групп, сообществ и других организаций, включенных в процесс исследования и/или производства). При этом она должна предоставлять содержательный доступ ко всем перечисленным знаниям и данным, т.е. удобную навигацию по всему информационному пространству системы и возможность поиска информации в терминах представляемой области знаний.

В системах такого класса онтология задает формальное описание области знаний, на основе которого строится информационное наполнение (контент) системы, выполняется интеграция в единое информационное пространство (ИП) релевантных информационных ресурсов и документов. На основе онтологии строится пользовательский интерфейс, обеспечивающий содержательный доступ к знаниям и данным, интегрированным в информационное пространство системы. В частности, в таком интерфейсе пользователь может использовать онтологию в качестве «проводника» для навигации по ИП системы, а также формулировать поисковые запросы, основными элементами которых являются понятия и отношения онтологии.

Заметим также, что при разработке интеллектуальных и информационных систем возможно использование онтологий, ранее разработанных для области знаний этих систем. Это позволяет переиспользовать уже проверенные на практике знания, что обеспечивает высокое качество создаваемых систем и их потенциальную интегрируемость с ранее разработанными.

Таким образом, онтологии могут использоваться как при разработке интеллектуальных и информационных систем, так и в качестве полноправного компонента в процессе их функционирования.

2. Средства формализации знаний на основе онтологий

Существует множество подходов к формализации знаний в интеллектуальных и информационных системах на основе онтологий. В этой главе будут рассмотрены наиболее популярные из них, а также подходы, разработанные в нашем коллективе.

2.1. Средства формализации знаний в программной среде SemP-TAO

Рассмотрим подход к формализации знаний, предлагаемый в программной среде SemP-TAO

[Загорулько Ю.А. и др., 1996], предназначенной для построения широкого класса интеллектуальных систем.

Модель представления знаний программной среды SemP-ТАО объединяет на основе объектно-ориентированного подхода такие классические средства представления и обработки знаний, как семантические сети и продукционные правила, а также недоопределенные модели [Нариньяни и др., 1998] и методы программирования в ограничениях. Благодаря такому сочетанию средств, в данной среде эффективно решаются задачи, которые требуют сочетания логического вывода и вычислений над неточно (частично) заданными значениями.

Язык представления и обработки знаний (ЯПОЗ) технологической среды SemP-ТАО включает два типа средств – средства для описания предметной области и средства для спецификации приложений, предназначенных для использования в рамках описанной предметной области.

Первая группа средств позволяет представлять предметную область в виде классов объектов и отношений, а также конструировать необходимые для этого новые типы данных и домены. Описания классов и отношений составляют онтологию, которая задает семантику и структуру семантической сети – основного средства представления декларативных знаний в SemP-ТАО.

Вторая группа средств позволяет описывать функциональность приложения, главным образом, в виде системы продукционных правил, работающей над семантической сетью. Язык содержит не только продукционные правила и средства оперирования семантической сетью, включая операторы создания объектов и бинарных отношений, их редактирования (изменения значений атрибутов объектов) и удаления из сети, но и средства динамического управления активацией продукционных правил, а также. Кроме того, язык включает традиционные средства императивного программирования, а также средства для создания эргономичных пользовательских интерфейсов.

Условие применения правила состоит из образца, включающего описание объектов и отношений, заданных в онтологии, а также глобальных и локальных предикатов, накладывающих ограничения, соответственно, на контекст, в котором применяется данное правило, и на значения аргументов отношений и атрибутов объектов, описанных в образце.

В правой части правила разрешается обращаться к операторам, оперирующим семантической сетью, а также другим операторам и функциям, разрешенным в языке. В частности, включение в правую часть правил средств динамического управления активацией правил позволяет обеспечить высокую гибкость управления процессом вывода и обработки информации.

Возможность использования в посылках и заключениях продукционных правил понятий (классов) и отношений, определенных в онтологии, значительно повышает уровень описания правил и их «понимаемость» экспертами.

2.2. Формализация знаний на основе дескриптивных логик

Дескриптивные логики (description logics) – это семейство логик, специально спроектированных для представления знаний о предметной области [Baader et al., 2003; Тузовский и др., 2005].

Дескриптивные логики (ДЛ) должны были прийти на смену таким средствам и формализмам представления знаний, как семантические сети, фреймы и исчисление предикатов, объединив их достоинства и избавившись от их недостатков. Известно, что семантические сети и фрейм-представления, обладая высокими по выразительности средствами для представления знаний о предметной области, не обеспечивают достаточно формального описания ПО. В то же время исчисление предикатов дает высокий уровень формализации описаний, но не предоставляет удобных средств для представления специфических знаний о предметной области.

Для того, чтобы ДЛ можно было использовать в качестве формализма представления знаний в интеллектуальных системах, необходимо, чтобы такие системы отвечали на запросы пользователей за разумное время. В связи с этим процедуры логического вывода, применяемые в ДЛ, в отличие, например, от алгоритмов доказательства теорем логик первого порядка, должны всегда завершаться, как при положительных, так и при отрицательных ответах. Поэтому ДЛ часто рассматривается как область разрешимых частей логики первого порядка.

Разрешимость и сложность задач вывода зависят от выразительной мощности конкретной дескриптивной логики, построенной в рамках концепции ДЛ. С одной стороны, очень экспрессивные ДЛ сталкиваются с задачами вывода высокой сложности, которые могут быть даже неразрешимыми. С другой стороны, очень слабые ДЛ (с эффективными процедурами вывода) могут не обладать достаточной выразительностью для представления важных понятий конкретной прикладной задачи. Определение баланса между выразительностью конкретной дескриптивной логики и сложностью задач вывода является одной из наиболее важных проблем при ее конструировании (или выборе).

Базовыми элементами ДЛ являются концепты (понятия) и роли. Каждый *концепт* представляет класс, категорию или сущность. *Роли* служат для описания свойств концептов и отношений между ними. В ДЛ также входят *конструкторы (операции)*, с помощью которых можно строить из

базовых элементов понятия и роли более высокого уровня.

База знаний, построенная средствами дескриптивной логики представляется в виде пары *Tbox* и *Abox*. *Tbox* (*terminological knowledge*) – это набор утверждений, описывающих терминологический словарь базы знаний, т.е. набор классов, их свойства и отношения между ними (интенциональные знания). *Abox* (*assertional knowledge*) представляет собой реализацию схемы классов в виде набора экземпляров, содержащих утверждения об экземплярах понятий (экстенциональные знания). По существу *Tbox* является моделью того, что *может* быть истинным, а *Abox* является моделью того, что *в настоящее время* является истинным.

Дескриптивный язык ДЛ имеет модельно-теоретическую семантику [Tarski, 1956]. Поэтому утверждения в *TBox* и в *ABox* могут отождествляться с формулами в ЛПП или, в некоторых случаях, незначительными их расширениями.

Благодаря тому, что ДЛ имеют формальную семантику, они позволяют не только описать терминологию и утверждения, но также предоставляют возможности по выполнению на их основе логического вывода (автоматического доказательства). Автоматическое доказательство обеспечивает ответы на такие запросы, как: выполнимость понятий (*satisfiability*) – может ли существовать некоторое понятие *C*, описание которого задано в *Tbox*; включение (*subsumption*) – является ли некоторое понятие *C* частным случаем (подмножеством) понятия *D*; согласованность (*consistency*) – является ли вся БЗ согласованной; проверка экземпляра (*instance checking*) – является ли некоторое утверждение истинным.

Понятия могут строиться из выражений. В состав операторов включается базовый набор операций (дополнение, объединение и пересечение множеств) и количественные ролевые ограничения.

С помощью автоматического доказательства можно формировать ответы на запросы к *Tbox* и *Abox*. Запросы к *Tbox* определяют классификацию и отношения между понятиями, а запросы к *Abox* определяют текущее состояние известных фактов. Выполнимость понятия является доказательством того, что понятие, или понятийное выражение является логически согласованным с БЗ. Кроме этого операция включения ДЛ (*subsumption*) может использоваться для обнаружения отношений класс-подкласс, которые заданы в БЗ неявно.

Таким образом, ДЛ определяют формальный язык для описания семантики ПО вместе с теорией доказательства (процедурами логического вывода).

Достоинством ДЛ является то, что они позволяют строить формальные декларативные языки, обладающие большой выразительной мощностью для представления знаний. Другое их

достоинство – применяемые в них процедуры логического вывода, как правило, завершаются, как при положительных, так и при отрицательных ответах. В то же время ДЛ трудны для освоения экспертами, которые не являются специалистами в области математики и программирования.

2.3. Средства формализации знаний подхода Semantic Web

Средства подхода Semantic WEB – это набор методов и технологий, предназначенных для представления информации в пригодном для машинной обработки виде. Эти средства позволяют представлять информацию в виде семантической сети, специфицированной с помощью онтологий. Благодаря стандартизованному представлению программа-клиент может непосредственно извлекать факты и делать из них логические заключения, используя протокол HTTP и идентификаторы ресурсов URI. Использование средств Semantic WEB стало в последние 10 лет очень популярным для разработки интеллектуальных приложений.

Наиболее выразительным средством описания онтологий для семантической паутины, является язык OWL (*Web Ontology Language*) [OWL, 2013]. В основе этого языка – представление действительности в модели данных «объект – свойство». OWL пригоден для описания не только веб-страниц, но и любых объектов действительности. Каждому элементу описания в этом языке (в том числе свойствам, связывающим объекты) ставится в соответствие URI.

OWL обеспечивает лучшую выразительность и более богатую семантику, чем многие другие языки описания структур, такие как XML, RDF, RDF Schema (RDF-S) [Лапшин, 2010] и другие.

Язык OWL имеет три диалекта (в порядке возрастания выразительности):

- OWL Lite является самым простым диалектом и поддерживает только классификационную иерархию понятий, элементарные ограничения и простейшие виды аксиом.
- OWL DL обеспечивает максимальную выразительность при сохранении полноты вычислений (все заключения гарантировано будут вычислимыми) и разрешимости (все вычисления производятся за конечное время). OWL DL включает все языковые конструкции OWL, но ограничивает свободу их использования (например, в то время как класс может быть подклассом многих классов, класс не может быть представителем другого класса). OWL DL так назван из-за его соответствия дескриптивным логикам (в частности, SHOIN(D)).
- OWL Full предоставляет максимальную выразительность и синтаксическую свободу RDF, но без гарантий вычислений за конечное время. Например, в OWL Full класс может рассматриваться

одновременно как собрание индивидов и как один индивид в своем собственном значении.

При использовании OWL для построения онтологий интеллектуальных или информационных систем наиболее часто используют диалекта OWL DL, поскольку имеющиеся машины вывода рассчитаны на поддержку OWL только в рамках данного диалекта.

Онтологии, написанные на языке OWL, включают в себя описание классов и их свойств, а также экземпляры классов (индивидов), совокупность которых вместе со связывающими их отношениями образует семантическую сеть.

Для записи OWL-онтологий могут использоваться разные форматы: RDF/XML, OWL/XML, Манчестерский синтаксис.

Resource Description Framework (далее – RDF) – это разработанная консорциумом W3C модель для представления данных, в особенности – метаданных. RDF представляет утверждения о ресурсах в виде, пригодном для машинной обработки.

Для поиска информации в онтологии используются такие языки как SPARQL и DL Query.

SPARQL (рекурсивный акроним от английского SPARQL Protocol and RDF Query Language) — это язык запросов к данным, представленным по модели RDF, а также протокол для передачи этих запросов и ответов на них. SPARQL является рекомендацией консорциума W3C и одной из технологий Semantic WEB. Существуют реализации языка запросов SPARQL для ряда языков программирования.

Язык DL Query предназначен для осуществления запросов к онтологии и основан на Манчестерском синтаксисе.

Логический вывод в OWL-онтологиях осуществляется на основе аксиом онтологии средствами одной из машин вывода, каковых уже разработано достаточное количество (Pellet, FaCT++, HermiT). Помимо аксиом можно задавать правила вывода, которые позволяют выводить информацию, не содержащуюся явным образом в онтологии. Для этого можно использовать язык SWRL (Semantic Web Rule Language), основанный на объединении языков OWL-DL и RuleML. Ядром RuleML является язык Datalog, который, в свою очередь, является синтаксическим подмножеством языка Prolog. Этим и объясняется то, что SWRL-правила состоят из хорновских дизъюнктов.

Для работы с OWL онтологиями в программных приложениях используются различные библиотеки. Наиболее популярными из них являются OWL API и Jena.

Для разработки онтологий создано множество редакторов. Наиболее популярным является Protégé [Protégé, 2013]. Protégé – это свободно распространяемый редактор онтологий с открытым

кодом и одновременно фреймворк для построения баз знаний. Онтологии, построенные в Protégé, могут быть экспортированы во множество форматов, включая RDF (RDF Schema), OWL, и XML Schema.

Protégé имеет открытую, легко расширяемую архитектуру за счет поддержки модулей расширения функциональности. Это позволяет настраивать редактор под нужды различных приложений. Популярность Protégé объясняется и тем, что он поддерживается значительным сообществом, включающим разработчиков и ученых, правительственных и корпоративных пользователей.

Таким образом, подход Semantic Web дает богатый набор средств формализации знаний, который постоянно расширяется и развивается благодаря поддержке консорциума W3C. Эти средства обладают не только значительной выразительностью, но еще и являются разрешимыми. Пока слабым звеном этим средств является язык SWRL, который позволяет описывать очень простые правила, выразительности которых недостаточно для разработки даже примитивных экспертных систем.

Другой проблемой в использовании средств, предоставляемых подходом Semantic Web, является их высокая сложность для простых пользователей, не знакомых с современными логиками. Это затрудняет использование этих средств в технологиях построения информационных и интеллектуальных систем, ориентированных на экспертов.

2.4. Средства формализации знаний в технологии построения порталов научных знаний

Рассмотрим подход к формализации семантики области знаний на основе онтологий, который использован в технологии построения порталов научных знаний [Загорулько и др., 2008].

Эта технология предназначена для разработки порталов научных знаний (ПНЗ) – специализированных интернет-порталов, обеспечивающих интеграцию и систематизацию научных знаний и информационных ресурсов определенной области знаний, а также содержательный доступ к ним из любой «точки» интернет-пространства. Портал научных знаний не только предоставляет доступ к собственным информационным ресурсам, но и поддерживать навигацию по заранее размеченным (проиндексированным) ресурсам, размещенным в сети Интернет.

Технология ориентирована на создание и сопровождение порталов экспертами, не являющимися специалистами в области информатики и программирования, и с минимальным участием (на первых этапах разработки) инженеров знаний. Важной особенностью технологии является обеспечение

возможности декларативной настройки портала на заданную область знаний не только в процессе создания, но и в ходе эксплуатации. Реализация такой возможности позволяет отслеживать динамику появления новых знаний и информационных ресурсов по тематике портала и, тем самым, обеспечит поддержку его актуальности и полезности.

Создание технологии с такими свойствами стало возможным благодаря выбору онтологии в качестве средства формализации семантики области знаний и использованию ее в качестве концептуальной основы и информационной модели портала знаний онтологии.

Для того, чтобы онтологии успешно выполняли функцию формализации семантики моделируемых областей знаний, требуются удобные и эффективные средства их представления, а также методологии (технологии), облегчающие процесс их построения и делающие их более надежными.

Прежде всего, для представления онтологии необходим формализм, обеспечивающий описание понятий моделируемой области знаний, их свойств и разнообразных семантических связей (отношений) между ними. Важным требованием к формализму описания онтологии является возможность выстраивания понятий в иерархию «общее–частное» и поддержка наследования свойств по этой иерархии. Этот формализм должен также предоставлять возможность задания ограничений на значения возможных свойств (атрибутов) объектов – экземпляров понятий онтологии.

Для повышения компактности описания области знаний, а также для поддержки дополнительного контроля за использованием отношений, формализм должен предоставлять возможность приписывания отношениям различных математических свойств, таких как симметричность, транзитивность, антисимметричность и др

Кроме того, для представления сложно-структурированной информации требуются бинарные отношения, снабженные дополнительными атрибутами, специализирующими связь между двумя сущностями (аргументами отношения). Примером такого отношения может быть, например, отношение, связывающее между собой персону и организацию, в которой эта персона работает. Дополнительные атрибуты здесь нужны для того, чтобы отразить, в какой должности работает данная персона в этой организации и с какого времени.

Такой формализм был разработан (его подробное описание можно найти в [Загорулько, 2013a]) и используется в качестве базиса при построении онтологий порталов научных знаний.

На рисунке 1 представлены основные компоненты данной технологии. Она включает в себя набор универсальных и базовых онтологий,

методологию построения онтологий и программные средства поддержки процесса построения и редактирования онтологий и тезауруса, заполнения и редактирования контента портала, средств поиска и навигации по онтологии и контенту, а также пользовательский интерфейс.

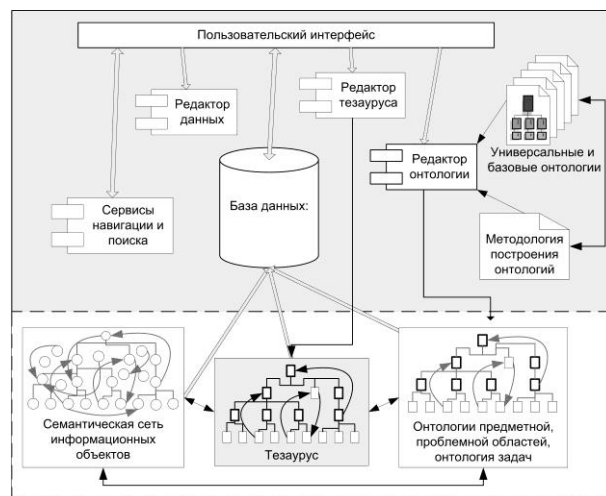


Рисунок 1 – Технология построения порталов научных знаний

Для обеспечения унифицированного представления разнородных знаний и данных, учета их связанности, а также поддержки функциональности ПНЗ предложена информационная модель, которая объединяет модели проблемной области и области (областей) знаний портала, а также построенную на их основе модель представления данных.

В качестве ядра, базового компонента информационной модели портала научных знаний была выбрана онтология, что позволило не только обеспечить его основную функциональность, но и сделало настраиваемым на любую область научных знаний.

Создание портала начинается с построения его онтологии. Согласно предложенной методологии эта онтология создается на основе базовых онтологий путем специализации и расширения имеющейся в них системы понятий. Это значительно облегчает создание онтологии и ее дальнейшее сопровождение.

В качестве базовых выбраны онтология научной деятельности и онтология научного знания, на основе которой строится онтология области знаний портала.

Онтология научной деятельности, выступающая в качестве онтологии проблемной области портала знаний, базируется на онтологии, предложенной в [Benjamins et al., 1998] для описания научно-исследовательских проектов и расширенной в [Загорулько и др., 2008] для применения к более широкому классу задач. Она включает базовые классы понятий, относящиеся к организации научной и исследовательской деятельности, такие как *Исследователь*, *Организация*, *Событие*, *Деятельность (Проект)*, *Публикация* и др. В эту

онтологию также включен класс *Информационный ресурс*, который служит для описания информационных ресурсов, представленных в сети Интернет.

Онтология научного знания фиксирует основные содержательные структуры, используемые для построения предметных онтологий. В частности, эта онтология содержит мета-понятия, задающие структуры для описания понятий конкретной области знаний, такие как *Раздел науки*, *Метод исследования*, *Объект исследования*, *Научный результат*.

Понятия базовых онтологий связаны между собой ассоциативными отношениями, выбор которых осуществляется не только исходя из полноты представления проблемной и предметной областей портала, но и с учетом удобства навигации по его контенту и поиска информации.

Понятия онтологии области знаний портала являются реализациями метапонятий онтологии научного знания и могут быть упорядочены в иерархию «общее-частное».

Вводя формальные описания понятий предметной области в виде классов объектов и отношений между ними, онтология портала задает структуры для представления реальных объектов и связей между ними. В соответствии с этим данные на портале представлены в виде семантической сети, т.е. как множество разнотипных информационных объектов и связей между ними, которые в совокупности образуют информационное содержание (контент) портала.

Для управления онтологиями используется редактор онтологий, реализованный как web-приложение и доступный через Интернет. Этот редактор проектировался и разрабатывался таким образом, чтобы он был понятен экспертам, прост и удобен в использовании. В частности, из-за этих требований мы отказались от такого популярного средства построения онтологий как редактор Protégé [Protégé, 2013].

Ввод информации осуществляется с помощью управляемого онтологией редактора данных, который позволяет создавать, редактировать и удалять информационные объекты (ИО) и связи между ними. Формы для ввода конкретных ИО и их связей автоматически генерируются по онтологии.

Содержательный доступ к систематизированным знаниям и информационным ресурсам заданной области знаний обеспечивается с помощью предоставляемых порталом развитых средств навигации и поиска, функционирование которых также базируется на онтологии. Благодаря этому основной сценарий работы пользователя с порталом состоит из выбора либо с помощью средств визуализации онтологии, либо с помощью механизма поиска объектов определенного класса, их просмотра, навигации по их связям (реализациям

отношений онтологии) и фильтрации списков таких объектов.

Таким образом, данная технология поддерживает разработку порталов научных знаний без участия разработчиков-программистов. Для создания портала требуются только инженеры знаний – специалисты в представлении знаний и эксперты – носители знаний в моделируемой области. Использование онтологий позволяет сделать процесс разработки портала научных знаний более технологичным, в частности, переиспользовать ранее полученные и формализованные знания.

2.5. Использование онтологий в технологии разработки тематических интеллектуальных научных интернет-ресурсов

Подход к построению порталов научных знаний получил развитие в технологии разработки тематических интеллектуальных научных интернет-ресурсов (ИНИР) [Загорулько и др., 2013b], которые, как и ПНЗ, предназначены для осуществления информационной поддержки научной и производственной деятельности в определенной области знаний. Основу ИНИР, как и порталов научных знаний составляют онтологии.

Главное отличие данной технологии от технологии ПНЗ состоит в существенном использовании семантических web-сервисов. Такие сервисы в технологии ИНИР выступают и в качестве внутренних функциональных компонентов ИНИР, расширяющих функциональность ИНИР, и в качестве внешних информационных ресурсов, накапливаемых и систематизируемых в ИНИР с тем, чтобы обеспечить к ним содержательный доступ.

Тематический ИНИР представляет собой доступную через Интернет информационную систему, обеспечивающую систематизацию и интеграцию научных знаний и информационных ресурсов определенной области знаний, содержательный эффективный доступ к ним (поиск и навигацию) и поддерживающую их использование при решении различных научных и производственных задач за счет предоставления соответствующих интерфейсов и сервисов.

Система знаний ИНИР базируется на формализмах онтологий и семантических сетей. При этом онтология составляет ядро системы знаний и наряду с описанием моделируемой области знаний содержит соотнесенное с ним описание структуры и типологии интегрируемых информационных ресурсов и методов интеллектуальной обработки содержащихся в них данных и знаний.

Семантическая сеть, структура которой определяется онтологией ИНИР, играет роль интеллектуального хранилища данных, в котором накапливается информация о релевантных научных информационных ресурсах и web-сервисах,

реализующих методы обработки содержащейся в них информации.

На основе онтологии и семантической сети организуется удобная навигация по научным знаниям и информационным ресурсам, интегрированным в ИНИР, а также содержательный поиск требующихся данных и средств их интеллектуальной обработки (представленных, в том числе, в виде web-сервисов).

Особенностью данного подхода является то, что ИНИР позволяет значительно сократить время, которое требуется для обеспечения доступа к необходимой информации и ее анализа, за счет предварительного поиска релевантных интернет-ресурсов и аккумуляции их описаний непосредственно в семантической сети ИНИР. С этой целью в программную оболочку ИНИР включается подсистема сбора онтологической информации (метаданных) о релевантных интернет-ресурсах, которая в своей работе опирается на онтологию и тезаурус области знаний ИНИР.

Система знаний ИНИР (см. рисунок 2) включает онтологию ИНИР, тезаурус области знаний ИНИР, семантическую сеть, служащую для представления информационного содержания (контента) ИНИР, а также интегрируемые в ИНИР информационные ресурсы и средства их интеллектуальной обработки (web-сервисы).



Рисунок 2 – Система знаний ИНИР

Онтология ИНИР состоит из трех взаимосвязанных онтологий, отвечающих за представление указанных выше компонентов знаний, а именно: онтологии области знаний ИНИР, онтологии научных интернет-ресурсов и онтологии задач и методов.

Онтология области знаний ИНИР строится на основе двух базовых онтологий, включенных в программную оболочку ИНИР, – онтологий научной деятельности и онтологий научного знания, унаследованных из технологии построения ПНЗ.

Основным классом онтологии научных интернет-ресурсов является класс *Информационный ресурс*, который предназначен для описания, представленных в сети Интернет информационных ресурсов. Набор атрибутов и связей этого класса основан на стандарте Dublin core [Hillmann., 1995-2013]. Его атрибутами являются: название ресурса, язык ресурса, тематика ресурса, тип доступа к ресурсу и т.п. Объекты этого класса могут быть связаны семантическими отношениями с другими информационными объектами, представляющими в контенте ИНИР организации, персоны, публикации, события, разделы науки и т.д.

Онтология задач и методов кроме описания задач, на решение которых нацелен ИНИР, и методов их решения включает также описания web-сервисов, реализующих методы обработки информации, содержащейся в интегрируемых в ИНИР информационных ресурсах. Описания таких web-сервисов базируются на онтологии OWL-S [OWL-S, 2004], предназначенной для описания семантических web-сервисов (Semantic Web Services) [Cabral, 2004] и представленной на языке OWL.

Таким образом, онтология ИНИР, вводя формальные описания понятий некоторой области знаний, типов интегрируемых информационных ресурсов и методов их интеллектуальной обработки в виде классов объектов и отношений между ними, одновременно задает структуры для представления информации о реальных объектах моделируемой области знаний, интегрируемых информационных ресурсах и методах и средствах обработки содержащейся в них информации. Данная информация хранится в контенте ИНИР в виде объектно-ориентированной семантической сети, типы информационных объектов и отношений которой определяются классами объектов и отношений, введенных в онтологии ИНИР.

Тезаурус в ИНИР служит для описания смысла понятий (терминов), используемых в моделируемой области знаний. При этом тезаурус позволяет задавать смысл понятий не только с помощью определений, но и посредством соотнесения их с другими понятиями. Благодаря этому тезаурус может применяться как при обработке пользовательских запросов, так и при поиске и аннотировании информационных ресурсов [Лукашевич, 2011], интегрируемых в ИНИР.

Тезаурус ИНИР строится на основе ядра тезауруса, изначально включенного в систему знаний ИНИР. Ядро тезауруса содержит описание понятий базовых онтологий, включая описание терминов, с помощью которых они представляются в интернет-ресурсах.

ИНИР имеет традиционную трехуровневую архитектуру, включающую уровень доступа к информации, уровень обработки информации и базовый уровень.

Первый уровень обеспечивается пользовательским интерфейсом – программным компонентом, предоставляющим конечному пользователю содержательный доступ к контенту ИНИР при решении его задач.

Уровень обработки информации обеспечивает все информационные потоки в ИНИР – от конструирования онтологии до обработки пользовательских запросов. Он включает модуль поиска информации в информационном пространстве ИНИР, средства разработки/настройки базы знаний ИНИР и управления его контентом, а также подсистему сбора онтологической информации (метаданных) об интернет-ресурсах.

Базовый уровень обеспечивает управление всеми данными и знаниями информационного пространства ИНИР. Он выполняет функции хранения и манипулирования данными (контентом ИНИР) и знаниями (онтологией и тезаурусом) с использованием средств стандартных СУБД (MySQL), технологий Semantic Web (OWL, RDF) и семантических web-сервисов (WSDL, OWL-S). В частности, в качестве хранилища данных используется свободно распространяемая версия системы Virtuoso (Virtuoso Open-Source Edition) [Virtuoso, 2013], обеспечивающая эффективную работу с большими объемами данных в RDF-формате (RDF Triple Store) [Erling, 2009]. Кроме того, эта система предоставляет возможность взаимодействия с web-сервисами.

3. Использование онтологий в ЭС и СППР

Как было показано выше, онтологии используются как при разработке ЭС и СППР, так и во время их функционирования.

3.1. Использование онтологий в медицинских ЭС. Экспертная система для подбора оздоровительной программы

Данная система предназначена для оказания помощи человеку при подборе подходящей для него оздоровительной программы.

На основе данных о конкретном человеке (антропометрические данные, список заболеваний и проблем, связанных с его здоровьем и самочувствием, преследуемые им цели) система предлагает ему наиболее подходящие виды физических нагрузок, диеты, восстановительные или профилактические процедуры. Кроме того, система может выдать справочную информацию о восстановительных комплексах, их показаниях и противопоказаниях.

Знания в системе представлены OWL-онтологией, фрагмент которой показан на рисунке 3.

OWL-онтология создана средствами редактора Protégé. Логический вывод в рассматриваемой системе реализуется на основе заданных в онтологии аксиом средствами машины вывода

Pellet. Аксиомы задают ограничения для понятий онтологии. Кроме того, в системе используются запросы на языке SPARQL, которые также записываются в терминах онтологии.

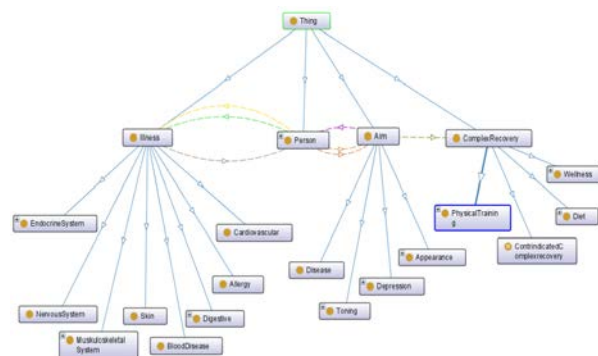


Рисунок 3 – Онтология комплексного восстановления организма человека.

3.2. Использование онтологий в системе поддержки принятия решений на нефтегазодобывающем предприятии

Покажем, как используются онтологии в разработанной нами СППР [Загорюлько и др., 2010], которая на основе анализа статической и динамической информации об объектах технологической инфраструктуры нефтегазодобывающего предприятия должна вырабатывать для ЛПР рекомендации по предотвращению аварийных ситуаций, улучшению показателей работы предприятия, о проведении планового технического обслуживания и/или экстренного ремонта объектов, о списании объектов и замене их новыми и пр.

Данная СППР имеет достаточно гибкую архитектуру, позволяющую подключать к ней различные методы решения задач и решатели. Эта гибкость во многом достигается благодаря тому, что в систему в качестве полноправного компонента включены две взаимосвязанные онтологии – онтология предметной области и онтология задач, обеспечивающие настройку системы на предметную область и типы решаемых задач.

Базовыми понятиями онтологии ПО (см. рисунок 4) являются: *Объект*, *Нормативно-справочный объект*, *Состояние*, *Результат*.

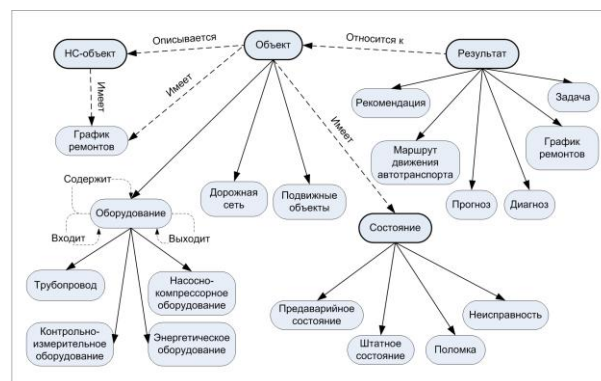


Рисунок 4 – Фрагмент онтологии ПО.

В классе *Объект* выделяются подклассы *Оборудование* (насосы, трансформаторы, трубопроводы и т.п.), *Подвижные объекты* (автоцистерны, грузовики, автобусы и другие виды автотранспорта) и *Дорожная сеть*. Объекты могут находиться в том или ином состоянии, для описания которого вводится класс *Состояние*.

Важное место в онтологии ПО занимает описание топологии технологической инфраструктуры предприятия: взаимное расположение и связность объектов задается отношениями «Содержит», «Входит», «Выходит».

Для контроля соответствия параметров объектов анализа нормативным значениям используется нормативно-справочная информация: для каждого понятия, представляющего некоторый вид оборудования или контейнера оборудования, заводится специальное понятие, которое мы будем называть нормативно-справочный объект или *НС-объект*. Этот *НС-объект* имеет тот же набор атрибутов, что и исходное понятие (объект), но числовые значения его атрибутов задаются парой чисел, которые определяют интервал изменения атрибута данного понятия согласно нормативу (техпаспорту).

Результатами работы СППР являются *Диагнозы* состояния оборудования и подвижных объектов, *Рекомендации* для ЛПР, *Прогнозы* изменения состояния объектов, решения транспортных задач (*Маршруты движения автотранспорта*), скорректированные *Графики ремонтов* и т.п. Отдельным результатом может быть порожденная исполняемым модулем поддержки принятия решений *Задача*, в которой, например, более детально анализируется тот или иной объект.

В данной системе онтология ПО, с одной стороны, выступает в качестве высокоуровневого интерфейса к внутреннему хранилищу данных, обеспечивая доступ к ним в виде объектов предметной области, с другой стороны, она определяет формат представления данных в самой СППР в виде тех же объектов предметной области (экземпляров понятий онтологии) и отношений между ними, что позволяет упростить и унифицировать обмен информацией между разнородными компонентами и модулями (решателями) СППР. В терминах онтологии также описываются экспертные правила (см. ниже).

Онтология задач (см. рисунок 5), описывая типы решаемых системой задач, определяет тем самым ее функциональность. Она включает описания задач, модулей, реализующих их решения, и используемых ими решателей.

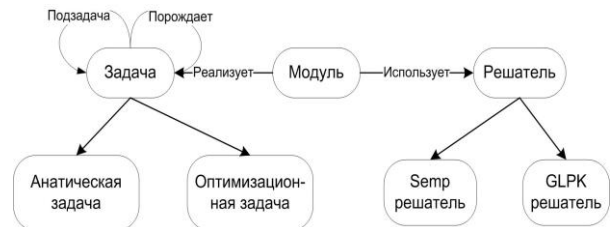


Рисунок 5 – Фрагмент онтологии задач.

Для решения аналитических задач используются модули поддержки принятия решений, реализованные в парадигме продукционной модели. При этом для спецификации входящих в эти модули продукционных правил, а также обеих онтологий используется язык представления и обработки знаний системы Semp-TAO, что позволяет использовать введенные в онтологиях понятия и отношения в посылках и заключениях продукционных правил. Для исполнения этих модулей поддержки принятия решений используется интерпретатор продукционных правил системы Semp-TAO (Semp-решатель).

Приведем пример продукционного правила:

FORALL NR: P: *Насос* (*Capacity*, *Energy_consumption*, *Efficiency*),
НС_Насос (*Efficiency_min*),
Описывается (P, NR),
Efficiency > *Efficiency_min* &
Tendency (*Capacity*) = “*падаем*” &
Tendency (*Efficiency*) = “*падаем*” &
Tendency (*Energy_consumption*) = “*растем*”
=>
Create Прогноз (*message*: “КПД достигнет критического уровня через” +
Critical_moment (*Efficiency*, *Efficiency_min*) +
“часов”, *parameters*: {<Pump, P>});
Create Задача (*name*: “Диагностика насоса”,
parameters: {<Pump, P>}).

Заметим, что в этом правиле *Capacity*, *Energy_consumption* и *Efficiency* являются временными рядами параметров насоса *подача*, *потребляемая энергия* и *кпд насоса P*, а NR – это нормативно-справочный объект насоса *P*.

Это правило говорит о том, что если для какого-либо насоса возрастает потребление электроэнергии при одновременном снижении кпд и подачи, и эти показатели, оставаясь в нормативных рамках, приближаются к их границам, то с определенной долей уверенности можно прогнозировать, что этот насос скоро выйдет из строя. При возникновении такой ситуации это правило выдаст соответствующее сообщение ЛПР и запустит задачу детальной диагностики проблемного насоса.

4. Использование онтологий в системах информационной поддержки научной и производственной деятельности

4.1. Использование онтологий в порталах научных знаний

Рассмотренная в разделе 2.4 технология построения порталов научных знаний была использована в рамках проектов по созданию научных интернет-порталов по компьютерной лингвистике и археологии, выполнявшихся на базе ИСИ СО РАН.

Разработанный совместно с Институтом археологии и этнографии СО РАН **археологический портал знаний** [Андреева и др., 2006] был создан для решения задачи интеграции накопленных знаний и информационных ресурсов по археологии, а также обеспечения содержательного доступа к ним. Этот портал рассчитан на широкий круг пользователей – от научных работников и преподавателей до студентов и школьников, интересующихся достижениями археологической науки.

Онтология области знаний портала включает четыре базовых иерархии:

- иерархию *Разделов науки*, отражающую иерархию основных направлений научной деятельности в археологии. Примерами таких направлений являются *Общая*, *Полевая* и *Реконструктивная археология*, в свою очередь подразделяющиеся на более частные;
- иерархию *Объектов*, определяющую объекты исследования в археологии, примерами которых могут являться археологические культуры, этносы, исторические личности, предметы быта, обнаруженные в процессе раскопок;
- иерархию *Методов исследования*, служащую для описания подходов, принципов, технологий, методик исследования, применяемых в археологии к определенному типу археологических объектов;
- иерархию *Научных результатов*, позволяющую описать результаты научной деятельности археологии, такие как открытия, гипотезы, новые законы, теории, исторические факты.

Зафиксированных в онтологии научных знаний метапонятий оказалось недостаточно для отражения хронологических и географических характеристик археологических знаний. Поэтому онтология научного знания была дополнена новыми классами понятий, характерными для исторических наук:

- класс *Археологический период* служит для датирования объектов исследования и определения хронологического положения разделов науки;
- класс *Местоположение* позволяет задавать географическую привязку разделов науки (например, для разделов *Океаническая археология*, *Археология Урала*) и историко-хронологические

связи между периодами времени и их географическим местоположением (например, при задании понятий Древняя Греция или Мезолит в Северной Европе).

Контент портала представляет знания об основных разделах археологии, ее объектах, методах и результатах исследования, которые объединены по географическому и хронологическому принципу, снабжены текстовыми описаниями и ссылками на публикации. Заданная в контенте портала информация позволяет организовать содержательный доступ к ресурсам, представляющим базы данных и архивы визуальных (графических) и библиографических документов, материалов деятельности археологических организаций, отчетов экспедиций, а также описания музейных коллекций.

Портал знаний по компьютерной лингвистике [Боровикова и др., 2008] разработан для организации эффективного доступа к лингвистическим ресурсам. Пользователями такого портала могли бы стать как научные работники, преподаватели и студенты, исследующие, преподающие и изучающие эту дисциплину, так и специалисты, разрабатывающие программные системы, предназначенные для обработки текстов, анализа и синтеза речи.

В качестве онтологии области знаний в данном портале выступает онтология компьютерной лингвистики (КЛ), построенная на основе онтологии научного знания.

Для отражения в исследуемом материале компьютерной лингвистики того аспекта, на который направлена научная деятельность, онтология научного знания была дополнена новым понятием *Предмет исследования*.

В результате исследований было выделено около 200 понятий компьютерной лингвистики, которые были организованы в следующие иерархии «общее–частное»:

- иерархию *Объектов исследования*, рассматривающую следующие объекты моделирования языка: *Речевое произведение* (РП), как объективную форму существования и использования естественного языка, и *Структурные языковые единицы* в составе РП, соответствующие различным языковым уровням – предложения, словосочетания, слова, морфемы, звуки и интонационные единства;
- иерархию *Методов исследования*, служащую для систематизированного описания инструментов исследования, применяемых в компьютерной лингвистике;
- иерархию *Разделов науки*, в основе которой лежит классификация базовых теоретических и прикладных направлений компьютерной лингвистики;
- иерархию *Научных результатов*, служащую для типизации и описания результатов научной

деятельности. В этой иерархии были выделены такие классы: *Технологии и программные продукты*, *Прикладные системы*, *Лингвистические ресурсы*. Последний класс делится на такие классы, как *Корпуса*, *Лингвистические БД*, *Онтологии*, *Словари* и *тезаурусы*;

- иерархию *Предметов исследования*, которая описывает *Процессы*, связанные с функционированием языковых единиц в коммуникации, и *Прикладные процессы*, имеющие практическую ценность, отвечающие определенному социальному запросу.

Так как портал посвящен компьютерной лингвистике, в его контенте в первую очередь представлены знания об основных разделах компьютерной лингвистики, о ее предметах и объектах исследования, используемых в ней моделях и методах.

Ресурсы компьютерной лингвистики представлены непосредственно результатами деятельности организаций и отдельных исследователей, полученными в ходе выполнения научных и коммерческих проектов. К таким ресурсам относятся как технологии, программные продукты, прикладные системы, так и лингвистические ресурсы: словари, корпуса (текстов и речи) и лингвистические БД. Для организации более эффективного доступа к таким ресурсам в контенте представлена информация о различных аспектах их разработки: организациях, персонах и проектах, с которыми связано их появление, а также о таких содержательных характеристиках ресурсов, как отнесенность к разделу науки, объекту или предмету исследования, методам исследования. Эта информация связывает ресурсы с остальными данными и знаниями, представленными в контенте портала, что позволяет пользователю выделить группы ресурсов, созданные, например, в ходе осуществления некоторой исследовательской деятельности (гранта, проекта, конкурса) или с использованием определенного метода исследования.

4.2. Использование онтологий в системе информационной поддержки разработчиков интеллектуальных СППР

Значительно повысить скорость и эффективность создания СППР, а также их качество, может хорошая информационная поддержка разработчиков [Загорулько Г., 2013]. Она должна обеспечить разработчиков концептуальным и методологическим базисом, предоставить им систематизированные знания о научной дисциплине "Теория принятия решений" и структурированные описания методов принятия решений и решателей или пакетов, реализующих эти методы. Полученная информация позволит разработчикам лучше понять стоящие перед создаваемой СППР задачи и подобрать наиболее подходящие методы и средства для их решения.

Для того чтобы разработчик СППР смог получить удобный доступ ко всей необходимой ему

при создании СППР информации и эффективно воспользоваться ею, он, прежде всего, должен иметь представление о принципах организации и систематизации такой информации, т.е. ее концептуальной основе. В качестве концептуальной основы информационной поддержки разработчика предлагается использовать онтологию задач и методов принятия решений (ЗиМППР). Рассмотрим, какими особенностями должна обладать онтология ЗиМППР.

Порталы научных знаний, рассматриваемые в предыдущем разделе, ориентированы на обслуживание одного типа пользователей, например, ученого или производственника. Этим определялись и структура онтологий портала, и требования к ним. Однако существует потребность в создании систем, отражающих интересы различных типов пользователей. Поэтому формализация семантики таких областей знаний должна учитывать несколько типов пользователей.

Одной из таких проблемных областей является разработка СППР. В этой области мы имеем, по крайней мере, три типа пользователей – эксперт, инженер знаний, программист. Каждый из этих пользователей на своем уровне выступает как ЛПР, и каждому из них нужна такая же поддержка, какая осуществляется порталом знаний. Но ситуация здесь осложняется тем, что мы как и прежде строим одну систему информационной поддержки, но рассчитанную на разные группы пользователей одновременно. Это ведет к существенному усложнению системы знаний, которая не может строиться путем включения отдельных онтологий для каждого типа пользователей, так как эти онтологии имеют общих частей не меньше, чем различных. И если эти онтологии строить независимо одна от другой, то это может привести к противоречиям в системе понятий, а в дальнейшем и к противоречивости всей системы знаний. В связи с этим в рассматриваемой в этом разделе системе информационной поддержки была предложена структура онтологии, отличная от структуры онтологий для традиционных ПНЗ.

Для того, чтобы свести профессиональную терминологию всех типов пользователей к общей системе понятий ЗиМППР должна иметь так называемую «вертикально-горизонтальную» структурную организацию, что позволит представлять интересующую область знаний в двух измерениях (см. рисунок 6).

Первое измерение («вертикальная структуризация») представляется в виде традиционной иерархической структуры, на каждом уровне которой интересующая область знаний описывается с разной степенью детализации.

Второе измерение («горизонтальная структуризация») задает описание области знаний с точки зрения разных типов специалистов, участвующих в процессе создания и использования СППР.



Рисунок 6 – Вертикально-горизонтальная организация онтологии ЗиМППР

В качестве общего ядра, на котором строится вертикально-горизонтальная онтология ЗиМППР, выступает метаонтология задач и методов, которая содержит описание таких базовых понятий поддержки принятия решений, как Задача, Метод, Модуль, Решатель, Входные данные, Результат, Ситуация, Проблемная ситуация, Альтернатива, Этап принятия решений, а также отношения между ними [Загоруйко Г., 2013а].

Онтология ЗиМППР тесно связана с понятиями предметных областей, для которых создаются СППР. Для того чтобы дать представление о задачах и методах, не вдаваясь в конкретику ПО, описания этих областей должны быть одинаково устроены. Это задается путем использования в качестве основы для построения онтологии конкретной предметной области метаонтологии ПО, включающей базовые понятия, являющиеся общими для всех ПО.

Для реализации информационной поддержки разработчиков СППР были использованы средства технологии создания порталов научных знаний, которая позволяет поддержать изложенные в докладе концептуальные и методологические принципы обеспечения информационной поддержки. На основе онтологии, описывающей предметную область, она позволяет строить структурированный контент, представляющий собой семантическую сеть объектов, привязанных к понятиям онтологии ПО, где предметная область – это поддержка принятия решений, а объекты – это конкретные задачи, методы, этапы и т.д.

Заключение

В докладе показано, что формализация семантики области знаний в виде онтологий – это весьма продуктивный подход. Он позволяет эффективно представлять как объективные знания (ориентированные на машинную обработку), так и субъективные знания (ориентированные на различные группы пользователей), необходимые для функционирования информационных и интеллектуальных систем.

Было отмечено, что к настоящему времени разработано большое число средств различного

уровня для формализации семантики – логики, языки, редакторы, среды.

В докладе также показано, что онтологии могут использоваться как при проектировании и разработке информационных и интеллектуальных систем, так и в качестве полноправного компонента во время функционирования системы.

При разработке информационных и интеллектуальных систем онтология может использоваться для:

- формирования и фиксации общего разделяемого всеми экспертами знания о предметной области;
- описания экспертных правил и прецедентов;
- явной концептуализации ПО, позволяющей описывать семантику данных;
- обеспечения возможности переиспользования знаний;
- описания функциональности системы (типов решаемых задач);
- разработки отдельных компонентов системы (в качестве их высокоуровневой спецификации);
- разработки хранилищ данных (в качестве высокоуровневой спецификации структуры хранилища);

В процессе функционирования систем перечисленного класса онтология может применяться для обеспечения:

- совместного использования разнородных данных и знаний в рамках одной системы;
- процесса решения задач, составляющих функциональность системы;
- лучшего понимания предметной области пользователями системы.

Работа выполнена при финансовой поддержке РФФИ (проект № 13-07-00422) и Президиума РАН (интеграционный проект СО РАН № 15/10 «Математические и методологические аспекты интеллектуальных информационных систем»)

Библиографический список

- [Андреева и др., 2006] О.А. Андреева, О.И. Боровикова, С.В. Булгаков, Ю.А. Загоруйко, Е.А. Сидорова, Б.Г. Циркин, Ю.П. Холушкин. Археологический портал знаний: содержательный доступ к знаниям и информационным ресурсам по археологии // Труды 10-й национальной конференции по искусственному интеллекту с международным участием - КИИ'2006. – Москва: Физматлит, 2006. –Т.3. С.832-840.
- [Ануреев и др., 2009] Ануреев И.С., Батура Т.В., Боровикова О.И., Загоруйко Ю.А., Кононенко И.С., Марчук А.Г., Марчук П.А., Мурзин Ф.А., Сидорова Е.А., Шилов Н.В. Модели и методы построения информационных систем, основанных на формальных, логических и лингвистических подходах / Отв. ред. А.Г. Марчук ; Рос. акад. наук, Сиб. отд-ние, Ин-т систем информатики им. А.П. Ершова. – Новосибирск: Изд-во СО РАН, 2009. ISBN 978–5–7692–1113–3. – 330 с.
- [Боровикова и др., 2008] Боровикова О.И., Загоруйко Ю.А., Загоруйко Г.Б., Кононенко И.С., Соколова Е.Г. Разработка портала знаний по компьютерной лингвистике // Труды 11-ой национальной конференции по искусственному

интеллекту с международным участием КИИ-2008. М.: ЛЕНАНД, 2008. Т.3. С.380-388.

[Варшавский и др., 2006] Варшавский, П.Р. Методы правдоподобных рассуждений на основе аналогий и прецедентов для интеллектуальных систем поддержки принятия решений / П.Р. Варшавский, А.П. Еремеев // Новости Искусственного Интеллекта, № 3, 2006, С. 39-62.

[Загорюлько Ю.А. и др., 1996] Попов И.Г.. Представление знаний в интегрированной технологической среде Semp-ТАО. // Проблемы представления и обработки не полностью определенных знаний. –Москва-Новосибирск, 1996. –С.59–74.

[Загорюлько и др., 2008] Загорюлько Ю.А., Боровикова О.И. Подход к построению порталов научных знаний // Автометрия. № 1, 2008, т. 44. С. 100–110.

[Загорюлько и др., 2010] Загорюлько Ю.А., Загорюлько Г.Б., Кравченко А.Ю., Сидорова Е.А. Разработка системы поддержки принятия решений для нефтегазодобывающего предприятия // Труды 12-й национальной конференции по искусственному интеллекту с международным участием – КИИ-2010. – Москва: Физматлит, 2010. Т.3. С.137-145.

[Загорюлько, 2013] Загорюлько Ю.А. О концепции интегрированной модели представления знаний // Известия Томского политехнического университета. – 2013. – Т. 322. – № 5. С. 98-103.

[Загорюлько, 2013а] Загорюлько Ю.А. Технологии разработки интеллектуальных систем, основанные на интегрированной модели представления знаний // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2013): материалы III Междунар. научн.-техн. конф. (Минск, 21-23 февраля 2013 г.) / редкол.: В. В. Голенков (отв. ред.). – Минск: БГУИР, 2013. С. 31–42.

[Загорюлько и др., 2013б] Ю.А. Загорюлько, Г. Б. Загорюлько, В.К. Шестаков, И.С. Кононенко. Концепция и архитектура тематического интеллектуального научного интернет-ресурса // Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды XV Всероссийской научной конференции RCDL'2013, Ярославль, Россия, 14-17 октября 2013 г. – Ярославль: ЯрГУ, 2013. С.57–62.

[Лапшин, 2010] Лапшин В.А. Онтологии в компьютерных системах. М.: Научный мир, 2010. – 224 с.

[Загорюлько Г., 2013] Загорюлько Г.Б. Обеспечение информационной поддержки разработчиков СППР // Информационные и математические технологии в науке и управлении / Труды XVIII Байкальской Всероссийской конференции «Информационные и математические технологии в науке и управлении». Часть III. – Иркутск: ИСЭМ СО РАН, 2013. С. 137–142.

[Загорюлько Г., 2013а] Загорюлько Г.Б., Загорюлько Ю.А. Подход к интеграции разнородных методов поддержки принятия решений для сложных задач // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2013): материалы III Междунар. научн.-техн. конф. (Минск, 21-23 февраля 2013 г.) / редкол.: В. В. Голенков (отв. ред.). – Минск: БГУИР, 2013. С. 265–268.

[Лукашевич, 2011] Лукашевич Н.В. Тезаурусы в задачах информационного поиска. –М.: Изд-во МГУ, 2011.

[Мартынов, 1974] Мартынов В.В. Семитологические основы информатики. –Минск: Наука и техника, 1974. –192 с.

[Нариньяни и др., 1998] Нариньяни, А.С. Программирование в ограничениях и недоопределённые модели / А.С. Нариньяни, В.В. Телерман, Д.М. Ушаков, И.Е. Швецов // Информационные технологии, 1998. – №7. С. 13-22.

[Петровский, 2009] Петровский А.Б. Теория принятия решений. Москва, Издательский центр «Академия», 2009.

[Попов, 1987] Попов Э.В. Экспертные системы: Решение неформализованных задач в диалоге с ЭВМ.. –М.: Наука. 1987. – 288 с.

[Тузовский и др., 2005] Тузовский А. Ф., Чириков С. В., Ямпольский В. З. Системы управления знаниями (методы и технологии). – Томск: Изд-во НТЛ, 2005. 260 с.

[Baader et al., 2003] Baader, F. The Description Logic Handbook: Theory, Implementation, Applications / F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider – Cambridge, 2003. 574 P.

[Benjamins et al., 1998] Benjamins V.R. and Fensel D. Community is Knowledge! in (KA)2 // Proceedings of the 11th Banff

Knowledge Acquisition for Knowledge-based Systems workshop, KAW'98 (Banff, Canada, April 1998). – Calgary: SRDG Publications, Department of Computer Science, University of Calgary, 1998.

[Cabral, 2004] L. Cabral, J. Domingue, E. Motta, T. Payne, F. Hakimpour. Approaches to semantic web services: an overview and comparisons // The semantic web: Research and applications, 2004. p. 225-239.

[Erling, 2009] Erling O., Mikhailov I. RDF Support in the Virtuoso DBMS // Networked Knowledge-Networked Media. – Springer Berlin Heidelberg, 2009. p. 7-24.

[Gruber, 1995] Gruber T. Toward Principles for the Design of Ontologies Used for Knowledge Sharing // International Journal of Human-Computer Studies. November 1995. Vol. 43. Issues 5–6. P. 907–928.

[Guarino, 1998] Guarino N. Formal Ontology in Information Systems // Formal Ontology in Information Systems. Proceedings of FOIS'98, Trento, Italy, June 6–8, 1998 / Ed. N.Guarino. Amsterdam: IOS Press, 1998. P. 3–15.

[Hillmann., 1995-2013] Hillmann D. Using Dublin Core. <http://dublincore.org/documents/usageguide/> (дата обращения: 14.12.2013)

[OWL-S, 2004] OWL-S: Semantic Markup for Web Services. <http://www.w3.org/Submission/OWL-S/> (дата обращения: 14.12.2013)

[OWL, 2013] OWL Web Ontology Language Guide. W3C Recommendation 10 February 2004. URL: <http://www.w3.org/TR/owl-guide/> (дата обращения: 14.12.2013).

[Protégé, 2013] Protégé. URL: <http://protege.stanford.edu/> (дата обращения: 14.12.2013).

[SWRL, 2013] SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission 21 May 2004. <http://www.w3.org/Submission/SWRL/> (дата обращения: 14.12.2013).

[Tarski, 1956] Tarski A. Logic, Semantics, Metamathematics. – Oxford University Press, 1956. – 258 p.

[Virtuoso, 2013] Virtuoso open-source edition. OpenLink Software. <http://www.openlinksw.com/wiki/main> (дата обращения: 14.12.2013)

ONTOLOGY-BASED FORMALIZATION OF SEMANTICS OF KNOWLEDGE AREAS IN INFORMATION AND INTELLIGENT SYSTEMS

Zagorulko Yu.A. *, Zagorulko G.B. *

** A.P. Ershov Institute of Informatics Systems
Siberian Branch of the Russian Academy of
Sciences, Novosibirsk, Russia*

zagor@iis.nsk.su

gal@iis.nsk.su

The paper discusses the role of ontology in intelligent and information systems, the ontology-based approaches to and means for formalizing the semantics of knowledge areas are presented and the examples of practical use of ontologies in systems of this class are discussed.

The paper shows that the formalization of the semantics of the knowledge area in the form of ontologies is a very productive approach. It allows you to effectively represent both objective knowledge (oriented to machine use) and subjective knowledge (targeted at different groups of users) of the modeled area, necessary for the operation and information intelligent systems.

In conclusion, it is concluded that the ontology can be used successfully at both development time and run time of information and intelligent system.



УДК 004.822:514

КАКОЙ ДОЛЖНА БЫТЬ ПАРАДИГМА РЕШЕНИЯ ЗАДАЧ НА ОСНОВЕ БАЗ ЗНАНИЙ?

Грибова В.В., Клещев А.С.

Федеральное государственное бюджетное учреждение науки Институт автоматизации и процессов управления Дальневосточного отделения Российской академии наук, г. Владивосток, Россия

gribova@iacp.dvo.ru,

kleshev@iacp.dvo.ru

По аналогии с парадигмой решения задач в рамках алгоритмического подхода и современными требованиями к разработке интеллектуальных систем предлагается парадигма решения задач на основе баз знаний. Далее анализируется соответствие существующих технологий предложенной парадигме.

Ключевые слова: интеллектуальные системы; семантические технологии; онтологии; базы знаний.

Введение

В настоящее время при решении задач на компьютере наиболее широко используется алгоритмический подход. Для этого подхода сложилась общепринятая парадигма решения задач. Однако далеко не все задачи могут быть решены в рамках алгоритмического подхода, что стимулирует разработку других подходов к решению задач. Одним из таких подходов является решение задач на основе баз знаний, который привел к появлению нового класса программных средств – интеллектуальных систем или систем с базами знаний. Вместе с тем, для этого подхода до сих пор не предложена парадигма решения задач аналогичная таковой для алгоритмического подхода.

В настоящей работе предлагается парадигма для решения задач на основе баз знаний и показывается, в какой степени существующие технологии разработки интеллектуальных систем ей удовлетворяют.

1. Парадигма решения задач в рамках алгоритмического подхода

Решение задач на компьютере обычно связано с необходимостью получения какой-либо информации. Формально эта потребность в информации, получаемой в результате решения задачи, выражается через определение множества Y всех возможных результатов решения этой задачи. Чтобы получить информацию $y \in Y$, необходима некоторая исходная информация, которая формально представляется исходными данными x

задачи. Множество всех допустимых исходных данных задачи, которое должно быть формально определено, обозначим через X . Формальная постановка задачи или ее спецификация есть предикат $P(x, y)$, где $x \in X$, а $y \in Y$. Для допустимой спецификации задачи должно быть доказано утверждение о существовании ее решения:

$$\forall x \in X \exists y \in Y P(x, y);$$

желательно также, чтобы для нее было справедливо (доказано) и утверждение о единственности ее решения:

$$\forall x \in X \exists! y \in Y P(x, y).$$

В случае, если справедливы оба эти утверждения, разрабатывается алгоритм решения задачи, который представляет собой реализацию на алгоритмическом языке такого всюду определенного функционального отображения $A: X \rightarrow Y$, что справедливо утверждение о правильности этого алгоритма:

$$\forall x \in X P(x, A(x)).$$

Если же для спецификации задачи утверждение о единственности ее решения не является справедливым, то обычно алгоритм ее решения представляет собой реализацию такого всюду определенного функционального отображения $A: X \rightarrow 2^Y$, что справедливы утверждения о нахождении этим алгоритмом всех решений задачи:

$$\forall x \in X \forall y \in A(x) P(x, y)$$

$$\text{и } \forall x \in X y \notin A(x) \rightarrow \neg P(x, y).$$

Если при решении задачи пользователь использует программу, правильно реализующую алгоритм A , то он отвечает за правильность входных данных x программы; в этом случае он получает правильный результат решения задачи $y = A(x)$.

Считается, что любая технология решения задач на компьютере в рамках алгоритмического подхода должна удовлетворять приведенной в этом разделе парадигме.

2. Парадигма решения задач на основе баз знаний

Построим парадигму решения задач на основе баз знаний по аналогии с парадигмой решения задач в рамках алгоритмического подхода, приведенной в предыдущем разделе, иллюстрируя результаты ее применения на двух типах интеллектуальных систем, достаточно далеких друг от друга – системах компьютерной диагностики заболеваний и системах поиска математических доказательств.

Любая интеллектуальная система предназначена для решения некоторой задачи, поэтому для нее должна быть сформулирована и формализована потребность в информации через определение множества возможных решений Y этой задачи. Для системы компьютерной диагностики заболеваний это множество диагнозов и их объяснений, а для системы поиска математических доказательств – множество полных доказательств. Поскольку при использовании интеллектуальных систем результат решения задачи представляется в некоторой концептуальной системе, определение множества Y содержит определение соответствующей системы понятий (онтологию), а $y \in Y$ представляет собой множество экземпляров этих понятий. При этом множество Y должно быть определено таким образом, чтобы предикат принадлежности $y \in Y$ был вычислимым, а форма представления информации $y \in Y$ должна быть понятной пользователям интеллектуальной системы без каких-либо посредников или дополнительной их подготовки. Последнее может быть достигнуто лишь в случае, если онтология, содержащаяся в определении множества Y представляет собой формализацию концептуальной системы пользователей.

Таким же образом следовало бы определить множество X допустимых исходных данных задачи. Для системы компьютерной диагностики заболеваний это должно быть множество всех реальных историй болезни, а для системы поиска математических доказательств – множество теорем. Однако из этих примеров видно, что множество X не может быть определено точно – множество всех возможных историй болезни и множество всех теорем нетривиальной математической теории неизвестны. Поэтому обычно определяется более широкое множество $X' \supseteq X$; для системы компьютерной диагностики заболеваний это множество объектов, имеющих вид историй болезни

и представимых на соответствующем искусственном языке, а для системы поиска математических доказательств – множество правильных математических предложений, также представимых на соответствующем языке.

Кроме того, в случае использования интеллектуальных систем таким же способом должно быть определено множество допустимых баз знаний $K(X', Y)$, а также спецификация задачи – предикат $P(x, k, y)$, где $x \in X'$, $k \in K(X', Y)$, $y \in Y$, причем предикаты принадлежности $x \in X'$ и $k \in K(X', Y)$ должны быть вычислимыми, а форма представления информации $x \in X'$ и $k \in K(X', Y)$ должна быть понятной экспертам и пользователям интеллектуальной системы без каких-либо посредников или дополнительной их подготовки. Последнее может быть достигнуто лишь в случае, если онтология, содержащаяся в определениях множеств X' и $K(X', Y)$ представляют собой формализацию концептуальных систем экспертов и пользователей. Для системы компьютерной диагностики заболеваний базой знаний являются описания проявлений различных заболеваний в зависимости от возможных, индивидуальных особенностей пациентов, воздействий различных событий (например, лечебных мероприятий) на проявления заболеваний, причинно-следственных отношений между заболеваниями и т.п. [Черняховская, 1983], а спецификация задачи компьютерной диагностики – описание отношения между историями болезни, базами знаний, диагнозами пациентов и их объяснениями [Клещев и др., 2005, Клещев и др., 2006]. Для системы поиска математических доказательств базой знаний может быть некоторая аксиоматическая система и множество правил вывода, а спецификацией задачи – описание отношения между теоремами, аксиоматическими системами, правилами вывода и доказательствами. Кроме того, онтологии X' и $K(X', Y)$ должны использоваться для управления формированием $x \in X'$ и $k \in K(X', Y)$, а результат вычисления последних отношений принадлежности должен контролироваться.

В случае интеллектуальных систем особенностью спецификации задачи является то, что для нее не является справедливым утверждение о существовании решения. Для системы компьютерной диагностики заболеваний неявно предполагается, что существует «правильная» база знаний $k^* \in K(X', Y)$ такая, что

$$\forall x \in X \exists y \in Y P(x, k^*, y);$$

однако эта «правильная» база знаний неизвестна, а для любой другой базы знаний утверждение о существовании решения не обязано быть справедливым. Кроме того, это утверждение может быть нарушено и для $x \in X' \setminus X$. Для системы поиска математических доказательств, напротив, «правильная» база знаний $k^* \in K(X', Y)$ известна, однако входные данные $x \in X'$ – это математические предложения, относительно которых неизвестно,

являются они теоремами (в этом случае $\exists y \in Y \ P(x, k^*, y)$) или нет (в этом случае такого $y \in Y$ не существует). Тем более, для интеллектуальных систем не является справедливым утверждение о единственности решения даже в случае его существования.

Хотя для спецификации задачи утверждение о существовании решения не является справедливым, для нее можно сформулировать более слабое утверждение:

$$\exists k^* \in K(X', Y) \forall x \in X \exists y \in Y \ P(x, k^*, y),$$

а в некоторых случаях и слабое утверждение о единственности решения:

$$\exists k^* \in K(X', Y) \forall x \in X \exists! y \in Y \ P(x, k^*, y).$$

Эти утверждения являются лишь предположениями и не могут быть доказаны.

Для системы компьютерной диагностики заболеваний – это предположения о существовании «правильной» базы знаний k^* , в терминах которой может быть объяснен правильный диагноз любой реальной истории болезни, и о единственности диагноза у каждого пациента, а для системы поиска математических доказательств – это предположение о том, что любое математическое предложение (теорема), истинное в любой модели аксиоматической системы k^* , имеет доказательство в этой аксиоматической системе (как правило, не единственное).

В предположении справедливости для спецификации задачи этого более слабого утверждения о существовании решения можно разрабатывать алгоритм решения задачи, который представляет собой реализацию на алгоритмическом языке такого частично определенного функционального отображения

$$A: \langle X', K(X', Y) \rangle \rightarrow Y$$

(или $A: \langle X', K(X', Y) \rangle \rightarrow 2^Y$), для которого справедливо (может быть доказано) «слабое» утверждение о правильности этого алгоритма:

$$\forall x \in X' \forall k \in K(X', Y) \ P(x, k, A(x, k)).$$

Для системы компьютерной диагностики заболеваний такой алгоритм для истории болезни x (если это реальная история болезни, полученная в результате наблюдения врачом реального пациента, то $x \in X$) и базы знаний $k \in K(X', Y)$ (например, сформированной экспертом) находит все возможные диагнозы пациента x и их объяснения в терминах базы знаний k . Для системы поиска математических доказательств такой алгоритм для математического предложения $x \in X'$ и базы знаний k^* , являющейся теорией некоторой аксиоматической системы, должен находить одно из возможных доказательств (если оно существует).

Поскольку «сильное» утверждение о существовании решения для спецификации задачи не является справедливым, а алгоритм решения

задачи $A(x, k)$ может применяться к любым входным данным $x \in X'$ и к любой базе знаний $k \in K(X', Y)$, этот алгоритм обладает некоторыми нежелательными свойствами. Для системы компьютерной диагностики заболеваний при некоторых исходных данных и базе знаний множество решений задачи $A(x, k)$ может не содержать «правильного» решения (т.е. такого, которое затем было верифицировано иным способом), содержать «неправильные» решения (т.е. такие, которые будут отвергнуты в процессе последующей верификации) или даже быть пустым (в случае, если исходные данные x противоречат базе знаний k). Для системы поиска математических доказательств, если математическое предложение $x \in X' \setminus X$ (т.е. не является теоремой), то алгоритм решения задачи $A(x, k^*)$ может работать бесконечное время (следствие теоремы Черча).

Если при решении задачи пользователь использует интеллектуальную систему с базой знаний $k \in K(X', Y)$, правильно реализующую алгоритм A , то он, как и раньше, отвечает за правильность входных данных x программы; однако в этом случае он может получить и неправильный, и неточный результат решения задачи $y = A(x, k)$, либо даже не получить никакого результата.

Для того, чтобы технология интеллектуальных систем была жизнеспособной, она должна включать процедуры, позволяющие уменьшать долю случаев, в которых система обладает указанными выше нежелательными свойствами [Грибова и др., 2010]. Эти процедуры могут быть разными для различных типов интеллектуальных систем. Приведем примеры таких процедур для двух рассматриваемых типов интеллектуальных систем.

Для системы компьютерной диагностики заболеваний назовем базой прецедентов репрезентативную выборку $\{\langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle\}$, где $x_i \in X$ – реальная история болезни, а $y_i \in Y$ – объяснение ее верифицированного диагноза (для $i = 1, \dots, n$). Будем называть эту базу прецедентов оценкой правильности базы знаний $k \in K(X', Y)$, $C(k) = \{\langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle\}$, если

$$\forall i \in \{1, \dots, n\} \ y_i \in A(x_i, k)$$

(аналогично можно определить оценку точности базы знаний). Будем говорить, что база знаний $k \in K(X', Y)$ приближается к «правильной» базе знаний k^* , если в оценке ее правильности $C(k)$ параметр n неограниченно растет. Процедуру, которая обеспечивает приближение базы знаний к «правильной» базе знаний, будем называть системой управления базой знаний. Она может быть автоматической, т.е. реализующей такое отображение, которое базе прецедентов $\{\langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle\}$ ставит в соответствие такую базу знаний $k \in K(X', Y)$, что эта база прецедентов оказывается оценкой точности этой базы знаний $C(k) = \{\langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle\}$ [Клещев и др., 2012]. Либо эта процедура может быть человеко-

машинной: для базы прецедентов $\{ \langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle \}$ система управления находит множество возможных баз знаний $k \in K(X', Y)$, такое, что эта база прецедентов оказывается оценкой точности каждой из этих баз знаний $C(k) = \{ \langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle \}$, а эксперт выбирает «наилучшую» базу знаний из этого множества по своему усмотрению.

Системы поиска математических доказательств эффективно могут применяться в задаче верификации моделей интуитивных доказательств [Клещев, 2011]. Назовем (простой) моделью интуитивного доказательства математического предложения $x \in X'$ конечную последовательность математических предложений $x_1, x_2, \dots, x_{n-1}, x_n, x$. Будем говорить, что интуитивное доказательство верифицировано с использованием базы знаний $k \in K(X', Y)$, если построены: доказательство $y_1 \in Y$ математического предложения $x_1 \in X$ с использованием базы знаний $k \in K(X', Y)$; доказательство $y_2 \in Y$ математического предложения $x_2 \in X$ с использованием базы знаний $k \cup \{x_1\}$; ...; доказательство $y_n \in Y$ математического предложения $x_n \in X$ с использованием базы знаний $k \cup \{x_1, x_2, \dots, x_{n-1}\}$; доказательство $y_{n+1} \in Y$ математического предложения $x \in X$ с использованием базы знаний $k \cup \{x_1, x_2, \dots, x_{n-1}, x_n\}$. Будем называть оценкой эффективности базы знаний $k \in K(X', Y)$ множество математических предложений, для которых система поиска математических доказательств с этой базой знаний построила доказательства. Системой управления базой знаний, расширяющей оценку эффективности базы знаний, будем называть процедуру, состоящую из двух действий: если доказано некоторое математическое предложение $x \in X$, то это предложение включается в базу знаний; если при этом построено доказательство $y \in Y$, то оно обобщается до метадоказательства my [Клещев, 2007], и это метадоказательство включается в базу знаний. В этом случае процедура построения доказательства математического предложения x_i , входящего в модель интуитивного доказательства предложения x , может состоять из четырех шагов: поиск в базе знаний предложения x_i' , для которого x_i является частным случаем (в случае успеха система управления базой знаний включает x_i в базу знаний); поиск в базе знаний метадоказательства my_i , конкретизацией которого является полное доказательство y_i предложения x_i (в случае успеха система управления базой знаний включает x_i в базу знаний); автоматический поиск полного доказательства y_i предложения x_i с использованием множества предложений и правил вывода, содержащихся в базе знаний (в случае успеха система управления базой знаний включает x_i в базу знаний, обобщает y_i до метадоказательства my_i и включает это метадоказательство в базу знаний); интерактивное построение полного доказательства y_i предложения x_i с использованием множества предложений и правил вывода, содержащихся в базе

знаний (в случае успеха система управления базой знаний включает x_i в базу знаний, обобщает y_i до метадоказательства my_i и включает это метадоказательство в базу знаний).

3. Соответствие технологий разработки интеллектуальных систем парадигме решения задач на основе баз знаний

Обсудим, в какой степени современные технологии разработки интеллектуальных систем соответствуют описанной выше парадигме. Рассмотрим следующие технологии: (а) основанную на использовании оболочек экспертных систем [Bachant et al., 1984, Compton, 1988]; (б) основанную на использовании WebProtege [Gennari et al., 2003, Musen, 2007]; (с) основанную на использовании платформы IACPaaS [Грибова и др., 2011, Грибова и др., 2013а, Грибова и др., 2013б].

Технологии (а) и (б) не поддерживают спецификацию множеств Y, X' и $K(X', Y)$ таким образом, чтобы предикаты принадлежности $y \in Y, x \in X'$ и $k \in K(X', Y)$ были вычислимыми. Технология (с) поддерживает такую спецификацию - множества Y, X' и $K(X', Y)$ специфицируются как метаязыки, для которого определена и поддерживается логическая семантика. Форма представления знаний понятна специалистам соответствующих предметных областей при использовании технологий (б) и (с).

Технологии (а) и (б) не предлагают аппарата для представления спецификации задач в форме предиката $P(x, k, y)$, где $x \in X', k \in K(X', Y), y \in Y$. Для технологии (а) разработаны методы управления базами правил, а для технологии (б) такая задача вообще не рассматривалась. Технология (с) предлагает аппарат для представления спецификации задач в форме предиката, с использованием которого можно сформулировать слабое утверждение о существовании решения задачи, а также слабое утверждение о правильности алгоритма решения задачи. Имеется опыт доказательства такого утверждения для достаточно сложного случая [Клещев и др., 2013]. Кроме того, разработан прототип системы автоматического управления базой знаний для системы компьютерной диагностики заболеваний [Клещев и др., 2012].

Инструментальная поддержка технологии (а) включает инструменты для формирования баз правил и проблемно-независимый решатель задач (оболочку); разработка интерфейса не поддерживается (фиксированный интерфейс является частью оболочки). Кроме того, эти инструменты не являются облачными, как и создаваемые с их помощью интеллектуальные системы.

Инструментальная поддержка технологии (б) включает редактор для поддержки формирования баз знаний в декларативной форме [Crubezy, 2004],

множество решателей некоторых классов [Chandrasekaran, 1986] задач и средства адаптации этих решателей к базам знаний. Только редактор является облачным. Проблема формирования баз знаний экспертами без посредников или дополнительного их обучения решена в этом редакторе не в полной мере.

Инструментальная поддержка технологии (с) включает редактор IWE для поддержки формирования баз знаний в декларативной форме (в котором проблема формирования баз знаний экспертами без посредников или дополнительного их обучения решена в полной мере), средства разработки решателей задач в виде систем взаимодействующих агентов, ориентированные на представление баз знаний, а также средства разработки интерфейсов таких решателей. Все эти средства являются облачными, как и создаваемые с их помощью интеллектуальные системы.

Заключение

В настоящее время осознана острая необходимость разработки технологии и инструментальной поддержки создания интеллектуальных систем. Стандартная технология, основанная на алгоритмическом подходе и соответствующие ей средства разработки не применимы к разработке интеллектуальных систем. В первую очередь это связано с тем, что в архитектуре интеллектуальных систем имеется дополнительный компонент – база знаний, которая должна разрабатываться и сопровождаться экспертами предметной области и быть им понятной. В тоже время для интеллектуальных систем не предложено парадигмы решения задач, в результате чего все технологии разрабатываются и развиваются, как правило, интуитивно. Это ведет к тому, что невозможно объективно оценить их и наметить пути дальнейшего развития данной области исследований. Предложенная в работе парадигма может стать, в случае ее принятия заинтересованным сообществом, основой для оценивания технологий и дальнейшего развития технологий и инструментальной поддержки разработки интеллектуальных систем.

Работа выполнена при финансовой поддержке ДВО РАН (инициативный научный проект 12-I-ОНИТ-04) и при финансовой поддержке РФФИ (проект 12-07-00179-а).

Библиографический список

- [Bachant et al., 1984] Bachant, J. McDermott, J. "R1 revisited: four years in the trenches", The AI Magazine (Fall 1984; Fall), pp.21-32
- [Chandrasekaran, 1986] Chandrasekaran B. Generic tasks in knowledge-based reasoning High-level building blocks for expert system design // IEEE Expert, 1986. 1(1). - Pp.23-30.
- [Compton, 1988] Compton, P. Horn, K. Quinlan, R. Lazarus, L. "Maintaining an expert system". Proceedings of the fourth Australian Conference on Applications of Expert Systems, (1988), pp.110-129.

[Crubezy, 2004] Crubezy M., Musen M.A. Ontologies in support of problem-solving // In S. Staab & R. Studer, (Eds.) Handbook on ontologies. - Berlin: Springer, 2004. - Pp. 321-341.

[Gennari et al., 2003] Gennari, J.H.; Musen, M.A.; Fergerson, R.W., etc. The evolution of Protégé: An environment for knowledge-based systems development/International Journal of Human-Computer Studies. 2003. 58(1):89-123

[Musen, 2007] Musen M. Technology for building intelligent systems: from psychology to engineering // Nebraska Symposium on Motivation. 2007. 02. 52. Pp. 145-84.

[Грибова и др. 2010] Грибова В.В., Клещев А.С., Шалфеева Е.А. Управление интеллектуальными системами // Известия РАН. Теории и системы управления. 2010. № 6. С. 122-137.

[Грибова и др., 2011] Грибова В.В., Клещев А.С., Крылов и др. Проект IASPaas – развиваемый комплекс для разработки, управления и использования интеллектуальных систем // Искусственный интеллект и принятие решений. – 2011. – №1.

[Грибова и др., 2013a] Грибова В.В., Клещев А.С. Технология разработки интеллектуальных сервисов, ориентированных на декларативные предметные базы знаний. Часть 1. Информационные ресурсы // Информационные технологии. 2013. №9. С. 7-11.

[Грибова и др., 2013b] Грибова В.В., Клещев А.С. Технология разработки интеллектуальных сервисов, ориентированных на декларативные предметные базы знаний. Часть 2. Решатель задач. Пользовательский интерфейс // Информационные технологии. 2013. №10. С. 10-14.

[Клещев, 2007] Клещев А.С. Модель аналогии между математическими доказательствами // Проблемы управления. – 2007. – №1. – С.20-24.

[Клещев, 2011] Клещев А.С. Операционная модель интуитивных доказательств // Проблемы управления. 2011. №1. С. 2-7.

[Клещев и др., 2005] Клещев А.С., Москаленко Ф.М., Черняховская М.Ю. Модель онтологии предметной области «Медицинская диагностика». Часть 1. Неформальное описание и определение базовых терминов // Научно-техническая информация. Сер. 2. – 2005. - № 12. – С. 1-7.

[Клещев и др., 2006] Клещев А.С., Москаленко Ф.М., Черняховская М.Ю. Модель онтологии предметной области «Медицинская диагностика». Часть 2. Формальное определение причинно-следственных связей, причин значений признаков и причин заболеваний // Научно-техническая информация. Сер. 2. – 2006. - № 2. – С. 19-30.

[Клещев и др. 2012] Клещев А.С., Смагин С.В. Задачи индуктивного формирования знаний для онтологии медицинской диагностики // Научно-техническая информация. Сер.2. - М.: ВИНТИ РАН, 2012. №1. С.9-21.

[Клещев и др., 2013] Клещев А.С., Москаленко Ф.М. Доказательство корректности алгоритма решения частной задачи медицинской диагностики // Информатика и системы управления. 2013. №2(36). С. 134-146.

WHAT SHOULD BE THE PARADIGM OF TASK SOLVING FOR THE KNOWLEDGE-BASED SYSTEMS?

Gribova V, Kleschev A.

*Institute of Automation and Control Processes,
Far Eastern Branch of Russian Academy of
Sciences, Vladivostok
gribova@iacp.dvo.ru,
kleschev@iacp.dvo.ru*

A paradigm for tasks solving based on knowledge bases is proposed by analogy with the paradigm of solving tasks within the algorithmic approach and modern requirements for the development of intelligent systems. Adequacy of the existing technologies to the new paradigm is analyzed.

Introduction

The algorithmic approach of tasks solving is most widely used. This approach has the generally accepted paradigm for tasks solving. However, not all problems can be solved using the algorithmic approach so other approaches are developing. One of such approaches is approach for tasks solving based on knowledge bases. However, this approach still does not have the paradigm similar to that for the algorithmic approach.

The paper proposes a paradigm for tasks solving based on knowledge bases and existing technologies are compared using this paradigm.

Main Part

The main includes three paragraphs. The first one describes the paradigm of task solving within the algorithmic approach. The second part is devoted to a new paradigm for tasks solving based on knowledge bases. In the third part existing technologies are analyzed in accordance with the new paradigm.

Task solving on a computer is usually associated with the need to obtain any information. Formally, this need for information that is obtained by task solving, is described in terms of the definition of the set Y of all possible results of solving this task. To obtain information $y \in Y$, we need input data x . The set of all possible input data of the task is X . A formal statement of the task or its specification is a predicate $P(x, y)$, where $x \in X$, and $y \in Y$. For the task specification a statement of existence of a solution must be proven. It is also desirable to prove a statement about uniqueness of its solution. If both these statements are true, an algorithm for task solving is developed.

The paradigm of tasks solving based on knowledge bases has the same sets X and Y . Moreover, intelligent systems have knowledge base, so in the same way a set of admissible knowledge bases $K(X, Y)$ should be determined, and the specification of the task - the predicate $P(x, k, y)$. A form of representation of information in the knowledge base must be understandable for domain experts and users of intelligent systems without any intermediaries or special training. In the case of intelligent systems feature the task specification is that it is not a fair statement about the existence of solutions. Although the specifications of the task a statement of the existence of solutions is not fair, for it is possible to formulate a weaker statement, and in some cases a weak statement about the uniqueness of a solution. These statements are only predictions and can not be proven.

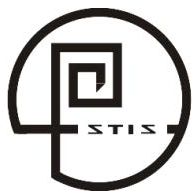
In assuming of validity for the task specification this weaker statement about existence of solutions the programmer can develop an algorithm for the task solving. If an intelligent system with a knowledge base implements the correct algorithm? Then the user is responsible for the correctness of input data, but in this case he can get wrong and inaccurate result of the task solving, or do not even get any result.

If we want the technology of intelligent systems to be viable, it must include procedures to reduce the proportion of cases in which the system has undesirable properties. These procedures may be different for different types of intelligent systems.

In the third section of the paper we discuss the adequacy of modern technologies for intelligent system development to the paradigm. Considered the following technologies: (a) a technology based on the use of expert system shells; (b) a technology based on the use of WebProtege; a technology based on the use of the platform IACPaaS. Technologies (a) and (b) does not support the specification of sets of input and output data, the knowledge base so that the predicates of accessories were computable. The technology (c) supports this specification. Form of knowledge representation is understandable for the domain experts in technologies (b) and (c). Technologies (a) and (b) not provide specifications of the task in the form of a predicate. Methods of control of the rule base have been developed for the (a) technology. This problem has not considered for the (b) technology. The technology (c) provides the apparatus for the task specification in the form of a predicate, using which it is possible to formulate a weak assertion of the existence of solutions, as well as weaker statement about the correctness of the algorithm for the task solving. Tools for the (a) technology includes tools for creation of a rule base and a problem-independent task solver (shell), the development of an interface is not supported. In addition, these tools and intelligence systems are not cloudy. Tool for the (b) technology includes an editor of knowledge bases, a set task solvers of some. The editor is cloudy. Tools for the (c) technology includes an editor of knowledge bases, tools for development of task solvers in the form of a set of agents and user interfaces. All of these tools are cloudy.

Conclusion

There is the urgent need to develop technologies and tools to support the process of development of intelligent systems. The general technology based on the algorithmic approach and the corresponding development tools do not applicable for intelligent system development. This is primarily due to the fact that the architecture of intelligent systems has an additional component - the knowledge base, which should be developed and maintained by domain experts and be understandable for them. At the same time a paradigm of task solving have not developed for intelligent systems, so all of the technologies are intuitively developing. This leads to the fact that it is impossible to objectively evaluate them and define ways of further development of this field of research. Proposed in the paper paradigm may become, if it is accepted by the community concerned, the basis for the evaluation of technologies and the further progress of technologies and tools for development of intelligent systems.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.96

ПРОЕКТИРОВАНИЕ ИНЖЕНЕРНЫХ РАСЧЕТОВ ДЛЯ ПОРТАЛОВ ЗНАНИЙ

Глоба Л.С.^{*}, Новогрудская Р.Л.^{*}

^{} Национальный технический университет Украины «Киевский политехнический институт»,
г. Киев, Украина*

lgloba@its.kpi.ua

rinan@ukr.net

В работе представлен подход к проектированию расчетных задач на инженерных порталах знаний, основанный на алгебре расчетов, которая позволяет получить требуемый общий расчет как результат выполнения определенной системы алгебраических формализмов. Алгебра расчетов описывает последовательность выполнения частных расчетов, при этом формализмы алгебры расчетов портала знаний представляются множеством операций заданного вида на хранимом наборе информационных и вычислительных ресурсов портала.

Ключевые слова: портал знаний, алгебра, расчетная задача.

Введение

В настоящее время наблюдается тенденция к систематизации и структуризации данных различных предметных областей, результатом которой является создание различных информационных систем, систем автоматизации либо управления производственной деятельностью, а также, порталов знаний, либо знаний предприятий (knowledge management portals порталов). Основной задачей таких систем является реализация методов поиска и связности элементов этих систем, направленная на оптимизацию времени доступа к информационным ресурсам. Такая задача влечет за собой ряд проблем:

- построение концептуальной модели системы,
- разработка формальной логики, описывающей мета-структуру элементов системы,
- построение метода поиска по информационному пространству системы

На сегодняшний день существует большое количество научных и исследовательских институтов, организаций, университетов, которые накапливают и хранят большие объемы технической и научной информации. Естественной потребностью для конечного пользователя является создание единой точки доступа к разнородным территориально-разнесенным информационным и вычислительным ресурсам. Важное требование в таком случае - возможность конечному

пользователю получить именно необходимую ему информацию за достаточно короткое время, невзирая на место ее расположения. Среди множества средств, предоставляющих такой доступ можно выделить электронные архивы, библиотеки, онлайн-системы, сайты и порталы. Наиболее эффективным средством из всех вышеперечисленных являются порталы знаний, поскольку они позволяют не только получить доступ к информационным ресурсам, хранящимся в их базах, но и, используя данные, знания и сервисы, реализовать определенные вычислительные задачи предметной области.

1. Введение в проблему

Эффективность использования порталов знаний следует из удобства пользования информацией, сосредоточенной в их среде. Это означает с технической стороны, что портал знаний должен обеспечивать интеграцию и совместную работу сервисов, баз данных и знаний, информационных хранилищ и вычислительных ресурсов. Следовательно, основной проблемой при их проектировании является построение корректной концептуальной модели и модели представления знаний [Kühn, 1998].

В зависимости от предметной области, который посвящен портал, необходимо определить присутствие в его среде информационных и вычислительных ресурсов. Если проектирование направлено на построение порталов научных знаний, тогда разработка его концептуальной

модели не составит труда, а в качестве модели представления знаний можно выбрать семантическую либо онтологическую модель [Боровикова, 2002], [Гаврилова, 2000]. Если же результат проектирования – инженерный портал, то возникает ряд дополнительных проблем: определение механизма связывания информационных ресурсов с вычислительными, описание характеристик вычислительных и информационных ресурсов в зависимости от рассматриваемой предметной области, определение результата выполнения вычислительного ресурса, описание последовательности взаимодействия вычислительных ресурсов для реализации более глобальных вычислений. Из этих проблем следует необходимость решения следующих задач:

- построение логической модели портала
- построение метода вывода в терминах логики портала
- разработка технологии проектирования портала

Таким образом, необходимо описать не только доступные пользователю на портале расчетные задачи, такие как сервисы вычислений, но и установить связи между конкретными сервисами (набором сервисов) для реализации конкретных рабочих процессов расчетных задач в рамках общих рабочих процессов портала, а также, установить связи сервисов с информационными ресурсами, что необходимо для решения конкретных производственных задач пользователя [Глоба, 2011].

В зависимости от параметров заданных пользователем и тематикой расчетной задачи она может быть выполнена с помощью различных алгоритмов. При построении реальных научных и инженерных задач используется их декомпозиция на подзадачи, которые в зависимости от направления общей задачи и параметров, которые используются при расчете, могут компоноваться в общий расчет динамически, причем одна из подзадач может использоваться в нескольких расчетах. Исходя из всего вышесказанного, возникает задача разработать и описать алгебру связности расчетных задач.

Для примера рассмотрим подход к проектированию специализированного интернет портала знаний в области Прочности Материалов.

2. Характеристика особенностей предметной области

Как уже было сказано, важное место в построении портала знаний занимает интеграция расчетных задач, которых существует огромное количество и которые занимают важное место при проведении различного рода исследований в предметной области. Существует необходимость осуществить корректное внедрение этих задач в информационную среду портала [Глоба, 2012].

Каждый расчет в предметной области Прочность Материалов состоит из множества подрасчетов. Результатом выполнения полного расчета является полная характеристика материала и вывод о возможности его использования при проектировании конкретной детали, как одной из составных частей сложной конструкции. Полный расчет обязательно состоит из двух расчетов: Выбор основных параметров и Поверочный расчет, которые, в свою очередь, состоят из нескольких групп расчетов. На этапе выбора основных параметров рассчитываются значения основных характеристик и параметров конструкции, а на этапе поверочного расчета - проверяется соответствуют ли определенные характеристики материала требованиям к соответствующим характеристикам конструкции.

Таким образом, расчеты, которые входят в группу расчета основных параметров, направлены, в основном, на расчет по заданным заранее общим правилам или формулам. Однако основной особенностью предметной области Прочность Материалов, и основным отличием ее расчетов от расчетных задач любой другой предметной области, является то, что под каждым конкретным подрасчетом группы поверочного расчета, понимается набор условий, которым должен соответствовать проверяемый материал. Такой набор условий установлен для того, чтобы можно было определить возможность использования материала для создания конструкции.

Из вышесказанного следует ряд специфических задач, связанных с разработкой расчетных задач портала «Прочность материалов». Во-первых, необходимо учесть логику связывания подрасчетов в общие расчеты, поскольку нет необходимости хранить одинаковые шаблоны для одних и тех же подрасчетов, которые могут использоваться в разных общих расчетах. Во-вторых, необходимо выделить специфические объекты предметной области, которые станут базовыми элементами связности расчетов, а также выделить на уровне логики основные операции, использующиеся в расчетных задачах портала. В-третьих, нужно разработать метаописания, характеризующие расчетные задачи портала знаний «Прочность Материалов».

В среде портала знаний можно выделить два типа элементов:

- Функциональные элементы – являются динамически изменяемыми элементами, и содержат последовательность действий внутри себя. К ним относятся расчетные задачи портала знаний, бизнес-процессы, исполняемые на портале и другие активные вычислительные компоненты.
- Информационные элементы – являются статическими элементами и не несет ни какой функциональной нагрузки. Они сами по себе не являются процессом взаимосвязанных вычислительных работ.

Множество метаописаний функциональных

элементов портала знаний предложено разбить на три подмножества:

- Стандартные метаописания – определяют\строятся согласно стандарту дублинского ядра, и описывают основные характеристики функциональных элементов.
- Специфические метаописания – метаописания связанные с предметной областью.
- Метаописания, устанавливающие связь, – используются для определения связи элемента с другими элементами, информационными ресурсами, расчетными задачами (функциональными элементами).

Выделим следующие метаописания, описывающие различные характеристики расчетных задач.

Стандартные метаописания

$$A_{st} = \{A_1, A_2, A_3, A_4, A_5\}:$$

Title (название, имя ресурса) – A_1 (атрибут)

Название атрибута	Описание атрибута
Title	Имя, данное ресурсу. Данный элемент, как правило, содержит формальное имя, под которым данный ресурс известен.

Identifier (идентификатор) – A_2

Название атрибута	Описание атрибута
Identifier	Уникальный идентификатор, рассматривается также как номер функционального элемента

Adress (адресс) – A_3

Название атрибута	Описание атрибута
Adress	Ссылка на место расположения ресурса

Subject (Предметная область) – A_3

Название атрибута	Описание атрибута
Subject	Узкая предметная область применения данного функционального элемента

Description (описание) – A_4

Название атрибута	Описание атрибута
Description	Текстовое описание функционального элемента

Data (данные) – A_5

Название	Описание атрибута
----------	-------------------

атрибута	
Data	Тип данных, которые содержит функциональный элемент

Специфические для предметной области метаописания

$$A_{st} = \{A_6, A_7, A_8, A_9, A_{10}\}$$

Parameters (параметры) – A_6

Название атрибута	Описание атрибута
Parameter	Параметры, которые употребляются при расчете

Loading (нагрузки) – A_7

Название атрибута	Описание атрибута
Loading	Нагрузки, о которых упоминается в расчете

Methods (методы) – A_8

Название атрибута	Описание атрибута
Methods	Методы расчетов, которые встречаются в расчете

Element (элемент) – A_9

Название атрибута	Описание атрибута
Element	Элемент, связанный с расчетом, или для которого проводится расчет

Measurement area (площади измерений) – A_{10}

Название атрибута	Описание атрибута
Measurement area	Площади конструкций, для которых актуальны приводимые в расчете формулы, либо для которых проводится расчет, либо которые встречаются в расчете

Метаописания, устанавливающие связь

$$A_{st} = \{A_{11}, A_{12}, A_{13}, A_{14}\}$$

Regulatory (нормативный документ, в котором описан расчет) – A_{11}

Название атрибута	Описание атрибута
Regulatory	Название нормативного документа (ГОСТ, ISO, Нормы расчета), который связан с функциональным элементом

Computation (ссылочный расчет) – A_{12}

Название атрибута	Описание атрибута
Computation	Расчет (функциональный элемент), с которым связан данный расчет

General computation (общий расчет) – A₁₃

Название атрибута	Описание атрибута
<i>General computation</i>	Общий функциональный элемент (если возможно его идентифицировать)

Material (материал)- A₁₄

Название атрибута	Описание атрибута
Material	Материал, который встречается в расчете

3. Алгебра расчетов

Приведем алгебру расчетов, которая позволяет получить требуемый общий расчет в результате выполнения определенной системы алгебраических формализмов, позволяющей задать последовательность выполнения частных расчетов. Формализмы алгебры расчетов для портала знаний представляются множеством операций заданного вида на хранимом наборе информационных и вычислительных ресурсов портала [Теленик, 1999] [Чень, 1983].

Элементы алгебры расчетов

Для возможности совместного использования различных расчетных задач, представленных на портале, и связанных с ними информационных ресурсов, упрощения процесса связывания расчетов в общий расчет рассмотрим элементы алгебры расчетов. Алгебра расчетов представляется формализмами, описывающими каждый расчет предметной области, и состоит из определения расчетов, операций над расчетами, соответствующих аксиом и теорем, позволяющих описать процесс обслуживания расчетов информационными ресурсами.

Пусть метаописание расчетных задач портала «Прочность материалов» задается как $A = \langle N, M \rangle$, где $N = \{N_1 N_2, \dots, N_n\}$ – множество имен атрибутов, $M = \{M_1 M, \dots, M_n\}$ – множества значений атрибутов и, таким образом, m_i^j – j значение из i-ого множества значений атрибутов. Под набором расчетов, которые входят в общий расчет, будем понимать отношение на множестве значений некоторого подмножества атрибутов расчета. Тогда схеме отношений R_k можно записать в виде $R_k = (R_1, R_2, \dots, R_n)$, где множество можно разделить на отношения, заданные на трех выделенных множествах расчетов в зависимости от выделенных типов метаописаний (стандартные метаописания, специфические для предметной

области метаописания, метаописания устанавливающие связь) [Теленик, 1999].

Условие расчета представляет собой любую логическую формулу, в которой:

Переменные – имена атрибутов $N_1 N_2, \dots, N_n$,

Константы – элементы соответствующих множеств $M_1 M, \dots, M_n$,

Предикатные символы – символы отношений $<> \leq \geq = \neq$,

Символы логических связок – символы логических операций $\wedge, \vee, -$.

Расчетом C_k назовем систему $\langle R_k A_k \rangle$, где R_k – отношение со схемой $R_k = (R_1, R_2, \dots, R_n)$, A_k – логическая формула условия.

Аксиомы алгебры расчетов

Опишем правила формирования расчета:

Аксиома 1. В набор расчетов для формирования общего расчета могут входить те, и только те расчеты, атрибуты которых удовлетворяют заданному условию.

$C_i = \{C_1, C_2, \dots, C_n \in C_i \mid \forall d_i \in A_i(M_i), \forall d_j \in A_j(M_j): d_i = d_j\}$
Выполнение условий связано с истинным значением логической формулы, определяющей заданное условие, при подстановке в нее значений метаописаний из множества расчетов, т.е. условие для расчета играет роль семантической связки или проверки, которая позволяет добавить в общий расчет только те расчеты, для которых это условие истинно.

Также, в алгебре расчетов можно выделить два типа расчетов:

- пустой расчет – $C_k = \langle R_k A_k \rangle$ в котором условие A_k принимает ложное значение.
- полный расчет – $C_k = \langle R_k A_k \rangle$ в котором условие A_k является конъюнкцией всех возможных условий базы.

Аксиома 2. Произвольный расчет портала знаний, заданный по схеме R_k – это расчет типа $C_k = \langle R_k A_k \rangle$.

Множество условий A_k представим как совокупность подмножеств условий – R_k и формул, которыми эти условия задаются – Q_k .

Условие формируется по сложному правилу, особенно когда оно играет роль механизма выбора для включения расчета в общий расчет.

Аксиома 3. Логическая формула условия и операнд условия:

- Если A имя метаописания, то A операнд условия.
- Если A значение метаописания, то A операнд условия.

- Если A и B – операнды условия, то каждое из выражений (A+B), (A-B), (A*B), (A:B), (-A) – операнды условий.
- Операнд не может быть получен никак кроме описанных выше способов.

В результате сказанного условие P_k можно определить следующим образом:

- Т-условие P_k .
- если A и B операнды условия P_k , то $A \Omega B$, где $\Omega = \{>, <, \leq, \geq, =, \neq\}$ – условие P_k .
- если A и B условия P_k , то (A+B), (A-B), (A*B), (A:B), (-A) – условия.

Рассмотрим выполнение логического правила: для того, чтобы объединить расчеты в общий, необходимо выполнение одного из следующих условий:

Все рассматриваемые расчеты объединяются в общий расчет.

Некоторые из рассматриваемых расчетов (множеств расчетов) не объединяются в общий расчет, остальные объединяются.

В общий расчет или объединяется одно множество расчетов, а другое нет, или же наоборот.

Других условий не существует.

Таким образом, условия выполнения расчетов определяется операциями алгебры логики: $\wedge, \vee, -$.

Необходимо доказать возможность применения теории алгебры логики для формирования условий инженерных расчетов на базе описанных выше операций [Takeuti, 1987], [Plaisted, 1981]..

Теорема 1. Пусть $C_1 = \langle R_1 A_1 \rangle$ и $C_2 = \langle R_2 A_2 \rangle$ – расчеты и $R_1 = R_2$. Расчет $C_3 = \langle R_3 A_3 \rangle$ тогда и только тогда является объединением

$C_1 \cup C_2$ расчетов C_1 и C_2 , когда $A_3 = A_1 \vee A_2$.

Доказательство. Построим логическую таблицу действий над кортежами расчетов C_1 и C_2 , по определению операции объединения. Таблица 1 строится для всех кортежей отношения, заданных схемой $R_1 = R_2$. Принцип заполнения логической таблицы основан на определении условия расчета и определении соответствующей операции над расчетом.

Таблица 1

Наличие в расчете	Условие расчета		
	C_1	C_2	C_3
В C_1 и C_2	Истина	Истина	Истина
Только в C_1	Истина	Ложь	Истина
Только в C_2	Ложь	Истина	Истина
Ни в C_1 , ни в C_2	Ложь	Ложь	Ложь

Рассмотрим принцип формирования первой строки таблицы 1. Поскольку кортеж содержится и в расчете C_1 и в расчете C_2 по определению, то условия расчета и условия для этого кортежа расчетов C_1 и C_2 примут соответственно истинные значения. По определению операции объединения расчетов, такой кортеж должен быть включен в расчет C_3 , что соответствует истинному значению условия C_3 . Аналогично получены остальные строки таблицы. Сопоставляя условия расчетов C_1 , C_2 и C_3 для всех возможных значений кортежей, можно сделать вывод, что условие A_3 определяется таблицей истинности, совпадающей для таблицы истинности, определяющей операцию «или» в алгебре логики

Теорема 2. Пусть $C_1 = \langle R_1 A_1 \rangle$ и $C_2 = \langle R_2 A_2 \rangle$ – расчеты и $R_1 = R_2$. Расчет $C_4 = \langle R_4 A_4 \rangle$ тогда и только тогда является пересечением

$C_1 \cup C_2$ расчетов C_1 и C_2 , когда $A_3 = A_1 \wedge A_2$.

Теорема 3. Пусть $C_1 = \langle R_1 A_1 \rangle$ и $C_1^* = \langle R_1, \forall \rangle$, где C_1^* содержит все кортежи схемы R_1 . Расчет

$C_3 = \langle R_3 A_3 \rangle$, является дополнением расчета C_1 к C_1^* тогда и только тогда, когда $A_3 = \neg A_1$

Аналогично приведённому доказательству теоремы, доказывается возможность использования операций конъюнкции («и») и отрицания («не»): для построения результирующего расчета.

Заключение

Представлен подход к проектированию последовательности выполнения расчетных задач на инженерных порталах знаний при динамическом формировании сложных инженерных расчетов. Предложена алгебра расчетов, которая позволяет получить требуемый общий расчет в результате выполнения определенной системы алгебраических формализмов и динамически сформировать последовательность выполнения частных расчетов. Алгебра формализмов для расчетов портала знаний представляется множеством операций заданного вида на хранимом наборе информационных и вычислительных ресурсов портала.

Библиографический список

- Порталы знаний предприятия [Электронный ресурс]. – Режим доступа: <http://www.hr-ua.com/2009/12/portaly-znaniy-predpriyatij/>
- [Kühn, 1998] Kühn O., Abecker A. Corporate Memories for Knowledge Management in Industrial Practice: Prospects and Challenges. 1998.
- [Боровикова, 2002] Боровикова О. И., Загорулько Ю. А. Организация порталов знаний на основе онтологий // Тр. междунар. семинара «Диалог 2002» Компьютерная лингвистика и интеллектуальные технологии (Протвино, 6–11 июня 2002 г.). М.: Наука, 2002. Т. 2. С. 76–82.
- [Гаврилова, 2000] Гаврилова Т. А., Хорошевский В. Ф. Базы знаний интеллектуальных систем. Учебник. — СПб.: Питер, 2000
- [Глоба, 2011] Глоба Л.С., Новогрудская Р.Л. Систематизация информационных ресурсов Интернет-портала «Прочность материалов», Вісник Харк. нац. ун-ту імені В. Н. Каразіна, серія

"Математичне моделювання. Інформаційні технології. Автоматизовані системи управління". - Вип. 16, № 927, 2011.

[Глоба, 2012] Глоба Л.С., Новогрудская Р.Л. Модели и методы интеграции информационных и вычислительных ресурсов // Труды второй международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» OSTIS-2012. Минск, 2012, С.

[Теленик, 1999] Теленик С.Ф., Логика представления вычислительных процессов в интеллектуальной системе SmartBase // Системные технологии. Системное моделирование технологических процессов: Сб. науч. Тр. – 1999. – Вып. 6. – С.131-139.

[Чень, 1983] Чень Ч., Ли Р., Математическая логика и автоматическое доказательство теорем. – М.: Наука, 1983. – 360с.

[Теленик, 1999] Теленик С.Ф., Логика организации взаимодействия в адаптивной технологии // Системные технологии. Системы и процессы обработки информации и управление: Сб. науч. Тр. – 1999. – Вып. 7. – С. 119-129.

[Такеути, 1987] Такеути Г. Теория доказательств. – М.: Мир, 1978. – 412с.

[Plaisted, 1981] Plaisted D.A., Theorem proving with abstraction // Artificial intelligence. - 1981. - Vol. 16. – P. 47-108.

THE DEVELOPMENT OF ENGINEERING CALCULATION FOR THE KNOWLEDGE PORTALS

Globa L.S., Novogrudskaya R.L.

*National thechnical university of ukraine «Kyiv
Politechnick Institute» , Kyiv, Ukraine*

lgloba@its.kpi.ua

rinan@ukr.net

The paper presents an approach to the development of computational tasks of the engineering knowledge portals. Algebra of calculations is proposed, that allows to obtain the required total calculation task as a result of a certain system of algebraic formalism describing the sequence of execution of private calculations. Algebra formalisms for calculation tasks of knowledge portal is represented by a set of operations of a given form to a stored set of information and computing resources of the portal.

Introduction

Currently, there is a tendency to systematization and structuring of data from different domains, that results in the development of various information systems, automation systems , or performance management systems, as well as knowledge portals, or knowledge management enterprise portals . The main objective of such systems is the implementation of search methods and connectivity elements. Such systems aimed at optimization of time access to information resources.

Such a task entails a number of problems that are based on the development of:

- a conceptual model of the system,
- the formal logic, describing meta-structure elements of the system,
- construction of the search method on information space of system.

Main Part

Efficiency of the use of knowledge portals should be focused to systematization of information in their environment. It means technically that knowledge portal should provide integration and collaboration of services, databases and knowledge, information storage and computing resources. Therefore , the main problem in their design is to develop a conceptual model and the correct model of knowledge representation [Kühn, 1998].

Based on these problems, the need to solve the following tasks: designing of a logical model of the portal, development of the method of search in terms of the logic of the portal, development of the technology for knowledge portal designing.

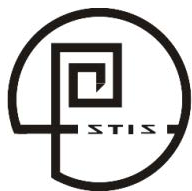
It is proposed to split metadescriptions set of functional elements of the knowledge portal into three subsets.

Based on described problems the algebra of calculation tasks is proposed, that allows to obtain the required general computational task as a result of a certain system of algebraic formalisms, that allow you to specify the sequence of execution of separate computational tasks. The formalisms of computation algebra for the knowledge portal represents a set of operations of a given form to a stored set of information and computing resources of knowledge portal.

To be able to share various computational tasks presented on the portal, and related information resources, simplifying the process of computations binding consider the overall calculation of the elements of calculation. Calculations algebra is seemed as formalism describing each calculation domain, and consists of determining the calculation operations, calculations relevant axioms and theorems.

Conclusion

An approach to the designing of computational tasks in engineering knowledge portals is presented. Proposed algebra of calculations, that allows to obtain the required general computational task as a result of a certain system of algebraic formalism describing the sequence of execution of separate computational tasks. Algebra of formalisms for computational tasks of the knowledge portal is represented by a set of operations of a given form to a stored set of information and computational resources of the portal.



УДК 004.8

ОДИН ЯЗЫК ДЛЯ СПЕЦИФИКАЦИИ ОНТОЛОГИЙ

Плесневич Г.С.

Московский энергетический институт (Национальный исследовательский университет)

г. Москва, Россия

salve777@mail.ru

Описывается некоторый язык для спецификации онтологий, входящий в систему «Бинарная Модель Знаний», которая предназначена для спецификации и интерпретации онтологий. Определен синтаксис, денотативная и операционная семантика этого языка. Рассматривается записанная в этом языке онтология, представляющая известную задачу Steamroller. Показано, как можно решить эту задачу, используя продукционный вывод.

Ключевые слова: искусственный интеллект, представление знаний, онтологии.

Введение

В работе [Плесневич, 2013] мы кратко описали систему «Бинарная Модель Знаний» (БМЗ) языков, предназначенных для спецификации онтологий. Представленные в БМЗ онтологии записываются с помощью языков структурной и логической спецификации понятий. Эти языки ориентированы на представление понятий.

Формальное понятие имеет имя C , универсум U^C и экстенционал Ext^C . Универсум U^C – это множество всех имен, которые могут обозначать примеры (экземпляры) понятия C . Экстенционал Ext^C есть пара (E^C, \sim^C) , где E^C – множество всех имен из U^C , которые обозначают примеры понятия (в данном контексте), а \sim^C – отношение эквивалентности на множестве E^C , определяющее *корреферентность* имен. (Два имени a и b корреферентны, если они обозначают один и тот же объект предметной области.)

Мы предполагаем, что универсум U^C любого понятия C включает множество Sur так называемых *суррогатов* #1, #2, #3, ... – системных идентификаторов для объектов моделируемых предметных областей. Мы также предполагаем, что для любого $x \in E^C$ существует единственный суррогат, корреферентный E^C .

В системе БМЗ имеется язык ЯСС структурной спецификации объектов, при помощи которого определяются универсумы понятий. Эти понятия в БМЗ являются либо *классами*, либо *бинарными связями*. Последние устанавливают отношения между объектами двух классов. компонентами которых служат выражения видов суррогаты.

Для спецификации экстенсионалов понятий моделируемой предметной области применяются логические предложения. Эти предложения используют термины тех структурных предложений языка ЯСС, которые определяют универсумы этих понятий.

Онтология – это (конечное) множество O структурных и логических предложений. Таким образом, онтология O делится на две части $O = O_{стр} \cup O_{лог}$, где $O_{стр}$ состоит из структурных предложений, а $O_{лог}$ – из логических предложений.

В системе БМЗ языки логической спецификации составляют иерархию по своей выразительности. В настоящей работе мы опишем некоторый язык логической спецификации, названный ЯЛС-0, который является в некотором смысле минимальным в этой иерархии.

1. Задача Steamroller

Сначала запишем в языке ЯЛС-0 и в языке ЯСС известную задачу Steamroller (паровой каток). Эта задача была сформулирована в 1978 году Л. Шубертом (L. Schubert) и была им предложена в качестве тестовой задачи для программ логического вывода, основанных на методе резолюции (см., например, [Walther 2003]).

Задача Steamroller состоит в доказательстве, что из следующих утверждений (а) – (л) логически следует утверждение (м).

(а) Волки, лисы, птицы, гусеницы и улитки – животные.

(б) Существуют волки, лисы, птицы, гусеницы и улитки.

(в) Злаки – растения.

(г) Злаки существуют.

(д) Каждое животное ест некоторые растения или ест всех животных, которые его меньше и которые едят некоторые растения.

(е) Гусеницы меньше птиц.

(ж) Птицы меньше лис.

(з) Лисы меньше волков.

(и) Волки не едят гусениц и не едят растений.

(к) Птицы едят гусениц, но не едят улиток.

(л) Гусеницы и улитки едят некоторые растения.

(м) Существует животное, которое ест некоторых животных, питающихся злаками.

Знание, содержащиеся в утверждениях (а) – (л), представим в следующей онтологии **O-St**, записанной в языке ЯСС U ЯЛС-0.

O-St = {

1. (Животное Менее Животное)
2. (Животное Ест Животное)
3. (Животное Ест Растение)
4. Волк, Лиса, Птица, Гусеница, Улитка ISA Животное
5. EXIST Волк, Лиса, Птица, Гусеница, Улитка
6. Злак ISA Растение
7. EXIST Злак
8. EACH Животное: X (ЕСТ SOME Растение OR Ест EACH Животное THAT (Менее X) AND Ест SOME Растение)
9. EACH Гусеница Менее EACH Птица
10. EACH Птица Менее EACH Лиса
11. EACH Лиса Менее EACH Волк
12. EACH Волк -Ест EACH Гусеница, Растение
13. EACH Птица (Ест EACH Гусеница AND - Ест EACH Улитка)
14. EACH Гусеница, Улитка Ест SOME Растение}

Мы видим, что предложения 4 – 14 в точности отвечают утверждениям (а) – (л). Утверждение (м) в языке ЯЛС-0 записывается так:

15. EXIST Животное THAT Ест SOME Животное THAT Ест EACH Злак.

В онтологии **O-St** предложения 1, 2 и 3 принадлежат языку ЯСС. Они задают структуру бинарных связей Менее и Ест. Остальные понятия Животное, Растение, Волк, Лиса и т.д. не имеют структуры.

Решение задачи Steamroller состоит в установлении, что из онтологии **O-St** логически следует предложение 15.

2. Синтаксис и семантика языка ЯЛС-0

Предложения языка ЯЛС-0 строятся из термов трех видов: С-термы, L-термы и Р-термы.

С-термы обозначают классы объектов. Они интерпретируются как подмножества множества Surr суррогатов. Примеры С-термов, содержащихся в онтологии **O-St**: Животное, Животное THAT Ест SOME Растение.

L-термы обозначают бинарные связи. Они интерпретируются как подмножества множества (Surr, Surr) всех пар суррогатов. Примеры L-термов:

Менее, Ест, -Ест, Ест SOME Растение, Менее X AND Ест SOME Растение

Р-термы обозначают одноместные предикаты, заданные на множестве Surr суррогатов. Примеры Р-термов: Менее X, Менее EACH Волк.

При описании синтаксиса и семантики выражений языка ЯЛС-0 мы будем использовать переменные, имеющие следующие значения:

- c, d – индивидные константы (имена объектов);
- C, D, E – имена классов;
- L, M, N – имена бинарных связей;
- P, Q, R – имена предикатов;
- $Name$ – простые (атомарные) имена классов, бинарных связей или предикатов;
- S – предложение языка ЯЛС-0;
- V – имена X, X_1, X_2, \dots ;
- T – имена **true** и **false** (или **1** и **0**), обозначающие истину и ложь.

Через « Exp », где Exp – какое-либо выражение, обозначим семантику этого выражения, т.е. его значение в данной интерпретации в универсуме Surr.

СИНТАКСИС С-ТЕРМОВ:

$C, D ::= -C \mid C \text{ AND } D \mid C \text{ OR } D \mid$
 $C ::= D \text{ THAT } P \mid L \text{ SOME } D \mid L \text{ ONLY } D \mid$
 $D : V \mid Name$

СЕМАНТИКА С-ТЕРМОВ:

« $-C$ » = $Surr \setminus \langle C \rangle$;
« $C \text{ AND } D$ » = « C » \cap « D »;
« $C \text{ OR } D$ » = « C » \cup « D »;
« $L \text{ SOME } C$ » = $\{x \in Surr \mid (\exists y \in \langle C \rangle) (x, y) \in \langle L \rangle\}$;
« $L \text{ ONLY } C$ » =
 $\{x \in Surr \mid (\forall y \in Surr) [(x, y) \in \langle L \rangle \rightarrow x \in \langle C \rangle]\}$;
« $C \text{ THAT } P$ » = $\{x \in \langle C \rangle \mid \langle P \rangle(x)\}$;
« $D : V$ » = « D »;

« $Name$ » – множество всех строчек символов начинающихся с заглавных букв.

СИНТАКСИС L-ТЕРМОВ:

$L, M ::= -L \mid L \text{ AND } M \mid L \text{ OR } M \mid INV(L) \mid$
 $Name$

СЕМАНТИКА L-ТЕРМОВ:

« $-L$ » = $(Surr, Surr) \setminus \langle L \rangle$;
« $C \text{ AND } D$ » = « C » \cap « D »;
« $C \text{ OR } D$ » = « C » \cup « D »;
« $INV(L)$ » = $\{(y, x) \mid (x, y) \in \langle L \rangle\}$.

СИНТАКСИС Р-ТЕРМОВ:

$P, Q ::= -P \mid P \text{ AND } Q \mid P \text{ OR } Q \mid L \text{ SOME } C \mid$
 $L \text{ EACH } C \mid (L \ V) \mid (L \ c) \mid Name$

СЕМАНТИКА Р-ТЕРМОВ:

$x \langle -P \rangle \Leftrightarrow \neg (x \langle P \rangle)$;
 $x \langle P \text{ AND } Q \rangle \Leftrightarrow x \langle P \rangle \wedge x \langle Q \rangle$;
 $x \langle P \text{ OR } Q \rangle \Leftrightarrow x \langle P \rangle \vee x \langle Q \rangle$;
 $x \langle L \text{ SOME } C \rangle \Leftrightarrow (\exists y \in \langle C \rangle) (x, y) \in \langle L \rangle$;
 $x \langle L \text{ EACH } C \rangle \Leftrightarrow (\forall y \in Surr)$
 $[(x, y) \in \langle L \rangle \rightarrow x \in \langle C \rangle]$;
 $x \langle (L \ V) \rangle \Leftrightarrow (x, V) \in \langle L \rangle$;
 $x \langle (L \ c) \rangle \Leftrightarrow (x, c) \in \langle L \rangle$.

Замечания. 1) Выражение $L \text{ SOME } D$ является С-термом или Р-термом в зависимости от контекста (положения в предложении). 2) Вместо обычной префиксной записи « P »(x) для результата

применения предиката « P » к объектной переменной x мы используем постфиксную запись $x \llbracket P \rrbracket$. 3) Символ операции AND (операции OR) используется как для обозначения конъюнкции (дизъюнкции) предикатов, так и для обозначения пересечения (объединения) классов.

СИНТАКСИС ПРЕДЛОЖЕНИЙ:

$S ::= -S \mid T \mid \text{EXIST } C \mid \text{EXIST } L \mid$
 $\text{NULL } C \mid \text{NULL } L \mid C \text{ ISA } D \mid C = D \mid \text{EACH } C P \mid$
 $\text{SOME } C P; \neg S = S.$

СЕМАНТИКА ПРЕДЛОЖЕНИЙ:

$\llbracket -S \rrbracket \Leftrightarrow \neg \llbracket S \rrbracket$; $\llbracket \text{EXIST } C \rrbracket \Leftrightarrow C \neq \emptyset$;
 $\llbracket \text{NULL } C \rrbracket \Leftrightarrow C = \emptyset$; $\llbracket C \text{ ISA } D \rrbracket \Leftrightarrow \llbracket C \rrbracket \subseteq \llbracket D \rrbracket$; $\llbracket C = D \rrbracket \Leftrightarrow \llbracket C \rrbracket = \llbracket D \rrbracket$; $\llbracket \text{EACH } C P \rrbracket \Leftrightarrow$
 $(\forall x \in \llbracket C \rrbracket) \llbracket P \rrbracket(x)$; $\llbracket \text{SOME } C P \rrbracket \Leftrightarrow$
 $(\exists x \in \llbracket C \rrbracket) \llbracket P \rrbracket(x).$

Следующие выражения мы рассматриваем как паттерны для получения примитивных предложений:

$C = -D \mid L = -M \mid \text{EXIST } C \mid \text{EXIST } L \mid$
 $\text{NULL } C \mid \text{NULL } L \mid C \text{ ISA } D \mid L \text{ ISA } M \mid C = D \mid$
 $L = M \mid E = C \text{ AND } D \mid E = C \text{ OR } D \mid N = L \text{ AND } M \mid N =$
 $L \text{ OR } M \mid C = D \text{ THAT } P \mid C = L \text{ SOME } D \mid$
 $C = L \text{ ONLY } D \mid M = \text{INV}(L) \mid Q = -P \mid R = P \text{ AND } Q \mid R =$
 $P \text{ OR } Q \mid P = L \text{ SOME } C \mid P = L \text{ EACH } C \mid$
 $P = (L \vee).$

Конкретное *примитивное предложение* получается путем подстановки в паттерн имен из *Name*.

Любое предложение языка ЯЛС-0 можно преобразовать в множество примитивных предложений. Покажем это на примере.

Пример 2.1. Возьмем предложение 15, т.е.

EXIST Животное THAT Ест SOME
 Животное THAT Ест EACH Злак,

и положим:

$P1 = \text{Ест EACH Злак},$
 $C1 = \text{Животное THAT Ест EACH Злак},$
 $P2 = \text{Ест SOME Животное THAT Ест}$
 $\text{EACH Злак},$
 $C2 = \text{Животное THAT Ест SOME}$
 $\text{Животное THAT Ест EACH Злак}.$

Тогда предложение 15 можно заменить на множество следующих примитивных предложений:

- $O_1 = \{$
1. $P1 = \text{Ест EACH Злак}$
 2. $C1 = \text{Животное THAT } P1$
 3. $P2 = \text{Ест SOME } C1$
 4. $C2 = \text{Животное THAT } P2$
 5. $\text{EXIST } C2\}.$

Множество O_1 примитивных предложений можно рассматривать как эквивалентное расширение предложения 15 в следующем смысле: любая модель для предложения 15 (т.е. интерпретация, при которой это предложение истинно) продолжается до модели множества O_1 и наоборот, сужение модели для O_1 на понятия, входящие в 15, является моделью этого предложения.

3. Продукционный вывод

Операционную семантику языка ЯЛС-0 мы определим в терминах продукций, действующих на базах фактах данной онтологии.

Мы дадим абстрактный синтаксис для языка продукций. Факты, относящиеся к классу C , будем записывать в виде $+a(C)$ и $-a(C)$, а факты, относящиеся к бинарной связи L , записывать в виде $+(c,d)(L)$ и $-(c,d)(L)$. Компоненты продукций также будем записывать в виде $+X(C)$, $-X(C)$, $+(X,Y)(L)$ и $-(X,Y)(L)$, где X и Y – переменные, используемые для сопоставления этих компонент с фактами.

Продукции действуют только на базах фактов для онтологий, составленных из примитивных предложений. Поэтому сначала произвольную онтологию нужно преобразовать в состоящую из примитивных предложений.

Обозначим через $O\text{-St}^*$ онтологию, полученную преобразованием всех предложений из $O\text{-St}$ в примитивные предложения. (Онтология $O\text{-St}^*$ состоит 36 примитивных предложений.)

С каждым из 24 паттернов для примитивных предложений мы связываем набор продукций. Укажем некоторые из них.

1. Паттерн $C = -D$.

Продукции:

$+X(C) \Rightarrow -X(D)$
 $-X(C) \Rightarrow +X(D)$
 $+X(D) \Rightarrow -X(C)$
 $-X(D) \Rightarrow +X(C)$

Например, первая продукция действует так: в базе фактов для класса C разыскиваются факты вида $+a(C)$ и, если такой факт есть, то к базе фактов для D присоединяем факт $-a(C)$.

2. Паттерн $C \text{ ISA } D$.

Продукции:

$+X(C) \Rightarrow +X(D)$
 $-X(D) \Rightarrow -X(C)$

3. Паттерн $\text{EXIST } C$.

Продукции:

$\Rightarrow +cj(C)$
 $+X(C) \Rightarrow \text{EXIST } C$

Здесь cj – новая сколемовская константа. Таким образом, результатом применения первой продукции является присоединение факта $+cj(C)$ к базе фактов для класса C .

4. Паттерн $E = C \text{ AND } D$.

Продукции:

$+X(E) \Rightarrow +X(C); +X(D)$
 $-X(E); +X(C) \Rightarrow -X(D)$
 $-X(E); +X(D) \Rightarrow -X(C)$
 $-X(C) \Rightarrow -X(E)$
 $-X(D) \Rightarrow -X(E)$

Например, первая продукция действует так: в базе фактов для класса E разыскиваются факты вида $+a(E)$ и, если такой факт есть, то к базе фактов для C присоединяем факт $+a(C)$ и к базе фактов для D присоединяется факт $+a(D)$. Вторая продукция действует так: в базе фактов для E и в базе фактов для C разыскиваются соответственно факты вида

$-a(E)$ и $+a(C)$; если такие два факта найдены, то к базе фактов для D присоединяется факт $-a(D)$.

5. Паттерн $C = D$ THAT P .

Продукции:

$+X(C) \Rightarrow +X(D); +X(P)$
 $-X(C); +X(D) \Rightarrow -X(P)$
 $-X(C); +X(P) \Rightarrow -X(D)$
 $-X(D) \Rightarrow -X(C)$
 $-X(P) \Rightarrow -X(C)$

7. Паттерн $C = L$ SOME D .

Продукции:

$+(X, Y)(L) \Rightarrow +X(C); +Y(D)$
 $+X(C); +(X, Y)(L) \Rightarrow +Y(D)$
 $+X(C) \Rightarrow +(X, cj)(L); +cj(D)$
 $-X(C) \Rightarrow -(X, xj)(L)$
 $-X(D) \Rightarrow -(xj, X)(L)$

В указанных продукциях C, D, E, P, L, M являются параметрами со значениями в множестве *Name*. Поэтому это продукции общего вида. Конкретные продукции получают тогда, когда параметры получают значения

В БМЗ интерпретатор продукций реализует тактику параллельного их исполнения «снизу вверх».

Пример 3.1. Возьмем онтологию, состоящую из предложений онтологии **O-St*** и онтологии O_1 , но без предложения EXIST C2 Пусть Prod – множество конкретных продукций, ассоциированных с этой онтологией. Процесс применения Prod начинается с пустой базы фактов $B_0 = \emptyset$. Тогда к B_0 применимы только продукции из Prod с паттерном EXIST C. В результате их применения получается новая база фактов

$B_1 = \{+c_1(\text{Волк}), +c_2(\text{Лиса}), +c_3(\text{Птица}), +c_1(\text{Гусеница}), +c_1(\text{Улитка}), +c_1(\text{Злак})\}$. К базе фактов B_1 применимы продукции из Prod с паттерном $C \text{ ISA } D$.

Продолжая применять продукции из Prod, мы на некотором шаге получим базу фактов, содержащую факт $+c_3(C2)$. Применив к этому факту продукцию $+X(C2) \Rightarrow \text{EXIST } C2$, получим EXIST C2, т.е. истинно предложение 15.

Заключение

В настоящей статье мы определили некоторый язык ЯЛС-0 для логической спецификации онтологий. Этот язык является расширением базового языка ALC дескриптивной логики [Baader, 2003]. На самом деле C-термы языка ЯЛС-0 (обозначающие классы объектов) записываются в так называемом манчестерском синтаксисе, который был предложен в [Horridge et al. 2006] для основанного на дескриптивной логике языка OWL. Для логического вывода в онтологиях, записанных в языке ЯЛС-0, мы предложили метод, основанный на применении продукций.

Работа выполнена при финансовой поддержке Российского Фонда Фундаментальных Исследований по проекту №14-07-00387.

Библиографический список

[Baader et al. 2003] Baader F., Calvanese D., McGuinness D., Nardi D., Patel-Schneider P. (eds.) Description Logic Handbook. – Cambridge University Press, 2003.

[Horridge et al. 2006] Horridge M., Drummond D., Goodwin J., Rector A., Stevens R., Wang H. The Manchester OWL Syntax // OWL Experience and Directions Workshop, 2006.

[Walther 2003] Walther C. A mechanical solution of Schubert's Steamroller by many-sorted resolution // Artificial Intelligence, .26 (2), 1985.

[Плесневич 2013] Плесневич Г.С. . Формальные онтологии //20-я Международная научно-техническая конференция "Открытые семантические технологии проектирования интеллектуальных систем" OSTIS-2012 (16 - 18 февраля 2012, Минск), Минск, 2012.

A LANGUAGE FOR SPECIFYING ONTOLOGIES

Plesniewicz G.S.

Moscow Power Engineering University, Russia

salve777@mail.ru

Some language for logical specification of ontologies is presented. This language is an extension of the description logic language ALC.

Introduction

We present some language ЯЛС-0 for specifying ontologies. The language belongs to the system “Binary Model of Knowledge” of concept-based languages. (The system is under development at the Applied Mathematics Department of the Moscow Power Engineering Institute.) The syntax, denotative and operational semantics of the language ЯЛС-0 are defined. As example, we have written in ЯЛС-0 the ontology for the well-known problem “Steamroller”. It is shown how to solve the problem by applying productions associated with the ontology.

Main part

The syntax of the language ЯЛС-0 comprises the rules for defining the so-called C-terms, L-terms and P-terms. C-terms denote classes of individual objects of a given problem domain, L-terms denote binary relations between individual objects, and P-terms denote unary predicates. C-terms have the same semantics as the semantics of concept descriptions in the language ALC. Therefore, ЯЛС-0 is an extension of ALC. Syntax of ЯЛС-0 is closed to the so-called Manchester syntax. Statements of ЯЛС-0 have the form almost exactly corresponded to natural language statements.

With any ontology O it is associated some set $\text{Prod}(O)$ of productions. We show how $\text{Prod}(O)$ can be applied for obtaining logical consequences from O .

Conclusion

In the paper we present some language ЯЛС-0 for logical specifying ontologies. The language is an extension of the basic description logic language ALC.



УДК 004.822:514

ПЕРСОНАЛЬНАЯ ОНТОЛОГИЯ ПРОФЕССИОНАЛЬНОГО ОПЫТА

Соснин П.И.*

* *Ульяновский государственный технический университет,
г. Ульяновск, Российская Федерация*

sosnin@ulstu.ru

В работе приводятся результаты исследований вопросов онтологизации персонального профессионального опыта. Специфику решений по созданию и использование персональной онтологии определяет ориентация на представление опыта в виде системы моделей прецедентов, освоенных и созданных субъектом деятельности в реализации совокупности проектов. Модели прецедентов построены по образцу интеллектуально обработанных «деятельностных рефлексов» с использованием вопросно-ответных рассуждений.

Введение

Деятельность – это форма жизни, в основу которой природа положила интеллектуально обработанные условные рефлексы, формирующие то, что позволяет ввести в типовые взаимодействия человека с окружающей средой «вторую сигнальную систему» [Данилова и др. 1997], то есть естественный язык (ЕЯ). Приписывание знаков ЕЯ обстоятельствам, которые распознаются с помощью органов чувств в определённой окружающей обстановке, открывает возможность для абстрагирования от этой обстановки до знакового образца, позволяющего управлять соответствующим типовым поведением человека.

Такое типовое поведение принято называть «прецедентом», а знаковый образец, по которому оно осуществляется «моделью прецедента». То, что называют «опытом человека» включает громадное количество «прецедентов», взаимодействие с которыми (для развития, поиска, доступа, настройки, рационализации и других целей) невозможно без их систематизации.

Разумно различать неявную *естественную систематизацию*, за существование которой отвечают естественные механизмы опыта человека, *естественно-искусственную систематизацию*, ответственность за которую несёт освоенная человеком система понятий, и *искусственную систематизацию*, материализация и использование которой осуществляется с помощью искусственных сред, например, компьютеризованного типа.

Первые два вида систематизации имеют отношение только к человеку, а третий вид

допускает его применение не только человеком, а, например, и программными (интеллектуальными) агентами, в различных приложениях. Именно к третьему виду относятся искусственные средства систематизации, получившие название «онтологий». За созданием любой «онтологии» всегда стоят задачи, для решения которых используются взаимодействия с опытом и его моделями.

В статье предлагается подход к формированию персональных онтологий, обслуживающих взаимодействие конкретного индивида с доступным ему опытом. Подход материализован инструментально в версии OwnWIQA вопросно-ответной моделирующей среды WIQA (Working In Questions and Answers) [Соснин и др., 2012].

1. Предварительные основы

Предлагаемый подход развивает наши исследования в проблемной области «Формирование и использование прикладных онтологий», результаты которых раскрыты в публикациях [Sosnin, 2010], [Соснин и др., 2010].

Специфику этих исследований, ориентированных на прикладные онтологии в проектировании автоматизированных систем (АС), определяют следующие установки:

1. Онтология конкретной прикладной области должна строиться в процессах проектирования АС, относящихся к этой области.

2. Функции основного информационного источника в построениях прикладных онтологий должны выполнять рассуждения лиц, вовлечённых в

процессы проектирования и использования их результатов.

3. То, что включается в онтологию, должно прямо или опосредовано найти материальное воплощение в процессах проектирования и/или их продуктах.

4. Включение онтологии в состав средств, используемых в разработках АС и их применениях, должно конструктивно приносить ожидаемые позитивные эффекты.

Отметим, что среди различных типов рассуждений был выделен класс вопросно-ответных рассуждений (QA-рассуждений), осуществляемых в процессах решения проектных задач. Выбор этого класса был обусловлен диалоговой природой сознания, обеспечивающего доступ к наличному опыту.

Именно отмеченная специфика установок привела к созданию инструментально-технологического комплекса WIQA.Net, на базе которого разработан ряд его приложений, например, для концептуального проектирования АС, документирования и обучающего сопровождения. В концептуальном проектировании «онтология проекта» применялась в оперативном предикатно-онтологическом контроле проектных решений [Соснин и др., 2012].

Еще одним приложением, разработанным на основе WIQA.Net и использующем этот комплекс, является «База Опыта проектной организации», в состав которой включена и её онтология.

Так что реальные построения и применения прикладных онтологий привели к решению исследовать возможности построения и использования персональных онтологий. Представленные выше установки были сохранены, но в их приложениях не только к задачам

проектирования, которые по силам одному человеку, но и к освоению опыта решения профессиональных задач, с которыми сталкивается конкретный индивид. Этот класс задач в разработке персонального инструментария OwnWIQA было решено считать подобными задачам проектирования, которые можно решать в операционной среде OwnWIQA.

К установкам, модифицированным для построения OwnWIQA, была добавлена ещё одна:

5. Владелец персональной онтологии является и разработчиком и пользователем продуктов, при создании которых он применяет построенную (и освоенную) им онтологию.

На разработку инструментария OwnWIQA, а также методологического обеспечения его применений существенное воздействие оказали публикации в области эмпирической программной инженерии, в первую очередь, раскрывающие моделирование и использование профессионального опыта [Basili et al. 2001], вопросно-ответных механизмов доступа к базам опыта [Henninger 2003] и научного подхода к экспериментированию в человеко-компьютерных средах [Sjoberg et al. 2007].

2. Вопросно-ответная память

В соответствии с установками предлагаемого подхода персональная «онтология» предназначена для систематизации моделей прецедентов, которые освоены индивидом и используются им в решении задач $Z=\{Z_k\}$, относящихся к области его профессиональных интересов.

В спецификациях «онтологий» и её материализации принципиальное место занимает вопросно-ответная память (QA-память), обобщённо представленная на рисунке 1.

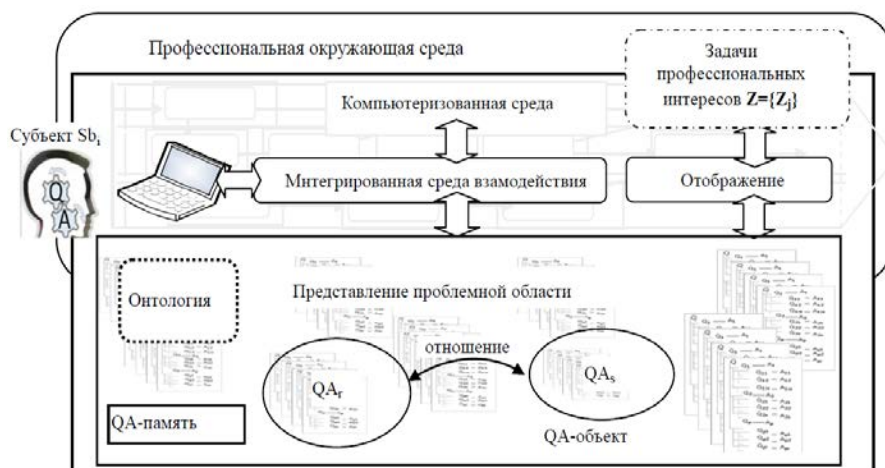


Рисунок 1 – Вопросно-ответная память инструментария OwnWIQA

QA-память – это подсистема инструментария OwnWIQA, предназначенная, в первую очередь, для моделирования QA-рассуждений, используемых индивидом (субъектом Sb_i) в процессах решения

задач. Конкретная QA-модель состоит из связанной совокупности моделей вопросов $\{Q_i\}$ и соответствующих им моделей ответов $\{A_i\}$, причём,

каждая из таких моделей загружена в определённую ячейку QA-памяти.

Каждая из ячеек QA-памяти создаётся оперативно для регистрации и хранения атрибутики очередной модели типа Q или A. Атрибутика включает базовый набор атрибутов $B(a_1, a_2, \dots, a_N)$, к которому владелец OwnWIQA (или короче I) может добавить полезные дополнительные атрибуты $AA(aa_1, aa_2, \dots, aa_M)$. Совокупность базовых атрибутов с операциями над ними образует интерактивный объект типа Q или A. Ответственность за операции над объектом, в которых используется дополнительная атрибутика, возлагается на субъекта Sb_i .

В число базовых атрибутов ячейки QA-памяти включены: уникальное имя модели (тип и индекс, приписываемый автоматически), например Q1.1.2, способное выполнять роль адреса ячейки; идентификатор создателя модели, то есть имя I; знаковая модель Q или A, в форме символьной строки; момент времени создания или модификации; имя проекта и другие атрибуты. Обобщённое представление Q-объекта приведено на рисунке 2.

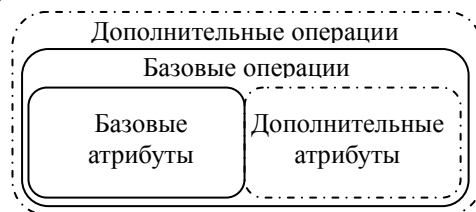


Рисунок 2 – Интерактивный Q-объект

Объекты A-типа имеют подобную структуру. Приписывание дополнительных атрибутов и программирование дополнительных операций осуществляется субъектом Sb_i в процессах решения задач, если это полезно (что отражено пунктирными границами). Кроме того, в инструментарий OwnWIQA встроены средства для решения ряда системных задач, в решениях которых применены дополнительная атрибутика с дополнительными операциями. К числу таких задач относятся, например, задачи формирования документов по шаблонам на базе QA-рассуждений, зарегистрированных в QA-памяти.

Отметим, что к любому интерактивному объекту можно прикрепить файлы, например, графические файлы, которые будут включаться в документы.

И всё же особой целостностью обладает пара ячеек QA-памяти, в которых содержатся взаимно дополнительные Q-объект и A-объект. С такой парой в QA-памяти связывается QA-объект, позволяющий представлять «предикативность» и «актуальное членение» в знаковых описаниях, например, «обстоятельств окружающей среды». За такой целостностью «взаимодополнительность вопроса и ответа», что позволяет приписывать характеристику «да» (например, истинности или

правдоподобия) согласованным текстовым представлениям вопроса Q_i и ответа A_i .

С другой стороны, любая коммуникативная единица, представленная знаками языка, например «предложение», открыта для её актуального членения на «тему» и «рему» [Иванов. 2010], указывающую на то «информационно новое», для выражения которого предложение создано. Актуальное членение проще всего выразить и зарегистрировать с помощью согласованных вопроса Q_i и ответа A_i .

Следовательно, с помощью QA-объекта можно представить в QA-памяти «переменную» (Q-объект) и её «значение» (A-объект), как для любого простого типа, так и для любого составного типа. Более того, «переменной» определённого типа с помощью дополнительных атрибутов можно приписать «характеристики типа», общепринятые в языках программирования, а также полезные «семантические характеристики».

За приписыванием «переменной» её «значения» стоит «предложение», с которым (с помощью дополнительных атрибутов) можно связать определённую «модальность» или их совокупность, например, приписав «значению» его «неопределённость», например, с помощью вероятностной меры.

А значит, в QA-памяти информационно представимы «объекты» и «обстоятельства» окружающей среды, причём с привязкой ко времени и с учётом измерений и идентификации их характеристик. Отображение двух объектов среды и связи между ними в QA-памяти обобщённо представлены на рисунке 1.

В QA-памяти представимы не только объекты предметной области $Z=\{Z_k\}$, но и процессы, структурированные алгоритмически. Так, например, шаг алгоритма, выраженный в виде псевдокодированного оператора, может быть загружен в поле описания Q-объекта. Ответом на такую интерпретацию вопроса логично считать результат выполнения оператора.

Отметим, что в общем случае «вопрос» может включать «подчиненные вопросы», что приводит к иерархически связанным совокупностям QA-объектов, представленных на рисунке 3.

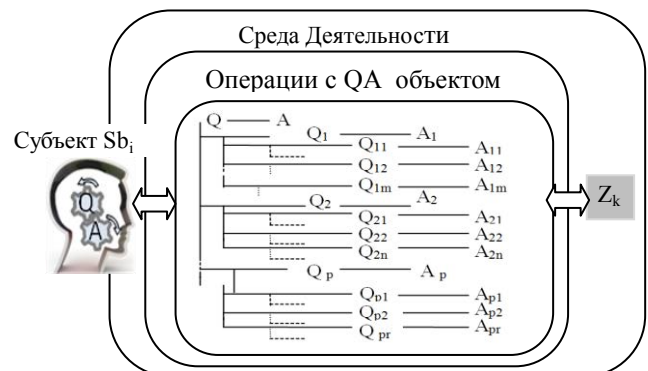


Рисунок 3 – Составной QA-объект

Отметим, что структура и содержание QA-объектов, определяются теми (профессиональными) задачами, которые приходится решать субъекту Sb_i . В частности, в QA-памяти представимы «требования» (requirements) и их «спецификация» в «проектах» или «материализация» в «продуктах», в создание которых субъект Sb_i внёс или вносит свой вклад, применяя доступный ему опыт.

3. Онтологизация персонального опыта

3.1. Ориентация на прецеденты

QA-память инструментальной среды OwnWIQA специфицирована и материализована так, чтобы в её ячейках можно было представлять решения задач $\{Z_k\}$, за которыми стоят «прецеденты», в осуществлении которых участвует субъект Sb_i . Более того, для «моделей прецедентов», которые хранятся в QA-памяти, должны использоваться нормативные формы, обобщённая структура которых приведена на рисунке 4.

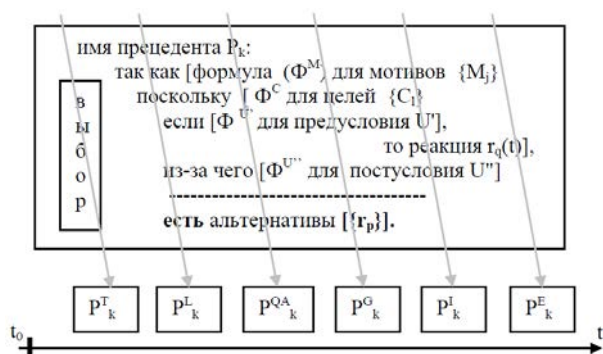


Рисунок 4 – Модель прецедента

Модель прецедента P_k включает: текстовую составляющую P_k^T , в виде постановки задачи; логическую составляющую P_k^L , формула которой представлена на рисунке 4; вопросно-ответную модель P_k^{QA} задачи Z_k ; графическое представление прецедента P_k^G ; исходный псевдокод P_k^I и исполняемый код P_k^E .

Нормативная модель прецедента построена таким образом, чтобы она раскрывала концептуальное содержание задачи и представляла её концептуальное решение (псевдокод решения). По этой причине в модели принципиальное место занимают конструкции на естественно-профессиональном языке L^P предметной области $\{Z_k\}$. Основными из этих конструкций являются текст T_k постановки задачи P_k^T и предложения $\{P_{ik}\}$ в модели P_k^{QA} , регистрирующей вопросно-ответный анализ этой задачи, в результате которого на все важные вопросы по её решению получены ответы, достаточные для построения логической схемы P_k^L и алгоритмического решения P_k^I .

В комплексе OwnWIQA сохранена ориентация активности его пользователя на «проекты», что позволяет выделять на множестве задач $\{Z_k\}$ их взаимосвязанные совокупности $S_m\{Z_k^m\}$, за каждой из которых стоит определённый «проект»,

выполненный или выполняемый субъектом Sb_i . Использование структуризации активности в единицах типа «проект» позволила заимствовать из комплекса WIQA.Net ряд средств для организации и управления процессами (псевдо) параллельного решения задач, а также заимствовать опыт создания и использования «онтологий проектов» [Соснин и др., 2012]. Так что в «персональной онтологии» $O_k = I(\{O_k^m\})$ интегрируются совокупность «онтологий проектов» $\{O_k^m\}$ субъекта Sb_i .

3.2. Лексика персональной онтологии

С множеством задач $\{Z_k\}$, решения которых ориентированы на прецеденты, связана определённая совокупность $\{T_k\} \cup \{P_{ik}\}$ употреблений языка L^P . Именно этот лингвистический материал в построениях персональной онтологии профессионального опыта должен занимать принципиальное место.

Следует отметить, что тексты и предложения из множеств $\{T_k\} \cup \{P_{ik}\}$ создаются в результате согласованного уточнения их версий, в которых применялась не только лексика моделей прецедентов $\{P_k\}$, но и лексика естественного языка L в том объёме, которым владеет субъект Sb_i . А значит из языка L логично отбирать всё то, что будет полезным для решения задачи онтологической систематизации множества моделей $\{P_k\}$.

В комплексе QwnWIQA для такого отбора используется толковый «Словарь», который строится субъектом Sb_i в процессе создания системы моделей $S(\{P_k\}, O^k)$, включающей его персональную онтологию O^k . В этом словаре содержатся и понятия онтологии, требующих их определения. Так что лексика онтологии формируется из лексики моделей прецедентов и лексики, используемой в толковом «Словаре».

3.3. Формирование приращений онтологии

В структуре персональной онтологии O^k выделяются взаимосвязанные неформальная и формализованная составляющие. Неформальная часть состоит из понятий $\{N_q\}$ онтологии с их толкованиями (на базе языков L и L^P), представленными в статьях «Словаря».

В каждой статье второй части «имя статьи» N_q связывается со следующими списками:

- списком $(g_{q1}, g_{q2}, \dots, g_{qn})$ имён дифференцирующих признаков, согласованным с определением имени N_q ;
- набором списков $\{(N_s, N_v, \dots, N_w)\}$ имён других статей, каждый из которых связан с определённым типом систематизации моделей прецедентов.

Выбранная версия представления онтологии $O^k = I(\{O_k^m\})$ упрощает её оперативное создание в процессе решения задач $\{S_m\{Z_k^m\}\}$, так как типовым приращением для O^k является O_k^m . Кроме того, каждая онтология O_k^m развивается за счёт приращений, основными из которых являются

имена объектов, зарегистрированных субъектом Sb_i в ячейках QA-памяти. По этой причине, формализованное содержимое онтологии O_k^m адресуется, что упрощает его включение в программную обработку, например, для доступа к моделям прецедентов.

В инструментальной среде WIQA.Net для выявления приращений использовался предикатно-онтологический контроль текстовых описаний проектных решений в определённых «проектах». Предложения такого описания (автоматизированно) переводились на прологоподобный язык, после чего выделенные предикаты проходили проверку на соответствие текущему состоянию онтологии проекта. Отрицательный результат проверки указывал либо на ошибку, либо на необходимость ввести в онтологию приращение. Средства и механизмы такого контроля детально раскрыты в публикациях [Соснин и др., 2010] и [Соснин и др., 2012].

В разработке комплекса OwnWIQA было решено отказаться от выявления приращений онтологии за счёт предикатно-онтологического контроля в связи с его сложностью (использование визуализируемого псевдофизического силового взаимодействия между членами предложений в текстах описаний) для потенциальных пользователей OwnWIQA.

Однако визуализацию предикатной структуры предложений было решено сохранить, предоставив субъекту Sb_i возможности её графического (block-and-line) представления (с помощью специализированного графического редактора). Для

выявления предикатной структуры субъект Sb_i может использовать морфологический анализатор, а в проверках механизмы онтологической фильтрации.

4. Средства создания персональной онтологии

Основным предназначением инструментария OwnWIQA является обслуживание концептуальной активности субъекта Sb_i в его профессиональной деятельности. Ориентация на профессиональную деятельность означает только то, что решаемые субъектом Sb_i задачи группируются и связываются в образования, которым можно приписать статус «проекта». Такая ориентация вводит *связность* в совокупности моделей прецедентов, а значит и *системность* в доступный для Sb_i опыт, так как часть этого опыта представлена моделями прецедентов.

Ориентация в OwnWIQA на разработку проекта привела к решению связать типовую единицу системности профессионального опыта с онтологией проекта, а интеграцию таких онтологий, то есть интеграцию $\{O_k^m\}$, связать с персональной онтологией O_k профессионального опыта субъекта Sb_i . Это же решение позволило отделить построение онтологий от действий по их интеграции, что и привело к нормативной концептуальной структуре «онтологии проекта», представленной на рисунке 5 в виде, который принято называть framework.

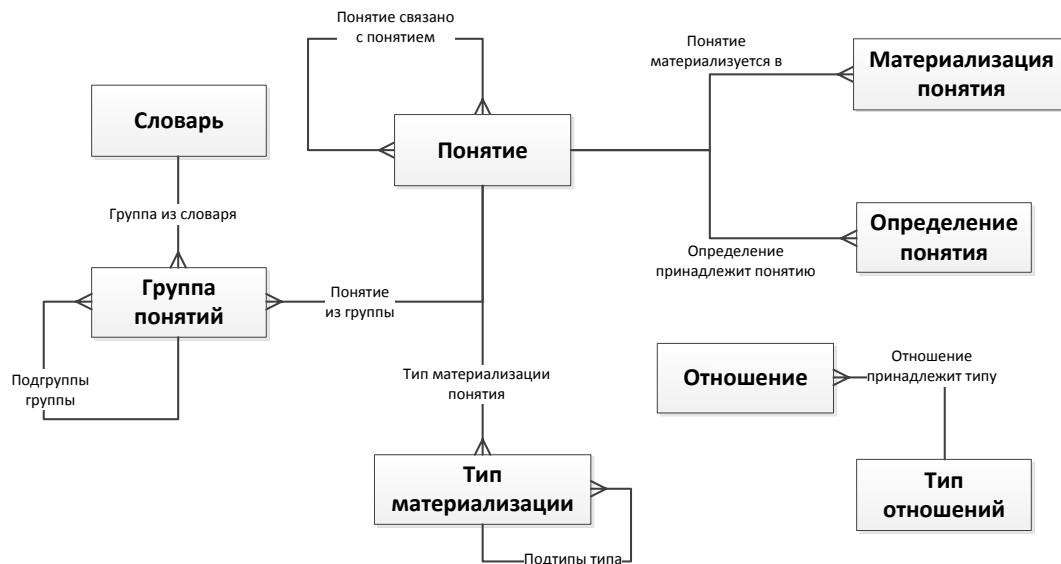


Рисунок 5 – Концептуальная структура онтологии

Так как о назначении и содержании «Словаря» было сказано выше, то пояснения в структуре необходимы только для «отношений» и «материализаций».

С различными «отношениями» в структуре типовой «онтологии проекта» связаны различные типы систематизации, которые можно ввести в

конкретную онтологию O_k^m . Приписывание отношениям имён и типов осуществляется субъектом Sb_i в операционной обстановке, фрагмент которой приведён на рисунке 6. Полезность типов, представленных в интерфейсной форме, была выявлена при создании и использовании «онтологий проектов» в инструментальной среде WIQA.Net. Отметим, что отоборанный набор типов

существенно богаче, чем те наборы типов, которые традиционно используются при создании онтологий.

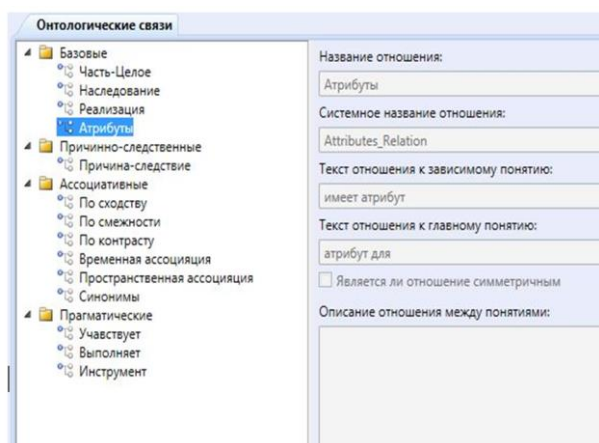


Рисунок 6 – Приписывание онтологических связей

Отметим, что отношениям можно не только приписать имена, выводящие на их семантику, но и прокомментировать. Такая возможность особо полезна для прагматических отношений, например, инструментального типа (связывающего объект с инструментами для его обработки или процесс со средствами его реализации).

«Материализация» введена в структуру онтологии для референциального связывания того, что стоит за понятием с «референтом», на которое понятие указывает при его употреблении в текстах или рассуждениях. Часть списка типовых «референтов» человеко-компьютерной среды, а также операционная среда их выбора и спецификации приведены на рисунке 7.

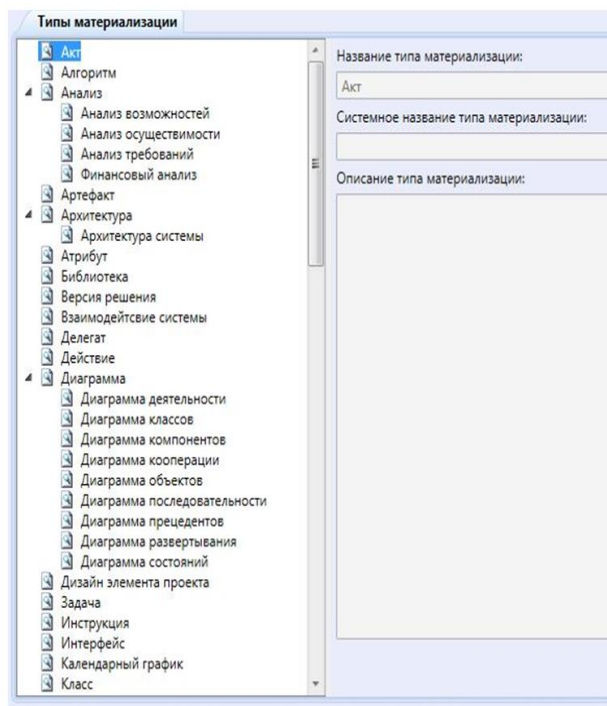


Рисунок 7 – Приписывание материализации

Учёт «материализации» повышает

ответственность субъекта Sb_i за выбор слов, используемых им в решении задач, способствует предотвращению ошибок, а также вводит в онтологию дополнительную систематизацию. Отметим, что список «типов материализации» открыт для включения новых составляющих.

Отметим и то, что для выполнения конкретной референциальной функции необходимо представить «референт» в среде OwnWIQA определённым файлом или указать «местоположение» референта в таком файле. Спецификация референции регистрируется с помощью интерфейсной формы, представленной на рисунке 8.

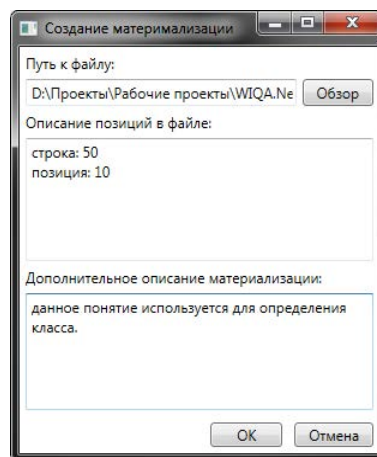


Рисунок 8 – Спецификация референта

Для того чтобы перейти к файлу, в котором материализовано понятие необходимо:

1. Подключить онтологию проекта, в котором находится понятие
2. В дереве онтологий выбрать необходимое понятие
3. Перейти на вкладку «Материализация»
4. В списке материализаций выбрать материализацию, к файлу которой необходимо перейти
5. Вызвать контекстное меню
6. Выбрать пункт «Перейти к файлу»

Такого рода инструкции имеются для типовых процедур OwnWIQA. В то же время для наиболее часто используемых процедур, возможно их выполнение за счёт активации определённых «горячих клавиш». Так, например, для перехода к файлу достаточно активировать клавиши $Ctrl + G$.

5. Реализация структур данных

Возможности создания и использования персональной онтологии в среде OwnWIQA обеспечиваются структурой данных, концептуальная схема которой приведена на рисунке 9 в форме, которая реализована в двух версиях, одна из которых ориентирована на создание онтологии и её использование субъектом Sb_i , а вторая открыта для её использования в псевдо-кодовых программах.

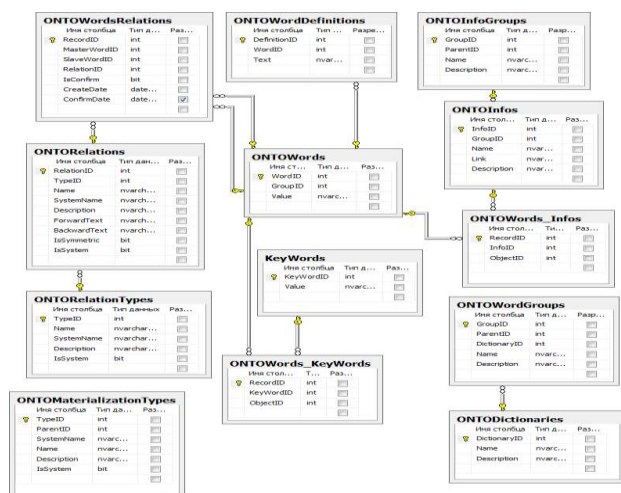


Рисунок 9 – Структура данных онтологии

Схема приводится с демонстрационными целями (без пояснений и спецификаций) только для того, чтобы показать масштабность информационной поддержки взаимодействий с онтологией. Её реализация по первой версии проведена по образцу программирования приложений с базами данных. Все интерфейсы, показанные выше, относятся к реализации именно этой версии.

Вторая версия представляет собой «клон» первой версии, загруженный как системный проект в QA-память комплекса OwnWIQA. Такая возможность обусловлена тем, что основу онтологии определяет иерархическая структура системы понятий. Системные отношения между понятиями представляются списками, которые регистрируются в полях описания соответствующих ячеек QA-памяти.

Подобный приём с клонированием фрагментов Базы данных используется в комплексе WIQA.Net для данных об организационной структуре коллектива проектировщиков [Sosnin, 2013].

6. Использование персональной онтологии

В комплекс OwnWIQA, встроены два базовых варианта применения персональной онтологии, первый из которых осуществляется в традиционных формах человеко-компьютерного взаимодействия. В рамках этого варианта субъект Sb_i имеет визуальный доступ к систематизации, которую он сам построил для совокупности моделей прецедентов, использованных или используемых им в определённых проектах. В таком взаимодействии принципиальным является проверенное и контролируемое использование лексики, согласованное с пониманием прецедентов.

Второй вариант открывает дополнительные возможности, во-первых, вопросно-ответного моделирования взаимодействий с онтологией, например, для проверки корректности её содержания и связей, и, во-вторых, для псевдо-

кодового программирования задач, в которых полезно применение онтологий.

К числу таких задач относится, например, поиск подходящей модели прецедента в сложившихся условиях. В OwnWIQA решение такой задачи возложено на программного агента, запрограммированного на псевдо-кодовом языке WIQA [Sosnin, 2013]. Обобщённая структура этого агента приведена на рисунке 10.

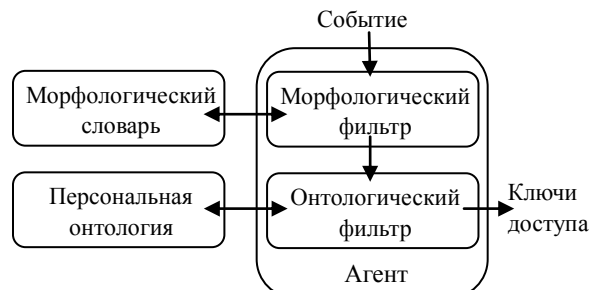


Рисунок 10 – Структура агента

Агент считает с «доски поиска» описание события, затребовавшего доступ к системе моделей прецедентов. Текст этого описания обрабатывается «морфологическим фильтром», который выделяет из описания список потенциальных «ключей доступа» к Базе прецедентов. «Онтологический фильтр» исключает из этого списка те лексемы, которые отсутствуют в онтологии.

7. Информационные источники

На формирование персонального профессионального опыта принципиальное воздействие оказывают информационные источники различных типов, доступные в компьютеризованных средах. Так, например, не следует изобретать собственные определения понятий, для хорошо освоенных проблемных областей. Их следует заимствовать из тех источников, которым можно доверять.

Именно такого рода понятия следует накапливать и систематизировать в той части онтологии, которая является общей для проектов субъекта Sb_i . Следует заметить, что, заимствуя устоявшиеся понятия, следует сохранить их связь с соответствующим источником или источниками. Такая функция в OwnWIQA возложена на компоненту «Информационные источники», концептуальная структура которой приведена на рисунке 11.

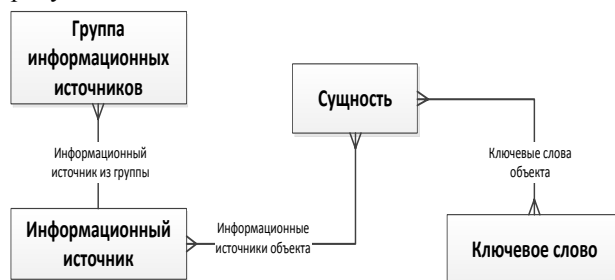


Рисунок 11 – Информационные источники

В концептуальной структуре присутствует блок «сущность», экземпляр которой может быть не только понятием, но и описанием определённого объекта. Независимо от того, что представляет собой экземпляр сущности, ему можно приписать «Ключевое слово» или совокупность таких слов, например для доступа в Интернет.

Заключение

В статье представлены средства, предназначенные для создания и использования персональной онтологии профессионального опыта в инструментально-моделирующей среде OwnWIQA. Инструментарий ориентирован на пользователя, который (в своих собственных проектах) решает задачи, а также представляет их решения в виде моделей прецедентов для повторного использования в будущих проектах. Он же, по ходу работ, систематизирует модели прецедентов, строя персональную онтологию опыта, соответствующую освоенным прецедентам.

Спецификации персональной онтологии специально согласованы с деятельностью в рамках проектов, что позволяет: разделять создавать онтологии проектов и объединяя их в единую систему; расширить набор типов онтологической систематизации; вводить референциальные связи составляющих онтологии с составляющими моделей опыта; использовать содержимое онтологии в программировании задач.

Библиографический список

- [Данилова и др. 1997] Данилова Н.Н., Крылова А.Л. Физиология высшей нервной деятельности / Н.Н. Данилова А.Л. Крылова – М.: Учебная Литература, 1997.
- [Иванов. 2010] Иванов Н. В. Актуальное членение предложения в текстовом дискурсе и языке/ Н. В. Иванов // Ozon.ru, Азбуковник, 2010
- [Соснин и др., 2010] Соснин П.И., Шамшев А.Б. Комплекс средств контроля семантики проектных задач и проектных решений / П.И. Соснин, А.Б. Шамшев // «Автоматизация процессов управления», 2010, № 3, – С. 55-62.
- [Соснин и др., 2012] Соснин П.И., Маклаев В.А. Инструментальные средства для спецификации концептуализаций в проектировании автоматизированных систем/ П.И. Соснин, В.А. Маклаев // Онтология проектирования, 2012. № 2, – С. 39-52.
- [Basili et al. 2001] Basili V. R., Lindvall M. Costa P., Implementing the experience factory concepts as a set of experience bases/ V. R. Basili, M. Lindvall, P. Costa// In Proc. of the International Conference on Software Engineering & Knowledge Engineering, 2001, pp. 102-10
- [Henninger et al. 2001] Henninger S., Tool Support for Experience-based Software Development Methodologies/ S. Henninger// Advances in Computers, vol. 59, 2003, pp. 29-82
- [Sjoberg et al. 2001] Sjoberg D.I. K., Dyba T., Jorgensen M., The Future of Empirical Methods in Software Engineering Research/ D.I. K. Sjoberg, T. Dyba, M. Jorgensen // In Proc. of Workshop Future of Software Engineering, 2007, pp. 358-378.
- [Sosnin, 2010] Sosnin P., Creation and Usage of Project Ontology in Development of Software Intensive Systems/ P. Sosnin// Polibits. – 2010 – № 2, pp. 132-146.
- [Sosnin, 2013] Sosnin P., A Scientifically Experimental Approach to the Simulation of Designer Activity in The Conceptual Designing of Software Intensive Systems/ P. Sosnin// IEEE ACCESS, vol. 1, 2013. C.488-504.

A PERSONAL ONTOLOGY OF PROFESSIONAL EXPERIENCE

P.I. Sosnin*

Ulyanovsk State Technical University, Ulyanovsk, Russia

sosnin@ulstu.ru

The paper presents an ontological systematization of the experience which supports a professional activity of a person. Specificity of the offered approach (to the creation of the personal ontology) is defined by using the precedent' model as a basic type of experience units. Models of precedents are constructed in accordance with the normative scheme during the project activity of the person. Solving the project tasks the person uses question-answer reasoning that supports access to the necessary experience.

Introduction

Human experience has a naturally artificial nature that is evolved in interactions of the person with surrounding. Such interactions are impossible without using the natural language providing a creation of precedents as typical units of the person behavior. Moreover, the language helps to form precedents' models and to register them in the experience. The paper is aimed at a systematization of the professional experience with the help of the personal ontology.

Main Part

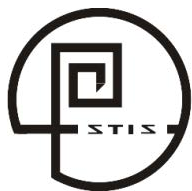
Offered approach is based on a reflection of the activity area to its presentation in a question-answer memory. In this case, all tasks that are solved in interactions with experience can be presented in a uniform way. The memory of such a type correlates with the hierarchical structure of the ontology.

The personal ontology is created and used by the person in accordance with a precedent framework which is adjusted on any task being solved. Moreover, any component of the project ontology includes the referential relation with the corresponding component of the precedent model.

The project ontology is also created on the base of the normative framework. This framework is developed as a number of operational windows which support the work of the person with the dictionary, relations of systematizations and means of references between ontology and models of precedents.

Conclusion

Specifications of the personal ontology are coordinated with activity within the projects that allow: separately to create project ontology for embedding it to the integral system; to expand a set of types of ontological systematizations; to use the referential relations among ontology and experience models; to use components of the ontology in programming of tasks.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

РОЛЬ ОНТОЛОГИИ В ПРОЕКТИРОВАНИИ ИНФОРМАЦИОННЫХ СИСТЕМ

Боргест Н.М.

Самарский государственный аэрокосмический университет им. С.П.Королева, г. Самара, Россия
borgest@ssau.ru

Институт проблем управления сложными системами Российской академии наук, г. Самара, Россия
borgest@yandex.ru

В работе показана определяющая роль онтологии в выборе средств, методов, инструментов и технологий при создании информационных систем. Отмечается взаимосвязь жизненных циклов онтологий и информационных систем. Приведены примеры построения онтологий, на основе которых разработаны соответствующие информационные системы в области проектирования и производства авиационной техники.

Ключевые слова: онтологии; информационные системы; жизненный цикл; проектирование.

Введение

В Коммюнике онтологического саммита (Ontology Summit 2013 Communiqué), подготовленного в мае 2013 года, в качестве проблемы отмечается отсутствие как общего мнения о методологии проектирования онтологий, так и согласия относительно методов оценки онтологий. Результатом проведенного группой международных экспертов анализа явился документ, в котором для оценки онтологий было предложено пять основных показателей качества: понятность, точность, техническое совершенство, адекватность и встроенность. Отмечено, что для обеспечения успеха при разработке, внедрении и использовании онтологий, их оценка должна проводиться на протяжении всего жизненного цикла (ЖЦ) [Neuhaus et al., 2013].

Проработка вопросов проектирования онтологий и связанное с ними исследование стадийности и формализмов в онтологическом пространстве предметной области (ПрО) напрямую влияет на все показатели и функционал разрабатываемых информационных систем (ИС), нацеленных на решение конкретных прикладных задач [Боргест и др., 2013].

В работе на примерах разработки системы автоматизированного проектирования самолета на этапе подготовки технических предложений и создания системы планирования производства авиационных двигателей для машиностроительного предприятия показана роль онтологий в определении структуры и функционала этих систем.

1. Онтологический анализ предметной области

Онтологический анализ ПрО – это разделение реального мира на составляющие и классы объектов, и определение их онтологий, или же совокупности фундаментальных свойств, которые определяют их изменения и поведение [Benjamin et al., 1994]. Онтологический анализ обычно начинается с составления словаря терминов, который используется при исследовании характеристик объектов и процессов, составляющих рассматриваемую систему, а также создания системы точных определений этих терминов. Кроме того, документируются основные логические взаимосвязи между соответствующими введенным терминам понятиями. Результатом этого анализа является онтология системы или же совокупность словаря терминов, точных их определений и взаимосвязей между ними.

Уже почти 20 лет процесс построения онтологии, согласно методологии IDEF5, рассматривается как состоящий из следующих действий или этапов [Benjamin et al., 1994]:

1. Изучение и систематизирование начальных условий. Этот этап устанавливает основные цели разработки онтологии, а также распределяет роли между участниками проекта.

2. Сбор и накапливание данных.

3. Анализ данных. Эта стадия заключается в анализе и группировке собранных данных с целью облегчения построения терминологии.

4. Начальное развитие онтологии. На этом этапе на основе отобранных данных формируется предварительная онтология.

5. Уточнение и утверждение онтологии.

Онтологический анализ ПрО представляет собой особый вид научной деятельности, в результате которой производится построение модели предметных знаний. Концептуальные составляющие предметных знаний являются онтологическими знаниями ПрО. В основе онтологического анализа лежит описание ПрО в терминах сущностей, отношений между ними и преобразования сущностей, которое выполняется в процессе решения некоторой задачи. При разработке онтологии ПрО проектные решения основываются на структурных свойствах класса. Онтология всегда отражает взгляд аналитика, т.е. всегда субъективна, а поэтому на сегодня не существует единственно правильного способа ее создания. Несмотря на это можно выделить некоторые основные принципы, на которых базируется создание онтологий. Это: ясность, согласованность, расширяемость, минимум влияния кодирования и онтологических обязательств [Гаврилова и др., 2000], [Найханова, 2008].

2. Модель жизненного цикла онтологий

На рисунке 1 представлено фактическое развитие методологии IDEF5 в виде обобщенной модели ЖЦ онтологий, включающей в себя 8 основных этапов или стадий «жизни» онтологий.

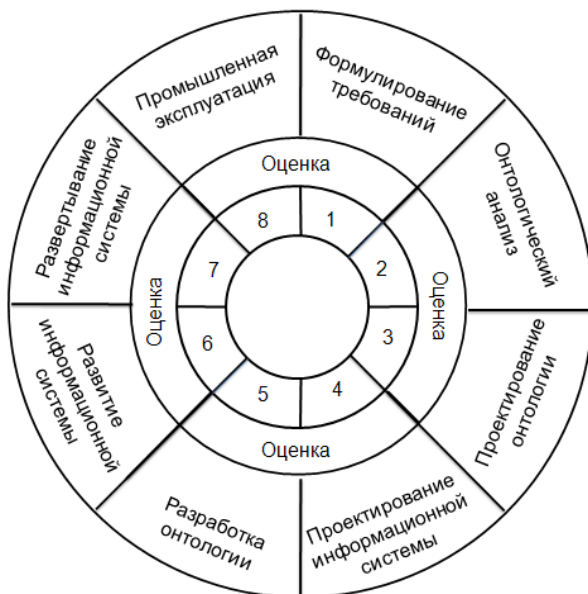


Рисунок 1 – Жизненный цикл онтологий [Neuhaus et al., 2013]

1. Формулирование требований
2. Онтологический анализ
3. Проектирование онтологий
4. Проектирование информационной системы
5. Разработка онтологий
6. Развитие информационной системы
7. Развертывание информационной системы
8. Промышленная эксплуатация

3. Жизненный цикл программного обеспечения

Жизненный цикл программного обеспечения (ПО) – это период времени, который начинается с момента принятия решения о необходимости создания программного продукта и заканчивается в момент его полного изъятия из эксплуатации [IEEE Std, 1990]. Основными стандартами жизненного цикла ПО являются ГОСТ 34.601-90 и ISO/IEC 12207:1995 (русский аналог — ГОСТ Р ИСО/МЭК 12207-99). Модель жизненного цикла ПО – это структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении ЖЦ. Модель ЖЦ зависит от специфики, масштаба и сложности проекта и специфики условий, в которых система создается и функционирует [IEEE Std, 1990].

Стандарт ISO/IEC 2382-1 определяет ИС как систему обработки информации, работающую совместно с организационными ресурсами, такими как люди, технические средства и финансовые ресурсы, которые обеспечивают и распределяют информацию. В узком смысле ИС это подмножество компонентов ИС, включающее базы данных, СУБД и специализированные прикладные программы. В этом случае ИС рассматривается как программно-аппаратная система, предназначенная для автоматизации целенаправленной деятельности пользователей и обеспечивающая в соответствии с заложенной в неё логикой обработки возможность получения, модификации и хранения информации [Маглинец, 2010].

4. Выбор онтологического редактора

При создании онтологий целесообразно использовать соответствующие инструменты, так называемые редакторы (конструкторы) онтологий или онтологические системы.

В настоящее время на рынке программных средств достаточно активно продвигаются более 50 редакторов онтологий и систем поддержки их жизненного цикла [Климань и др., 2010]. Для сравнительного анализа онтологических систем обычно в качестве критериев рассматривают: открытость системы, удобство работы (интуитивность), наличие доступного материала для самообучения, возможность импортирования данных из СУБД, поддержка формата OWL.

5. Стандарты для построения онтологий

Развитие работ в области создания ИС и онтологий, накопление успешных практик и обобщающих методик подталкивает специалистов ПрО, баз данных и онтологов к разработке стандартов, фиксирующих полезные начинания. Некоторые примеры таких успешных «рекомендаций» рассмотрены ниже.

ISO 10303 (STEP). Стандарты ISO 10303 определяют средства описания (моделирования)

промышленных изделий на всех стадиях ЖЦ. Проект STEP развивается с середины 80-х годов прошлого века. Единообразная форма описаний данных о промышленной продукции обеспечивается введением в STEP языка Express, инвариантного к приложениям. В стандартах STEP использован ряд идей, ранее воплощенных в методиках информационного IDEF1X и функционального IDEF0 проектирования. В рамках STEP предпринята попытка создания единых информационных моделей (онтологий) целого ряда приложений. Эти модели получили название прикладных протоколов. В качестве альтернативного языка для обмена геометрическими и техническими данными о промышленных изделиях используется язык разметки XML. В 2004 г. компаниями Dassault Systemes и Lattice Technology предложено подмножество 3D XML языка XML, которое получает все большую популярность для межсистемных обменов в CALS-технологиях.

ISO 22745 (eOTD). Международная ассоциация управления кодами электронной коммерции (ECCMA) активно продвигает стандарты ISO 22745 (Системы промышленной автоматизации и интеграции), ISO 8000 (Качество информационных технологий) и eOTD (открытый технический словарь). Стандарт ISO 22745 включает в себя словарь, представляющий собой совокупность терминов, определений и концепций, применяемых для описания отдельных объектов, организаций, адресов, товаров и услуг. В комплексе стандартов ISO 22745 описаны элементы данных, относящиеся к конкретным классам и парам значений свойств.

Открытый технический словарь позволяет точно определить свойства в соответствии с данными ISO 10303, определять информацию и обмениваться данными с партнерами из других стран без искажения смысла данных.

ISO (P-Lib). ISO 13584 Parts Library - это серия международных стандартов для представления и обмена доступными для компьютерной интерпретации данными о поставляемых компонентах и комплектующих изделиях (узлах, деталях). Стандарт ISO 13584 PLIB включает в себя 7 разделов: общий обзор и основополагающие принципы; концептуальная модель библиотеки деталей; основные ресурсы; логическая модель библиотеки поставщика; данные о поставщике; программный интерфейс к данным; методология структуризации классов (семейств) деталей. В отличие от стандарта ISO 10303 STEP, предназначенного для описания конкретного экземпляра продукта, стандарт PLIB позволяет описывать классы продуктов (компонентов и комплектующих).

ISO 15926. Наиболее перспективным стандартом организации онтологических баз данных является ISO 15926. Данный стандарт определяет структуру объектов. В нем специфицируется модель данных, определяющая значение сведений о ЖЦ в едином контексте. Эталонная модель данных (справочная

онтология) отражена в библиотеке справочных данных (RDL). Интеграция приложения в информационное пространство требует приведения в соответствие классов и атрибутов прикладной модели этого приложения к классам и атрибутам эталонной модели (операционная онтология).

6. Тезаурус - семантическая основа интеллектуального помощника проектанта

Современные ИС становятся не только инструментом, но и «помощником» проектанта [Боргест и др., 2012]. Благодаря высокой степени формализации процессов проектирования, компьютерные системы могут выполнять всё больший набор функций. На сегодняшний день умение работать с САПР – необходимый навык инженера при решении им проектных задач. Современные САПР – это сложнейшие программные комплексы, освоение которых может затрудняться из-за сложного интерфейса. Для преодоления этой проблемы ведущие разработчики САПР ставят перед собой задачи, включая создание систем, способных общаться с пользователем на тривиальном языке, что позволит значительно облегчить освоения сложных САПР. Наличие в арсенале САПР тезауруса ПрО приблизит решение указанной проблемы.

6.1. Модель сущностей и отношений

В качестве основы метамодели для Робота-проектанта используется тезаурус предметной области. Тезаурус представляет собой систематизированную совокупность понятий определенной отрасли науки, отражающих логические связи между терминами (рисунок 2). Эти связи основываются на классовой иерархии, родо-видовых и ассоциативных связях.

Основными единицами тезаурусов являются термины предметной области. Термин является одним или большим числом слов, обозначающих понятие. Понятие рассматривается как единица мысли, формируемая мысленно для отражения всех или некоторых свойств конкретного или абстрактного, реально существующего или мысленного объекта. Понятия существуют как абстрактные сущности, независимо от терминов, которые их выражают. Понятие предметной области обычно имеет несколько возможных вариантов лексического представления в тексте, которые рассматриваются как синонимы. Среди таких синонимов выбирается дескриптор-термин, который рассматривается как основной способ ссылки на понятие в рамках тезауруса. Другие термины из синонимичного ряда, включенные в тезаурус, называются аскрипторы. Они используются как вспомогательные элементы, помогающие найти подходящие дескрипторы.

Тезаурус дает единые, исчерпывающие данные о конкретном объекте. Целями использования

тезауруса в работе-проектанте являются: обеспечение общей терминологии для ПрО с целью совместного использования всеми пользователями; получение точных и непротиворечивых определений каждого термина; обеспечение задания семантики с помощью множества аксиом, которые автоматически позволяют получать ответ на множество вопросов о ПрО.

Помимо задач интеграции тезаурус может использоваться как основа для реализации голосового интерфейса. Содержимое тезауруса может быть использовано в качестве ключевых слов для подпрограммы голосового взаимодействия.

6.2. Установление родо-видовых отношений

Родо-видовая связь устанавливается между двумя дескрипторами, если объём понятия нижестоящего дескриптора входит в объём понятия вышестоящего дескриптора (ГОСТ 7.25-2001). Например, «крыло» и «оперение» являются одним из видов «несущих поверхностей», «несущие поверхности», в свою очередь, являются одним из видов «конструктивных частей самолёта», а «конструктивные части самолёта» - одним из видов «конструкций». На рисунке 3 показана также эквивалентная связь между терминами «фюзеляж» и «корпус самолёта».

6.3. Создание ассоциативных связей

Отношение ассоциации является неиерархическим и наиболее трудно определяемым. Ассоциативное отношение является объединением отношений, не входящих в иерархические отношения или в отношения синонимии. Допускается включать в ассоциативное отношение все виды отношений, кроме синонимии и отношения «род-вид». Основное назначение установления ассоциативных отношений между дескрипторами тезауруса - указание на дополнительные дескрипторы, полезные при индексировании или поиске. Например, «самолёт» - «конструкция», «самолёт» - «двигатель», «самолёт» - «параметры» и т.д. Между этими терминами нельзя установить иерархические родо-видовые отношения (например, «конструкция» не является одним из видов «самолёта») и они не являются синонимами, но состоят в отношениях принадлежности: «самолёт» имеет «конструкцию», «самолёт» имеет «двигатель», «самолёт» имеет «параметры» и т.д. В таком случае устанавливаются ассоциативные связи. В Protégé отношения ассоциации отображаются пунктирными линиями, стрелки показывают направление, в котором связаны термины (рисунок 4).

Ассоциативное отношение между двумя дескрипторами стоит устанавливать, если при употреблении одного термина другой термин подразумевается. Один термин может быть необходимым элементом определения другого

термина, например, термин «конструкция» составляет необходимую часть определения термина «самолёт». Отношения ассоциации включаются в онтологию тогда, когда нет возможности установить иерархические отношения или отношения синонимии.

В онтологию «Проектирование самолёта» включены 15 видов ассоциативных связей. Из них можно выделить несколько типов связей: принадлежности - «имеет», функциональные - «выполняет преобразование», «обеспечивает», «предназначен для», «разрушает», «создаёт», «назначает», «выбирает», «производится», «вызывает», «участвует», характеристические - «характеризует», «показывает», «представляет собой», «является». При создании онтологии количество видов ассоциативных связей минимизировалось с целью универсальности использования их для различных терминов.

Тезаурус «Самолет» создан на языке OWL, который позволяет отображать сложные связи и отношения рассматриваемой ПрО в информационной модели, а также использовать тезаурус для представления сложных информационных структур.

7. Онтология машиностроения

Онтология предприятия наиболее полно описана в монографии [Шведин, 2010]. Продукция машиностроительного предприятия характеризуется высокой степенью сложности и состоит из нескольких уровней сборок. При этом и детали, и компоненты, как правило, проходят различные маршруты и производятся несколькими цехами. В производстве сложной техники, такой как самолет или авиационный двигатель, производственные траектории объединяют целые сети предприятий отрасли, включая и межотраслевые связи и отношения. Системы оперативно-производственного планирования на основе мультиагентных технологий, позволяют найти и осуществить такие решения, которые обеспечивают параллельное выполнение нескольких заказов [Скобелев, 2013]. В такой системе одновременно должны осуществляться процессы распределения ресурсов, планирования, оптимизации, мониторинга и контроля выполнения заказов в режиме реального времени. Для создания, и, в дальнейшем, для полного и безошибочного функционирования такой системы необходима базовая онтология выбранной ПрО.

Цель создаваемой онтологии машиностроения выступить в роли семантической основы разрабатываемой ИС, предназначенной для производственного планирования, помочь в обосновании определения методов и технологий решения задач.

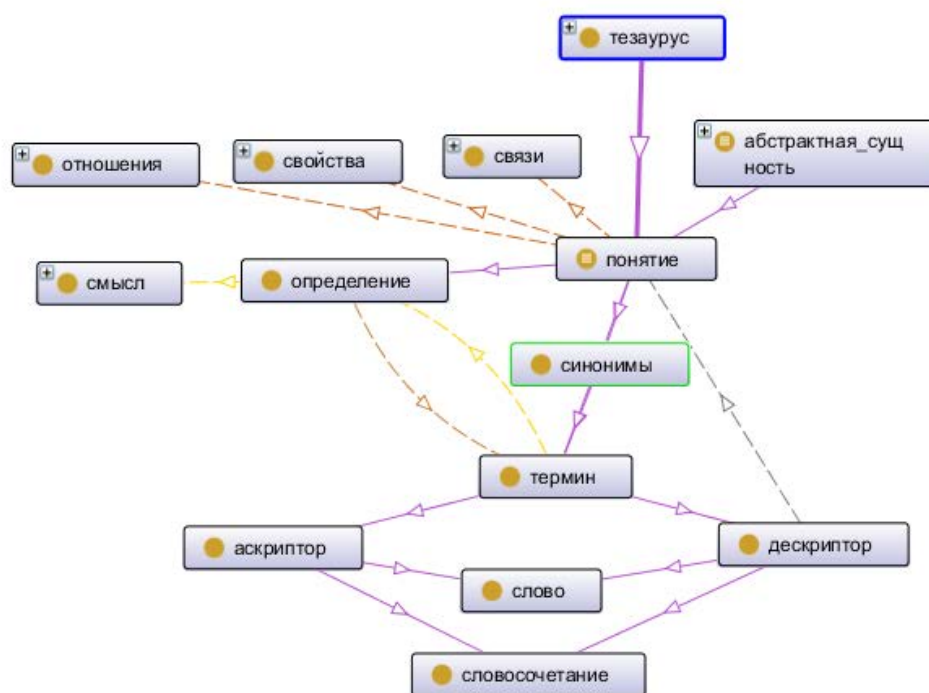


Рисунок 2 – Модель сущностей и отношений в тезаурусе

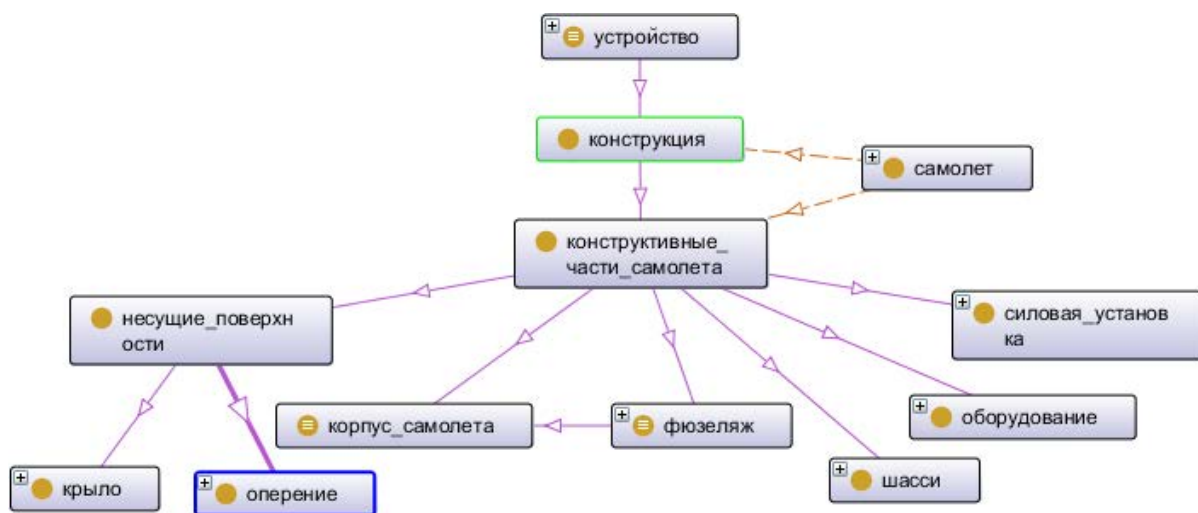


Рисунок 3 – Установление иерархической родо-видовой связи между терминами

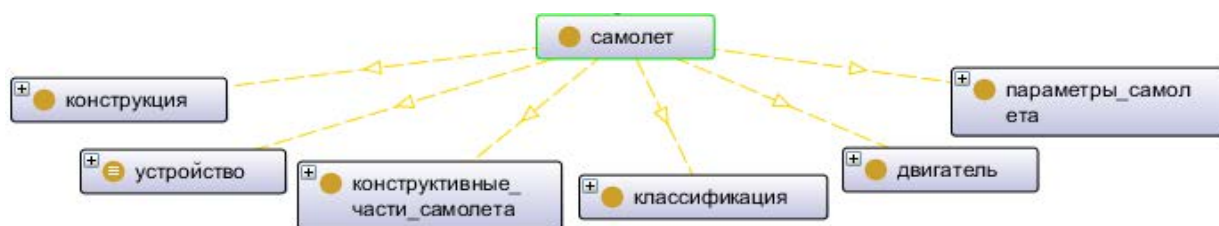


Рисунок 4 – Установление ассоциативных связей принадлежности между терминами

На первом этапе был определен верхний уровень иерархии и выделены основные сущности предприятия. Основным критерием для включения сущности в верхний уровень иерархии является отсутствие у нее надкласса. В онтологии машиностроительного предприятия созданы шесть подклассов [Шустова, 2013]:

1. Предприятие и все его сущности (цеха, отделы, бюро и проч.) являются объектами управления,
2. Управление предприятием (начальники, мастера, инженеры, и др.) – субъекты управления,
3. Документы – средства управления,
4. Материальные ресурсы – ресурсы для управления,

5. Выпускаемая продукция – результат деятельности (управления),
6. Процессы – процессы, выполняемые сущностями.

Предприятия различных отраслей промышленности имеют разную производственную структуру. В цехах создаются соответствующие бюро и службы: планово-диспетчерское бюро, бюро организации труда и заработной платы, техническое бюро, бухгалтерия, службы технолога, механика, энергетика и др.

Основными (типовыми) цехами современных машиностроительных предприятий являются цеха:

- заготовительного производства;
- инструментального производства;
- механосборочных работ;
- испытаний изделий;
- ремонта изделий.

Кроме описания деятельности подразделений в онтологию предприятия заносятся комментарии, которые доступны для просмотра как в основном окне, так и в контекстном меню блока семантической сети.

Разработанная НПК «Разумные решения» для ОАО «Кузнецов» на основе созданной базовой онтологии машиностроения мультиагентная система производственного планирования «Smart Factory» функционирует как автономная система. Она может быть интегрирована с существующими программами, такими как: программы по складскому учету, расчету заработной платы, системами бухгалтерского учета и отчетности и др. Система может применяться на машиностроительных заводах, которые характеризуются постоянными инновациями, существенной сложностью изделий и высокой динамикой изменений в среде, а также априорной неопределенностью спроса и предложения, требующей адаптивности при реакции на события в режиме реального времени.

Заключение

Работа в части создания прототипа интеллектуального помощника проектанта поддержана Программой развития НИУ СГАУ в рамках НИР по теме: «Разработка и развитие технологии генерации знаний на основе использования CAE/CAD/CAM/PLM-систем на примере создания элементов прототипа робота-проектанта в предметной области «самолет», в части создания базовой онтологии машиностроения поддержана Минобрнауки РФ в рамках государственного контракта № 07.524.12.4022 по теме: «Разработка распределенной интеллектуальной системы согласованного управления производственными цехами корпораций машиностроительных предприятий, построенных по сетевому принципу», а также входит в число

фундаментальных работ по онтологическому инжинирингу ИПУСС РАН.

Библиографический список

[Боргест и др., 2013] Онтологии: современное состояние, краткий обзор / Н.М. Боргест, М.Д. Коровин // Онтология проектирования. – 2013. - №2. - с. 49-55.

[Боргест и др., 2012] Робот-проектант: фантазия и реальность / Боргест Н.М., Громов А.А., Громов А.А., Морено Р.Х., Коровин М.Д. Шустова Д.В. и др. // Научный журнал «Онтология проектирования» №4(6) – 2012 - с.73-94

[Гаврилова и др., 2000] Базы знаний интеллектуальных систем: учебное пособие для вузов / Гаврилова Т.А., Хорошевский В.Ф. // СПб.: Питер, 2000. - 382с.

[Климань и др., 2010] Критерии принятия решений при выборе редактора онтологий. / Д. Климань, В. Любченко // Information Models of Knowledge.ITHEA. Kiev, Ukraine – Sofia, Bulgaria, 2010-2011 с.

[Маглинец, 2010] Анализ требований к автоматизированным информационным системам : учебное пособие / Ю. А. Маглинец. – М. : Интернет-Ун-т информ. технологий : БИНОМ. Лаборатория знаний, 2010. – 200 с.

[Найханова, 2008] Технология создания методов автоматического построения онтологий с применением генетического и автоматного программирования: монография / Найханова, Л.В. // г. Улан-Удэ: Изд-во БНЦ СО РАН, 2008. - 244с.

[Скобелев, 2013] Ситуационное управление и мультиагентные технологии: коллективный поиск согласованных решений в диалоге / Скобелев П.О. // Онтология проектирования. – 2013. – №2. с.26-49.

[Шведин, 2010] Онтология предприятия: экспириентологический подход: Технология построения онтологической модели предприятия / Шведин Б.Я. – М: Ленанд, 2010. – 240 с.

[Шустова, 2013] Онтологии инженерной деятельности – семантическая основа разрабатываемых систем / Шустова Д.В.// Научно-информационный межвузовский журнал "Аспирантский вестник Поволжья", № 7-8, 2013.

[Benjamin et al., 1994] IDEF5 Method Report. /Perakath C. Benjamin, Christopher P. Menzel, Richard J. Mayer, Florence Fillion, Michael T. Futrell, Paula S. de Witte, Madhavi Lingineni. // Information Integration for Concurrent Engineering (IICE). Contract: F33615-C-90-0012. Knowledge Based Systems, Inc. 1994. - p.187.

[Neuhaus et al., 2013] Ontology Summit 2013 Communiqué: Towards Ontology Evaluation across the Life Cycle, Fabian Neuhaus, Amanda Vizedom, Matthew West, Peter Yim // 2013, 13 p. http://ontolog.cim3.net/cgi-bin/wiki.pl?OntologySummit2013_Communique

[IEEE Std, 1990] 610.12-1990 - IEEE Standard Glossary of Software Engineering Terminology. - Available from: <http://standards.ieee.org/findstds/standard/610.12-1990.html>

ROLE OF ONTOLOGIES IN INFORMATION SYSTEMS DESIGN

Borgest N.M.

*Samara State Aerospace University named after
S.P. Korolev, Samara, Russia*

borgest@ssau.ru

*Institute of Control of Complex Systems of the
Russian Academy of Sciences, Samara, Russia*

borgest@yandex.ru

The work shows the prominent role of ontologies in the choice of means, methods, tools and technologies in creation of informational systems. Noted the relationship of the life cycles of ontologies and information systems. We present examples of ontologies and relevant information systems in the design and manufacture of aeronautical engineering.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

МОДЕЛИРОВАНИЕ ДИНАМИКИ ПРОЦЕССОВ ПОСЛЕДОВАТЕЛЬНОСТЬЮ ОДНОРОДНЫХ СЕМАНТИЧЕСКИХ СЕТЕЙ НА ОСНОВЕ АНАЛИЗА ПОСЛЕДОВАТЕЛЬНОСТИ ОПИСЫВАЮЩИХ ПРОЦЕСС ТЕКСТОВЫХ ВЫБОРОК

Харламов А.А. *, Ермоленко Т.В. **, Жонин А.А. ***

** Институт высшей нервной деятельности и нейрофизиологии РАН, г.Москва*

kharlamov@analyst.ru

*** Институт проблем искусственного интеллекта, г. Донецк, Украина*

etv@iai.dn.ua

**** Государственный научно-исследовательский институт информационных технологий и телекоммуникаций «Информика», г.Москва*

neurofish@yandex.ru

Описанный в статье подход к моделированию динамики процессов основан на технологии автоматического смыслового анализа текстовой информации. В процессе обработки текста формируется ассоциативная сеть, ключевые понятия которой, в том числе, лексические маркеры анализируемого процесса, ранжируются их смысловым весом. Взвешенный статусом маркера на шкале «хорошо-плохо», этот вес дает значение вклада маркера в характеристику состояния процесса. Изменение нормированной суммарной для всех маркеров характеристики процесса от временного среза к временному срезу и характеризует направленность процесса.

Ключевые слова: автоматическая обработка текста, ассоциативная (однородная семантическая) сеть, моделирование динамики процессов, лексические маркеры.

Введение

Огромный объем личной информации, в том числе, текстовой, которой делятся пользователи, отслеживается не только спецслужбами, друзьями или случайными знакомыми; за потенциальными или действующими клиентами следят банки и микрофинансовые организации, которые стремятся оценить кредитоспособность заемщика. Целью такого отслеживания является поиск и анализ сообщений по заданной тематике [Конторович, 2011].

Помимо соцсетей источниками информации для анализа могут быть блогеры, количество читателей постов которых сравнимо с аудиторией СМИ, и наконец, собственно СМИ.

Располагая инструментарием для автоматического определения эмоциональной и оценочной окраски текста, можно обследовать выборки текстов блогосферы значительного объема. Зная тематическую принадлежность или другие характеристики исследуемых текстов, можно определять, какие сегменты блогосферы связаны с выражением положительных или отрицательных

оценок и эмоций. Таким образом, анализируя эмоциональную окраску последовательности (вчера, сегодня, завтра) текстовых выборок из социальных сетей, в результате можно получить динамику развития социального процесса (улучшение, ухудшение, стабильно).

Выявление в документе эмоционально окрашенной лексики и эмоциональной оценки объектов автором является основной задачей анализа тональности текста или Sentiment analysis – развивающегося направления компьютерной лингвистики. Эмоциональная оценка, выраженная в тексте, также называется тональностью, или сентиментом текста [Розин и др., 2011].

Большинство существующих методов эмоциональной оценки основано на использовании словаря эмоционально окрашенных слов [Kerstin Denecke, 2008], [Thelwall et al., 2010] и словаря символов, обозначающих эмоции. В словарных методах каждое слово обладает весом, характеризующим его эмоциональную окрашенность.

К статистическим методам выявления таких терминов относятся способы анализа текста,

реализованные на основе нейросетевых алгоритмов. Одним из хорошо зарекомендовавших себя методов статистического анализа, реализующего глобальный анализ текстов является метод на основе формализма искусственных нейронных сетей из нейроподобных элементов с временной суммацией сигналов, используемого для формирования статистического портрета текста; и формализма искусственных нейронных сетей Хопфилда, используемого для смысловой перенормировки весов ключевых понятий в тексте [Харламов и др., 2008], [Харламов и др., 2011]. Этот подход обладает достаточным быстродействием и не зависит от языка и предметной области.

В результате анализа текста из него автоматически извлекается индекс в виде сети основных понятий и их связей с весовыми характеристиками. В качестве смыслового портрета текста рассматривается не просто список ключевых слов, а сеть понятий – множество ключевых слов или устойчивых словосочетаний связанных между собой. Каждое понятие получает некоторый вес, отражающий значимость этого понятия в тексте. Связь между понятиями тоже имеет вес. Использование связей позволяет более точно взвешивать понятия текста. Для извлечения лексических маркеров используются лингвистические фильтры.

Текст как законченная мысль автора описывает некоторую ситуацию. Поэтому корректный смысловой анализ текста позволяет выявить ключевые понятия текста и их ранги (смысловые веса) в этом тексте. Если имеется множество текстов, описывающих динамику некоторой ситуации: один текст – вчера, второй – сегодня, третий – завтра. Тогда анализ каждого из этих текстов позволяет выявить ранги ключевых понятий, и состояний ситуации вчера-сегодня-завтра. Таким образом, ассоциативная (однородная семантическая) сеть, полученная на основе анализа текста, который описывает состояние ситуации в текущий момент времени, включающая в свой состав ключевые понятия текста в их взаимосвязях и с их (ключевых понятий и их связей) весовыми характеристиками (рангами), дополненная такими же сетями, полученными в предыдущие и последующие моменты времени, является моделью динамики ситуации. Объединение одноименных понятий в такой последовательности сетей (вдоль оси времени) горизонтальными связями, дополненное рангами этих понятий в соответствующие моменты времени, показывает динамику их участия в ситуации. При этом ближайшие ассоцианты ключевых понятий (понятия, отстоящие в сети от анализируемого понятия на одну связь) позволяют проинтерпретировать динамику ключевых понятий (влияние на него сопутствующих обстоятельств).

Последовательность таких семантических сетей, описывающих последовательные во времени выборки текстов, с их (сетей, следовательно, и

текстов) суммарными численными характеристиками, является моделью процесса в его динамике. При этом из каждой сети фильтруется только та ее часть, которая содержит лексические и психолингвистические маркеры, характеризующие моделируемый процесс (и их ближайшие ассоцианты). Эти лексические и психолингвистические маркеры имеют свои смысловые веса в сети. Суммарные численные значения весов маркеров, взвешенных их статусом на шкале «хорошо-плохо» характеризуют состояние процесса в текущий момент времени. А в последовательности сетей возникает динамика смысловых рангов соответствующих понятий, что и моделирует динамику социального процесса: улучшение, ухудшение, стабильно.

1. Теоретические основы моделирования динамики процессов на основе анализа последовательности текстовых выборок

Предлагаемый в работе количественный анализ направленности процессов основывается на двух основных подходах.

Для анализа процесса используется автоматическое формирование семантической сети текста (корпуса текстов), содержащего информацию, касающуюся процесса. При этом автоматически извлекаемые из текста, среди прочих ключевых понятий, лексические и психолингвистические метки ранжируются в зависимости от их значимости в тексте. Суммарный ранг этих меток определяет состояние процесса (хорошо-нейтрально-плохо).

Рассматриваются последовательные срезы текстовых корпусов, с построением их семантических сетей (и ранжированием меток), что позволяет выявлять динамику исследуемого процесса как изменение суммарного ранга меток, характеризующих процесс, от среза к срезу.

Рассмотрим эти два подхода более подробно.

1.1. Формирование ассоциативной (однородной семантической) сети текста

Анализ статистики слов и их связей в тексте позволяет реконструировать внутреннюю структуру текста, и таким образом, сформировать описание семантики предметной области текста.

Статистический анализ выявляет ключевые понятия текста - слова или устойчивые словосочетания с их частотой встречаемости в тексте. Важной особенностью используемого подхода, является возможность автоматически устанавливать взаимосвязи между выявленными ключевыми понятиями текста. При выявлении связей учитывается статистика попарного появления слов в смысловых фрагментах исследуемого материала. Далее статистические показатели ключевых слов пересчитываются в семантические веса, при этом учитываются

подобные характеристики ключевых понятий с ними связанных, а также учитываются численные показатели связей.

После пересчета статистических характеристик текста в семантические, ключевые понятия, которые не релевантны структуре текста, получают малый вес, а наиболее представительные наделяются высоким рангом. Полученная семантическая сеть позволяет производить различные виды анализа текстовой информации. Сеть отражает внутреннюю структуру текста, значимость выделенных ключевых понятий, а также, показывает степень связанности понятий в тексте. Такое представление текста получается полностью автоматически.

Технология реализует следующую обработку информации:

- сегментацию слов и предложений текста на основе графематических правил;
- нормализацию грамматических форм слов и вариаций словосочетаний. Выявление корневых основ ключевых понятий;
- фильтрацию в тексте семантически несущественных, вспомогательных слов: удаляются предлоги, числительные и самые общеупотребимые слова с широким значением;
- выявление ключевых понятий текста (слов и словосочетаний), и их взаимосвязей в тексте;
- вычисление их относительной (в тексте) значимости – рангов ключевых понятий;
- формирование представления семантики текста в форме семантической сети ключевых понятий.

До собственно статистического (с элементами лингвистики) анализа текста осуществляется его первичная обработка. Задачей первичной обработки текста является подготовка его к статистическому анализу. Подготовка текста заключается в очистке его от нетекстовых символов, а также в корректной обработке таких единиц текста как аббревиатуры, инициалы, заголовки, адреса, номера, даты, указатели времени.

Сегментация предложений позволяет разбить текст на участки, которые могут содержать терминологические словосочетания предметной области и избежать выделения неадекватных словосочетаний на стыках таких участков. В результате предобработки (с использованием морфологического анализа) близкие по форме слова и словосочетания приводятся к одинаковой форме (нормализуются).

Ключевые понятия предметной области (слова и словосочетания) выделяются с использованием частотного анализа текста. В процессе формирования частотного портрета текста подсчитывается частота встречаемости слов в тексте.

Сформированное таким образом представление лексики текста подвергается затем пороговому преобразованию по частоте встречаемости. Порог

отражает степень детальности описания текста. В процессе статистического анализа выделяются устойчивые термины и терминологические словосочетания, которые служат далее в качестве элементов для построения семантической сети. При этом в составе частотного портрета текста общеупотребительные слова, а также словосочетания, содержащие только общеупотребительные слова, опускаются.

Первичная (частотная) сеть формируется из выявленных на предыдущем этапе ключевых понятий за счет использования ассоциативных связей (попарной встречаемости) этих слов в смысловых фрагментах текста. В качестве критерия для определения наличия ассоциативной связи между парой понятий используется частота их совместной встречаемости в одном смысловом фрагменте текста (например, в предложении). Превышение частотой попарной встречаемости ключевых понятий некоторого порога позволяет говорить о наличии между понятиями ассоциативной (семантической) связи, а совместные вхождения понятий в предложения с частотой меньше порога считаются просто случайными.

Элементы полученного таким образом частотного портрета текста (однородной семантической) – ассоциативной – сети и их связи имеют числовые характеристики, отражающие их относительный вес в данном тексте, соответствующий частоте их встречаемости в тексте.

Для более точной оценки семантических весов понятий используются веса всех связанных с ними понятий, т.е. веса целого семантического сгущения. В результате итеративной процедуры перенормировки наибольшие веса получают ключевые понятия, связанные с наибольшим числом других понятий с большим весом, то есть те понятия, которые стягивают на себя смысловую структуру текста. Полученные таким образом смысловые веса ключевых понятий показывают значимость этих понятий в тексте.

1.2. Выявление динамики процесса

Полученная сеть представляет собой семантический (структурный) портрет текста (корпуса текстов). Если текст, или корпус текстов описывает некоторую структуру (научную разработку, предметную область, социологическую ситуацию), то сформированная таким образом семантическая сеть представляет собой семантический срез этой структуры в момент написания текста.

Семантическая сеть, построенная на тексте, написанном позже, и описывающем ту же структуру, может отличаться от первой, поскольку представляет текст, релевантный состоянию описываемого процесса на момент времени более поздний, чем предыдущее. Сеть может содержать те же ключевые понятия, но может не содержать

некоторых из них, которые выбыли из описываемой структуры, а может включать в себя другие понятия, которые появились в описываемой текстом структуре за это время. И, главное, весовые характеристики содержащихся в сети понятий могут отличаться от их весовых характеристик, какие были в первой сети.

Соединим одинаковые ключевые понятия обеих сетей связями, толщина которых будет пропорциональна весу ключевого понятия. Если понятия в обеих сетях имеют одинаковый вес, связь имеет одинаковую толщину от сети к сети. Если понятия имеют разные веса, связь, их соединяющая, либо утолщается, либо утоньшается, демонстрируя динамику состояний ключевых понятий, и, таким образом, динамику состояний сети в целом.

Если мы возьмем тексты следующего временного среза, и построим еще одну сеть, и присоединим ее к двум предыдущим, то будем иметь картину разворачивания структуры (научной разработки, предметной области, социологической ситуации) во времени. И так сколько угодно шагов. Такая модель динамики процесса наглядна, удобна для исследования (сеть как статический смысловой срез исследуемой структуры представляет собой удобный для навигации по нему объект в силу ассоциативности связей между ключевыми понятиями), и обладает числовыми характеристиками, что делает ее удобной для аналитического исследования процессов, и, как следствие, удобной для автоматического анализа.

Наконец, для того, чтобы исследовать конкретный процесс в его динамике, выберем ключевые понятия семантической сети, которые являются лексическими и психолингвистическими метками этого процесса, и будем исследовать динамику развития количественных характеристик этих понятий. Как и другие ключевые понятия, эти понятия могут появляться вновь, появляться последовательно на разных временных срезах, и наконец, исчезать. Могут также меняться от временного среза к срезу их численные характеристики. То есть они ведут себя как обычные ключевые понятия в динамике.

Удалим все ключевые понятия всех сетей, кроме упомянутых меток. В этом случае оставшаяся часть модели динамики текстов становится моделью динамики исследуемого процесса. Причем, суммарная числовая характеристика оставшихся ключевых понятий сети характеризует состояние процесса в текущий момент времени, а их изменение, от временного среза к срезу, характеризует динамику процесса во времени.

1.3. Формализм подхода

Для формирования однородной семантической (ассоциативной) сети создается частотный портрет текста, содержащий информацию о частоте встречаемости ключевых понятий текста, представленных как корневые основы

соответствующих слов, или их устойчивых сочетаний, встречающихся в тексте, а также об их совместной (попарной) встречаемости в смысловых фрагментах текста (например, в предложениях). Частотный портрет, таким образом, содержит информацию о частоте встречаемости ключевых понятий и их попарной (в терминах их ассоциативной связи) встречаемости в тексте. Использование хопфилдоподобного алгоритма позволяет перейти от частоты встречаемости к смысловому весу (вес связей при этом остается неизменным).

Эта обработка включает несколько этапов. На этапе первичной обработки из текста удаляется нетекстовая информация, текст сегментируется на слова и предложения, из текста удаляются стоп-слова, рабочие и общеупотребимые слова, а оставшиеся слова подвергаются морфологической обработке. Морфологическая обработка производится с использованием заранее подготовленного морфологического словаря (словаря флексивных морфем) – словаря первого уровня – $\{B_i\}_1$. В результате формируется словарь второго уровня – $\{B_i\}_2$ – словарь корневых основ (и устойчивых словосочетаний).

На следующем этапе строится частотный портрет текста, то есть выявляются частоты p_i встречаемости корневых основ B_{i2} ключевых понятий (полученных в результате морфологического анализа) и их устойчивых сочетаний, и частоты p_{ij} их попарной встречаемости в предложениях текста. Одновременно формируется словарь третьего уровня $\{B_i\}_3$ – словарь пар слов.

На третьем этапе, частоты встречаемости перенормируются в смысловые веса с использованием хопфилдоподобной итеративной процедуры. В результате итеративной процедуры перенормировки наибольшие веса получают ключевые понятия, связанные с наибольшим числом других понятий с большим весом, то есть те понятия, которые стягивают на себя смысловую структуру текста.

$$w_i(t+1) = \left(\sum_{i \neq j} w_i(t) w_{ij} \right) \sigma(\bar{E}) \quad (1)$$

здесь $w_i(0) = p_i$; $w_{ij} = p_{ij} / p_j$ и $\sigma(\bar{E}) = 1/(1 + e^{-k\bar{E}})$ – функция, нормирующая на среднее значение энергии всех вершин сети \bar{E} , где p_i – частота встречаемости i -го слова в тексте, p_{ij} – частота совместной встречаемости i -го и j -го слов в фрагментах текста. В дальнейшем эти весовые характеристики корневых основ используется для выявления предложений текста,

содержащих наиболее важную информацию в тексте.

В результате получается так называемая ассоциативная (однородная семантическая) сеть N как совокупность несимметричных пар понятий $\langle c_i c_j \rangle$, где c_i и c_j – ключевые понятия, связанные между собой отношением ассоциативности (совместной встречаемости в некотором фрагменте текста). Иначе семантическую сеть можно представить в виде множества звездочек $\langle c_i \langle c_j \rangle \rangle$, где $\langle c_j \rangle$ – множество ближайших ассоциантов ключевого понятия c_i .

Под семантической сетью N понимается множество несимметричных $\langle c_i c_j \rangle \neq \langle c_j c_i \rangle$ пар ключевых понятий $\langle \langle c_i c_j \rangle \rangle$, где c_i и c_j – понятия, связанные между собой отношением ассоциативности (совместной встречаемости в некоторой ситуации):

$$N \cong \{ \langle c_i c_j \rangle \}. \quad (2)$$

Семантическая сеть, описанная таким образом, может быть переописана как множество так называемых звездочек $\langle c_i \langle c_j \rangle \rangle$:

$$N \cong \{ z_i \} = \{ \langle c_i \langle c_j \rangle \rangle \} \quad (3)$$

Под звездочкой $\langle c_i \langle c_j \rangle \rangle$ понимается конструкция, включающая главное событие c_i , связанное с множеством событий-ассоциантов c_j , которые являются семантическими признаками главного события, отстоящими от главного события на одну связь. Связи направлены от главного события к событиям-ассоциантам.

Последовательность одноименных звездочек, принадлежащих разным временным смысловым срезам – семантическим сетям, называется элементарным процессом π :

$$\pi = z_i(t_1) \Rightarrow z_i(t_2) \Rightarrow z_i(t_3) \Rightarrow \dots \quad (4)$$

где $z_i(t_k)$ – конкретная звездочка в момент времени t_k . Вес ключевого понятия в текущий момент времени, определяющий его ранг в семантической сети – $w_i(t_k)$.

События-ассоцианты c_j главного понятия звездочки c_i являются его семантическими признаками, и позволяют интерпретировать его содержательно на каждом шаге процесса.

2. Описание информационной модели оценки направленности процесса на основе однородной семантической (ассоциативной) сети

Информационная модель включает в свой состав:

- модуль поиска релевантной исходной информации (текстов из открытых источников);
- модуль извлечения из подготовленных текстов общей для них семантической сети;
- модуль оценки направленности анализируемого процесса.

Модуль поиска релевантных для обработки текстов осуществляет поиск по заданным источникам текстов, удовлетворяющих условиям поставленной задачи. В том числе, задается регион, для которого предполагается проведение исследований, временной промежуток $\Delta T_l = (t_{l\text{end}} t_{l\text{beg}})$, $l = 1..L$, который принимается за l -й временной срез (один из L), а также лексические и психолингвистические маркеры M , характеризующие предметную область исследуемого процесса. Последние задаются экспертом, характерны строго для своего процесса (или нескольких процессов, если они исследуются совместно), и, в конечном итоге, определяют качество анализа. Зато информационная модель абсолютно не зависит от предметной области. Ей все равно, что представлять.

Процесс поиска текстов для каждого маркера $M_k, k = 1..K$ осуществляется отдельно, при заданных остальных (место и время) общих для всех маркеров параметрах поиска. Полученные на этапе поиска тексты обрабатываются (по отдельности) с формированием для каждого текста семантической сети N (см. следующий этап), исключительно для ранжирования текстов в корпусе текстов – результатов поиска относительно релевантности именно этому маркеру. Для этого в каждом тексте вычисляется смысловой вес $r_i = w_i$ заданного маркера. Он определяет релевантность текста этому маркеру.

Далее, для каждого маркера отбираются тексты, ранг которых по этому маркеру оказывается выше заданного порога $r_j \geq h_{\text{отбора}}$. Эти тексты, в совокупности по всем маркерам, составляют корпус текстов, подлежащих обработке на следующем этапе.

Модуль формирования семантической сети корпуса текстов N предусматривает несколько процедур в своем составе. Исходные тексты претерпевают предобработку, в процессе которой из них удаляется нетекстовая информация, удаляются также стоп-слова, рабочие и общеупотребимые слова, то есть слова, которые не несут смысла в этом корпусе текстов. Помимо этого, в процессе морфологического анализа, все словоформы

приводятся к своим корневым основам, чтобы увеличить достоверность результатов последующей статистической обработки

Процедура частотного анализа формирует частотный портрет текста в виде первичной ассоциативной сети, в которой участвуют в качестве вершин оставшиеся после предобработки леммы слов, связанные между собой связями, полученными из анализа попарной встречаемости слов в смысловых фрагментах текста (например, предложениях). Как вершины сети, так и их связи имеют числовые характеристики – частоты их (вершин - p_i , и связей - p_{ii}) встречаемости в анализируемом тексте. Эти числовые характеристики нам понадобятся после перенормировки для оценки рангов наших маркеров описываемого текстом процесса.

Процедура перенормировки итеративно пересчитывает частотные характеристики вершин сети (ключевых понятий текста) в смысловые веса таким образом, что понятия (вершины сети), связанные с большим числом других понятий, увеличивают свой вес, в ущерб весам других понятий. Понятия, несущие в тексте максимальную смысловую нагрузку, становятся максимально весомыми. Они как бы стягивают на себя структуру текста. Становятся главными темами текста.

Далее, работает последняя процедура обработки информации, которая выявляет численные характеристики маркеров их суммированием для конкретного временного среза текстов, и выявляет динамику изменений полученных таким образом обобщенных характеристик процесса от временного среза к временному срезу.

На этом этапе выбранные экспертом лексические и психолингвистические маркеры исследуемого процесса M_k , которые были заданы экспертом на первом этапе работы модели, фильтруют полученную на предыдущем этапе семантическую сеть $N = \{< c_i < c_j >>\}$. Из нее удаляются все вершины, кроме понятий-маркеров c_i , а также ближайших ассоциантов маркеров – вершин семантической сети (понятий), отстоящих от понятий-маркеров на один шаг $< c_j >$, являющихся их семантическими признаками.

Каждому маркеру на каждом временном срезе ставится в соответствие его смысловой вес $w_i = 0..100$, полученный на предыдущем этапе, который становится рангом этого маркера M_i для этого временного среза l .

Для всех маркеров вычисляется произведение Π_i статуса маркера (на шкале «хорошо-плохо») $S_i = (-1, 0, +1)$ на его ранг:

$$\Pi_i = S_i * r_i \quad (5)$$

И полученные для каждого маркера M_i произведения Π_i суммируются по всем маркерам:

$$\Pi = \sum \Pi_i \quad (6)$$

Таким образом, получается суммарная характеристика $\Pi(l)$ временного среза l оцениваемого процесса.

Далее строится график суммарных характеристик $\Pi(l)$ срезов. Или, в другом представлении, визуализируется ряд семантических сетей, включающих помимо маркеров также их семантические признаки (ближайшие ассоцианты) $c_i < c_j >$. При этом центральное понятие-маркер c_i в сети визуализируется размером, пропорциональным его рангу, цвет – соответствующий его статусу на шкале «хорошо-плохо», и все центральные понятия связываются между собой связями своего цвета вдоль оси времени.

Таким образом, на графике представляются основные тенденции каждого маркера во времени, а их ближайшие ассоцианты позволяют проинтерпретировать получившуюся картину.

3. Информационное моделирование оценки направленности процесса

В качестве примера представлено построение модели оценки направленности процесса на примере текстов по тематике внутренней политики РФ на основе новостных материалов портала newsru.com в свете отношений правительства и общества. Модель включает в свой состав:

- тексты из открытых источников, относящиеся к одной теме и разным временным срезам;
- семантические сети, построенные для корпусов текстов каждого среза;
- оценку направленности социального стресса, основанную на характеристиках лексических и психолингвистических маркеров.

Поиск релевантных текстов для обработки начинается с поиска по заданным источникам текстов, удовлетворяющих условиям поставленной задачи. Процесс поиска текстов для каждого маркера $M_k, k = 1..K$ осуществляется отдельно, при заданных остальных (место и время), одинаковых для всех маркеров, параметрах поиска. Полученные на этапе поиска тексты обрабатываются (по отдельности) с формированием семантической сети N (см. следующий этап), исключительно для ранжирования их в корпусе текстов – результатов поиска относительно релевантности именно этому маркеру. Для этого в каждом тексте вычисляется смысловой вес $r_j = w_i$ заданного маркера. Он определяет релевантность текста к этому маркеру.

Задачей моделирования оценки направленности процесса будет построение модели оценки направленности процесса социальной напряженности на примере упомянутых текстов по тематике внутренней политики РФ.

Перечень выбранных экспертным путём маркеров: "конфликт" (статус -2), "консенсус" (статус 1), "согласие" (статус 2), "бесконфликтность" (статус 2). Ранг отражает, с содержательной стороны, степень выраженности отношений согласия между правительством и обществом, чем выше ранг – тем более выражено согласие. С формальной вычислительной стороны ранг отражает вклад веса каждого маркера в итоговую интегральную оценку процесса социального стресса.

Тематические термины процесса: "правительство", "общество".

Временные срезы: сентябрь и октябрь 2013 года.

Выбранные тексты: выбрано 22 новостных текста за сентябрь 2013 и 17 новостных текстов за октябрь 2013. Не во всех текстах встречаются все выбранные маркеры. Надёжность оценки выбранного среза прямо зависит от объема корпуса текстов с выбранными маркерами в данном срезе.

Далее все тексты для каждого среза объединяются в единый текст и обрабатываются модулем автоматического анализа. В результате обработки строится итоговая семантическая сеть, содержащая среди понятий маркеры, для которых были вычислены их семантические веса (ранги).

Наконец, выявляются численные характеристики маркеров, их суммированием для конкретного временного среза текстов, и выявлением динамики изменений полученных таким образом обобщенных характеристик процесса от среза к срезу.

Таблица 1 – Оценка направленности процесса

Маркер	Статус	Ранг 09.2013г.	Вклад 09.2013	Ранг 10.2013	Вклад 10.2013
конфликт	-2	65	-130	79	-158
консенсус	1	72	72	98	98
согласие	2	54	108	93	186
бесконфликтность	2	83	166	52	104
Итоговая сумма:			216		230

Итоговые суммы являются интегральной оценкой процесса социальной напряженности по данной тематике для данного временного среза.

Однако бессмысленно рассматривать интегральные оценки в отрыве от динамики их изменения, поскольку сумма статусов участвующих в рассмотрении маркеров не сбалансирована и почти наверняка имеет отклонение в ту или иную сторону, что приведёт к отклонению интегральной оценки в случайном направлении. Состав маркеров одинаков для разных временных срезов и, поэтому, разница интегральных оценок между срезами не подвержена этому эффекту - случайные компоненты вклада взаимно уничтожаются.

Среднеквадратичное отклонение интегральной оценки, вычисленной методом бутстрепа (формирование случайной подвыборки) составляет 8.7. Оценка динамики, произведенная на данной выборке с данным распределением, и с данным среднеквадратичным отклонением, является достоверной.

Разница интегральных оценок есть оценка направленности процесса социальной напряженности. Показателем качества модели является устойчивость оценки направленности относительно добавления или удаления маркера сходного с имеющимися типа. Это означает, что предпочтительны модели с большим числом оцениваемых маркеров, и большее число тематических текстов должно подвергаться анализу. Возможна "калибровка" модели с целью вычисления среднего отклонения интегральной оценки при наличии устойчивости модели (признаваемой экспертно). Для этого следует принять некоторый временный срез за "0", точку отсчета и значение интегральной оценки в этом срезе вычитать в дальнейшем из интегральных оценок прочих временных срезов. Во всяком случае, необходимо иметь ввиду, что данный пример имеет чисто технический характер демонстрации работы алгоритма.

Оценка модели: данная модель имеет несбалансированный вклад множества рассматриваемых маркеров (общий вклад статусов маркеров равен +3, средний +0.75) и на рассматриваемой выборке новостных текстов по тематике взаимоотношений общества и правительства (задаваемых ключевыми словами "правительство" и "общество") обнаруживает положительную направленность процесса социальной напряженности от временного среза – «сентябрь 2013г.» до временного среза – «октябрь 2013г.».

Заключение

Описанный в статье подход к моделированию динамики процессов основан на хорошо зарекомендовавшей себя технологии автоматического смыслового анализа текстовой информации. В процессе обработки текста формируется ассоциативная сеть, ключевые понятия которой, в том числе, лексические и психолингвистические маркеры анализируемого процесса, ранжируются их смысловым весом.

Умноженный на статус маркера на шкале «хорошо-плохо», этот вес дает значение вклада маркера в характеристику состояния процесса. Изменение суммарной для всех маркеров характеристики процесса от временного среза к временному срезу характеризует направленность процесса. Приведенный пример не демонстрирует качество обработки, а лишь иллюстрирует работу механизма оценки. Предлагаемый подход является для эксперта инструментом моделирования процесса, настраивая который, подстраивая под свои представления, под реальный процесс, он может добиться адекватности моделирования.

Работа была выполнена в рамках НИР «Исследование методов интеллектуального анализа полуструктурированной информации и информационного моделирования направленности процессов социального стресса на основе данных из открытых источников» (при финансовой поддержке Министерства образования и науки Российской Федерации по Госконтракту от 10 октября 2013г. № 14.514.11.4114).

Библиографический список

- [Конторович, 2011] Конторович С.Д., Литвинов С.В., Носко В.И. Методика мониторинга и моделирования структуры политически активного сегмента социальных сетей [Электронный ресурс] / С.Д. Конторович, С.В. Литвинов, В. И. Носко // «Инженерный вестник Дона», 2011, №4. – Режим доступа: <http://ivdon.ru/magazine/archive/n4y2011/642/2/1428> (доступ свободный) – Загл. с экрана. – Яз. рус.
- [Розин и др., 2011] Розин М.Д., Свечкарев В.П., Конторович С.Д., Литвинов С.В., Носко В.И. Исследование социальных сетей как площадки социальной коммуникации рунета, используемой в целях предвыборной агитации [Электронный ресурс] / М.Д. Розин, В.П. Свечкарев, С.Д. Конторович, С.В. Литвинов, В.И. Носко // «Инженерный вестник Дона», 2011, №1. – Режим доступа: <http://ivdon.ru/magazine/archive/n1y2011/397> (доступ свободный) – Загл. с экрана. – Яз. рус.
- [Харламов и др., 2008] Харламов, А.А. Перестройка модели мира, формируемой на материале анализа текстовой информации с использованием искусственных нейронных сетей, в условиях динамики внешней среды. / А.А. Харламов, В.В. Раевский // Речевые технологии, N 3, 2008. С. 27-35.
- [Харламов и др., 2011] Харламов, А.А. Семантические сети как формальная основа решения проблемы интеграции интеллектуальных систем. Формализм автоматического формирования семантической сети с помощью преобразования в многомерное пространство / А.А. Харламов, Т.В. Ермоленко // Материалы международной научно-технической конференции OSTIS-2011, Минск БГУИР. С. 87-96.
- [Kerstin, 2008] Kerstin, Denecke Using SentiWordNet for multilingual sentiment analysis / Denecke Kerstin // IEEE 24th International Conference on Data Engineering Workshop. 2008. Pp. 507-512.
- [Thelwall et al., 2010] Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., & Kappas, A. Sentiment strength detection in short informal text / M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, A. Kappas // Journal of the American Society for Information Science and Technology, 2010. Pp. 2544–2558.

MODELING OF PROCESS DYNAMICS BY SEQUENCE OF HOMOGENOUS SEMANTIC NETWORKS ON THE BASE OF TEXT CORPUS SEQUENCE ANALYSIS

Kharlamov A.A. *, Yermolenko T.V. **,
Zhonoin A.A. ***

**Institute of Higher Nervous Activity and
Neurophysiology of Russian Academy of Sciences,
Moscow*

kharlamov@analyst.ru

*** Institute of Artificial Intelligence Problems,
Donetsk, Ukraine
etv@iai.dn.ua*

**** State Institute of Information Technologies
and Telecommunications «Informika», Moscow
neurofish@yandex.ru*

The represented approach of dynamic process modeling is based on the technology of automatical semantic text analysis. An associative network is forming during text processing. Its key notions, including lexical and psycholinguistic markers of the analyzed process, are ranked by their semantic weights. The weight being multiplied by marker status value at the scale of “good-bad” gives its contribution to the process stage characteristic. Transformation (dynamics) of the accumulated for all of the markers process characteristics from one period of time to another one characterizes a direction of the process.

Introduction

Process dynamics modeling (social process for example) needs to form the process model. This model is usually formed by hand. Automatization of the process could cut the cost of the model. Just now everybody can extract the information about sentiments from texts about analyzed process. There is two more steps to model of the process dynamics. We need the instrument which automatically shows the process statics. And then we must to show the results on the timeline.

Main Part

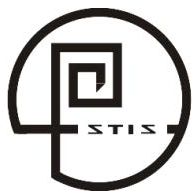
Associative (homogeneous semantic) network is such an instrument. Its nodes are the main concepts of analyzed text – lexical and psycholinguistic markers of the process. Also the network includes weight characteristics of the concepts and their relationships. This weights are characterized rank of the concepts in the text. The network can be constructed automatically by TextAnalyst program instrument.

The sum of such concept ranks weighted by their status (at the scale “good-bad”) characterizes quantitatively the order of the process at the given time period. The process markers are selected by expert. The expert also fixes their status at the scale “good-bad”.

Such value sequence which extracted automatically at the base of text corpus characterizes analyzed process dynamics at the timeline (yesterday-today-tomorrow).

Conclusion

In the paper the approach to automatical modeling of process dynamics at the base of automatical text corpus analysis for the sequential points at the timeline are shown. Formalism of the approach on the base of automatical semantic text analysis and algorithm of process dynamic analysis are shown also. The algorithm action is explained by example.



УДК 004.822:514

ПРАВИЛА ФОРМИРОВАНИЯ ТЕРМИНОЛОГИЧЕСКИХ КЛАСТЕРОВ

Мальковский М.Г., Соловьев С.Ю.

Факультет ВМК МГУ имени М.В.Ломоносова, г.Москва, Россия

malk@cs.msu.su

soloviev@glossary.ru

В работе рассматривается задача кластеризации терминологической сети и предлагается двухэтапный метод ее решения. На первом этапе отбираются кандидаты в центры кластеров, часть из них отсеивается на втором этапе, а для оставшихся центров формируются искомые кластеры. Принципы решения основных подзадач кластеризации формулируются в виде трех групп правил. Для проверки работоспособности предложенного подхода построена стратегия управления правилами, посредством которой удалось вполне успешно разделить на кластеры терминологическую сеть УТП.

Ключевые слова: терминологическая сеть, понятие, кластер.

Введение

Научное определение того или иного термина явно или неявно предполагает существование родственных терминов, образующих в совокупности терминосистему проблемной области [Шелов, 2003]. Структуры терминосистем представимы в виде совокупности семантических связей, допускающих объединение в единую терминологическую сеть [Мальковский и др., 2012].

Терминологическую сеть можно рассматривать как естественную (в некотором смысле) надстройку над множеством определений терминов. С формальной точки зрения терминологическая сеть представляет собой семантическую сеть, узлами которой являются определения терминов, а дугами – экземпляры бинарных отношений из заранее фиксированного набора допустимых отношений.

В терминологических сетях:

- каждая дуга представляет собой упорядоченную пару узлов, помеченную символом отношения; если для дуги не оговаривается ее родовая принадлежность, то в записи такой дуги метка опускается;
- набор допустимых отношений обязательно содержит родовидовые отношения, которым соответствуют дуги $(A, B)_P$, где A – вид, B – род;
- понятийным узлом называется узел в который заходит хотя бы одна дуга;
- потомками понятийного узла A называются понятийные узлы B , связанные с A дугой (B, A) ;

- каждый понятийный узел имеет уникальное имя, которое одновременно служит наименованием понятия;

- как правило, наименование понятия есть общее наименование объектов, составляющих его объем: “Анемометры”, “Варочные печи”, “Именные ценные бумаги” и т.д., но “Российская Федерация”, “Ботаника” и пр.

С ростом терминологической сети увеличивается количество интегрированных в нее терминосистем, а у пользователя возникает парадоксальная, на первый взгляд, проблема потери ориентации, вызванная с нерасчлененностью сети на крупные фрагменты-кластеры. Заметим, что кластеризация имеет смысл и для всей терминологической сети, и для ее отдельных частей. Фактически кластеризация всей терминологической сети сводится к восстановлению составляющих ее терминосистем.

1. Подход к кластеризации

Кластеризацию терминологической сети предлагается разделить на два последовательных этапа. На первом этапе строится подмножество понятийных узлов W , именуемых кандидатами в центры кластеров. На втором этапе некоторые кандидаты из рассмотрения исключаются, а оставшиеся в множестве W центры порождают искомые кластеры.

Терминологический кластер (далее просто кластер) с центром A есть множество $K(A | W)$, состоящее из самого узла A , а также из других узлов B , отличных от центров кластеров, но соединенных с A путем из выделенных дуг (см. раздел 2).

Каждый центр A однозначно определяет множество подчиненных ему центров $S(A | W)$. По определению множество $S(A | W)$ составляют узлы B из $W \setminus \{A\}$, соединенные выделенной дугой (A, B) с некоторым узлом X из $K(A | W)$.

Количество узлов кластера $K(A | W)$ будем обозначать $k(A | W)$. Единственным параметром кластеризации является целое число MiN – минимально допустимое количество узлов в кластере.

Если W – множество кандидатов в центры кластеров, то на втором этапе кластеризации для исключения избыточных центров применяются два правила.

Правило 1.1 Исключить B из W , если (а) $k(B | W) < MiN$ и (б) $S(B | W) = \emptyset$.

Правило 1.2 Исключить B из W , если (а) $k(B | W) < MiN$ и (б) для всех узлов A из $S(B | W)$ выполняется неравенство $MiN \leq k(A | W)$.

В результате применения каждого правила множество W изменяется: $W \rightarrow W_{new}$, что порождает необходимость перевычислять после каждого применения кластеры $K(A | W_{new})$ и подчиненные центры $S(A | W_{new})$.

При анализе терминологических сетей существенно используются специальные отношения между терминами, узлами и дугами. Приведем эти отношения.

Во-первых, будем говорить, что (многословный) термин x подчинен (многословному) термину y , если термин x является развитием термина y . Примерами отношения подчиненности являются следующие пары терминов:

x = “Промышленные аварии” и y = “Аварии”;
 x = “Централизованная библиотечная система” и y = “Библиотечные системы”;
 x = “Скорость света в вакууме” и y = “Вакуум”.

Отношение подчиненности позволяет выделить в терминологической сети собственный подкласс дуг, отвечающих синтаксическому способу терминообразования [Гринев-Гриневиц, 2008].

С формальной точки зрения термин x , состоящий из слов x_1, x_2, \dots, x_g , подчинен термину y , состоящему из слов y_1, y_2, \dots, y_h , если существует однозначная функция

$$f : \{y_1, y_2, \dots, y_h\} \rightarrow \{x_1, x_2, \dots, x_g\}$$

такая, что для всех $i = 1, \dots, h$ слова y_i и $f(y_i)$ отличаются формальными суффиксами.

Во-вторых, будем говорить, что дуга (A, B) является терминологически связанной, если имя узла A подчинено имени узла B . Из общего количества дуг, связывающих понятийные узлы, терминологически связанные дуги составляют 20%. Типичным примером дуги, не удовлетворяющей условию терминологической связанности, является дуга (A, B) , в которой узлы A и B именуются “Акции” и “Ценные бумаги”.

В-третьих, будем называть модельной диаграммой подсеть терминологической сети составленную из двух путей, не имеющих общих узлов, за исключением общего начала и общего конца. Общий вид модельной диаграммы представлен на рисунке 1.

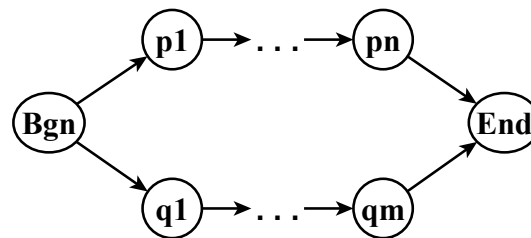


Рисунок 1 – Модельная диаграмма <n,m>

Сложность модельной диаграммы есть пара целых чисел $\langle n, m \rangle$, где n – количество внутренних узлов одного пути, а m – количество узлов второго пути, причем в такой паре всегда выполняется неравенство $n \leq m$. Из двух оценок сложности $\langle n, m \rangle$ и $\langle a, b \rangle$ оценка $\langle n, m \rangle$ считается меньшей, если $n + m < a + b$ или $n + m = a + b$, но $n < a$.

В связи с отсутствием в терминологических сетях кратных ребер, наименьшая оценка сложности модельных диаграмм есть величина $\langle 0, 1 \rangle$ (рисунок 2а). А оценка, непосредственно предшествующая минимуму, есть $\langle 1, 1 \rangle$ (рисунок 2б). Модельные диаграммы позволяют ввести оценки структурной сложности для дуг.

В-четвертых, будем называть структурной сложностью дуги минимальную сложность модельных диаграмм, содержащих эту дугу. Если дуга не входит ни в одну модельную диаграмму, то ее структурная сложность полагается равной $\langle N, N \rangle$, где N – общее количество узлов терминологической сети. Очевидно, что структурная сложность каждой из трех дуг модельной диаграммы $\langle 0, 1 \rangle$ есть величина $\langle 0, 1 \rangle$.

2. Выделенные дуги

Определенные сложности при кластеризации вызывают понятийные узлы, имеющие две и более исходящих дуг. При определенных обстоятельствах такой узел и все его потомки неоднократно попадают в различные кластеры, что негативно сказывается на структурных связях между кластерами. По этой причине для целей кластеризации все дуги терминологической сети подразделяются на выделенные и прочие. По определению:

- если узел имеет единственную исходящую дугу, то такая дуга является выделенной;
- если узел B имеет несколько исходящих дуг, то выделенная дуга выбирается из исходящих применением правил 2.1–2.4.

Правило 2.1 При выборе выделенной дуги отдать предпочтение терминологически связанным дугам, если таковые имеются.

Правило 2.2 При выборе выделенной дуги отдать предпочтение дугам минимальной структурной сложности.

Правило 2.3 При выборе выделенной дуги отдать предпочтение дугам (B, X) , если узел X является кандидатом в центры кластеров – элементом множества W , и если такие дуги имеются.

Правило 2.4 При выборе выделенной дуги отдать предпочтение родовидовым связям, то есть дугам $(B, X)_p$, если таковые имеются.

Правила 2.1–2.4 устроены таким образом, что они сокращают число исходящих дуг, претендующих стать выделенными. В худшем случае, когда правило не находит предпочтительных дуг, состав претендентов не изменяется. Для практического использования правил необходимо установить порядок их применения.

3. Центры кластеров

Исследования по терминоведению, а также анализ терминологических сетей выявили некоторое количество свойств-закономерностей присущих терминосистемам [Шелов, 2003], [Мальковский и др., 2013]. При обнаружении центров кластеров свойства терминосистем используются в иной роли – в роли правил обнаружения терминосистем. Перемена местами посылок и следствий не проходит бесследно. Построенные из закономерностей правила кластеризации порождают определенное количество ложных центров, а в некоторых случаях вообще не позволяют выявить имеющийся (истинный) центр кластера. В связи с этим для построения множества W кандидатов в центры искомым кластерам предлагаются несколько взаимодополняющих правил.

Правило 3.1 Квалифицировать узел X как возможный центр кластера, если в терминологической сети существуют терминологически дуги (B, X) .

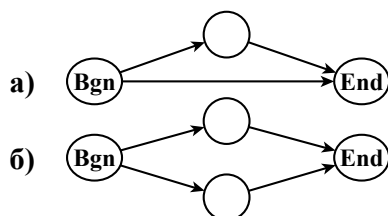


Рисунок 2 – Модельные диаграммы $\langle 0,1 \rangle$ и $\langle 1,1 \rangle$

Правило 3.2 Квалифицировать узел X как возможный центр кластера, если X является конечным узлом End в некоторой модельной диаграмме структурной сложности $\langle 0,1 \rangle$ (см. рисунок 2а).

Правило 3.3 Квалифицировать узел X как возможный центр кластера, если X является

концевым узлом End в некоторой модельной диаграмме структурной сложности $\langle 1,1 \rangle$ (см. рисунок 2б).

Правило 3.4 Квалифицировать узел X как возможный центр кластера, если в терминологической сети найдутся по крайней мере три дуги $(A, X)_p$, $(B, X)_p$ и (C, X) .

Правило 3.5 Квалифицировать узел X как возможный центр кластера, если X не имеет исходящих дуг.

Правила 3.1–3.5 набирают кандидатов в центры кластеров. Следующие два правила отбраковывают заведомо непригодных кандидатов.

Правило 3.6 Исключить узел X из состава кандидатов в центры кластеров, если в X имеет три и менее потомков.

Правило 3.7 Исключить узел X из состава кандидатов в центры кластеров, если в терминологической сети найдутся – см. рисунок 3 – три дуги (A, X) , (B, X) и (X, C) такие, что
 $\Rightarrow A, B, C$ – понятийные узлы;
 \Rightarrow дуга (X, C) не является терминологически связанной; однако
 \Rightarrow имя узла A подчинено имени узла C ;
 \Rightarrow имя узла B подчинено имени узла C .

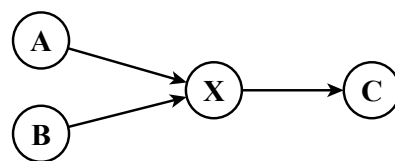


Рисунок 3 – Терминологически связанный фрагмент

Последнее правило применимо, например, в ситуации, когда
имя A есть “Атмосферное давление”,
имя B есть “Атмосферные осадки”,
имя X есть “Метеорологические элементы”,
имя C есть “Атмосфера”.
Здесь понятие X “блокирует” синтаксические связи между терминами A и C , а также между B и C , хотя A , B и C несомненно принадлежат одному терминологическому кластеру.

4. Алгоритм кластеризации УТП

Любой алгоритм кластеризации, построенный на базе правил кластеризации, реализует тот или иной порядок их выполнения. Работоспособности предложенного подхода подтверждается алгоритмом кластеризации терминологической сети УТП [Мальковский и др., 2012], насчитывающей около 10 тысяч понятийных узлов. В качестве исходных данных алгоритм использует собственно УТП и целочисленный параметр Min . Результатом работы алгоритма является набор терминологических кластеров. Двухэтапная организация вычислений имеет вид:

Этап 1. Последовательно построить:

- множество W_1 с помощью правила 3.1;
- множество W_2 с помощью правила 3.2;
- множество $W_{12} = W_1 \cup W_2$;
- множество W_3 с помощью правила 3.3;
- множество W_4 с помощью правила 3.4;
- множество $W_{34} = W_3 \cup W_4$;
- множество W_5 с помощью правила 3.5;
- множество W_6 с помощью правила 3.6;
- множество W_7 с помощью правила 3.7;
- множество кандидатов в центры кластеров

$$W = ((W_{12} \cap W_{34}) \cup W_5) \setminus (W_6 \cup W_7).$$

Этап 2. Последовательно выполнить действия:

- исключить из W часть кандидатов в центры кластеров посредством правила 1.1;
- исключить из W часть кандидатов в центры кластеров посредством правила 1.2;
- для каждого центра B , сохранившегося в W , построить терминологические кластеры $K(B|W)$.

Процедура построения выделенных дуг, неявно задействованная на втором этапе, последовательно применяет к набору исходящих дуг правило 2.1, правило 2.2, правило 2.3 и, наконец, правило 2.4. Искомая выделенная дуга считается построенной, если после применения очередного правила множество исходящих дуг сократилось до одной дуги.

Метод построения терминологической сети УТП позволяет (хотя и с оговорками) проследить происхождение терминов, а значит, позволяет выявить истинные кластеры, пригодные для проверки результатов кластеризации.

Множества W_{12} и W_{34} , построенные на первом этапе, содержат практически идентичные подмножества центров истинных кластеров, однако сильно различаются в части ложных центров. По этой причине их пересечение, фигурирующее в окончательном вычислении W , позволяет избавиться от значительного количества (от 50%) ложных центров.

По результатам проверки алгоритма кластеризации УТП установлено, что при $Min = 19$ подтверждаются около 90% истинных кластеров, а остальные 10% кластеров нуждаются в дополнительном анализе.

Заключение

Важнейшей особенностью описанных правил кластеризации является их интерпретируемость, что позволяет создавать алгоритмы кластеризации с заданными свойствами.

Вообще говоря, особенности кластеризации существенно зависят от выбора конкретной терминологической сети. Вместе с тем, подход к кластеризации через постулирование закономерностей позволяет надеяться, что однажды построенный алгоритм будет вполне устойчив к изменениям сети. По этой причине разработка

универсального алгоритма кластеризации представляется необязательной.

Библиографический список

- [Гринев-Гриневич, 2008] Терминоведение / С.В.Гринев-Гриневич – М.: Академия, 2008.
- [Мальковский и др., 2012] Мальковский М.Г., Терминологические сети / М.Г.Мальковский, С.Ю.Соловьев // OSTIS-2012. Материалы конференции. С. 77-82
- [Мальковский и др., 2013] Мальковский М.Г., Исследование родовидовых отношений в терминологических сетях / М.Г.Мальковский, С.Ю.Соловьев // OSTIS-2013. Материалы конференции. С. 147-152
- [Шелов, 2003] Термин. Терминологичность. Терминологические определения / С.Д.Шелов – СПб.: Филологический факультет СПбГУ, 2003.

RULES FOR TERMINOLOGICAL CLUSTERS CREATIONS

Malkovsky M.G., Soloviev S.Y.

Lomonosov MSU, Moscow, Russia

malk@cs.msu.su

soloviev@glossary.ru

In work rules of registration of articles on conference OSTIS (Open Semantic Technologies for Intelligent Systems) are resulted. The Organizing Committee recommends to use the given file as a template for registration of articles

Introduction

We consider the problem of constructing clusters in terminological networks. Meaningful clusters help the user to quickly navigate through the network.

Main Part

We propose a method of clustering that consists of two steps. At the first step we select candidates for cluster centers. In a second step we remove false centers and build clusters.

Three groups of rules form the basis of clustering. Rules sifting false clusters are included in the first group. Conflict resolution rules for outgoing arcs are included in the second group. The rules for selection of candidates for the cluster centers are included in the third group.

Clustering algorithm is implemented to test the method. Results of calculations confirmed the high quality of clustering.

Conclusion

In general, clustering terminological network is a method of scientific research, and the development of a complete clustering algorithm completely optional. Moreover, the advantage is that you can connect the new rules.



УДК 004.822:514

НАПОЛНЕНИЕ СЕМАНТИЧЕСКИХ СЛОТОВ РЕЛЯЦИОННО-СИТУАЦИОННОГО ФРЕЙМА НА ПРИМЕРЕ ТАТАРСКИХ СИНТАКСЕМ

Сулейманов Д.Ш., Гатиатуллин А.Р.

*НИИ «Прикладная семиотика» Академии наук Республики Татарстан,
Казанский (Приволжский) Федеральный университет, г. Казань, Россия*

dvdt.slt@gmail.com

agat1972@mail.ru

Эта работа является продолжением статей Д.Ш.Сулейманова, А.Р. Гатиатуллина с описанием реляционно-ситуационной системы, которая используется авторами для описания семантических аспектов татарских морфем и синтаксем. Здесь описывается новый способ представления реляционно-ситуационной системы в виде совокупности фреймов, представляющих собой сценарии, описывающие некие универсальные семантические ситуации, и названные авторами реляционно-ситуационными фреймами (РСФ). В данной статье наполнение РСФ иллюстрируется примерами из татарского языка.

Ключевые слова: реляционно-ситуационная система; семантические универсалии, синтаксемы, реляционно-ситуационный фрейм.

Введение

В работах [Сулейманов и др., 2012] [Сулейманов и др., 2013] описаны фрагменты реляционно-ситуационной системы, которая состоит из совокупности семантических сценариев, называемых нами семантическими универсалиями. Эти семантические сценарии представляют собой разновидность неоднородных семантических сетей [Осипов, 1997]. В рамках этой модели, текст предложения представляется в виде ситуации, с определенным набором участников и отношений между ними.

В указанных работах представлено использование реляционно-ситуационной системы для описания семантического аспекта структурно-функциональной модели татарской аффиксальной морфемы [Сулейманов и др., 2003].

Кроме модели татарской морфемы данную систему можно использовать для решения множества других задач компьютерной обработки татарских текстов. Это задачи многоязычного интеллектуального поиска, машинного перевода, решения проблемы снятия неоднозначности и целого ряда других задач, в которых требуются элементы семантико-синтаксического анализа.

В данной работе описывается новая форма представления реляционно-ситуационной системы в виде совокупности реляционно-ситуационного

фреймов. Подобная форма представления семантических сценариев выбрана с целью придания реляционно-ситуационной системе большей строгости, упорядоченности и конструктивности, соответственно, технологичности, и дальнейшей автоматизации обработки информации с использованием современных средств управления базами данных и знаний.

Таким образом, в настоящее время исследования ведутся в двух направлениях. С одной стороны, разрабатываются РСФ, являющиеся в большей степени языко-независимыми, представляя собой семантические универсалии, отражающие практически глубинную ментальность здравого смысла в определенном контексте, с другой стороны, разработка концептуально-функциональной модели татарских синтаксем, в которой для описания семантического аспекта используется модель реляционно-ситуационных фреймов. Синтаксемы используются для заполнения слотов РСФ в соответствии с их семантико-синтаксическими характеристиками.

1. Модель татарской синтаксемы

В задачах автоматической обработки текста в полном объеме предполагается, как правило, реализация этапов морфологического, синтаксического, семантического и

прагматического анализа. Семантика рассматривается, зачастую, как надстройка над синтаксисом и, соответственно, семантического анализа следует за синтаксическим анализом. В работах по теории коммуникативной грамматики Золотовой [Золотова, 2004] предполагается тесная связь синтаксиса и семантики в языке, их взаимодействие при выполнении коммуникативной функции языка. В ее работах приводится описание языковых единиц, называемых синтаксемами, которые обладают свойствами, характеризующими взаимодействие морфологических, семантических и функциональных признаков.

В работах Золотовой описываются синтаксеммы русского языка. Нами предлагается использовать синтаксеммы для описания семантико-синтаксических структур тюркских предложений. Однако, учитывая языковые особенности тюркских языков, особенно богатую морфологическую структуру тюркских словоформ, синтаксеммы в тюркских языках также будут иметь структуру отличную от синтаксеммы русского языка.

Далее описывается модель татарской синтаксеммы со следующей структурой:

1. **Номер синтаксеммы**
2. **Основное слово синтаксеммы**
 - a. Синтаксический тип
 - b. Морфологический тип
 - c. Морфемная структура
 - d. Аналитическая форма
3. **Главное слово**
 - a. Морфологический тип
 - b. Морфологическая структура
4. **Зависимое слово**
 - a. Морфологический тип
 - b. Морфологическая структура
5. **Значение**
 - a. Тип ситуации
 - b. Роль синтаксеммы

Рассмотрим элементы этой структуры.

Каждому типу синтаксеммы присваивается свой уникальный номер, который указывается в параметре – **Номер синтаксеммы**.

Основное слово синтаксеммы – слово, которое играет определенную роль в реляционно-ситуационной системе и заполняет некий слот в соответствующем РСФ.

Синтаксеммы в рассматриваемой модели по своей структуре могут быть образованы как отдельными словоформами, так и сочетанием из нескольких словоформ. Синтаксеммы из нескольких словоформ – это несвободные словосочетания, составляющие единство с точки зрения семантики, которые в зарубежной литературе обозначаются как многословные выражения – MultiwordExpression, или многословные конструкции.

Параметр **Синтаксический тип** используется для описания признака – является ли синтаксема однословной или многословной конструкцией. Для однословной синтаксеммы этот параметр имеет значение W, для аналитической формы- AF, а для аналитической конструкции- АК.

В описываемой модели синтаксеммы “Аналитические формы” – это все конструкции, в которых основа первой словоформы не фиксирована в отношении к остальным словоформам конструкции. Это означает, что на это слово накладываются не лексические, а только морфологические ограничения. Например, то что словоформа слева должна иметь морфему направительного падежа –ГА.

Поверхностная форма: *урманга кадәр, бакчага кадәр*

Морфемная структура: [N]+ГА кадәр.

В модели татарской синтаксеммы эта конструкция будет заполнять следующие параметры:

1. **Основное слово синтаксеммы:** *
- 1.1. **Синтаксический тип:**AF
- 1.2. **Морфологический тип:**N
- 1.3. **Морфемная структура:**-ГА; -ЛАР-ГА; -ЫМ-ГА; -Ың-ГА; -сы-ГА; -Ыбыз-ГА; -Ыгыз-ГА; -ЛАр-Ым-ГА; -ЛАр-Ың-ГА; -ЛАр-Ыбыз-ГА; -ЛАр-Ыгыз-ГА
- 1.4. **Аналитическая форма:** кадәр.

Символ „*” означает, что для рассматриваемой синтаксеммы с данным значением в параметре «Синтаксический тип» в позиции «Основное слово синтаксеммы» может находиться словоформа с любой основой и типом, указанным в параметре 2.2.

Параметр “Морфологический тип” определяет, какого типа аффиксальные морфемы могут присоединяться к этому типу основы, и этот параметр может принимать одно из следующих значений: N, A, V, S.

Здесь N – показатель основ, к которым могут присоединяться аффиксы имен существительных;

A – показатель типа основ, к которым могут присоединяться аффиксы имен существительных, имен прилагательных и наречий;

V – показатель типа основ, к которым могут присоединяться аффиксы глаголов;

S – показатель типа основ, к которым могут присоединяться только модальные аффиксы. Для татарского языка это аффиксальные морфемы –мы, -мыни, -дыр.

Параметр “Морфемная структура” определяет набор аффиксальных морфем, при наличии которых эта синтаксема будет выражать конкретную роль в заданной ситуации. Если в этой

словоформе синтаксемы нет аффиксальных морфем, то это поле остается пустым.

Параметр **2.4 Аналитическая форма** содержит вспомогательные слова этой конструкции, в данном примере это послелог *кадәр*.

Рассмотрим параметр “**Главное слово**”. Этот параметр показывает, к какой лексико-семантической группе должно относиться слово, с которым основное слово данной синтаксемы находится в отношении синтаксической зависимости. В Институте прикладной семиотики Академии наук РТ в настоящее время ведется разработка базы данных с иерархической системой лексико-семантических групп глаголов татарского языка.

В нашей модели лексико-семантическая группа (ЛСГ) – это группа слов одной части речи, объединенная одним словом-идентификатором или устойчивым словосочетанием, значение которого полностью входит в значение остальных слов группы и которое может заменять остальные слова в некоторых контекстах.

Сходство слов, входящих в одну ЛСГ, не исчерпывается только наличием у них общих лексических признаков, а также проявляется в наличии типовой сочетаемости, связанности с определенными структурно-семантическими моделями предложений, в закономерностях соотношений парадигматики внутри группы, в единстве словообразовательных характеристик, с направлением развития многозначности, которая часто носит регулярный характер, и т.д.

Для главного слова также указывается его морфологический тип, который может принимать значения, аналогичные морфологическому типу основного слова. Морфологическая структура содержит список аффиксальных морфем, которые должны быть у главного слова. Так, например, к такому типу аффиксальных морфем глагольных основ относятся залоговые аффиксы.

Возможны случаи, когда основное слово в данной морфологической форме выражает определенное значение только при наличии некоторого зависимого слова. Для таких случаев в модели татарской синтаксемы нами предусмотрен параметр – Зависимое слово.

Приведем пример заполнения базы данных для синтаксем, основным словом в которых является аналитическая форма: *кыяфәтенә карата матур ‘по внешности красив’*.

1. Номер синтаксемы: 22.
2. Основное слово: *
 - a. Синтаксический тип: W
 - b. Морфологический тип: N
 - c. Морфемная структура: ГА
 - d. Аналитическая форма: карата
3. Главное слово:
 - a. Морфологический тип: A

- b. Морфологическая структура: *
4. Зависимое слово
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение
 - a. Тип ситуации: quality_object
 - b. Роль синтаксемы: focus

Кроме описания структуры и контекста использования синтаксемы необходимо определить значение, выражаемое этой синтаксемой. Для этих целей предусмотрен разрабатываемый авторами реляционно-ситуационный фрейм.

2. Реляционно-ситуационный фрейм

Реляционно-ситуационный фрейм представляет собой реализацию типовой ситуации, состоящей из названия ситуации и набора слотов, которые являются ролями конструктивных элементов этой ситуации. Заполнение слотов производится конкретными татарскими синтаксемами, описанными выше в данной работе. РСФ общего вида имеет следующее представление:

```
SituationSi
  Role1: SintaxemI1;
  Role2: SintaxemI2;
  ...
  Role 3: SintaxemIN;
End_Situation
```

Выбор конкретной ситуации, соответственно, определяет и выбор конкретного типа РСФ, называемого базовым фреймом, с соответствующими слотами-ролями, заполненными определенными значениями-синтаксемами.

Например, базовый РСФ для ситуации, выражающей действие по пространственному перемещению, будет иметь следующий вид:

```
Situation 7.3: action_local
  Object:      Sintaxem 5;
  Old_local:   Sintaxem121;
  New_local:   Sintaxem119, 120;
  Direction:   Sintaxem118;
  Route:       Sintaxem124, 125, 126;
  Interval:    Sintaxem129;
  Time:        Sintaxem78;
  Period:      Sintaxem97;
  Instrument:  Sintaxem99;
End_Situation
```

Авторами статьи проведена семантическая классификация контекстов в зависимости от типов отношений, участвующих в формировании глубинного смысла данного контекста. На основе этой классификации разработана система, состоящая из 60 базовых реляционно-ситуационных фреймов.

Далее рассмотрим пример заполнения РСФ Situation 7.3: action_local конкретными синтаксемами в качестве слотов.

Синтаксема5 как значение слота Object.

1. Номер синтаксемы: 5
2. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: W
 - b. Морфологический тип: N
 - c. Морфемная структура: „“
 - d. Аналитическая форма: „“
3. Главное слово: ЛСГ („действия“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: -
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение
 - a. Тип ситуации: action
 - b. Роль синтаксемы: subject

Синтаксема121 как значение слота Old_local:

1. Номер синтаксемы: 121
2. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: W
 - b. Морфологический тип: N
 - c. Морфемная структура: -ДАН; -ЛАР-ДАН; -ЫМ-ДАН; -ЫЦ-ДАН; -СЫ-ДАН; -ЫБЫЗ-ДАН; -ЫГЫЗ-ДАН; -ЛАР-ЫМ-ДАН; -ЛАР-ЫЦ-ДАН; -ЛАР-ЫБЫЗ-ДАН; -ЛАР-ЫГЫЗ-ДАН
 - d. Аналитическая форма: „“
3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: -
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение
 - a. Тип ситуации: action_local
 - b. Роль синтаксемы: old_local

Синтаксемы119 и 120 как значения слота New_local:

1. Номер синтаксемы: 119
2. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: W
 - b. Морфологический тип: N
 - c. Морфемная структура: -ГА; -ЛАР-ГА; -ЫМ-ГА; -ЫЦ-ГА; -СЫ-ГА; -ЫБЫЗ-ГА; -ЫГЫЗ-ГА; -ЛАР-ЫМ-ГА; -ЛАР-ЫЦ-ГА; -ЛАР-ЫБЫЗ-ГА; -ЛАР-ЫГЫЗ-ГА
 - d. Аналитическая форма: „“
3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: -
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение
 - a. Тип ситуации: action_local

b. Роль синтаксемы: new_local

1. Номер синтаксемы: 120
2. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: AF
 - b. Морфологический тип: N
 - c. Морфемная структура: -ГА; -ЛАР-ГА; -ЫМ-ГА; -ЫЦ-ГА; -СЫ-ГА; -ЫБЫЗ-ГА; -ЫГЫЗ-ГА; -ЛАР-ЫМ-ГА; -ЛАР-ЫЦ-ГА; -ЛАР-ЫБЫЗ-ГА; -ЛАР-ЫГЫЗ-ГА
 - d. Аналитическая форма: кадр
3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: -
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение
 - a. Тип ситуации: action_local
 - b. Роль синтаксемы: new_local

Синтаксема 118 как значение слота Direction:

1. Номер синтаксемы: 118
2. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: AF
 - b. Морфологический тип: N
 - c. Морфемная структура: -ГА; -ЛАР-ГА; -ЫМ-ГА; -ЫЦ-ГА; -СЫ-ГА; -ЫБЫЗ-ГА; -ЫГЫЗ-ГА; -ЛАР-ЫМ-ГА; -ЛАР-ЫЦ-ГА; -ЛАР-ЫБЫЗ-ГА; -ЛАР-ЫГЫЗ-ГА
 - d. Аналитическая форма: таба
3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: -
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение
 - a. Тип ситуации: action_local
 - b. Роль синтаксемы: direction

Синтаксемы 124 и 125 как значение слота Route:

1. Номер синтаксемы: 124
3. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: AF
 - b. Морфологический тип: N
 - c. Морфемная структура: -ЛАР; -ЫМ; -ЫЦ; -СЫ; -ЫБЫЗ; -ЫГЫЗ; -ЛАР-ЫМ; -ЛАР-ЫЦ; -ЛАР-ЫБЫЗ; -ЛАР-ЫГЫЗ
 - d. Аналитическая форма: аша
3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: -
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение

- a. Тип ситуации: action_local
 - b. Роль синтаксемы: route
1. Номер синтаксемы: 125
 2. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: AF
 - b. Морфологический тип: N
 - c. Морфемная структура: -ЛАР; -Ым; -Ың; -сЫ; -ЫбЫз; -ЫгЫз; -ЛАР-Ым; -ЛАР-Ың; -ЛАР-ЫбЫз; -ЛАР-ЫгЫз
 - d. Аналитическая форма: аркылы
 3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
 4. Зависимое слово: -
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
 5. Значение
 - a. Тип ситуации: action_local
 - b. Роль синтаксемы: route

1. Номер синтаксемы: 126
2. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: AF
 - b. Морфологический тип: N
 - c. Морфемная структура: -ЛАР
 - d. Аналитическая форма: буйлап
3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: -
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение
 - a. Тип ситуации: action_local
 - b. Роль синтаксемы: route

Как видно из структуры РСФ и синтаксем с номерами 124, 125, 126, один и тот же слот могут заполнять любая из приведенных в списке синтаксем. Например:

Синтаксема 124, описывает конструкции с послелогом *аша‘через’*: *урман аша бара ‘идет через лес’*.

Синтаксема 125, описывает конструкции с послелогом *аркылы‘через’*: *урман аркылы бара ‘идет через лес’*.

Синтаксема 126, описывает конструкции с послелогом *буйлап‘вдоль’*: *юлбуйлап бара ‘идет вдоль дороги’*.

Синтаксема как значение слота Interval:

1. Номер синтаксемы: 129
2. Основное слово: ЛСГ („меры пространства“)
 - a. Синтаксический тип: W
 - b. Морфологический тип: N
 - c. Морфемная структура: -нЫ
 - d. Аналитическая форма: „“

3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: ЛСГ („число“)
 - a. Морфологический тип: N
 - b. Морфологическая структура: „“
5. Значение
 - a. Тип ситуации: action_local
 - b. Роль синтаксемы: interval

Синтаксема 129 как значение слота Instrument:

1. Номер синтаксемы: 129
2. Основное слово: ЛСГ („физические объекты“)
 - a. Синтаксический тип: W
 - b. Морфологический тип: N
 - c. Морфемная структура: ЛАР; -Ым; -Ың; -сЫ; -ЫбЫз; -ЫгЫз; -ЛАР-Ым; -ЛАР-Ың; -ЛАР-ЫбЫз; -ЛАР-ЫгЫз
 - d. Аналитическая форма: белән
3. Главное слово: ЛСГ („перемещение“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: ЛСГ („число“)
 - a. Морфологический тип: -
 - b. Морфологическая структура: -
5. Значение
 - a. Тип ситуации: action_local
 - b. Роль синтаксемы: instrument

Синтаксемы как значение слота 97 и 98 Period:

1. Номер синтаксемы: 97
2. Основное слово: ЛСГ („единицы времени“)
 - a. Синтаксический тип: AF
 - b. Морфологический тип: N
 - c. Морфемная структура: „“
 - d. Аналитическая форма: буе
3. Главное слово: ЛСГ („действие“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: ЛСГ („число“)
 - a. Морфологический тип: N
 - b. Морфологическая структура: „“
5. Значение
 - a. Тип ситуации: action
 - b. Роль синтаксемы: period

1. Номер синтаксемы: 98
2. Основное слово: ЛСГ („единицы времени“)
 - a. Синтаксический тип: W
 - b. Морфологический тип: N
 - c. Морфемная структура: -ДА
 - d. Аналитическая форма: „“
3. Главное слово: ЛСГ („действие“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: ЛСГ („число“)
 - a. Морфологический тип: N
 - b. Морфологическая структура: „“
5. Значение
 - a. Тип ситуации: action
 - b. Роль синтаксемы: period

В структуре СРФ показано, что слот Period также могут заполнять несколько типов синтаксем. Рассмотрим примеры с экземплярами этих синтаксем.

Синтаксема 97 – ел буе бара „идет в течение года“.

Синтаксема 98 – өч елда барды „шел три года“.

Синтаксема 78 как значение слота Time:

1. Номер синтаксемы: 78
2. Основное слово: ЛСГ („моменты времени“)
 - a. Синтаксический тип: W
 - b. Морфологический тип: N
 - c. Морфемная структура: „“
 - d. Аналитическая форма: „“
3. Главное слово: ЛСГ („действие“)
 - a. Морфологический тип: V
 - b. Морфологическая структура: *
4. Зависимое слово: „“
 - a. Морфологический тип: N
 - b. Морфологическая структура: „“
5. Значение
 - a. Тип ситуации: action
 - b. Роль синтаксемы: time

Представление текста в виде РСФ позволяет использовать технологии для работы с фреймами и находить смысловую близость двух разных текстов. Эта особенность используется как при интеллектуальном поиске, так и как механизм снятия неоднозначностей в задачах машинного перевода.

Заключение

В данной работе предложены реляционно-ситуационные фреймы как механизм для реализации семантико-синтаксических моделей анализа татарского простого предложения. Очевидно, практическая реализация семантико-синтаксического анализатора татарского предложения на базе этих моделей требует подготовки целого ряда лингвистических ресурсов. В их число входят базы данных с иерархическими классификациями Лексико-семантических групп, с семантическими сценариями, словарей синтаксем, а также словарей семантических ролей глаголов. На данный момент для татарского языка этих ресурсов не существует, или они находятся на стадии разработки.

Поддержан грантом РФФИ («Разработка комплексных моделей данных на основе ситуационного анализа текстов в задачах многоязычного поиска»), проект № 13-07-00494.

Библиографический список

[Сулейманов и др.(1), 2012] Сулейманов Д.Ш., Гатиатуллин А.Р. Семантические универсалии в системе описания значений морфем // Сборник Трудов конференции «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2011), Минск: Издательство БГУИР, 2012. – С.391-396.

[Сулейманов и др., 2013] Сулейманов Д.Ш., Гатиатуллин А.Р., Вагапов Д.Р. Семантические универсалии в системе описания значений морфем // Сборник Трудов конференции «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2013), Минск: Издательство БГУИР, 2013. – С.391-396.

[Осипов, 1997] Осипов Г.С. Приобретение знаний интеллектуальными системами: Основы теории и технологии. – М.: Наука, Физматлит, 1997.

[Сулейманов и др., 2003] Сулейманов Д.Ш., Гатиатуллин А.Р. Структурно-функциональная модель татарских морфем. Казань, 2003. 212 с.

[Золотова, 2004] Золотова Г.А., Ониненко Н. К., Сидорова М. Ю. Коммуникативная грамматика русского языка. Институт русского языка РАН им. В. В. Виноградова, М. 2004 – 544 с.

FILLING OF SEMANTIC SLOTS IN THE RELATIONAL-SITUATIONAL FRAME ON THE EXAMPLE OF TATAR SINTAXEMES

Suleymanov D. Sh., Gatiatullin A. R.

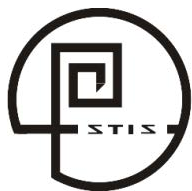
*Scientific Research Institute "Applied Semiotics"
Academy of Sciences of the Republic of Tatarstan,
Kazan Federal University
Kazan, Russia*

dvdt.slt@gmail.com

agat1972@mail.ru

This work is a continuation of the articles by D. Suleymanov and A. Gatiatullin, which describe a relational-situational system that is used by the authors to convey the semantic aspects of Tatar morphemes and syntaxemes. It presents a new way of displaying the relational-situational system as a set of frames representing scenarios, which describe some universal semantic situation s and which are named by the authors as relational-situational frames (RSF). In this article, the filling of RSF is illustrated on the examples from the Tatar language.

Key words: relational-situational system, semantic universals, syntaxeme, relational-situational frame.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

СЕМАНТИЧЕСКИЕ ТЕХНОЛОГИИ ДЛЯ МАТЕМАТИЧЕСКИХ ЭЛЕКТРОННЫХ РЕСУРСОВ

Невзорова О.А.

*Научно-исследовательский институт «Прикладная семиотика» АН Республики Татарстан,
Казанский (Приволжский) федеральный университет,
г. Казань, Россия
onevzoro@gmail.com*

В статье приведен обзор семантических технологий, разработанных для представления и обработки математических знаний на крупных мировых порталах, а также оригинальные результаты проекта семантической публикации документов электронной коллекции «Известия ВУЗов. Математика».

Ключевые слова: связывание данных; пространство открытых связанных данных.

Введение

Концепция семантического веба предполагает семантическое структурирование пространства Интернет-данных в целях использования его различными программными агентами. Основными прагматическими задачами текущего этапа развития семантического веба является задача унификации (совместимости данных) данных, а также задача связывания данных из разных источников данных.

Проект Linked Open Data¹ (LOD) является наиболее значимым по результатам примером принятия и применения принципов Linked Data (Связанные данные). Проект LOD стартовал в январе 2007 года и был поддержан Информационно-образовательной группой Семантической Паутины Консорциума W3C. В настоящее время в проекте участвуют не только исследователи и разработчики университетских научно-исследовательских лабораторий, но и ряд крупных организаций, таких как BBC, агентство Рейтер и библиотеки Конгресса США. Успех проекта, скорее всего, обусловлен его открытостью и актуальностью. На сегодняшний день технологии Linked Data все более широко используются производителями первичных данных. Ссылки RDF соединяют данные из различных источников в единый глобальный граф RDF и позволяют браузерам и поисковым роботам Связанных данных перемещаться между источниками данных.

Технологии Linked Data применяются для

совместного использования данных, относящихся к широкому спектру различных тематических доменов, а также данных, относящихся к различным доменам. Характерным примером и прототипом междоменных Связанных данных является DBpedia – набор данных, автоматически извлекаемый из общедоступного ресурса Википедия.

Разработанные решения проекта LOD и связанные с ним семантические технологии позволяют определить новое направление – семантическая публикация (semantic publishing) данных. Однако наибольшие успехи в этом направлении связаны с именно с публикацией структурированных данных, а не текстов (неструктурированных данных). Следующий шаг, несомненно, должен быть сделан в области развития семантических технологий обработки неструктурированных текстов.

Семантическая публикация предполагает увеличение числа вычислительных компонент текста, т.е. извлечения из текста компонент определенной семантики для последующей обработки (в том числе связывания). Математические тексты выгодно отличаются наличием стандартов и языков разметки, высокой структурированностью, наличием электронных математических библиотек, а также программных средств для автоматической обработки отдельных компонентов математических текстов.

В настоящей статье будут рассмотрены некоторые ключевые решения и крупные проекты, связанные с рассматриваемой задачей семантической публикации математических текстов.

¹ <http://lod2.eu>

1. Проект Mizar

Цель проекта Mizar, стартовавшего в 1973 году, была связана с задачами компьютеризации в области профессиональной математики. С 1989 года в рамках международного сотрудничества (Университет Shinshu, Nagano и Университет Alberta, Edmonton) разрабатывается система Mizar, которая поддерживает базу данных по математике, включающую более 9400 определений математических концептов, а также свыше 49000 теорем. Система Mizar использует формальный язык Mizar для представления математических статей.

Математическая библиотека Mizar (MML) в настоящее время является одной из крупнейших формальных математических библиотек и представляет собой коллекцию Mizar-статей (статей, подготовленных на формальном языке системы Mizar), содержащих определения, теоремы и доказательства. В проекте Mizar хорошо структурированные Mizar-статьи подготавливаются для публикации в журнале «Formalized Mathematics» [<http://mizar.org/fm/>]. Версия MML 1.166.1132 содержит свыше 1000 статей, включающих более чем 50000 теорем и свыше 10000 определений. Формальный язык Mizar использует логику первого порядка и теорию множеств Тарского-Гротендика. Язык Mizar поддерживает очень выразительную и гибкую систему типов, которая допускает зависимые типы, в том числе ограничения на предикаты. Однако замкнутость Mizar-статей не позволяет использовать новые методы и технологии управления математическим знанием (MKM - the mathematical knowledge management). Поэтому следующий шаг в направлении семантического представления был сделан в работе [Iancu et al., 2013], в которой предложены методы трансляции MML в формализмы OMDoc/LF [Kohlhase, 2006]. OMDoc (Open Mathematical Documents) является XML-форматом представления математических знаний, в котором эксплицитно задаются структуры формул и контекстные зависимости для машинной обработки. OMDoc использует Эдинбургский синтаксис логической структуры (LF - Edinburgh Logical Framework) для формализации языка Mizar. Это преобразование позволило расширить выразительные возможности OMDoc и пересмотреть ряд разработанных ранее архитектурных решений.

В системе Mizar можно выделить два языковых уровня представлений: уровень шаблонов (pattern-level), для которого характерен синтаксис, ориентированный на человека, причем для формул используется нотация LaTeX, а также семантический уровень (уровень конструктора) для внутреннего представления данных.

OMDoc является контекстно-зависимым форматом разметки и моделью данных для математических документов. В представлении выделяются три уровня абстракции:

- уровень объектов (Object Level). OMDoc использует языки OpenMath и MathML в качестве стандартов для разметки формул. Типы, термины и формулы Mizar соотносятся с этим уровнем;

- уровень утверждений (Statement Level). OMDoc предлагает оригинальную разметку для представления определений и утверждений математической теории. Определения, теоремы, схемы, нотации Mizar соотносятся с этим уровнем;

- уровень теории (Theory Level). OMDoc предлагает оригинальную разметку для кластеризации множеств утверждений в теории, а также определения отношений между ними (включение, морфизмы). Mizar-статьи соотносятся с этим уровнем.

Ядро OMDoc включает структурные отношения между различными математическими концептами. При этом авторы сознательно избегают фиксирования языковых примитивов для отношений и абстрагируются от специфических математических оснований. Таким образом, OMDoc является универсальным форматом представления, достаточно простым для управления. OMDoc не представляет точных аналогий к усложненным определениям Mizar, например, не существует адекватного представления свойства «теоремности» (theoremhood), которое определяет семантику формул Mizar. Расширение OMDoc на такие языково-специфические черты выполнено в прагматической версии OMDoc (Pragmatic OMDoc). Прагматический OMDoc позволяет вводить сложные представления, которые, с одной стороны, имеют формальную семантику, а с другой – интуитивно понятны человеку. В частности, семантика таких языково-специфических расширений определяется целиком внутри ядра OMDoc, что позволяет создавать «прагматические словари», настроенные на различные области. Прагматический уровень объектов обеспечивается посредством параметризации основной структуры, в которой формализуется синтаксис и семантика. Так, логическая структура LF определяется как OMDoc-теория, логика (логика первого порядка в Mizar) как OMDoc-теория с метатеорией LF, которая в свою очередь является метатеорией для языка реальных объектов (теория множеств Тарского-Гротендика). Mizar-статьи представляются как расширения этой теории. Соответствующая метатеория порождает прагматическую семантику объектной теории. Для обеспечения прагматического уровня утверждений используются шаблоны утверждений, которые представляют собой новый тип утверждений с конкретным синтаксисом. Семантика шаблонов определяется в терминах ядра OMDoc. Так, например, стандартная логика первого порядка определяется на основе трех типов шаблонов для функциональных символов, предикатных символов и аксиом (теорем). Полный список (свыше 30 шаблонов) для представления Mizar в OMDoc можно найти в [Iancu

et al., 2011]. Представление Mizar в OMDoc является критически важным результатом для трансляции MML в OMDoc.

Трансляция MML является сложным процессом, включающим следующие стадии:

- конвертирование miz-формата в xml;
- трансляция xml в классы языка Scala, с помощью которых моделируется уровень конструктора языка Mizar;
- трансляция классов Scala на OMDoc;
- запуск базового алгоритма MML-инструментария, который детализирует преобразование прагматического OMDoc в ядро OMDoc.

В результате выполнения трансляции система Mizar может взаимодействовать с другими OMDoc-приложениями, среди которых наибольший интерес представляет возможность индексирования транслированной библиотеки в поисковой системе MathWebSearch [<http://search.mathweb.org/>]. Следующим направлением разработки является интеграция системы Mizar на основе новой платформы MizarWiki [Urban, 2010]. Кроме того, следует отметить, что в транслированной библиотеке Mizar пока не поддерживается трансляция доказательств теорем (транслируются только утверждения теорем), пока это направление также является актуальным.

2. Парадигма активных документов (Active Document Paradigm - API) для публикации семантических данных

Проекты публикации семантических данных широко различаются от простой публикации документов с RDFa аннотациями, отражающих метаданные и междокументные ссылки до фреймворков, которые поддерживают организационно-техническое обеспечение документов, адаптированных для пользователей, которые представляют контент и инструменты, осуществляющие взаимодействия с семантической информацией контента. В дальнейшем, следуя [David et al., 2011] введем понятие «общее пространство контента» (content commons), которое включает семантически-аннотированные документы вместе с семантическими онтологиями. Эта информация может быть использована для различных семантических сервисов, например, для компиляции, визуализации, навигации, информационного поиска и др. Таким образом, соответствующее приложение может интегрировать эти сервисы, чтобы построить исполняемый документ, удовлетворяющий пользовательским предпочтениям. Такой фреймворк получил название ADP (Active Document Paradigm), далее будет рассматриваться реализация ADI в системе Planetary [Kohlhase, 2012]. Система Planetary является Web 3.0 системой, предназначенной для семантического аннотирования коллекции в различных областях науки, технологии, инженерии

и математики (STEM – Science, Technology, Engineering, Mathematics). Документы, опубликованные в системе Planetary, становятся гибкими адаптивными интерфейсами к общему пространству контента, включая объекты предметной области, контексты и отношения. Это решение обеспечивает встроенную поддержку пользователя посредством расширяемого набора пользовательских взаимодействий с документами на основе различных сервисов, которые эксплицируют данные (и таким образом делают их машиночитаемыми) в общем пространстве контента. Можно указать ряд успешных реализаций данной концепции: от архивов научной литературы [ArX], математической энциклопедии [Plac], создаваемой сообществом, системы курсов PantaRhei [Koh+] до портала формальной логики [Plaa].

Концепция активных документов использует идею модульной организации контента. Нижний уровень включает атомные «модули», т.е. объекты контента, которые соотносятся с некоторым элементарным уровнем представления и единственной темой. Например, для системы управления учебными курсами набор атомных модулей образуют обучающие объекты, для энциклопедии – отдельные статьи. Следующий уровень в иерархии – уровень «монографии» (модуль имеет хорошую структуру, одного или группу авторов, единую тематику содержания). Как объект контента монография включает ряд других объектов (модулей): предисловие, главы, оглавление, списки таблиц и рисунков, благодарности и пр. Монографии составляют коллекции, в которых возникают дополнительные модули (комментарий редакторов). Конкретными коллекциями в различных областях являются энциклопедии, академические журналы, материалы конференций, учебные курсы в системе управления курсами. Верхний уровень – уровень библиотеки предоставляет доступ к коллекциям. Практически, библиотека обеспечивает базовый URI для web-ресурса. В концепции Semantic Web библиотека представляет собой службу, которая объединяет ресурсы, адресуемые своими URL.

Приведем пример из [David et al., 2011], поясняющий различия между объектами контента и документами, подготовленными из них, как презентации. Для внутреннего представления объектов контента в системе Planetary используется OMDoc, при этом автор использует язык TeX (вариант LaTeX, позволяющий добавлять семантические аннотации в исходный файл). TeX-представление может быть сконвертировано в OMDoc посредством конвертора LaTeXXML [Ginev, 2011] для управления в Planetary. Подготовка TeX-документа производится в среде TeXIDE [Jucovschi&Kohlhase, 2010], которая обеспечивает такие сервисы, как семантико-синтаксическое выделение, завершение/поиск команд и управление взаимосвязями модулей.

Представление-презентация содержит дополнительные текстовые маркеры, которые вносят дополнительную информацию (например, параметр для нумерации определений по тексту). Другим отличием является привязка к термину его определения с указанием имени термина и модуля, в котором он определяется. При этом возможен импорт модуля с помощью специального отношения `\importmodule`. При этом гиперссылка к определению разрешается с помощью регулярной URI-ссылки. Отметим, что объекты контента вносят вклад в содержимое других объектов, расположенных выше по иерархии. Например, определяемые понятия запускают для пополнения индексное приложение (соответствующий модуль), которое размещается в конце монографии. Или названия секций вносятся во вступительную часть и т.п.

Основное преимущество модели ADP заключается в возможности поддержки режима раздельной компиляции и динамического связывания, а также повторного использования контекстных объектов. При этом по требованию пользователей в системе Planetary можно порождать контексты, удовлетворяющие запросам. В архитектуре Planetary производится преобразование структуры контента, закодированного в STeX, в формат активных документов в виде XHTML+MathML+RDFa. При этом исходные модули компилируются в контекстно-независимые формы, которые затем связываются в соответствующую конфигурацию под конкретную программу.

3. Проект формального математического хранилища на wiki-платформе (MathWiki)

Проект MathWiki [Alama et al., 2011] - интегральный проект на основе wiki-архитектуры, интегрирующий сервисы хостинга проектов и системы управления версиями, специализированные редакторы и средства интерактивной верификации, семантические сервисы для доказательства теорем (Coq proof assistant) с библиотекой CoRN (Constructive Coq Repository at Nijmegen). В настоящее время проект поддерживает только Mizar и Coq, однако авторы стремятся интегрировать в систему Mizar новые инструменты, например, сервис для доказательства tmEgg [Geuvers et al., 2006], а также систему Proviola [Tankink et al., 2010]. Цель проекта - развитие формального математического wiki-портала с соответствующим набором инструментов, связывание формальных математических текстов с соответствующими неформальными представлениями в Википедия, PlanetMath [http://planetmath.org], Wolfram MathWorld [http://mathworld.wolfram.com]. Для системы Mizar в некоторой степени реализовано связывание с Википедия MML-объектов (около двухсот объектов).

4. Семантическое аннотирование данных математической коллекции

Для Казанского университета актуальным является подготовка наборов связанных данных на основе статей научных журналов, издаваемых в университете. Подготовка математического набора связанных данных выполняется на основе разработанных программных инструментов, реализующих комплексный технологический процесс подготовки RDF-набора данных [Невзорова и др., 2012]. В качестве экспериментальной коллекции использовалась статья журнала «Известия ВУЗов. Математика» за 1997-2009 г.г.

Основными функциями разработанного программного прототипа для публикации данных в облаке LOD являются:

- индексирование математических статей в формате LaTeX в виде LOD-совместимых RDF-данных;
- извлечение метаданных статьи в виде концептов онтологии AKT Portal Ontology²;
- извлечение логической структуры документа с использованием онтологии Mocassin;
- извлечение экземпляров математических сущностей в виде концептов онтологии OntoMath^{Pro} и связывание с ресурсами DBPedia;
- распознавание семантики формул через связывание полученных экземпляров математических сущностей с математическими выражениями и формулами в тексте;
- установление взаимосвязи между опубликованными RDF-данными и существующими наборами данных LOD.

Разработанная технология имеет следующие отличительные особенности:

- математический RDF-набор строится на основе коллекции математических статей на русском языке. Применение разработанной технологии к коллекции текстов на других национальных языках требует замены используемого NLP-модуля системы “OntoIntegrator” [Nevzorova&Nevzorov, 2009] на соответствующий NLP-модуль, ориентированный на обработку математических текстов;
- для построения математического RDF-набора разработаны специальные онтологические ресурсы: онтология профессиональной математики OntoMath^{Pro}; онтология семантики структурных элементов математической статьи (расширение онтологии OMDoc);
- построенный RDF-набор помимо метаданных статей включает специальные семантические знания: знания, формируемые в результате специальной обработки математических формул - семантического связывания текстовых определений переменных формул с их

² <http://www.aktors.org/ontology/>

символьными обозначениями; знания, связанные с идентификацией в тексте экземпляров онтологии профессиональной математики *OntoMath^{Pro}*; знания о структурных элементах математической статьи.

Архитектура прототипа программной системы включает 8 модулей, которые могут быть сгруппированы в следующие подсистемы:

- преобразование формата;
- аннотирование текста;
- семантическое аннотирование;
- аннотирование метаданных;
- генерация RDF;
- связывание.

Подробное описание модулей дано в [Nevzorova et al., 2013]. Кратко отметим ряд важных особенностей разработанных семантических технологий.

Семантическое аннотирование математических текстов базируется на онтологии³ проекта *Mocassin⁴* и онтологии профессиональной математики *OntoMath^{Pro}*. В процессе разработки онтологии использовались различные терминологические источники: классические книги, интернет-ресурсы (Wikipedia, Cambridge Mathematical Thesaurus), научные статьи журнала «Известия Вузов. Математика», а также личный опыт профессиональных математиков Казанского Федерального Университета. Онтология *Mocassin* разработана на языках OWL2/RDFS⁵, которые обеспечивают богатые выразительные возможности, а также теоретические и практические средства вывода, например, с использованием современных машин вывода таких, как *Pellet⁶* и *FaCT++⁷*. Онтология *OntoMath^{Pro}* разработана на языках OWL-DL/RDFS и содержит 3450 классов, 6 типов свойств объектов, 3630 экземпляров свойства IS-A и 1140 экземпляров остальных свойств.

Объектами семантического аннотирования также являются формулы связанные с формулами фрагменты текста, задающие описания переменных формул. Внутри указанных фрагментов выделяются кандидаты в экземпляры классов *OntoMath^{Pro}*, которые затем связываются с соответствующими формулами с помощью оригинального алгоритма, реализованного в виде расширения *GATE⁸*.

Связывания построенного RDF-набора данных с существующими наборами данных в облаке LOD выполняется с использованием системы *Silk⁹*, в частности выполнено связывание классов онтологии *OntoMath^{Pro}* с ресурсами *DBpedia*

[Невзорова&Кириллович, 2013]. Разработанный прототип был апробирован на коллекции математических статей (общий объем коллекции более 1300 статей) для демонстрации возможностей предложенного подхода. Оценки результатов, приведенные в [Nevzorova et al., 2013] позволяют делать вывод об эффективности принятых решений.

Заключение

Рассмотренные в настоящем разделе крупные проекты организации математических порталов и семантические технологии представления и обработки математических знаний позволяют выделить следующие направления исследований и разработок:

- формализация уровня представлений. Предлагаемые решения содержат широкий спектр языков представления исходных математических текстов – форматы LaTeX, STeX, XML, специализированные формальные языки, а также программные средства конвертации языков;
- разработка специализированных онтологических ресурсов и различных схем описания данных, используемых в наборах LOD;
- разработка специализированных семантических сервисов для обработки математических текстов. Назначение сервисов весьма многообразно - обработка формул, проверка доказательств, семантический поиск, извлечение математических объектов и др.

Благодарности

Исследование выполнено при финансовой поддержке РФФИ, проект # 11-07-00507.

Библиографический список

- [Iancu et al., 2013] Mihnea Iancu, Michael Kohlhase, Florian Rabe, Josef Urban. The Mizar Mathematical Library in OMDoc: Translation and Applications; pp. 191-202 in *Journal of Automated Reasoning* (50:2); Springer Verlag, 2013.
- [Kohlhase, 2006] Kohlhase M. OMDoc – An Open Markup Format for Mathematical Documents [Version 1.2]. – Berlin: Springer Berlin Heidelberg New York, 2006. – 428 p.
- [Iancu et al., 2011] M. Iancu, M. Kohlhase, F. Rabe. *Translating the Mizar Mathematical Library into OMDoc format*; Technical Report KWARC Report-01/11, Jacobs University Bremen, 2011.
- [Urban, 2010] Josef Urban, Jesse Alama, Piotr Rudnicki, and Herman Geuvers. *A wiki for Mizar: motivation, considerations and initial prototype*. In *Intelligent Computer Mathematics*, 10th International Conference. Proceedings, volume 6167 of *Lecture Notes in Computer Science*. Springer, 2010. Pp.155-169.
- [David et al., 2011] Catalin David, Deyan Ginev, Michael Kohlhase, Bogdan Matican, Stefan Mirea. A Framework for Modular Semantic Publishing with Separate Compilation and Dynamic Linking; in Alexander Castro, Christoph Lange, Evan Sandhaus, Anita Waard, eds.: *1st Workshop on Semantic Publication (SePublica)*, , 2011-05-30 in *Hersonissos, Crete, Greece*; CEUR Workshop Proceedings 721, 2011.
- [Kohlhase, 2012] Michael Kohlhase. The Planetary Project: Towards eMath3.0; pp. 448–452 in Johan Jeuring, John A. Campbell, Jacques Carette, Gabriel Reis, Petr Sojka, Makarius Wenzel, Volker Sorge, eds.: *Intelligent Computer Mathematics, Conferences on Intelligent Computer Mathematics (CICM)*, 2012-07-09/2012-07-14 in *Bremen, Germany*; LNAI 7362, Springer Verlag 2012.

³ The ontology is accessible via URL:

<http://c11.niimm.ksu.ru/ontologies/mocassin> (login/password: demo/demokpfu)

⁴ <http://code.google.com/p/mocassin/>

⁵ <http://www.w3.org/TR/owl2-rdf-based-semantics/>

⁶ <http://clarkparsia.com/pellet/>

⁷ <http://owl.man.ac.uk/factplusplus/>

⁸ <http://bit.ly/c11-gate-morph-formula>

⁹ <http://www4.wiwiw.fu-berlin.de/bizer/silk/>

[Arx] *arXMLiv Build System*. URL: <http://arxivdemo.mathweb.org>

[Plac] *PlanetMath.org Math for people, by the people*. URL: <http://planetmath.org>

[Plaa] *Logic Atlas and Integrator*. URL: <http://logikatlas.omdoc.org/planetary>

[Koh+] Michael Kohlhase et al. *Planet GenCS*. URL: <http://gens.kwarc.info>

[Ginev, 2011] Deyan Ginev et al. The LATEXML Daemon: A LATEX Entrance to the Semantic Web, submitted 2011. URL: <https://kwarc.eecs/iu-bremen.de/repos/arXMLiv/doc/cicm-systems11/paper.pdf>

[Jucovschi&Kohlhase, 2010] Constantin Jucovschi, Michael Kohlhase. sTeXIDE: An Integrated Development Environment for sTeX Collections; pp. 336–344 in Serge Autexier, Jacques Calmet, David Delahaye, Patrick D. F. Ion, Laurence Rideau, Renaud Rioboo, Alan P. Sexton, eds.: *Intelligent Computer Mathematics*; LNAI 6167, Springer Verlag 2010.

[Alama et al., 2011] Jesse Alama, Kasper Brink, Lionel Mamane, and Josef Urban. Large Formal Wikis: Issues and Solutions. *Intelligent Computer Mathematics, Lecture Notes in Computer Science*, Volume 6824, 2011, pp. 133-148.

[Geuvers et al., 2006] Herman Geuvers, Lionel Elie Mamane A Document-Oriented Coq Plugin for TEXmacs. In: Libbrecht, P. (ed) MathUI Workshop, MKM 2006 Conference, Wokingham, UK (2006), <http://www.activemath.org/~paul/MathUI06/>

[Tankink et al., 2010] Carst Tankink, Herman Geuvers, James McKinna, Freek Wiedijk. Proviola: A Tool for Proof Re-animation. In: *Intelligent Computer Mathematics, Lecture Notes in Computer Science*, Volume 6167, 2010, pp. 440-454.

[Nevzorova et al., 2013] Olga Nevzorova, Nikita Zhiltsov, Danila Zaikin, Olga Zhibrik, Alexander Kirillovich, Vladimir Nevzorov, Evgeniy Birialtsev Bringing Math to LOD: A Semantic Publishing Platform Prototype for Scientific Collections in Mathematics // *The Semantic Web – ISWC 2013. Lecture Notes in Computer Science*. Volume 8218, 2013, pp. 379-394.

[Невзорова&Кириллович, 2013] Невзорова О.А., Кириллович А.В. Технологии связывания данных в пространстве Открытых данных на примере математической коллекции // *Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2013): материалы II Междунар. научн.-техн. конф. (Минск, 21-23 февраля 2013 г.) / Редкол.: В. В. Голенков (отв. ред.) [и др.]. – Минск: БГУИР, 2013. С. 91-96.*

[Stamerjohanns et al., 2010] Stamerjohanns H., Kohlhase M., Ginev D., David C., Miller B. Transforming Large Collections of Scientific Publications to XML // *Mathematics in Computer Science*. – Heidelberg: Springer, 2010. – P. 299–307.

[Schraefel et al., 2004] Schraefel M., Shadbolt N., Gibbins N. CS AKTive Space: Representing Computer Science on the Semantic Web // *Proceedings of WWW 2004*. – N.Y.: ACM Press New York, 2004. – P. 384–392.

SEMANTIC TECHNOLOGIES FOR MATHEMATICAL RESOURCES

Nevzorova O.A.

Research Institute of Applied Semiotics of the Academy of Sciences of Tatarstan Republic, Kazan (Volga Region) Federal University, Kazan, Russia

onevzoro@gmail.com

The article provides an overview of semantic technologies developed for the submission and processing the mathematical knowledge on the mathematical Internet portals, as well as the original results of semantic publishing on the Web for the articles of "Izvestiya Vysshikh Uchebnykh Zavedenii. Matematika" journal.



УДК 004.822:514

**ПОДХОДЫ К ОПИСАНИЮ ОТНОШЕНИЯ ЧАСТЬ-ЦЕЛОЕ В
ОНТОЛОГИЧЕСКИХ РЕСУРСАХ**

Лукашевич Н.В.

** Московский государственный университет им.М.В. Ломоносова
г. Москва, Российская Федерация*

louk_nat@mail.ru

Статья посвящена описанию принципов моделирования отношения *часть-целое* в разных онтологических ресурсах и тезаурусах. В частности, представлено решение по представлению это отношения в тезаурусе русского языка РуТез, создаваемого как ресурс для автоматической обработки документов. При моделировании этого отношения в компьютерных ресурсах важным вопросом является обеспечение транзитивности этого отношения. Свойство транзитивности обычно постулируется в классических аксиомах философской мереологии, однако на практике из-за разнообразия подвидов отношения *часть-целое*, сложности дифференциации этого отношения от близких видов отношений и т.п. вопрос должен исследоваться специальным образом.

Ключевые слова: отношение *часть-целое*, онтология, тезаурус, онтологическая зависимость

Введение

Отношение *часть-целое* является одним из самых известных и полезных в разных предметных областях.

Особенностью отношения *часть-целое* является разнообразие его проявлений. Наиболее типичными объектами, к которым применяется это отношение, являются физические объекты. Но также это отношение может устанавливаться и между сущностями, длящимися во времени, между группами сущностей, ролями и процессами и др. другое [Cruse, 1986; Gerstl, Pribennow, 1996; Winston et al., 1987].

При моделировании этого отношения в компьютерных ресурсах важным вопросом является обеспечение транзитивности этого отношения. Свойство транзитивности обычно постулируется в классических аксиомах философской мереологии, однако на практике с транзитивностью возникают проблемы. В связи с этим при моделировании этого отношения в конкретных компьютерных ресурсах приходится принимать специализированный набор решений.

В данной статье будет рассмотрена проблема нарушения транзитивности отношения *часть-целое* и подходы к объяснению и устранению этого нарушения (раздел 2). В разделе 3 будут представлены подходы к установлению отношения

часть-целое в различных онтологических ресурсах. В разделе 4 мы рассмотрим правила представления этого отношения в лингвистической онтологии тезаурусе РуТез, назначением которой является применение в автоматической обработке текстов.

**1. Аксиомы и определение отношения
*часть-целое***

В классической мереологии обычно никак не ограничиваются подвиды отношения *часть-целое* и постулируются три аксиомы для отношения *часть-целое* P [Simons, 1987; Varzi, 2006]: рефлексивность (все является частью самого себя), антисимметричность (ничто не является частью своих частей), транзитивность: (части частей являются частями целого).

Поскольку отношение обладает свойством рефлексивности, то выделяются еще строгое отношение *часть-целое*, т. е. когда рассматриваются только части, не равные своему целому PP :

$$PP(x, y) \stackrel{\text{def}}{=} P(x, y) \wedge \neg P(y, x)$$

Строгое отношение *часть-целое* является отношением строгого порядка, т.е. выполняются соотношения антирефлексивности, асимметричности и транзитивности.

Для определенности в дальнейшем в качестве отношения *часть-целое* будет рассматриваться именно строгое отношение *часть-целое* PP .

В лингвистике для определения отношения *часть-целое* широко используются лингвистические тесты, т. е. некоторые заданные предложения, в которые подставляются анализируемые сущности. При этом *часть* обычно называется меронимом, а *целое* – холонимом. Естественным тестом для определения меронимии является предложение *X – это часть Y*, которое должно звучать нормально для *X* и *Y*, интерпретируемых как родовые понятия: *палец – это часть руки, страница – это часть книги* [Cruse, 1986].

2. Проблемы транзитивности отношения *часть-целое* vs. подвиды отношения *часть-целое*

Многие авторы отмечают, что если применять лингвистические тесты, то возникают серьезные проблемы с транзитивностью отношения *часть-целое*, например, рассмотрим следующую совокупность утверждений: *рука – это часть дирижера, дирижер – это часть оркестра*, но странно, если сказать, что *рука – это часть оркестра* [Cruse, 1986; Winston et al., 1987; Motschnig-Pitrik, Kaasboll, 1999].

Рассматривая разные виды отношения *часть-целое*, авторы обычно подчеркивают, что проблемы с транзитивностью связаны со смешением разных видов отношений *часть-целое*. В работе [Winston et al., 1987] приводится разделение отношения *часть-целое* по на основе признаков функциональности, гомеомерности (принадлежность части и целого одному и тому же семантическому типу, например *оазис – пустыня*), отделмости. Авторы этой работы объясняют проблемы с транзитивностью следующим образом: пока используется один тип отношения, то *часть-целое* всегда транзитивно. Однако когда смешиваются различные отношения меронимии, то возникает проблема с транзитивностью.

В работе [Cruse, 1986] подчеркивается, что правильно сформированная иерархия состоит из элементов одного и того же типа. Так, если элемент меронимии – физический объект, то и все другие элементы меронимии должны быть такими же. Если один элемент является географической областью, то и другие должны быть такими же (так, Вестминстерское аббатство не является частью Лондона); если один элемент – абстрактное сущестительное, то и другие должны быть такими же.

В работе [Motschnig-Pitrik, Kaasboll, 1999] предлагается выделить те отношения *часть-целое*, которые, комбинируясь, дают приемлемые результаты транзитивности, и отделить те отношения *часть-целое*, которые могут привести к ошибочным транзитивным заключениям. Если моделировать такие отношения как *член/коллекция, материал/объект* отношениями, отличными от отношений *часть-целое*, то авторы утверждают, что оставшиеся типы отношений демонстрируют

транзитивное поведение, даже если комбинируются произвольным образом. Таким образом, группа отношений компонент/объект, порция/масса, фаза/деятельность, место/местность может быть названа базовыми отношениями *часть-целое*. В рамках любой комбинации базовых отношений *часть-целое* действует правило транзитивности, независимо от комбинации конкретных видов отношений.

Другое мнение высказывается в философской работе [Varzi, 2006]. Автор работы утверждает, что проблемы с транзитивностью отношения *часть-целое* и приводимые контрпримеры связаны с неявным сужением понятия «часть» в обыденной речи. То, что ручка двери, являясь функциональной частью двери, может не рассматриваться как функциональная часть дома, не означает, что ручка не является вообще частью дома. Напротив, ручка двери проявляет все обычные свойства частей: масса ручки является частью массы дома; она занимает часть пространства, занятого домом; она будет уничтожена, если уничтожить дом; если уничтожить ручку двери, то и дом будет поврежден.

Если рассмотреть пример: *рука дирижера – дирижер – оркестр*, то также можно видеть, что масса руки является частью массы оркестра, рука дирижера занимает часть пространства, занимаемого оркестром; если будет повреждена рука дирижера, это может вызвать и (может быть, даже серьезные) проблемы с функционированием оркестра.

Сужение понятия «часть» заключается в том, что на интерпретацию понятия «часть» накладываются дополнительные условия (т.е. дополнительное требование, что часть должна быть функциональной и т.п.) и при этом, действительно, свойство транзитивности может не выполняться. При этом не транзитивно именно дополнительное ограничение, а не обобщенное отношение в целом.

3. Классификация отношений *часть-целое* по онтологическим основаниям

Помимо классификации отношений *часть-целое* по семантическим основаниям, существуют еще классификации этого отношения по онтологическим свойствам, т.е. на основе анализа сосуществования части и целого [Guarino, 1992; Guizzardi, 2005]. Как будет показано ниже такая классификация будет полезной для обеспечения транзитивности отношения *часть-целое*. Для этого введем отношения онтологической зависимости.

3.1. Отношения онтологической зависимости

Для выявления онтологической зависимости нужно ответить на следующий вопрос: может ли сущность (*X*) существовать сама по себе, или подразумевает существование чего-либо еще (*Y*). Так, свойство белизны зависит от вещества, например, от куска бумаги, тогда и только тогда,

когда это свойство не может существовать без этого куска бумаги. Существует много форм онтологической зависимости.

Если рассматривать онтологическую зависимость конкретной сущности, то можно выделить специфическую зависимость и родовую зависимость [Gangemi et al., 2003; Masolo et al., 2004; Masolo et al., 2003].

При *специфической зависимости* (SD) конкретная сущность e_1 зависит от другой конкретной сущности e_2 , если необходимо, чтобы e_2 существовал, если e_1 – существует:

$$SD(e_1, e_2) \models_{def} ((\exists t \text{ } pre(e_1, t)) \wedge \wedge \forall t (pre(e_1, t) \rightarrow pre(e_2, t)))$$

где $pre(e_i, t)$ – предикат существования сущности e_i в заданное время t [Masolo et al., 2003].

Например, существование конкретного человека зависит от существования его мозга, кроме того, мозг не может быть заменен на другой мозг, т.е. это специфическая зависимость.

Отношение специфической зависимости между конкретными сущностями может быть естественно перенесено на специфическую зависимость между понятиями (CSD), т.е. понятие c_1 является специфически зависимым от понятия c_2 , если все экземпляры c_1 специфически от c_2 , т.е.

$$CSD(c_1, c_2) \models_{def} (\forall e_1 \in E(c_1) \exists e_2 (e_2 \in E(c_2) \wedge SD(e_1, e_2)))$$

При *родовой зависимости* (generic – GD) существование конкретной сущности зависит от существования конкретных сущностей, относящихся к некоторому понятию c :

$$GD(e, c) \models_{def} ((\exists t \text{ } pre(e, t)) \wedge (\forall t (pre(e, t) \rightarrow \exists e_c (e_c \in E(c) \wedge pre(e_c, t))))$$

При *родовой зависимости* между понятиями (CGD):

$$CGD(c_1, c_2) \models_{def} (\forall e_1 \in E(c_1) GD(e_1, c_2))$$

Так, в настоящее время существование конкретного человека зависит от существования его сердца родовой зависимостью, поскольку сердце может быть пересажено, но существование класса человеческих сердец необходимо.

3.2. Анализ зависимости существования части и целого

Отношения онтологической зависимости могут быть использованы для анализа отношений *часть-целое*.

Так, выделяется отношение *существенной части* – EP. Экземпляр e_1 – является существенной частью

экземпляра e_2 , если e_2 специфически зависит от e_1 :

$$EP(e_1, e_2) \models_{def} SD(e_2, e_1) \wedge PP(e_1, e_2) = \forall t (pre(e_2, t) \rightarrow PP(e_1, e_2))$$

где PP – предикат быть частью: e_1 является частью e_2 . Таким образом, для каждого конкретного человека его мозг является существенной частью. Отметим, что здесь для краткости записи используется предположение, что отношение часть-целое обсуждается только для существующих объектов [Simons, 1987], т.е.

$$\forall e_1, e_2, t (P(e_1, e_2) \rightarrow pre(e_1, t) \wedge pre(e_2, t))$$

В современном мире конкретному человеку может быть пересажено другое сердце, но человек обязательно должен иметь сердце. Такое отношение отражается посредством понятия *обязательной части* – MP. Конкретная сущность e_1 является обязательной частью конкретной сущности e_2 , если e_1 является экземпляром понятия c , и e_2 имеет родовую зависимость от понятия c :

$$MP(c, e_2) \models_{def} (\forall t (pre(e_2, t) \rightarrow \exists e_1 (e_1 \in E(c) \wedge PP(e_1, e_2))))$$

Часть e_1 называется *неотделимой частью* e_2 , если e_1 специфически зависит от e_2 , и e_1 является частью e_2 :

$$IP(e_1, e_2) \models_{def} (\forall t (pre(e_1, t) \rightarrow PP(e_1, e_2)))$$

Примером неотделимой части является мозг человека, который не может существовать вне своего целого.

Как уже указывалось, сердце человека может быть отделено от конкретного человека и пересажено другому человеку. Но при этом должна существовать сама категория людей. Такая зависимость называется *обязательным целым* MW.

$$MW(e_1, c) \models_{def} (\forall t (pre(e_1, t) \rightarrow \exists e_2 (e_2 \in E(c) \wedge PP(e_1, e_2))))$$

Все эти отношения могут быть перенесены на отношения между понятиями.

- отношение *существенной части* между понятиями (CEP):

$$CEP(c_1, c_2) \models_{def} (\forall e_2 (e_2 \in E(c_2) \rightarrow \exists e_1 (e_1 \in E(c_1) \wedge EP(e_1, e_2))))$$

- отношение *обязательной части* между понятиями (CMP);

- отношение *неотделимой части* между понятиями (CIP);

- отношение *обязательного целого* между понятиями (CMW).

4. Подходы к описанию отношения *часть-целое* в формальных и лингвистических онтологиях

В различных онтологических ресурсах принимаются разные решения по принципам описания отношений *часть-целое*. Рассмотрим некоторые подходы к представлению отношения в формальных онтологиях и лингвистических онтологиях (тезаурусах).

В онтологии SUMO [Niles, Pease, 2003] отношения *часть-целое* определены только над осязаемыми (tangible) пространственными сущностями – объектами. Такое ограничение не является типичным для общей мереологии. В этой онтологии отношение *часть-целое* подразделяется на следующие подвиды: член, компонент, кусок (piece), собственно часть, поверхностная часть. Поверхностные части делятся на поверхность, верх, низ и бок.

В онтологии OpenCyc (<http://www.cyc.com/cyc-2-1/intro-public.html>) отношение *часть-целое* определяется в очень обобщенном смысле. Единственное ограничение на аргументы отношения заключается в том, что они должны быть конкретными сущностями. Отношение *часть-целое* включает такие подвиды, как пространственные части, временные части, «концептуальные» части (например, содержать_информацию), члены группы и т.п.

В онтологии DOLCE [Masolo et al., 2003] отношение «объект–материал этого объекта» (ваза–глина) рассматривается как отдельное отношение «составляет» (constitute), не являющееся отношением *часть-целое*:

X составляет Y тогда и только тогда, когда X может быть субстратом после разрушения Y.

Такое решение связано с тем, что объект (ваза) и материал, из которого сделан объект, считаются различными сущностями. Если предположить, что между глиной и вазой существует отношение *часть-целое*, то глина должна совпасть с вазой, поскольку у глины и вазы совпадают части, а значит, и по аксиомам мереологии глина и ваза совпадают.

В информационно-поисковых тезаурусах отношения *часть-целое* входят в состав иерархических отношений. Иерархические отношения обычно рассматриваются как несимметричные и транзитивные. При установлении иерархических отношений важна независимость от контекста. В частности, в тех случаях, когда имеется множественная принадлежность части к целому, то между такими терминами не должно устанавливаться иерархическое отношение. Между такими дескрипторами может быть установлено отношение ассоциации. Например, карбюраторы являются частями не только автомобилей. Поэтому дескрипторы *КАРБЮРАТОР* и *АВТОМОБИЛЬ* не

должны быть связаны отношением *часть-целое* в информационно-поисковом тезаурусе [Will, 2004].

Впрочем, нужно отметить, что последовательное применение данной рекомендации в каком-либо информационно-поисковом тезаурусе практически не встречается. Для простоты описания отношений *часть-целое* рекомендуется в основном описывать жесткие иерархические системы, как иерархию географических регионов или вложенность военных подразделений [Z39.19, 2005]. Таким образом, с точки зрения разработки информационно-поисковых тезаурусов не рекомендуется описывать как отношения *часть-целое* такие отношения, как *рука – музыкант*, (поскольку руки не только у музыкантов); *дерево – лес* (поскольку деревья растут не только в лесу).

Отметим также, что никаких требований на зависимость целого от части (таких как существенная часть или обязательная часть) в рекомендациях информационно-поисковых тезаурусов не накладывается.

Подход к отношениям *часть-целое* в тезаурусе английского языка WordNet принципиально другой, поскольку отношения *часть-целое* устанавливаются в WordNet на основе лингвистического теста (см. раздел 2) [Miller, 1998].

Внутри отношения *часть-целое* дополнительно выделяются отношения *быть элементом* (человек – часть человечества) и *быть сделанным из* (стекло – часть стеклянного изделия). Синсет-часть может быть сопоставлен большому количеству синсетов-целое, как, например, *point* (острие) может быть у *стрелы, ножа, иголки, карандаша, булавки* и т.п.

Таким образом, при всей кажущейся очевидности принципов установления отношения *часть-целое*, распространенности этого отношения в различных предметных областях, среди исследователей и авторов ресурсов нет единства в трактовке отношения *часть-целое*.

5. Отношение *часть-целое* в тезаурусе РуТез

Лингвистическая онтология тезаурус РуТез предназначена для использования в автоматической обработке текстов (демонстрационная версия тезауруса расположена по адресу <http://www.labinform.ru/ruthes/index.htm>). К особенностям лингвистической онтологии можно отнести широту охвата различных предметных областей [Лукашевич, 2011], а также то, что единицами данного ресурса являются понятия, а не конкретные сущности (экземпляры). Полный объем онтологии составляет в настоящее время 54 тыс. понятий, около 160 тыс. текстовых входов на русском языке.

При описании отношения *часть-целое* в ЛО были сделаны усилия, чтобы обеспечить транзитивность этого отношения. Если обсуждать

свойства транзитивности и наследования для отношения *часть-целое* в ресурсе, предназначенном для автоматической обработки текстов в информационно-поисковых приложениях, то наиболее важной операцией, которую необходимо обеспечить, является релевантность обсуждения частей обсуждению целого. То есть необходимо описывать отношения *часть-целое* так, что если текст или его некоторый фрагмент посвящен обсуждению части, то можно предполагать, что этот текст (или его фрагмент) будет релевантен и обсуждению целого [Лукашевич, 2011].

Важным условием для обеспечения такого наследования является зависимость существования части от существования целого (ср. [Artale et al., 1996]). Действительно, если все существование некоторой части связано с существованием целого, то и тексты, обсуждающие эту часть, будут иметь непосредственное отношение и к целому, даже если это целое в тексте явно не упомянуто.

Этим требованием, в частности, обеспечивается выполнение рекомендаций руководств и стандартов по разработке информационно-поисковых тезаурусов [Will, 2004; Z39.19, 2005] в том, что описание иерархических отношений должно быть независимо от контекста их упоминания. Описание таких независимых от контекста, «надежных» отношений в ресурсах, предназначенных для автоматической обработки текстов, имеет большое значение, поскольку в автоматическом режиме часто бывает невозможно использовать контекст для подтверждения существования того или иного отношения.

Зависимость части от целого не влечет эксклюзивность части по отношению к целому, т. е. того, что у части ровно одно непосредственное целое. Так, например, локоть является частью руки человека и одновременно частью костной системы, при этом локоть является зависимой частью и для руки человека, и для костной системы.

Таким образом, при описании отношения *часть-целое* в онтологии РуТез применяются принципы, на основе которых в предыдущем разделе формализована рекомендация стандартов по информационно-поисковым тезаурусам: существование экземпляров понятия-части c_1 зависит от существования экземпляров целого c_2 специфической или родовой онтологической зависимостью, т.е. экземпляры понятия-части c_1 представляет собой неотделяемые части для экземпляров понятия-целого c_2 или экземпляры понятия c_2 являются обязательным целым для экземпляров c_1 :

$$целое_{до}(c_1, c_2) =_{def} (CIP(c_1, c_2) \vee CMW(c_1, c_2))$$

При условии принятия аксиомы транзитивности базового отношения *часть-целое*, данная модификация отношения также является транзитивной.

В результате в создаваемых по данной модели лингвистических онтологиях реально работает вывод по транзитивности отношений *часть-целое*, что является новым достижением для лингвистических онтологий, поскольку в тезаурусе WordNet транзитивность отношения *часть-целое* не предполагалась, а в рекомендациях по информационно-поисковым тезаурусам это отношение сводилось к весьма узкому набору случаев, из-за чего такой вывод не мог играть значительной роли.

Накладывая вышеперечисленные условия установления отношения *часть-целое*, мы принимаем достаточно широкую трактовку отношения *часть-целое*: - между физическими объектами (*балкон зала – зрительный зал*); - между регионами (*Европа – Евразия*); между веществами (*амидная группы – амиды*); между множествами (*батальон – рота*); между частями текста (*строфа – стихотворение*); между процессами (*номер представления – представление*)

Также отношения *часть-целое* устанавливаются для связей между сущностями, одна из которых является внутренней, зависимой от другой таких как:

- характерные свойства (*водоизмещение – судно*)(ср. [Artale et al., 1996, Уемов, 1963]);

- роль в процессе (*инвестор – инвестирование*)(ср. [Loebe, 2005; Masolo et al., 2004]);

- участник сферы деятельности – сфера деятельности (*машиностроительный завод – машиностроение*).

Приведем пример вывода на основе свойства транзитивности:

целое (ОБВИНЯЕМЫЙ ПО ДЕЛУ, СУДЕБНОЕ ОБВИНЕНИЕ)

\wedge *целое* (СУДЕБНОЕ ОБВИНЕНИЕ, СУДЕБНЫЙ ПРОЦЕСС)

\rightarrow *целое* (ОБВИНЯЕМЫЙ ПО ДЕЛУ, СУДЕБНЫЙ ПРОЦЕСС)

В информационных системах такие цепочки часто интерпретируются следующим образом: если в тексте обсуждается обвиняемый по делу, то этот текст релевантен и таким темам, как судебное обвинение, судебный процесс и т.д.

В результате в создаваемых по данной модели лингвистических онтологиях реально работает вывод по транзитивности отношений *часть-целое*, что является новым достижением для лингвистических онтологий, поскольку в тезаурусе WordNet транзитивность отношения *часть-целое* не предполагалась, а в рекомендациях по информационно-поисковым тезаурусам это отношение сводилось к весьма узкому набору случаев, из-за чего такой вывод не мог играть значительной роли.

Заключение

В данной статье мы проанализировали подходы к представлению отношений *часть-целое* в компьютерных ресурсах и представили подход к установлению отношений *часть-целое* в лингвистической онтологии тезауруса RuТез. В данном ресурсе отношение часть-целое между понятиями устанавливается только в случае онтологической зависимости части от целого, что помогает обеспечивать транзитивность этого отношения.

Библиографический список

- [Artale et al., 1996] Artale A., Franconi E., Guarino N., Pazzi L. Part-Whole Relations in Object-Centered Systems: An Overview // Data and Knowledge Engineering. 1996. V.20. pp. 347-383.
- [Cruse, 1986] Cruse D. Lexical Semantics. Cambridge, University Press. 1986.
- [Gangemi et al., 2003] Gangemi A., Navigli R., Velardi P. The OntoWordNet project: extension and axiomatisation of conceptual relations in Wordnet // In Proceedings of International Conference on Ontologies, Databases and Applications of Semantics (ODBASE). 2003.
- [Gerstl, Pribennow, 1996] Gerstl P., Pribennow S. A conceptual theory of part-whole relations and its applications // Data and Knowledge Engineering. 1996. V.20. P. 305-322.
- [Guarino, 1992] Guarino N. Concepts, attributes and arbitrary relations // Data Knowledge Engineering. 1992. V.8. P. 249-261.
- [Guizzardi, 2005] Guizzardi G. Ontological foundations for structural conceptual models. CTIT-PhD-thesis Series, No 05-74, 2005.
- [Loebe, 2005] Loebe F. Abstract vs. Social Roles: A Refined Top-level Ontological Analysis // In Proceedings of the 2005 AAAI Fall Symposium 'Roles, an Interdisciplinary Perspective: Ontologies, Languages, and Multiagent Systems AAAI Press, 2005. P.93-100.
- [Masolo et al., 2004] Masolo C., Vieu L., Bottazzi E. Catenacci C., Ferrario R., Gangemi A., Guarino N. Social roles and their descriptions // In Proceedings of the Ninth International Conference on the Principles of Knowledge Representation and Reasoning. AAAI Press. 2004.
- [Masolo et al., 2003] Masolo C., Borgo S., Gangemi A., Guarino N., Oltramari A., Shneider L. WonderWeb. Final Report. Deliverable D18. 2003.
- [Miller, 1998] Miller G. Nouns in WordNet // WordNet – An Electronic Lexical Database / Fellbaum, C (ed). The MIT Press, 1998. P. 23-47.
- [Motschnig-Pitrik, Kaasboll, 1999] Motschnig-Pitrik R., Kaasboll J. Part-Whole Relationship Categories and their Application in Object-Oriented Analysis // IEEE TSE. 1999. v. 11(5). P.779-797.
- [Niles, Pease, 2003] Niles I., Pease A. Linking Lexicons and Ontologies: Mapping WordNet to the Suggested Upper Merged Ontology // In Proceedings of the IEEE International Conference on Information and Knowledge Engineering. 2003. P.412-416.
- [Simons, 1987] Simons P. Parts. A study in Ontology. Oxford University Press, 1987.
- [Varzi, 2006] Varzi A. A Note on Transitivity of Parthood // Applied Ontology. 2006. V. 1, N 2. P. 141-146.
- [Will, 2004] Will L. Thesaurus consultancy // The thesaurus: review, renaissance and revision / Sandra K. Roe and Alan R. Thomas, editors. New York, London : Haworth, 2004. 209p.
- [Winston et al., 1987] Winston M., Chaffin R., Herrmann D. A Taxonomy of Part-Whole Relations // Cognitive Science, 11. 1987. P. 417-444.
- [Z39.19, 2005] Z39.19. 2005.Guidelines for the Construction, Format and Management of Monolingual Thesauri. – NISO.
- [Лукашевич, 2011] Лукашевич Н.В. Тезаурусы в задачах информационного поиска. М.: Изд-во Московского Университет, 2011.
- [Уемов, 1963] Уемов А.И. Вещи, свойства и отношения. М., 1963.

DESCRIPTION OF PART-WHOLE RELATIONS: APPROACHES IN ONTOLOGICAL RESOURCES

Loukachevitch N.V.

Lomonosov Moscow State University

louk_nat@mail.ru

The paper described approaches to representation of part-whole relations in computer resources. Special attention is given to rules for establishing part-whole relations in RuThes linguistic ontology.

Introduction

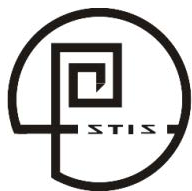
The part-whole relation plays an important role in many domains. The necessity to describe this relation appears during development of such different resources as information-retrieval thesauri, linguistic resources for natural language processing, ontologies, in object-oriented programming. For computer applications the transitivity of part-whole relations is especially important because it can be used in logical inference.

Main Part

One of interesting features of this relation is its diversity, it can be established between such different entities as physical objects, geographical regions, properties and states, collections and sets and others. Philosophical branch that studies part-whole relations – mereology – does not restrict subtypes of parts and wholes establishing several main axioms for the relation. The absence of restrictions on participants of part-whole relations leads to another problem when this relation is difficult to distinguish from similar relations. Besides, developers often encounter visible cases of violations of part-whole relation transitivity.

The important point for the part-whole relation study is co-existence and dependence of existence of the part and its whole. In our opinion this consideration is important for description of this relation retaining its transitivity.

In this paper we consider main problems of description of the part-whole relation in computer resources, its definitions in philosophy and linguistics. Then we will describe the principles of establishing part-wholes relations in linguistic ontology – thesaurus of Russian language RuThes, intended for natural language processing in information-retrieval applications such as automatic conceptual indexing, automatic text categorization, automatic summarization and others.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 519.711.2

УСЛОВНО-РЕФЛЕКТОРНАЯ ОСНОВА ЗАПОМИНАНИЯ ЗНАНИЙ

Стефанюк В.Л

*Институт проблем передачи информации Российской академии наук,
г. Москва, Россия*

stefanuk@iitp.ru

Модель «стопки книг», предложенная М.Л.Цетлиным 1963 году, нашла многочисленные применения в компьютерах и в поисковых системах интернета. В настоящей работе приводятся соображения в пользу определенной коррекции модели стопки книг, с учётом сведений из нейрофизиологии, восходящих к работам И.П.Павлова. Эту коррекцию особенно легко осуществить в модели памяти со случайной выборкой, близкой по характеристикам модели стопки книг. После коррекции подобные системы позволят более обоснованно запоминать такие сведения, как знания по семантике текстов (В.В.Мартынов) или знания об отношениях обусловленности (В.Л.Стефанюк и Л.В.Савинич).

Ключевые слова: модели памяти, стопка книг; случайная память; условный рефлекс, семиотика, семантика.

Введение¹

По мере развития и совершенствования компьютерных приемов, искусственного интеллекта и робототехники, бросается в глаза несомненное дистанцирование от естественной Природы. Хотя авторы часто пытаются создать нечто подобное наблюдаемому в живой Природе (машины, проявляющие творческий интеллект, танцующие роботы, компьютеры с высокой степенью параллелизма вычислений, носящих семантический характер), достигается это путем создания специальных имитационных моделей, по их устройству имеющих мало общего с тем, что наблюдается в Природе. Главная трудность – это отсутствие готовых рецептов или хотя бы подсказок, которые поставлялись бы этологами, биологами, физиологами и нейрофизиологами.

Поэтому мы решили написать небольшой доклад, посвященный одному из «тяжелых» научных вопросов – вопросу организации запоминания наблюдений. Конечная цель при этом – дать определенные рекомендации по усовершенствованию процессов создания и использования баз знаний (БЗ). Цель эта – высокая, но, двигаясь по направлению к ней, мы пока вынуждены заниматься построением некоторых

концептуальных моделей, которые могут породить самую разнообразную оценку и даже критику, к которой автор заранее настроен благожелательно.

Вопросы

Естественно, мы начнем с некоторых вопросов, обращенных к И.П.Павлову – основателю всего того, что сегодня обычно относят к физиологии и нейрофизиологии.

Дело в том, что когда мы запоминаем некоторое событие, то в нашей памяти, разумеется, сохраняется не это событие, каким оно было физически, а его слепок, так сказать, описание или знак. Причем этот знак может быть довольно сложным по своей структуре. Когда собака выделяет желудочный сок по звонку, не видя перед собою мяса, то звонок заставляет её вспомнить мясо, которого перед нею просто нет, т.е. она не видит этого мяса в реальности, не чувствует исходящего от него аромата и т.п.

Безусловный рефлекс поглощения пищи, в данном случае мяса, на фоне «музыки звонка» запускает целую цепь очевидных событий – откусывание, пережевывание с выделением слюны, переваривание пищи, для чего и выделяется желудочный сок, регистрируемый в опытах академика Павлова. (Конечно, это определенное упрощение, поскольку все эти этапы также носят сложный рефлексорный характер.)

В результате выработки условного рефлекса [Вилли, 1968] собака почему-то не набрасывается на

¹ Работа выполнена при финансовой поддержке РФФИ (проект № 12-07-00209-а) и при поддержке по программе президиума РАН «Информационные, управляющие и интеллектуальные технологии и системы».

мясо, которого в действительности нет, не запускает жевательный рефлекс (видимо потому, что в пасти у нее ничего нет!), но, тем не менее, выделяет желудочный сок, несмотря на то, что ни от пищевода, ни от желудка не поступает информации о том, что предстоит что-то переваривать.

Получается, что желудочный сок начинает выделяться собакой независимо от того, есть мясо, или его нет, т.е. желудочный сок начинает выделяться раньше всего, даже прежде, чем собака будет видеть то, что можно съесть. Это входит в состав условного рефлекса.

Условный рефлекс — это приобретенный рефлекс, свойственный отдельному индивиду (особи). Они возникают в течение жизни особи и не закрепляются генетически (не передаются по наследству). Они возникают при определенных условиях и исчезают при их отсутствии. Формируются на базе безусловных рефлексов при участии высших отделов мозга. Условно-рефлекторные реакции зависят от прошлого опыта, от конкретных условий, в которых формируется условный рефлекс [Вилли, 1968].

В связи с этим, возникают следующие вопросы:

1. Почему по звонку не запускается в работу, например, жевательный рефлекс?
2. Как долго «держится» созданный условный рефлекс?
3. Какое мясо давали собаке – новое или обычное для неё, стандартное?
4. Была ли собака голодна перед процедурой выработки у неё условного рефлекса?

Не исключено, что ответы на эти вопросы содержатся в детальном описании экспериментов И.П.Павлова, но в литературе наших дней этих ответов пока мною не найдено.

Правда, кое-какая информация имеется в интернете. Так, для формирования условного рефлекса необходимо:

- Наличие пары раздражителей: безусловного раздражителя и индифферентного (нейтрального) раздражителя, который затем становится условным сигналом;
- Определенная сила раздражителей. Безусловный раздражитель должен быть настолько сильным, чтобы вызывать доминантное возбуждение в центральной нервной системе. Индифферентный раздражитель должен быть привычным, чтобы не вызывать ярко выраженного ориентировочного рефлекса.
- Неоднократное сочетание раздражителей во времени, причем первым должен воздействовать индифферентный раздражитель, затем безусловный раздражитель. В дальнейшем действие 2-х раздражителей продолжается и заканчивается одновременно. Условный рефлекс возникнет в том случае, если

индифферентный раздражитель станет условным раздражителем, т.е. он сигнализирует о действии безусловного раздражителя.

- Постоянство окружающей среды — выработка условного рефлекса требует постоянства свойств условного сигнала.

На наш взгляд, всё дело в том, что жевательный условный рефлекс требует для своего формирования наличия мяса в пасти собаки, а не только звонка.

Слюновыделительный условный рефлекс для своего формирования требует зрительного подтверждения или наличия запаха мяса, кроме звонка. (Трудно удержаться от замечания, что в некоторых западных книгах, где упоминаются опыты И.П.Павлова, говорится, что по звонку у собаки «выделяется слюна»!)

Для рефлекса желудочного сока всё упомянутое не важно, и именно поэтому замечательному русскому ученому пришлось вживлять в желудок собаки специальную фистулу. (Кстати, можно прочесть в литературе, что он отрабатывал соответствующую непростую операцию в течение 10 лет.)

Таким образом, запоминание может как стимулироваться, так и дестимулироваться, или, как, сказано выше, для выработки условного рефлекса «нужны определенные условия».

Известно, что условные рефлексы лежат в основе *приобретенного поведения*. Это наиболее простые программы. Окружающий мир постоянно меняется, поэтому в нём могут успешно жить лишь те, кто быстро и целесообразно отвечает на эти изменения. По мере приобретения жизненного опыта в коре полушарий складывается система условно-рефлекторных связей.

Модель стопки книг

Покажем, что с учетом описанной схемы формирования условных рефлексов, эта модель памяти [Стефанюк, 2004], построенная М.Л.Цетлиным, и которая к настоящему времени нашла множество полезных применений, нуждается в некоторой корректировке.

Полезно процитировать учебник биологии [Вилли, 1968], в котором говорится: «Пока еще невозможно дать полное физиологическое объяснение психологическим явлениям мышления, памяти и научения».

А также: «В последние годы была выдвинута гипотеза о том, что память связана с синтезом РНК и что каждое запоминаемое событие кодируется в центральной нервной системе специфическими последовательностями нуклеотидов в РНК». Правда, затем сказано: «Однако физиологическая и биологическая основа памяти остается невыясненной».

И далее: «Некоторые опыты, например такие, в

которых животные научаются проходить через лабиринт, чтобы получить пищу или избежать электрических ударов, подтвердили роль «проб» и «ошибок» в научении».

Пожалуй, мы также не смогли бы указать, как точно реализуется модель стопки книг в нервной системе, но эта модель обладает многими свойствами, близкими к тому, что было описано выше. Прежде всего, заметим, что в [Стефанюк, 2004] описана также наша модель «случайной памяти», которая организована совершенно иначе, чем «стопка книг», но имеет ровно такие же «внешние» проявления. Последняя модель во многих случаях оказывается более удобной для математического исследования.

Вернемся к «стопке книг». Коррекция модели базируется на предполагаемых ответах на заданные выше вопросы.

Наверх стопки кладется книга только в случае, если факт, требующий запоминания, окрашен эмоционально². Если такой окраски нет, то положение факта в стопке сохраняется.

В более гибкой организации вводится степень эмоциональной окраски, и факт переносится в стопку тем выше, чем выше уровень эмоциональной окраски, вплоть до самого верхнего уровня. Причем уровень эмоциональной окраски зависит от множества факторов, не только внешних, но и внутренних, таких как физиологическое и психологическое состояние особи.

Просмотр же стека, в поиске интересующего исследователя факта, «который хочется сейчас вспомнить», как обычно, ведется в порядке сверху вниз.

Особенно легко добиться такой организации в нашей модели памяти со случайной выборкой (см., например, [Стефанюк, 2004]).

Организация памяти в виде так скорректированной «стопки книг» или «случайной памяти» хорошо согласуется с результатами, упоминаемыми в работе [Тинберген, 1985], отвечая на некоторые поставленные там вопросы:

- Большинство животных реагирует лишь на немногие стимулы из множества возможных в каждой данной ситуации.
- Но каким образом главные формы поведения подавляют остальные, практически неизвестно.
- Как работает этот механизм и где он локализован в мозгу – предмет дальнейших исследований.

В следующих двух разделах для конкретности кратко перечислены те модели знания, которые нас интересуют в первую очередь [Martynov, 1983],

² Эмоция – это кратковременное переживание (Л.В.Савинич, частное сообщение).

[Савинич Л.В & Стефанюк, 2007] и для которых мы предназначаем ту модель хранения знаний и схему её усовершенствования, о которых только что говорилось.

Универсальный семантический код

В УСК все глаголы разбиты на классы, например, глаголы типа внешнее движение (воздействует, трогает и т.п.) или внутреннее движение (колеблется, прорывается и т.п.), а также глаголы типа создание, сохранение, уничтожение, противодействие.

Тройка САО – это сочетание Субъект, Акция, Объект. Так фраза «Человек идет» отражается в такой тройке следующим образом. «Человек (С) остающийся (А) самим собой (О) в пространстве».

Как отмечает сам автор, язык УСК построен в терминах отношений, а последние трактуются так, как это принято в семиологии (см. ниже).

В работе [Martynov, 1983] автор утверждает, что УСК – это язык полной экспликации смысла, т.е. каждый комбинаторный тип строки символов его элементов имеет один и только один смысл. Ограничения на формирование строки не зависят от фрагментов проблемной области, которая подвергается описанию. Там же утверждается, что эта система позволяет формировать новые концепции и строить упомянутые выше гипотезы.

Видимо в целях достижения этого уровня автору и пришлось создавать все новые версии языка УСК, всё более точно отражающие семантику фраз.

Семантика и семиотика

В работе [Бенвенист, 1974] сказано: «С того времени, как Пирс и Соссюр, эти два полярно различных гения, ничего не зная друг о друге и почти одновременно пришли к мысли о возможности самостоятельной науки о знаках и способствовали разработке ее основ, возникла важнейшая проблема, которая в условиях царящей в этой области неразберихи не получила еще окончательной формулировки и, собственно, не была отчетливо поставлена: каково место языка среди знаковых систем?»

Говоря о языке людей, Соссюр пишет: «Язык есть система знаков, выражающих идеи, а, следовательно, его можно сравнивать с письмом, с азбукой для глухонемых, с символическими обрядами, с формами учтивости, с военными сигналами и т. д. и т. п. Он только наиважнейшая из этих систем».

В работе [Бенвенист, 1974] отмечается, что язык во всех своих аспектах предстает как двойственная сущность: являясь институтом социальным, он реализуется индивидом; в аспекте речи он непрерывен, но состоит из отдельных единиц.

И затем: «Можно... сказать, что знаки целиком

произвольные лучше других реализуют принцип семиологического процесса; вот почему язык, самая сложная и самая распространенная из всех систем выражения, вместе с тем и наиболее характерная из них всех; в этом смысле лингвистика может служить прототипом вообще всей семиологии, хотя язык - только одна из многих семиологических систем».

Далее говорится: «Даже если ограничиться чисто эмпирическим перечнем, становится ясным, что вся наша жизнь заключена в сети знаков и мы обусловлены ими до такой степени, что нельзя было бы упразднить ни одну из них без того, чтобы не поставить под угрозу равновесие и общества и отдельного человека. Эти знаки порождаются и множатся в силу внутренней потребности»

Э. Бенвенист заявляет: «От знака к высказыванию нет перехода ни путем образования синтагм (syntagmaton), ни каким-либо другим. Их разделяет непреходимая грань. Поэтому следует признать, что в языке есть две разные области, каждая из которых для своего изучения требует отдельного аппарата понятий. Для области, названной нами семиотической, основу исследования составит сосюрровская теория языкового знака. Семантическую же область следует рассматривать отдельно. Для ее исследования необходим новый аппарат понятий и определений».

На пути к разрешению этой непростой проблемы, по-видимому, и находился лингвист В.В.Мартынов.

Нам представляется, что использование в качестве знаний *отношений обусловленности* [Савинич & Стефанюк] и методов искусственного интеллекта открывает больше возможностей для порождения знаний, полезных как для человека, так и для машины. Однако, кроме проблемы создания таких знаний, остается малоисследованной задача такого способа хранения знаний в базе данных, который позволил бы извлекать знания, в первую очередь, адекватные сложившейся ситуации.

Именно эта задача решается в настоящей статье, в которой сделана попытка скорректировать модель памяти с учетом некоторых свойств условных рефлексов, наблюдаемых в живой природе.

Заключение

Учет нейрофизиологических особенностей формирования условных рефлексов позволил усовершенствовать математические модели памяти, приблизив их характеристики к природным.

Библиографический список

- [Бенвенист, 1974] Бенвенист Э. Общая лингвистика. - М., 1974. - С. 69-89.
[Вилли, 1968] К.Вили. Биология, М: Мир, 1968
[Martynov, 1983] Victor V.Martynov, USC-3: New Variant of a Language for Representing Knowledge and Effecting Calculations,

Artificial Intelligence. Proceedings of the IFAC Symposium, Oxford-New-York-Toronto-Sydney-Frankfurt, 1983.

[Савинич & Стефанюк, 2007] Савинич Л.В., Стефанюк В.Л. Представление конструкций со значением обусловленности. Вторая международная конференция «Системный анализ и информационные технологии (САИТ-2007)», Т. 1, с. 171-173, М:URSS, 2007.

[Савинич & Стефанюк, 2008] Савинич Л.В., Стефанюк В.Л. Выражение обусловленности в естественном языке. «Информационные технологии и вычислительные системы», №1, С. 30-37, 2008.

[Стефанюк, 2004] Стефанюк, В.Л. Локальная организация интеллектуальных систем. Модели и приложения. М.: Физматлит, 2004, 328 с.

[Тинберген, 1985] Н.Тинберген, Поведение животных, М: Мир, 1985.

[Цетлин, 1963] Цетлин М.Л. Конечные автоматы и моделирование простейших форм поведения // Успехи математических наук, 1963, Т. 18, № 4, с. 3-28.

[Stefanuk, 1974] Stefanuk V.L. A model of random pattern matching in a data base // MIP-R-110, University of Edinburgh, 1974 15p.

CONDITIONAL-REFLEXIVE BASES FOR REMEMBERING OF KNOWLEDGE

Stefanuk V.L.

Institute for Information Transmission Problems of the Russian Academy of Sciences, Moscow, Russia
stefanuk@iitp.ru

The pile of book model by Tsetlin (1963) has found multiple applications in computer and in the internet search engines. The correction of the model is proposed due to certain neurophysiology evidences stemmed from Ivan Pavlov research. Resulting model may be used for more adequate memorization of the knowledge of text semantics.

Keywords: memory models, pile of books; random memory; conditional reflexes, semiotics, semantics.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 621.311:658.26

САПР ТП С РАЗВИВАЮЩЕЙСЯ БАЗОЙ ЗНАНИЙ

Г.Б. Бурдо, Б.В. Палюх, Е.В. Воробьева

Тверской государственный технический университет, г. Тверь, Россия

gbtms@yandex.ru

pboris@tstu.tver.ru

evgeniya.vor813@gmail.com

В данной статье рассмотрена проблема построения системы автоматизированного проектирования технологического процесса (САПР ТП) с развивающейся базой знаний (РБЗ). Проанализированы имеющиеся автоматизированные системы проектирования технологических процессов. Выявлена и обоснована необходимость построения САПР ТП с РБЗ. Приведены принципы построения САПР ТП с РБЗ. Показана возможность использования семантических сетей для развития баз знаний.

Ключевые слова: автоматизированного проектирования технологического процесса; развивающаяся база знаний; технологическая подготовка производства; модель накопления и обобщения информации; семантические сети.

Введение

Большинство современных машиностроительных предприятий характеризуется многономенклатурностью выпускаемых изделий, частой сменой видов изделий при одновременном уменьшении времени на технологическую подготовку (ТПП) производства. Поэтому, в настоящее время активно происходит реализация элементов искусственного интеллекта (ИИ) в САПР ТП [Бурдо и др., 2009; Бурдо и др., 2010; Бурдо и др., 2013; Евгеньев, 2001], позволяющих значительно сократить длительность ТПП. К сожалению, одного этого направления развития автоматизированных систем проектирования технологических процессов явно недостаточно.

Основы построения САПР ТП с РБЗ

На основе анализа [Обухов и др., 2002; Яблочников, 2009; Бурдо и др., 2010; Бурдо и др., 2013; Кондаков, 2007; Евгеньев, 2001 и др.] уточнены принципы, полагаемые в основу создания САПР ТП с развивающейся базой знаний: 1) информационная и временная интеграция с АПУТП; 2) интеграция с системами поддержки жизненного цикла изделия; 3) реализация в САПР ТП процедур накопления и обобщения информации (опыта проектирования). Последнее мероприятие предполагает реализацию процедур накопления и обобщения опыта применения критериев (1) и обобщения и накопления опыта проектирования (2). Наиболее интересны для рассмотрения

процедуры обобщения (ОБ) и накопления (Н) опыта, состоящие в том, что каждое технологическое решение ($ТР_i$) запоминается с соответствующим множеством признаков детали ($МП_k$), включающих множество структур $\{S_{ik}\}$ и параметров $\{P_{jk}\}$ детали, и множества организационно-производственных признаков ОРП_k, определяющих организационные и производственные условия выполнения ТПр, ЦФ, оборудование, инструмент, оснастку:

$$ТР_i \leftrightarrow \langle \{S_i\}, \{P_i\}, ОРП_k \rangle, \quad (1)$$

$$МП_k = \langle \{S_i\}, \{P_i\}, ОРП_k \rangle. \quad (2)$$

Для каждого вида решения $ТР_i$ накапливаются множества $\{МП_k\}$, обобщением которых получается технологический образ O . Выявляется интервал технологических решений, $\Delta ТР_i$, соответствующий одному и тому же образу. $ТР_i^n$, проверенные в ТП, должны отвечать нижеперечисленным условиям:

$$ТР_i^n \leftrightarrow \langle \{S_i^n \in \bigcup S_i\}, \{P_j^n \in \Delta P_j\}, \{ОРП^n \in \Delta ОРП\} \rangle. \quad (3)$$

В этом случае решение считается достоверным. Технологический образ получается обобщением опыта:

$$TR_i^I \times TP_i \times \{MP_K\} \rightarrow O_i, \quad (4)$$

и характеризуется определенными интервалами допустимых значений $\{\Delta MP\}$ и $\{\Delta OP_j\}$. Образ O_i :

$$O_i = \langle \{S = \{\cup S_i\}, \{\Delta P_j\}, \{\{\Delta OP_j\}_i\} \wedge \{HD\}\} \rangle, \quad (5)$$

где $\{HD\}$ - множество номеров и названий детали, для которых проектировались технологические процессы.

Накопление и обобщение опыта следует производить поэтапно, начиная с минимального состава элементов в множествах MP_K , постепенно расширяя их до полных объемов, строя иерархию образов. Полное MP_K позволяет непосредственно выбирать TR_i любого уровня. Частным случаем накопления и обобщения является их межуровневая процедура:

$$TR_i^j \leftrightarrow \{TR_i^{j-1}, MK'_k\}; \quad (6)$$

$$TR_i^j \times TR_i^I \times MP'_K \times TR_i^{j-1} \rightarrow O_i^j; \quad (7)$$

$$O_i^j = \{TR_i^{j-1}, MP'_K\} \quad (8)$$

где j -номер уровня, MP'_K - подмножество множества MP_K , необходимое для синтеза TR_i^j на основе решения предыдущего уровня TR_i^{j-1} .

Процедуры Н и ОБ решений строятся по уровням, каждому соответствует образ O . На первом уровне накапливается и обобщается информация по методам получения заготовок МЗ, маршрутам обработки МОП и этапам Э изготовления детали. Для методов получения заготовки (3), признаками MP_1^I , однозначно определяющими ее выбор, являются:

$$Z_i \leftrightarrow \langle \{T D_i, S_i, \{PK_{ji}\}, N, MD, T, \{OP_j\}, \{CF, HD\}_K \} \rangle = MP_{1B}^I \quad (9)$$

где Z_i - i -й метод получения заготовок; TD_i - тип детали; S_i - граф связей поверхностей, определяющий контур заготовки, $\{PK_{ji}\}$ - его размерные параметры; N - годовая программа выпуска; MD - материал детали, m - масса; HD - номер детали; CF - целевая функция. Процедура формирования образа:

$$\{Z_i\} \times \{MP_{1B}^I\} \rightarrow \{O_{1i}^I\}, \quad (10)$$

$$O_{1i}^I = \langle \{ \{TD_i\}, \{S_i\}, \{\Delta PK_{ji}\}, \Delta N, \{MD\}, m, \{OP_j\}, \{HD\}, \{CF\} \}_i \rangle. \quad (11)$$

Маршрут обработки поверхностей определяется параметрами MP_{2m}^I .

$$MOPI \leftrightarrow \langle \{ \{VP_m, Z_i, СПЗ_i, \{PP_j\}_m, S_m, \{TP_j\}_m, ШП_m, MD, N, ФМ_m, \{OP_j\}, \{HD\} \} \rangle = MP_{2m}^I, \quad (12)$$

где VP_m - вид поверхности; $СПЗ_i$ - системные параметры заготовки, включающие размерные связи между ее поверхностями и точность размеров; $\{PP_j\}_m$ - множество размеров поверхностей; S_m - связи поверхностей с другими; $\{TP_j\}_m$, $ШП_m$ - множества, определяющие параметры точности размера и взаимного расположения; $ФМ$ - физико-механические свойства поверхности.

$$\{MOPI\}^I \times \{MOPI\} \times \{MP_{2m}^I\} \rightarrow \{O_{2i}^I\}, \quad (13)$$

$$O_{2i}^I = \langle \{ \{VP_m, Z_i, \{СПЗ_i\}, S = \cup S_m, \{\Delta TT_j\}, \{PP_j\}_m, ФМ_m, \{\Delta ШП_j\}, \Delta N, \{MD\}, \{OP_j\} \} \rangle. \quad (14)$$

Практической проверке подлежат точность и шероховатость обработанной поверхности. Этапы обработки \mathcal{E}_K определяются следующими признаками MP_{1K}^3 :

$$\mathcal{E}_K \leftrightarrow \langle \{ \{TD_i, Z_i, СПЗ_i, \{PP_{jm}\}, S_d, \{OP_j\}, ЦФ, MD, \{ФМ_m\}, N, \{TP_{jm}\}, \{ШП_m\}_{jk} \} \rangle. \quad (15)$$

Процедура обобщения-создания образа:

$$\{\mathcal{E}_K\}^I \times \{\mathcal{E}_K\} \times \{MP_{1K}^3\} \rightarrow \{O_{3K}^I\}. \quad (16)$$

Образ O_{3K}^I :

$$O_{3K}^I = \langle \{ \{TD_i\}, \{Z_i\}, \{СПЗ_i\}, \{\Delta PP_{jm}\}, S_d^B, \{MD\}, \{ФМ_m\}, \{\Delta N\}, \{OP_j\}, \{\Delta TT_{jm}\}, \{\Delta ШП_{jm}\}, \{HD\} \} \rangle, \quad (17)$$

где S_d^B - размерные связи технологических баз детали. Маршрут обработки детали определяется признаками MP_{1i}^2 :

$$M_i \leftrightarrow \langle \{ \{TD_i\}, S_d^B, \{Z_i\}, \{СПЗ_i\}, MD, \{ФМ_m\}, \{N\}, \{OP_j\}, \{TP_{jm}\}, \{ШП_{jm}\}, \{OP\} \} \rangle = MP_{1i}^2, \quad (18)$$

где $\{OP\}$ - множество определяющих размеров детали (длина, ширина, высота, приведенный диаметр и т.д.) - граф общих размеров детали.

$$\{M_i\}^n \times \{M_i\} \times \{MP_1^2\} \rightarrow \{O_{li}^2\}, \quad (19)$$

$$O_{li}^2 = \{ \langle \{TD_i\}, S = \cup S_d^B, \{Z_i\}, \{СПЗ_i\}, \{НД\}, \{ΔN\}, \{ΔTP_{jm}\}, \{ΔПП_{jm}\}, \{МД\}, \{ΔОР\}, \{ФМ_m\}, \{ОП_j\}, \{ЦФ\} \rangle \} \quad (20)$$

Операционная технология определяется признаками MP_{li}^3 :

$$ОП_i \leftrightarrow \langle \{TD_i, S_d, Z_i, СПЗ_i, \{ОП_j\}, \{ТР_{jm}\}, \{ШР_{jm}\}, МД, \{ФМ_m\}, ЦФ, \{ОР\} = MP_{li}^3 \rangle, \quad (21)$$

где S_d - размерные связи поверхностей детали. Процедура получения образа:

$$\{ОП_i\}^n \times \{ОП_i\} \times MP_{li}^3 \rightarrow O_{li}^3. \quad (22)$$

$$O_{li}^3 = \{ \langle \{TD_i\}, S = \cup S_d, \{Z_i\}, \{СПЗ_i\}, \{ΔШР_{jm}\}, \{МД\}, \{ФМ_m\}, \{ΔN\}, \{ОП_i\}, \{ΔTP_{jm}\}, \{ОР\}, \{ЦФ\}, \{НД\} \rangle \}. \quad (23)$$

Параметры режимов резания определяются признаками MP_{li}^4 :

$$РР_i \leftrightarrow \langle \{ВП_i, Z_i, СПЗ_i, \{РП_j\}, \{ТР_j\}, ШП_i, \{ОП_j\}, НД, ФМ, МД\} = MP_{li}^4, \quad (24)$$

$$\{РР_i\}^n \times \{РР_i\} \times \{M_i^4\} \rightarrow \{O_i^4\}; \quad (25)$$

$$O_{li}^4 = \{ \langle \{ВП_i, Z_i, СПЗ_i, \{ΔРР_j\}, \{ΔТТ_j\}, ΔШП_i, \{ОП_j\}, НД, ФМ, МД \rangle \} \quad (26)$$

Управляющие программы определяются признаками MP_2^4 :

$$УП_i \leftrightarrow \langle \{TD_i, S_d^n, S_d', \{ОП_j\}, \{ТР_{jm}\}, \{ШР_m\}, МД, \{ФМ_m\}, S_B, \{РП_i\} \rangle = MP_{2i}^4, \quad (27)$$

где S_d^n - размерные связи детали, соответствующие состоянию после обработки; S_d' - те же самые, но до обработки; S_B - временная структура выполнения переходов в ОП. Для быстрого поиска программ с целью их корректировки достаточно ТД₁, S_d^n, S_d' . Обобщение опыта (корректировка УП) выполняется по результатам их отработки в ТП:

$$\{УП_i\}^n \times \{УП_i\} \times \{M_2^4\} \rightarrow \{O_4^2\}; \quad (28)$$

$$O_2^4 = \langle \{TD_i, S_d', S_d^n, \{НД\} \rangle \}. \quad (29)$$

Добавляя оставшиеся неучтенными признаками, можно получать соответствующие подмножества образа O_2^4 . На этом уровне следует выполнять ОБ данных по фактическим затратам времени на выполнении операции в двух направлениях.

Первое: определение, сравнение и корректировка времени на выполнение конкретной операции в целом:

$$T_{шткi}^P \times \{T_{шткi}^Ф\} \rightarrow T_{штк}^У, \quad (30)$$

где $T_{шткi}$ - штучно-калькуляционное время выполнения i -й операции конкретной детали; индекс $p, ф, у$ - расчетное, фактическое, уточненное значение.

Второе: анализ затрат времени и их уточнение на выполнение множества элементов различных операций, характеризуемых параметрами:

$$\{Z_j\}: \{T_{шткi}^P\} \times \{T_{шткi}^Ф\} \times \{Z_i\} \rightarrow t_j^У, \quad (31)$$

где $t_j^У$ - множество уточненных составляющих штучно-калькуляционного времени, соответствующих множеству элементов операций Z_j . На основе образов обеспечивается проектирование по методу типизации на всех уровнях, вплоть до полных аналогов.

Указанные процедуры можно реализовать с использованием семантической технологии проектирования интеллектуальных систем [Голенков и др., 2011 и др.].

Модель накопления и обобщения информации в базах знаний приведена на рисунке 1.

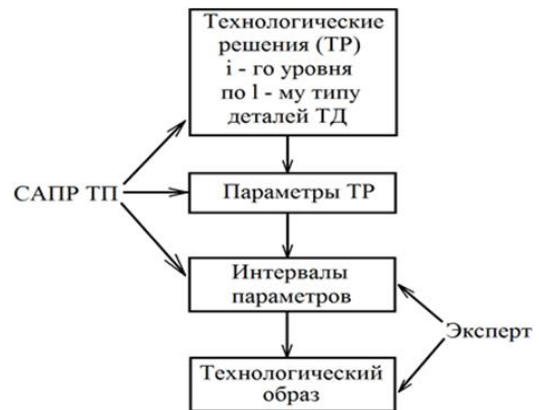


Рисунок 1 - Модель накопления и обобщения информации

Запоминаются технологическое решение (информационная модель ТПр i -го уровня) и перечисленные выше определяющие его признаки (параметры). Решения группируются по конструктивно-технологически сходным деталям, определяются и оцениваются интервалы признаков. Решение и интервалы признаков (параметров) заносятся в базу знаний, в дальнейшем технологическое решение любого уровня выбирается сравнением значений признаков детали с интервалами значений признаков в базе знаний.

САПР ТП выполняет накопление и определение интервалов параметров, оценка и обобщение осуществляется с участием экспертов.

Заключение

Представленная модель накопления и обобщения информации позволяет построить САПР ТП с элементами ИИ, которая позволит сократить время ТПП. Обобщение и накопление знаний выполняются при активном участии экспертов и с учетом результатов отработки ТП в ТП. Реализация предлагаемой САПР ТП с РБЗ может осуществляться в программной среде известных отечественных САПР ТП.

Библиографический список

[Бурдо и др., 2009] Бурдо Г.Б., Палюх Б.В. Повышение эффективности управления технологическими подразделениями в условиях единичного и мелкосерийного производства. / Г.Б. Бурдо, Б.В. Палюх // Вестник Донского ГТУ, 2009, т.9, №4, с.659-666.

[Бурдо и др., 2010] Бурдо Г.Б., Палюх Б.В., Рагозин Программные средства имитационного моделирования размерной структуры технологических процессов / Г.Б. Бурдо, Б.В. Палюх, Г.И. Рагозин // Программные продукты и системы. - 2010.-№1(89).-С.82 - 85.

[Бурдо и др., 2013] Бурдо Г.Б., Палюх Б.В., Испирян Н.В., Исаев А.А., Бурдо В.Г. Автоматизация технологической подготовки многоменклатурного производства. / Г.Б. Бурдо, Б.В. Палюх, Н.В. Испирян, А.А. Исаев, В.Г. Бурдо // Механика и физика процессов на поверхности и в контакте твердых тел и деталей технологического и энергетического оборудования: межвузовский сборник науч. тр. / Тверской гос. техн. ун-т-Тверь, 2013.-Выпуск 6.-С.106-110.

[Евгеньев, 2001] Евгеньев Г.Б. Системология инженерных знаний: учеб. пособие для вузов / Г.Б. Евгеньев - М.: Изд-во МГТУ им. Н.Э. Баумана, 2001. - 376 с.

[Яблочников, 2009] Яблочников Е.И. Организация единого информационного пространства технической подготовки производства с использованием PDM SmarTeam/ Е.И. Яблочников // Информационные технологии в проектировании и производстве. - 2009.-№3.- С. 22-29.

[Обухов и др., 2002] Обухов Е.И., Гайфуллин Б.Н. Автоматизация систем управления предприятиями стандарта ERP-MRP -11./ Е.И. Обухов, Б.Н. Гайфуллин -Интерфейс-Пресс, 2002.-286 с.

[Кондаков, 2007] Кондаков А.И. САПР технологических процессов/ А.И. Кондаков - М.: Издательский центр "Академия", 2007.-272с.

[Голенков и др., 2011] Голенков В.В., Гулякина Н.А. Принципы построения массовой семантической технологии компонентного проектирования интеллектуальных систем / В.В. Голенков, Н.А. Гулякина // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2011): мат. Междунар. научн.-техн. конф. – Минск: БГУИР, 2011. – С. 21-58.

METHODOLOGICAL BASES OF CONSTRUCTION CAD TA WITH DEVELOPING THE KNOWLEDGE BASE

Burdo G.B., Palyukh B.V., Vorobyeva E.V.

*Tver State Technical University (TvSTU),
Tver, Russia*

gbtms@yandex.ru

pboris@tstu.tver.ru

evgeniya.vor813@gmail.com

This article considers the problem of constructing a system of computer-aided design process (CAD TA) developing knowledge base (DKB). Analyzed existing industrial automation management system process. Spotted and the necessity of CAD TP with the DKB. The principles of CAD TP with the DKB. The possibility of using semantic networks for the development of knowledge bases.

Key words: computer-aided design process, developing the knowledge base, technological preparation of production, accumulation model and summarizing information, semantic networks.

Introduction

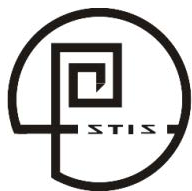
Most modern engineering enterprises characterized by diversified manufactured products, frequently changing kinds of products while reducing the time for technological training (TTP) production. Today there is a realization active elements of artificial intelligence (AI) in CAD TP [Burdo et al., 2009; Burdo et al., 2010; Burdo et al., 2013; Evgenyev, 2001:] to significantly reduce the duration of the TTP, but one of the directions of development of CAD TP is not enough.

Main Part

The principles of CAD TP developing knowledge base, developed procedures for the accumulation and synthesis of experience in the design process in CAD TP. The accumulation of experience design (solutions - technology) and intervals of attributes (parameters) and details of organizational and technological conditions for which designed technology, automatically accumulate CAD TP. With the participation of experts receive technological image (assessment and consolidation of decisions), recorded in the knowledge base. Technology image includes both the structure and parameters details (attributes details) and the parameters of organizational and technological conditions for implementation technologies (organizational and technological features). In the future, the technological solution of any level is selected by comparing the values of attributes parts at intervals of values of corresponding image features of technological knowledge base. CAD TP performs accumulation intervals and definition of parameters, evaluation and synthesis is carried out with the participation of experts.

Conclusion

The presented model of accumulation and synthesis of information allows one to construct CAD TP with elements of AI, which will reduce the time of TTP. Synthesis and accumulation of knowledge carried out with the active participation of experts and taking into account the mining TPD in TP. Implementation of the proposed CAD TP with DKB could take place in a software environment known domestic CAD TP.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822: 004.912

ПРИМЕНЕНИЕ КГТВ-АЛГОРИТМА ДЛЯ НАУЧНЫХ ТЕКСТОВ

Ландэ Д.В.^{*}, Снарский А.А.^{*}, Ягунова Е.В.^{**}

^{}Институт проблем регистрации информации НАН Украины,
Национальный технический университет Украины «Киевский политехнический институт»,
г. Киев, Украина*

DWLandé@gmail.com

ASnarskii@gmail.com

*^{**}Санкт-Петербургский государственный университет,
Санкт-Петербург, Россия*

Iagounova.Elena@gmail.com

Описывается применение предложенной авторами методики построения компактифицированного графа горизонтальной видимости (КГТВ) для выявления в научных текстах тех слов, которые определяют не только их структурную связность, но и информационную структуру. Сравниваются результаты, полученные на основе применения КГТВ-алгоритма и алгоритма построения простой сети слов.

Ключевые слова: опорные слова; научные тексты; граф горизонтальной видимости; информационная структура.

Введение

Построение сетей слов (Language Networks), узлами которых являются термины – слова или словосочетания, фрагменты естественного языка, уже традиционно позволяет выявлять структурные элементы текста, без которых он теряет свою связность [Ferrer-i-Cancho, 2001], [Ландэ и др., 2013]. При этом некоторые из важных структурных элементов текста оказываются также информационно-значимыми, определяющими информационную структуру [Солганик, 1991], [Черняховская, 1983]. Такие элементы, (будем называть их «опорными словами») могут использоваться для построения информационных портретов [Ягунова, 2010], идентификации не достаточно четко определенных компонент текста, таких как коллокации, сверхфразовые единства [Giora, 1983], [Ягунова, 2012].

Авторами были проведены исследования применения алгоритма КГТВ для выявления опорных слов для художественных текстов [Landé etc, 2013], однако вопрос о возможности его применения для массивов научных текстов относительно небольшого объема (в частности, материалов конференций) до сих пор остается открытым.

Графы горизонтальной видимости

В рамках теорий цифровой обработки сигналов (Digital Signal Processing) и сложных сетей (Complex Network) [Albert, 2002], [Strogatz, 2001] предложено несколько методов построения сетей на основе числовых рядов, среди которых можно назвать семейство методов построения графов видимости, в частности, так называемый граф горизонтальной видимости (Horizontal Visibility Graph – HVG) [Gutin, 2011], [Luque, 2009]. Эти подходы также позволяют строить сетевые структуры на основании текстов, в которых отдельным словам или словосочетаниям некоторым специальным образом ставятся в соответствие некоторые начальные числовые весовые значения.

При построении сетей слов в данной работе использована дисперсионная оценка важности слов [Ortuño etc, 2002]. Пусть текст состоит из N слов ($n=1, \dots, N$, n – порядковый номер слова в тексте, позиция слова). Обозначим средний интервал (количество слов) между появлениями слова A в тексте через $\langle \Delta A \rangle$, а средний квадрат значений этих интервалов через $\langle \Delta A^2 \rangle$. Дисперсионная оценка слова A – σ_A рассчитывается по формуле:

$$\sigma_A = \frac{\sqrt{\langle \Delta A^2 \rangle - \langle \Delta A \rangle^2}}{\langle \Delta A \rangle}.$$

По сути, дисперсионная оценка позволяет отделить слова, встречающиеся в тексте относительно равномерно (для равномерно распределенных слов эта оценка равна нулю), от слов, распределенных неравномерно. Т.е. это оценка различительной, дискриминантной силы слов, в частности, для информационного поиска. Идея дисперсионной оценки очень близка к традиционной оценке TFIDF, однако более корректно применима к полным единичным текстам, а не к массивам текстов, как TFIDF.

Сеть слов на основе КГТВ-алгоритма

В отличие от обычных числовых рядов, изучаемых в рамках цифровой обработки сигналов, ряды из цифровых значений, соответствующих словам, преобразуются в графы, узлам которых соответствуют не только цифровые значения, но сами слова, выражающие определенное смысловое значение.

Сеть слов с использованием алгоритма горизонтальной видимости строится в три этапа. На первом на горизонтальной оси отмечается ряд узлов, каждый из которых соответствует словам в порядке появления в тексте, а по вертикальной оси откладываются весовые численные оценки (визуально – набор вертикальных линий, см. рис.1).

На втором этапе строится традиционный граф горизонтальной видимости. При этом считается, что между узлами существует связь, если они находятся в «прямой видимости», т.е. если их можно соединить горизонтальной линией, не пересекающей никакую другую вертикальную линию. Этот геометрический критерий можно записать следующим образом: два узла (слова) слова, например, B с весом σ_B и C с весом σ_C соединены ребром, если $\sigma_B, \sigma_C > \sigma_X$ для всех слов X с весом σ_X , расположенных между словами B и C .

На третьем этапе, полученная на предыдущем этапе сеть компактифицируется. Все узлы с данным словом, например словом A , объединяются в один узел. Все связи таких узлов также объединяются. Важно отметить, что между любыми двумя узлами при этом остается не более одного ребра, кратные ребра изымаются. В результате получается новая сеть слов – *компактифицированный граф горизонтальной видимости* (см. рисунок 1).

На заключительном этапе формирования КГТВ также отфильтровываются слова, важные для согласованности текста, но не имеющие информационной значимости. Для этого использовался так называемый «стоп-словарь», сформированный на основе агрегации информации, доступной по адресам:

- [http://code.google.com/p/stop-words/source/browse/trunk/stop-words/stop-](http://code.google.com/p/stop-words/source/browse/trunk/stop-words/stop-words/stop-words-russian.txt?spec=svn3&r=3)

[words/stop-words-russian.txt?spec=svn3&r=3](http://code.google.com/p/stop-words/source/browse/trunk/stop-words/stop-words-russian.txt?spec=svn3&r=3)

- <https://github.com/punbb/langs/blob/master/Russian/stopwords.txt>
- <http://www.ranks.nl/stopwords/russian.html>
- <https://trac.mysvn.ru/punbb/punbb/browser/trunk/Russian/stopwords.txt>

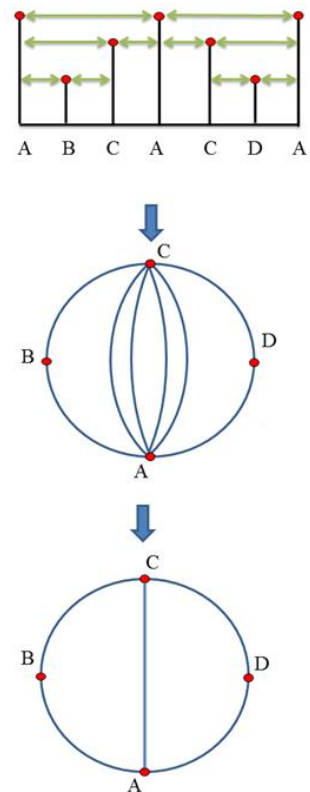


Рисунок 1 – Этапы построения компактификационного графа горизонтальной видимости

Следует отметить, что в рамках предложенного алгоритма не рассматривалась нормализация слов, т.е. в качестве разных узлов иногда использовались различные словоформы одного и того же слова.

Эксперимент

В качестве корпуса научных текстов авторами использовались труды международной научной конференции «Корпусная лингвистика – 2008», в частности тексты таких докладов:

- Е.А. Сидорова «Подход к построению предметных словарей по корпусу текстов» (длина текста – 10946 символов, включая пробелы);
- В.И. Шестопалова, Т.И. Петрова, М.А. Болгов «Региональный вариант живой русской речи как объект корпусной лингвистики» (12948 символов);
- В.Ф. Выдрин «На пути к электронному корпусу языка Бамана: обозначение тонов» (21633 символов);
- Е.В. Падучева «Прямая и косвенная диатеза

16	СЕР	16	ОСОБЕННОСТИ
17	РЕШЕНИЕ	17	НКРЯ
18	ЯВЛЯЕТСЯ	18	ВОЗМОЖНОСТИ
19	НКРЯ	19	СБАЛАНСИРОВАННОС
20	ВОЗМОЖНОСТ	20	ТЬ
21	И	21	ОСНОВА
22	РАЗМЕЩЕНИЕ	22	ИССЛЕДОВАНИЕ
23	ДАННЫЕ	23	ВЫДЕЛЯЕМ
24	УЧЕТ	24	ТЕРРИТОРИАЛЬНЫЙ
25	ПРИНЦИП	25	СЧИТАЕМ
26	ПОДКОРПУС	26	СОЦИАЛЬНЫЙ
	ИССЛЕДОВАНИЕ		СИТУАЦИЯ

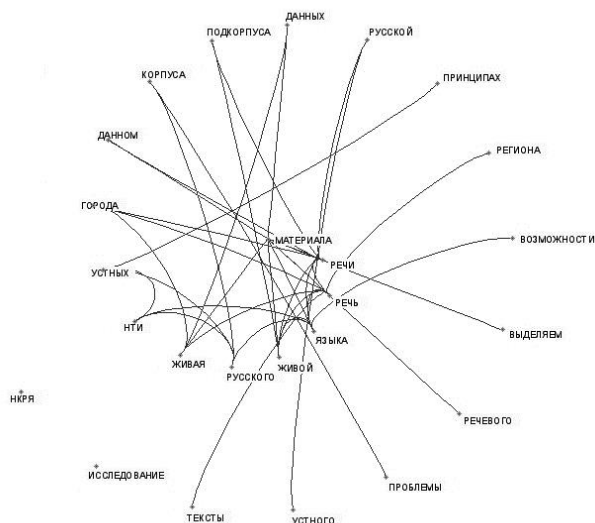


Рисунок 3 – центральный фрагмент КГТВ по докладу В.И. Шестопаловой, Т.И. Петровой и М.А. Болгова

Средний по размеру доклад В.Ф. Выдрина (таблица 3, рисунок 4) посвящен электронному корпусу языка Бамана. В состав наиболее весомых узлов в этом случае попали такие термины: БАМАНА, ТОН, АРТИКЛЬ, ЯЗЫК, ТЕКСТ, СЛОВО, СТЕПЕНЬ. Как можно видеть, в данном случае абсолютное большинство выбранных слов имеет специальный, информационно значимый для данного доклада характер.

Таблица 3 – Наиболее весомые узлы КГТВ и простой сети языка по тексту доклада В.Ф. Выдрина

№	КГТВ	№	Простая сеть слов
1	БАМАНА	1	БАМАНА
2	ТОН	2	ТОН
3	АРТИКЛЬ	3	СЛОВО
4	ЯЗЫК	4	ОКАЗЫВАЕТСЯ
5	ТЕКСТ	5	АРТИКЛЬ
6	СЛОВО	6	ТЕКСТ
7	СТЕПЕНЬ	7	ФАКТИЧЕСКИ
8	DUMESTRE	8	ЯВЛЯЕТСЯ
9	ВАМВАРА	9	ОБОЗНАЧАТЬ
10	ОКАЗЫВАЕТСЯ	10	ЯЗЫК
11	ЯВЛЯЕТСЯ	11	ПУБЛИКАЦИИ
12	ФРАЗА	12	ОБРАЗОМ
13	ЭЛЕМЕНТ	13	СВЯЗИ
14	ИМЕЮТСЯ	14	СУЩЕСТВЕННО
15	НИЗКИЙ	15	СТЕПЕНЬ

16	СВЯЗИ	16	ОБОЗНАЧЕНИЕ
17	КОМПОНЕНТА	17	НИЗКИЙ
18	ПРИЛАГАТЕЛЬНЫХ	18	КОРПУСА
19	ОБОЗНАЧЕНИЕ	19	КОЛИЧЕСТВО
20	ФАКТИЧЕСКИ	20	ВАМВАРА
21	ТОНАЛЬНЫЙ	21	ТОНАЛЬНЫЙ
22	ОТСУТСТВИЕ	22	КОМПОНЕНТА
23	МАЛИ	23	ИМЕЮТСЯ
24	КОРПУС	24	ДОМ
25	КЛАСС	25	ЭЛЕМЕНТА
26	ГЛОССАРИЙ	26	ВЫСОКИЙ

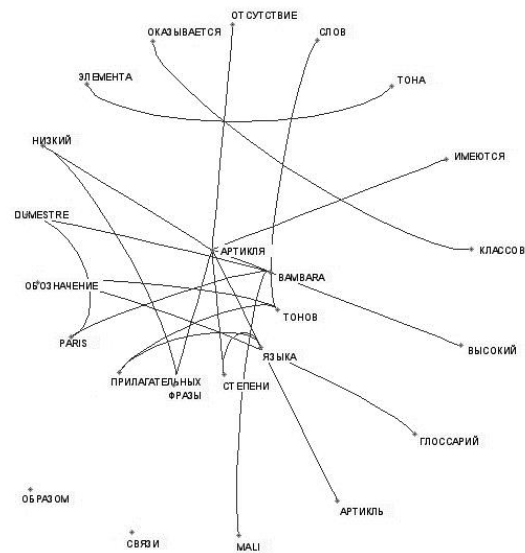


Рисунок 4 – центральный фрагмент КГТВ по докладу В.Ф. Выдрина

Самый большой из рассматриваемых по размеру доклад Е.В. Падучевой (таблица 4, рисунок 5) посвящен прямой и косвенной диатезе ментального глагола. В состав наиболее весомых узлов в данном случае попали многие из рассматриваемых глаголов, которые также являются информационно значимыми для данного доклада.

Таблица 4 – Наиболее весомые узлы КГТВ и простой сети языка по тексту доклада Е.В. Падучевой

№	КГТВ	№	Простая сеть слов
1	ИВАН	1	МНЕНИЕ
2	АКЦЕНТ	2	ГЛАГОЛ
3	ПРЕДПОЛАГАТЬ	3	ЗНАНИЯ
4	МНЕНИЕ	4	ПРЕДПОЛАГАТЬ
5	ГЛАГОЛ	5	МАША
6	ПАДУЧЕВА	6	ПОДОЗРЕВАТЬ
7	ПОДОЗРЕВАТЬ	7	ИВАН
8	МАША	8	ВОПРОС
9	ДОГАДЫВАТЬСЯ	9	ПРЕСУППОЗИЦИЯ
10	ПОДЧИНЯТЬ	10	АКЦЕНТ
11	ЗНАНИЯ	11	ПАДУЧЕВА
12	ПРЕСУППОЗИЦИЯ	12	ПРОПОЗИЦИОНАЛЬНЫЙ
13	АКТАНТ	13	КОНТЕКСТ
14	ДИАТЕЗ	14	ИМЕЕТ
15	ДОКАЗАТЬ	15	ГОВОРЯЩИЙ
16	КОНТЕКСТ	16	ДУМАТЬ
17	СИТУАЦИЯ	17	ДОКАЗАТЬ
18	ВОПРОС	18	ЯВЛЯЕТСЯ
19	ГОВОРЯЩИЙ	19	ПРОПОЗИЦИЯ
20	ПРЯМОЙ		

dedicated to the 50th anniversary of the Institute of Mathematics and Computer Science, 20-23 aug. 2013, Chi.in.u, Moldova: Proceedings IIS / Institute of Mathematics and Computer Science, 2013. - P. 108-113.

[Luque, 2009] Luque B., Lacasa L., Ballesteros F., Luque J. Horizontal visibility graphs: Exact results for random time series // Physical Review E, 2009. - P. 046103-1-046103-11.

[Strogatz, 2001] Strogatz S.H. Exploring Complex Networks // Nature, 2001. - 410. - P. 268-276.

[Ortuño etc, 2002] Ortuño M., Carpena P., Bernaola P., Muñoz E., Somoza A.M. Keyword detection in natural languages and DNA // Europhys. Lett, 2002, - 57(5). - P. 759-764.

APPLICATION OF THE CHVG-ALGORITHM FOR SCIENTIFIC TEXTS

*Lande D.V. *, Snarskii A.A. *, Yagunova E.V.***

** Institute for Information Recording NAS of
Ukraine, NTUU "Kiev Polytechnic Institute",
Kiev, Ukraine*

DWLande@gmail.com

ASnarskii@gmail.com

*** Saint-Petersburg State University,
St.-Petersburg, Russian Federation*

Iagounova.Elena@gmail.com

In work describes the application of the CHVG-algorithm for scientific texts. A CHVG-algorithm for identify the words that define the information structure of the text is proposed. It was found that the networks constructed in such way have a property that among the nodes with largest degrees there are words that determine not only a text structure communication, but also its informational structure.^{1 1}

Introduction

Along with "linear" text analysis, construction of a net with text elements such as words and word combinations as its nodes can help reveal the structural elements of a text which make it coherent. Finding those structural elements which also have informational significance and form informational structure of a text is an important problem.

The originality of the research is contained within the application of horizontal visibility graph used in digital signal processing to scientific texts. The proposed algorithm enables to extract the words which not only have informational significance but also are important for text coherence.

Main Part

According to the horizontal visibility algorithm, language network is built in three stages. First, a series of nodes are plotted on the horizontal axis with uniform spacing, each node corresponding to a word, in the order the words appear in the text. At the same time numeric weights (dispersion estimated value) corresponding to the words are plotted on the vertical axis. At the second stage a traditional horizontal visibility graph is built. Visibility is considered for the highest points of the nodes columns. An edge is put between the nodes, if there is a visible connection between them, e.g., if they

can be connected by a horizontal line which does not cross any column.

At the third stage language network is compactified. All the nodes with the given word are merged into one node. All the edges of such nodes are also merged. It is important to note that multiple edges are removed, and there is no more than one edge left between every two nodes. As a result, we have a new language network – a compactified horizontal visibility graph.

For the sake of comparison we analyzed the simplest types of language networks where on the first stage of the algorithm neighboring words in the text are connected, and on the second one network compactification takes place. At the same time we have maximum degree nodes for the maximum frequency words which are of great importance for text coherence and of little interest for the informational structure of a text.

Conclusion

The following results were obtained from studying the language networks:

– An algorithm for constructing compactified horizontal visibility graph (CHVG) was proposed.

– Language networks for different scientific texts are built on the basis of dispersion estimated values and CHVG.

– For scientific texts the CHVG nodes with maximum degree correspond to the words which not only provide for text coherence but also determine its informational structure and the semantics of the pieces of literature.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 519.816

ПОСТРОЕНИЕ ИНТЕГРАЛЬНОГО ПОКАЗАТЕЛЯ ОЦЕНКИ ПЕРСПЕКТИВНОСТИ ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА

Лобанов В.Н.^{*}, Петровский А.Б.^{**}

^{*} ОАО «НИИ вычислительных комплексов им. М.А. Карцева», г. Москва, Россия

fisher_1980@mail.ru

^{**} Институт системного анализа РАН, г. Москва, Россия

pab@isa.ru

В работе рассматривается процедура построения интегрального показателя оценки перспективности вычислительного комплекса, основанная на последовательном агрегировании исходных показателей в иерархическую систему составных критериев.

Ключевые слова: агрегирование критериев; вычислительный комплекс; иерархическая система критериев; интегральный показатель; составной критерий.

Введение

В настоящее время различные научные и прикладные задачи все чаще решаются с помощью высокопроизводительных вычислительных комплексов, которые выступают реальной альтернативой дорогостоящих суперкомпьютеров. Примерами таких задач являются обнаружение и отслеживание целей в радиолокации и гидроакустике, моделирование быстротекущих процессов в физике и ряд других задач, связанных с обработкой больших массивов информации за короткое время.

Стоимость создания вычислительных комплексов постоянно снижается, благодаря массовому выпуску стандартных комплектующих изделий и растущей конкуренции среди производителей. Современные относительно недорогие микропроцессоры, сетевые технологии и периферийные устройства позволяют строить на их основе разнообразные конфигурации вычислительных комплексов, удовлетворяющие требованиям пользователей по энергопотреблению, габаритам, массе и др. Построение вычислительных систем из стандартных компонент привело к тому, что на рынке представлено много комплексов различных конфигураций. Поэтому перед пользователем, которому нужно решить собственную прикладную задачу, встает непростая проблема сравнения и

выбора наиболее предпочтительной конфигурации вычислительного комплекса.

Сравнение и выбор вычислительных комплексов представляет собой сложную, слабо структурируемую и плохо формализованную задачу [Ларичев, 2006; Петровский, 2009]. Это обусловлено тем, что вычислительные комплексы характеризуются большим числом показателей, а выбор осуществляется по многим количественным и качественным критериям. Вместе с тем, как правило, вариантов конфигураций бывает немного, и такие варианты оказываются несравнимыми друг с другом по своим показателям, что не позволяет применять известные методы принятия решений для выбора лучшего варианта. Возможные подходы к многокритериальному выбору вычислительных кластеров изложены в работах [Лобанов и др., 2012; Лобанов и др., 2013; Ройзензон, 2005].

В данной работе описан выбор наиболее предпочтительного вычислительного комплекса из имеющегося перечня по интегральному показателю, характеризующему перспективность комплекса, который строится несколькими способами с помощью технологии ПАКС (Последовательное агрегирование Классифицируемых Состояний) [Петровский и др., 2012]. Многоэтапная технология ПАКС, исходя из предпочтений лица, принимающего решение (ЛПР), и/или знаний эксперта, позволяет сформировать несколько

иерархических систем с различными наборами составных критериев, которые объединяют большое число исходных характеристик рассматриваемых объектов в один или несколько итоговых критериев, имеющих количественные и/или качественные порядковые шкалы оценок.

Человеку, в силу особенностей его физической памяти, легче оперировать небольшой по объему информацией. На практике для этого бывает достаточно от трех до семи показателей. Итоговые критерии в компактной форме передают содержательный смысл исходных характеристик и дают возможность обосновать выбор наиболее предпочтительного варианта. Поэтому сокращение числа критериев оценки позволяет ЛПР не только решить задачу выбора, но и упрощает анализ полученных результатов.

Вместе с тем при слишком малом числе итоговых критериев и недостаточно продуманном агрегировании показателей на предыдущих уровнях иерархии могут возникать ситуации, когда все рассматриваемые варианты при их очевидном различии имеют одинаковые оценки. Осуществить выбор в таких случаях невозможно.

Модифицированная технология ПАКС-М [Petrovsky et al, 2013], ориентированная на решение задачи выбора с помощью нескольких методов, предоставляет ЛПР возможности избежать возникновения противоречий на разных этапах процедуры агрегирования критериев. Сопоставление результатов, полученных несколькими методами для разных иерархических систем критериев, позволяет провести анализ итоговых результатов, сравнить системы критериев между собой, выбрать наиболее предпочтительную систему и оценить качество сделанного выбора.

1. Построение нескольких критериев оценки вычислительного комплекса

Рассмотрим решение задачи выбора вычислительного комплекса с помощью технологии ПАКС-М, построив несколько иерархических систем составных критериев оценки путем агрегирования исходных характеристик. Построим сначала иерархические системы с небольшим числом итоговых критериев, а затем объединим их в единственный интегральный показатель оценки перспективности вычислительного комплекса.

В качестве исходных характеристик комплексов были выбраны следующие группы показателей.

ХМ. Технические характеристики модуля (частота ядра процессора; разрядность ядра процессора; количество потоков; количество ядер процессора; объем поддерживаемой процессором оперативной памяти; количество процессоров в модуле; объем оперативной памяти модуля;

наличие ускорителя универсальных вычислений; дисковая память модуля; наличие в модуле оптического накопителя данных).

ВХ. Вычислительные характеристики комплекса (число модулей в комплексе; скорость обмена между модулями; наличие встроенных средств ввода-вывода; наличие бесперебойного питания; программные характеристики комплекса; возможность модернизации технических и программных средств комплекса).

КХ. Конструкционные характеристики комплекса (размеры комплекса (высота, глубина, ширина); масса комплекса; защищенность от помех)

ЭХ. Эксплуатационные характеристики комплекса (энергопотребление; уровень шума; тепловыделение; условия эксплуатации (температура, влажность); наработка на отказ).

ПК. Производительность комплекса.

СИ. Стоимость изготовления комплекса.

Для каждого исходного показателя была сформирована вербальная шкала оценок с двумя или тремя градациями. Например, производительность комплекса оценивалась как $ПК^0$ – высокая (>2000 Гфлопс); $ПК^1$ – средняя ($2000-500$ Гфлопс); $ПК^2$ – низкая (<500 Гфлопс).

Процедура агрегирования характеристик включает несколько этапов:

- разбиение исходных показателей на группы, в которых критерии близки друг другу по смыслу или являются неотъемлемыми характеристиками какого-либо составного критерия;
- построение дерева агрегирования на основе сформированных групп критериев;
- формирование порядковых шкал оценок для составных критериев с указанием диапазона оценок исходных показателей для каждой градации на шкале.

В работе [Лобанов и др., 2013] для выбора вычислительного комплекса были построены иерархические системы составных критериев, агрегирующие исходные показатели в пять итоговых критериев: ПК. Производительность комплекса; СИ. Стоимость изготовления комплекса; ВХ. Вычислительные характеристики комплекса; КХ. Конструкционные характеристики комплекса; ЭХ. Эксплуатационные характеристики комплекса.

При построении дерева агрегирования критериев предварительно были определены исходные показатели, которые рассматривались в качестве итоговых критериев. Такими показателями стали производительность комплекса и стоимость изготовления комплекса. Использование СИ и ПК как итоговых критериев связано с их особой ролью. СИ является единственным показателем в перечне, характеризующим материальные затраты на приобретение вычислительного комплекса, а ПК обеспечивает сравнение скорости вычислений

комплексов. Остальные исходные показатели комплекса были объединены в составные критерии. Показатель «Технические характеристики модуля» был включен как составная часть в вычислительные характеристики комплекса.

При агрегировании исходных показателей в составные критерии конструирование градаций шкал составных критериев выполнялось на основе комбинаций градаций оценок исходных показателей. Конструировать каждую градацию шкалы составного критерия можно различными способами, используя разные методы для построения самой шкалы составного критерия, выбирая разные числа градаций и диапазоны изменения переменных на градациях шкал критериев предыдущего или данного уровня иерархии исходя из предпочтений ЛПР или знаний эксперта.

При формировании системы критериев градации шкалы составного критерия конструировались тремя разными способами с помощью метода стратификации кортежей с назначением разных диапазонов изменения градации на шкале. Каждую такую систему критериев удобно рассматривать как выражение точки зрения некоторого ЛПР/эксперта. Иными словами, можно считать, что при использовании любой из иерархических систем критериев каждый вариант комплекса оценивался не одним, а тремя независимыми экспертами.

В представленном примере три вычислительных комплекса ВК1, ВК2 и ВК3 сравнивались по пяти критериям: ПК, СИ, ВХ, КХ, ЭХ. Результаты многокритериальной оценки комплексов даны на рисунке 1. Справа на рисунке приведен формат представления оценок по каждому критерию: столбец – оценки соответствующего комплекса, данные тремя экспертами, строка – оценки трех комплексов, данные соответствующим экспертом.

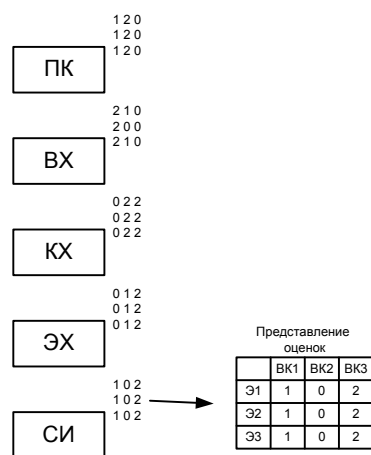


Рисунок 1 – Оценка комплексов по пяти критериям

Выбор наиболее предпочтительного комплекса проводился с помощью трех методов группового многокритериального выбора: метода АРАМИС, лексикографического упорядочивания по градациям критериальных оценок и метода взвешенных сумм рангов [Петровский, 2009]. Полученные этими

методами ранжировки были обобщены при помощи процедуры Борда. Итоговое упорядочение комплексов имело вид: $ВК1 > ВК2 \approx ВК3$. Таким образом, комплекс ВК1 оказался предпочтительнее комплексов ВК2 и ВК3, которые можно считать примерно равноценными.

Система с итоговыми критериями ПК, СИ, ВХ, КХ и ЭХ дала возможность выбрать наиболее предпочтительный вариант комплекса и объяснить результаты выбора, а также использовать эти критерии в качестве промежуточных при построении системы с единственным интегральным показателем агрегирования.

2. Построение единственного интегрального показателя оценки вычислительного комплекса

Рассмотрим теперь процедуру построения единственного интегрального показателя КК. Категория комплекса, характеризующего его перспективность и имеющего три вербальные градации оценок: $КК^0$ – перспективный комплекс, $КК^1$ – современный комплекс, $КК^2$ – устаревающий комплекс. Интегральный показатель формировался четырьмя разными способами путем агрегирования различных сочетаний промежуточных критериев, в качестве которых выступали приведенные выше критерии:

ПК. Производительность комплекса;
СК. Стоимость изготовления комплекса;
ВХ. Вычислительные характеристики комплекса;
КХ. Конструкционные характеристики комплекса;
ЭХ. Эксплуатационные характеристики комплекса.

Построенные системы агрегирования критериев были использованы для выбора наиболее предпочтительного вычислительного комплекса.

2.1 Первый способ агрегирования критериев

Критерии ПК и ВХ объединяются в составной критерий «Вычислительный потенциал комплекса» $ВП=(ПК, ВХ)$. Критерии КХ и ЭХ объединяются в составной критерий «Затраты на обслуживание комплекса» $ЗО=(КХ, ЭХ)$. Критерий СИ на данном этапе не включается в составной критерий ЗО, так как во-первых, предполагается, что приобретение или изготовление комплекса связано с разовыми затратами, а обслуживание комплекса требует периодических затрат (каждый месяц). Во-вторых, затраты на обслуживание комплекса приходятся на время, когда комплекс уже запущен в эксплуатацию и его владелец получает какую-то прибыль, за счет которой эти затраты будут компенсироваться.

Далее критерии ВП, ЗО и СИ объединяются в единственный интегральный показатель «Категория комплекса» $КК=(ВП, ЗО, СИ)$. Система составных критериев представлена на рисунке 2.

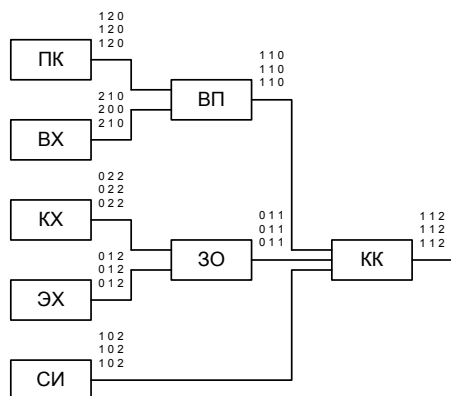


Рисунок 2 – Первый способ агрегирования критериев

Шкалы оценок для критериев ВП, ЗО и КК формировались следующим образом:

ВП. Вычислительный потенциал комплекса

$ВП^0$ – высокий (ПК⁰, ВХ⁰), (ПК⁰, ВХ¹), (ПК¹, ВХ⁰);
 $ВП^1$ – средний (ПК¹, ВХ¹), (ПК¹, ВХ²), (ПК², ВХ¹);
 $ВП^2$ – низкий (ПК², ВХ²).

ЗО. Затраты на обслуживание комплекса

$ЗО^0$ – низкие (КХ⁰, ЭХ⁰), (КХ⁰, ЭХ¹), (КХ¹, ЭХ⁰);
 $ЗО^1$ – высокие (КХ¹, ЭХ¹), (КХ¹, ЭХ²), (КХ², ЭХ¹), (КХ², ЭХ²).

КК. Категория комплекса

$КК^0$ – перспективный (ВП⁰, ЗО⁰, СИ⁰), (ВП⁰, ЗО¹, СИ¹), (ВП⁰, ЗО¹, СИ⁰), (ВП¹, ЗО⁰, СИ¹), (ВП¹, ЗО¹, СИ¹), (ВП¹, ЗО⁰, СИ⁰), (ВП⁰, ЗО⁰, СИ²);
 $КК^1$ – современный (ВП¹, ЗО¹, СИ¹), (ВП¹, ЗО⁰, СИ¹), (ВП⁰, ЗО¹, СИ²), (ВП², ЗО¹, СИ⁰), (ВП², ЗО⁰, СИ¹), (ВП², ЗО⁰, СИ²), (ВП², ЗО¹, СИ²).

Сравнение комплексов по интегральному показателю КК. Категория комплекса показало, что $ВК1 \approx ВК2 > ВК3$. Это не согласуется с результатами сравнения комплексов по пяти критериям и связано с тем, что введенный промежуточный критерий ВП уравнивал между собой комплексы ВК1 и ВК2, что привело к другому результату. Кроме того, критерии ВП и ЗО усложняют объяснение полученного результата. Построенная первым способом система агрегирования критериев не позволяет внятно объяснить результаты.

2.2 Второй способ агрегирования критериев

Критерии ПК, ВХ и КХ объединяются в составной критерий «Вычислительный потенциал комплекса» $ВП = (ПК, ВХ, КХ)$. Критерии ЭХ и СИ не группируются, так как они характеризуют денежные затраты на разных этапах.

Далее критерии ВП, ЭХ и СИ объединяются в единственный интегральный показатель «Категория комплекса» $КК = (ВП, ЭХ, СИ)$. Система составных критериев представлена на рисунке 3.

Шкалы оценок для критериев ВП и КК формировались следующим образом:

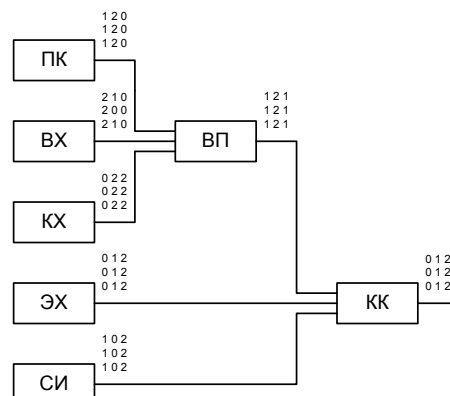


Рисунок 3 – Второй способ агрегирования критериев

ВП. Вычислительный потенциал комплекса

$ВП^0$ – высокий (ПК⁰, ВХ⁰, КХ⁰), (ПК⁰, ВХ⁰, КХ¹), (ПК⁰, ВХ¹, КХ⁰), (ПК¹, ВХ⁰, КХ⁰);
 $ВП^1$ – средний (ПК⁰, ВХ¹, КХ¹), (ПК¹, ВХ⁰, КХ¹), (ПК¹, ВХ¹, КХ⁰), (ПК¹, ВХ¹, КХ¹), (ПК⁰, ВХ⁰, КХ²), (ПК⁰, ВХ², КХ⁰), (ПК², ВХ⁰, КХ⁰), (ПК², ВХ⁰, КХ¹), (ПК², ВХ¹, КХ⁰), (ПК⁰, ВХ², КХ¹), (ПК¹, ВХ², КХ⁰), (ПК⁰, ВХ¹, КХ²), (ПК¹, ВХ⁰, КХ²);
 $ВП^2$ – низкий (ПК⁰, ВХ², КХ²), (ПК², ВХ⁰, КХ²), (ПК², ВХ², КХ⁰), (ПК¹, ВХ², КХ²), (ПК², ВХ¹, КХ²), (ПК², ВХ², КХ¹), (ПК², ВХ², КХ²).

КК. Категория комплекса

$КК^0$ – перспективный (ВП⁰, ЭХ⁰, СИ⁰), (ВП⁰, ЭХ⁰, СИ¹), (ВП⁰, ЭХ¹, СИ⁰), (ВП¹, ЭХ⁰, СИ⁰), (ВП⁰, ЭХ¹, СИ¹), (ВП¹, ЭХ⁰, СИ¹), (ВП¹, ЭХ¹, СИ⁰), (ВП⁰, ЭХ⁰, СИ²), (ВП⁰, ЭХ², СИ⁰), (ВП², ЭХ⁰, СИ⁰);
 $КК^1$ – современный (ВП¹, ЭХ¹, СИ¹), (ВП², ЭХ⁰, СИ¹), (ВП², ЭХ¹, СИ⁰), (ВП⁰, ЭХ², СИ¹), (ВП⁰, ЭХ¹, СИ²), (ВП¹, ЭХ⁰, СИ²), (ВП¹, ЭХ², СИ⁰);
 $КК^2$ – устаревающий (ВП⁰, ЭХ², СИ²), (ВП², ЭХ⁰, СИ²), (ВП², ЭХ², СИ⁰), (ВП², ЭХ², СИ¹), (ВП², ЭХ¹, СИ²), (ВП¹, ЭХ², СИ²), (ВП², ЭХ², СИ²).

При сравнении комплексов по показателю КК оказалось, что $ВК1 > ВК2 > ВК3$. Результаты оценки комплексов по показателю КК близки к полученным при оценке комплексов по пяти критериям. По шкале показателя КК комплекс считается перспективным, если общая сумма градаций оценок по промежуточным критериям не превышает значения 2 даже при наличии двух средних оценок или одной низкой оценки по промежуточным критериям.

Однако полученные результаты не совсем согласуются с результатами сравнения комплексов по пяти критериям. Построенная вторым способом система агрегирования критериев также не позволяет внятно объяснить результаты.

2.3 Третий способ агрегирования критериев

Критерии ВХ, КХ и ЭХ объединяются в составной критерий «Обобщенные характеристики комплекса» $ОХ = (ВХ, КХ, ЭХ)$. Критерии ПК и СИ не группируются.

Затем критерии ПК, СИ и ОХ объединяются в один интегральный показатель «Категория комплекса» $KK=(ПК, СИ, ОХ)$. Система составных критериев представлена на рисунке 4.

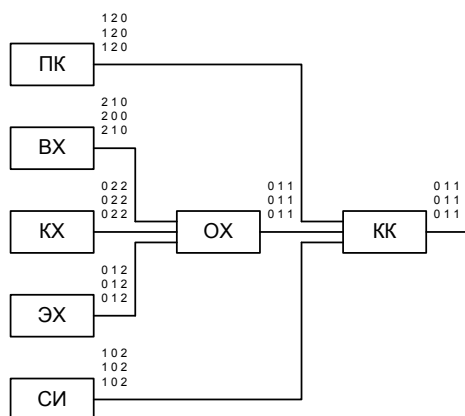


Рисунок 4 – Третий способ агрегирования критериев

Шкалы оценок для критериев ОХ и КК формировались следующим образом:

ОХ. Обобщенные характеристики комплекса

$ОХ^0$ – высокие $(ВХ^0, КХ^0, ЭХ^0), (ВХ^0, КХ^0, ЭХ^1), (ВХ^0, КХ^1, ЭХ^0), (ВХ^1, КХ^0, ЭХ^0), (ВХ^0, КХ^1, ЭХ^1), (ВХ^1, КХ^0, ЭХ^1), (ВХ^1, КХ^1, ЭХ^0), (ВХ^0, КХ^0, ЭХ^2), (ВХ^0, КХ^2, ЭХ^0), (ВХ^2, КХ^0, ЭХ^0);$

$ОХ^1$ – средние $(ВХ^1, КХ^1, ЭХ^1), (ВХ^2, КХ^0, ЭХ^1), (ВХ^2, КХ^1, ЭХ^0), (ВХ^0, КХ^2, ЭХ^1), (ВХ^1, КХ^2, ЭХ^0), (ВХ^0, КХ^1, ЭХ^2), (ВХ^1, КХ^0, ЭХ^2), (ВХ^0, КХ^2, ЭХ^2), (ВХ^2, КХ^2, ЭХ^0), (ВХ^2, КХ^0, ЭХ^2);$

$ОХ^2$ – низкие $(ВХ^2, КХ^2, ЭХ^1), (ВХ^2, КХ^1, ЭХ^2), (ВХ^1, КХ^2, ЭХ^2), (ВХ^2, КХ^2, ЭХ^2);$

КК. Категория комплекса

$КК^0$ – перспективный $(ПК^0, СИ^0, ОХ^0), (ПК^0, СИ^0, ОХ^1), (ПК^0, СИ^1, ОХ^0), (ПК^1, СИ^0, ОХ^0), (ПК^0, СИ^1, ОХ^1), (ПК^1, СИ^0, ОХ^1), (ПК^1, СИ^1, ОХ^0), (ПК^0, СИ^0, ОХ^2), (ПК^0, СИ^2, ОХ^0), (ПК^2, СИ^0, ОХ^0);$

$КК^1$ – современный $(ПК^1, СИ^1, ОХ^1), (ПК^0, СИ^1, ОХ^2), (ПК^1, СИ^0, ОХ^2), (ПК^0, СИ^2, ОХ^1), (ПК^1, СИ^2, ОХ^0), (ПК^2, СИ^0, ОХ^1), (ПК^2, СИ^1, ОХ^0);$

$КК^2$ – устаревающий $(ПК^1, СИ^1, ОХ^2), (ПК^1, СИ^2, ОХ^1), (ПК^2, СИ^1, ОХ^1), (ПК^0, СИ^2, ОХ^2), (ПК^2, СИ^0, ОХ^2), (ПК^2, СИ^2, ОХ^0), (ПК^1, СИ^2, ОХ^2), (ПК^2, СИ^1, ОХ^2), (ПК^2, СИ^2, ОХ^1), (ПК^2, СИ^2, ОХ^2);$

При сравнении комплексов по показателю КК оказалось, что $ВК1 > ВК2 \approx ВК3$. Сравнение по критерию ОХ показало, что комплекс ВК1 ненамного опережает комплексы ВК2 и ВК3, а те, в свою очередь, считаются равноценными. Сравнение комплексов по показателю КК дало результаты, согласующиеся с результатами сравнения комплексов по пяти критериям.

Полученные результаты можно объяснить следующим образом. По составному критерию ОХ комплекс ВК1 получает высокую оценку, так как, несмотря на низкую оценку по вычислительным характеристикам, он имеет лучшие оценки по конструкционным и эксплуатационным

характеристикам, в то время как комплексы ВК2 и ВК3 заметно уступают ему по этим параметрам. При агрегировании критериев ПК, СИ и ОХ в единый показатель агрегирования КК общая предпочтительность комплексов не меняется.

Построенная третьим способом система агрегирования критериев не содержит противоречий и дает понятное объяснение полученных результатов. При построении иерархической системы с одним интегральным показателем, удалось получить иерархическую систему с тремя промежуточными критериями ПК, ОХ, СИ. Оценки, характеризующие вычислительные комплексы по этой системе критериев, также согласуются с оценками по пяти критериям.

2.4 Четвертый способ агрегирования критериев

Все пять критериев сразу объединяются в единый показатель агрегирования «Категория комплекса» $КК=(ПК, ВХ, КХ, ЭХ, СИ)$. Система составных критериев представлена на рисунке 5.

При агрегировании пяти критериев шкала интегрального показателя КК. Категория комплекса формировалась следующим образом. Комплекс считался перспективным ($КК^0$), если сумма градаций оценок критериев, входящих в показатель КК, не превышала значение 4. Комплекс считался современным ($КК^1$), если сумма градаций оценок критериев, входящих в показатель КК, равнялась 5 или 6. Комплекс считался устаревающим ($КК^2$), если сумма градаций оценок критериев, входящих в показатель КК, превышала значение 6.

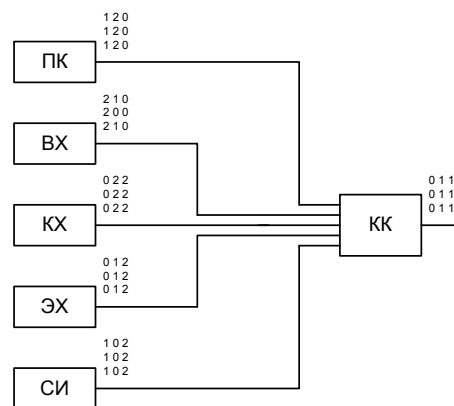


Рисунок 5 – Четвертый способ агрегирования критериев

При сравнении комплексов по показателю КК оказалось, что $ВК1 > ВК2 \approx ВК3$, что согласуется с результатами сравнения комплексов по пяти критериям и сравнением по третьей системе агрегирования критериев.

Полученные результаты можно объяснить следующим образом. Сопоставляя оценки комплексов по пяти критериям, можно заметить, что комплекс ВК1, по общему мнению экспертов, имеет две высокие оценки по критериям КХ и ЭХ (оценки по градациям соответствуют значению 0),

две средние оценки по критериям ПК и СИ (оценки по градациям соответствуют значению 1), а также одну низкую оценку по критерию ВХ (оценка по градации соответствует значению 2). Суммарное значение градаций оценок, присвоенных каждым из экспертов, равняется 4. Комплексы ВК2 и ВК3 по общей совокупности оценок уступают комплексу ВК1 и примерно равноценны.

В построенной системе агрегирования критериев есть только один недостаток – при объединении пяти критериев в один интегральный показатель достаточно сложно сформировать шкалу оценок с диапазоном значений для каждой градации порядковой шкалы показателя.

Заключение

При агрегировании критериев немаловажную роль играет установление семантических связей между исходными показателями и составными критериями. Неудачно сформированные отношения между ними могут привести к противоречиям при объяснении результатов. Действительно, две из четырех иерархических систем критериев с единственным показателем, построенные первым и вторым способом, дают результаты, которые различаются как между собой, так и отличаются от результатов оценки комплексов по пяти критериям. Напротив, результаты, которые получены по иерархическим системам критериев, построенным третьим и четвертым способом, согласуются между собой и с оценками комплексов по пяти критериям и имеют понятные объяснения.

Совпадение результатов для иерархических систем с различной степенью агрегирования критериев дает возможность ЛПР выбрать наиболее удобную систему критериев, либо совместно применять построенные системы критериев с целью повышения обоснованности получаемого решения при выборе наиболее предпочтительной конфигурации вычислительного комплекса.

Отметим также, что практически по всем построенным системам агрегирования критериев вычислительный комплекс ВК1 оказался наиболее предпочтительным по совокупности своих характеристик. Это свидетельствует о достаточно высокой надежности предложенного подхода к многокритериальному выбору сложной системы.

Работа частично поддержана Российским фондом фундаментальных исследований (проект 14-07-00916).

Библиографический список

- [Ларичев, 2006] Ларичев, О.И. Вербальный анализ решений / О.И. Ларичев, под ред. А.Б. Петровского // М.: Наука, 2006.
- [Лобанов и др., 2012] Агрегирование критериев в задаче выбора вычислительного кластера / Лобанов В.Н., Петровский А.Б. // Теория и практика системного анализа: Сборник трудов Всероссийской молодежной конференции. – Белгород: ИД «Белгород», 2012. – С.107-111.

[Лобанов и др., 2013] Выбор вычислительного кластера, основанный на агрегировании многих критериев / Лобанов В.Н., Петровский А.Б. // Вопросы радиоэлектроники, сер. ЭВТ, 2013. – Вып.2. – С.39-54.

[Петровский, 2009] Петровский, А.Б. Теория принятия решений / А.Б. Петровский // М.: Издательский центр «Академия», 2009.

[Петровский и др., 2012] Многокритериальный выбор с уменьшением размерности пространства признаков: многоэтапная технология ПАКС / Петровский А.Б., Ройзензон Г.В. // Искусственный интеллект и принятие решений, 2012. – № 4. – С.88-103.

[Ройзензон, 2005] Ройзензон, Г.В. Многокритериальный выбор вычислительных кластеров. / Г.В. Ройзензон // Методы поддержки принятия решений. Труды Института системного анализа Российской академии наук. / Под ред. С.В. Емельянова, А.Б. Петровского. Т.12. – М.: Едиториал УРСС, 2005. – С.68-94.

[Petrovsky et al, 2013] Selection of Computing Cluster by Aggregated Complex Criteria / Petrovsky A.B., Lobanov V.N., Zaboloeva-Zotova A.V. // Advances in Decision Technology and Intelligent Information Systems / Ed. by K.J. Engemann, G.E. Lasker. – Tecumseh: The International Institute for Advanced Studies in Systems Research and Cybernetics, 2013. – Vol. XIV. – Pp. 27-31.

CONSTRUCTION OF AN INTEGRAL INDEX FOR ASSESSING PROSPECTS OF COMPUTING COMPLEX

Lobanov V.N. *, Petrovsky A.B. **

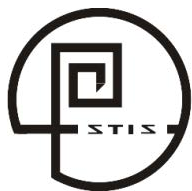
* *M.A. Kartsev Scientific Research Institute
for Computer Complexes Inc., Moscow, Russia*
fisher_1980@mail.ru

** *Institute for Systems Analysis,
Russian Academy of Sciences,
Moscow, Russia,*
pab@isa.ru

A comparison of different computing complex configurations and selection of the most preferable hardware and software options is a weakly formalized and ill-structured complex task. Computational complexes are characterized by a large number of indicators and selected by many criteria both quantitative and qualitative. The paper considers a construction of an integral index for assessing prospects of computing complex based on sequential aggregating initial attributes in the composite criteria system.

In order to choose the preferable computing complex, were used a multi-stage technology PAKS-M that is the modified version of the PAKS method (abbreviation of Russian words: Consequent Aggregation of Classified Situations). PAKS-M allows build several hierarchical trees with different degree of criteria aggregation by many methods of decision making. Using various trees of composite criteria allows us to simplify multicriteria comparison of complexes, select the most preferable system of criteria, and analyze the obtained results.

This work was partially supported by the Russian Foundation for Basic Research (project No 14-07-00916).



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.272:43+004.272.32

НЕКОТОРЫЕ ЗАКОНОМЕРНОСТИ И ОБЪЕКТИВНЫЕ ОГРАНИЧЕНИЯ РЕАЛИЗАЦИИ АЛГОРИТМОВ СЕМАНТИЧЕСКОЙ ОБРАБОТКИ НА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ С МАССОВЫМ ПАРАЛЛЕЛИЗМОМ

Вереник Н.Л.^{*}, Сейткулов Е.Н.^{**}, Гирель А.И.^{*}, Татур М.М.^{*}

^{} Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

nick.verenik@gmail.com

tatur@i-proc.com

*^{**} Евразийский национальный университет имени Л.Н. Гумилева
г. Астана, Казахстан*

seitkulov_y@enu.kz

В настоящей работе приводится описание разрабатываемой архитектуры проблемно-ориентированного семантического процессора, рассматривается формат данных и система команд процессора, основные принципы функционирования. Приводится ряд закономерностей и объективных ограничений архитектуры. Вкратце рассмотрен пример использования процессора для решения задачи поиска пути в графе.

Ключевые слова: семантическая обработка информации; семантическая сеть; проблемно-ориентированный процессор; массовый параллелизм.

Введение

Одним из наиболее перспективных способов формализации функционирования произвольной интеллектуальной системы является использование семантических сетей, с помощью которых могут быть представлены как знания системы (база знаний системы), так и алгоритмы интеллектуальной обработки информации, используемые данной системой. Семантическая сеть по своей сути является некоторой графовой структурой, элементы которой наделяются дополнительным смыслом. Вычислительная система, на базе которой функционирует семантическая сеть, как правило, является, так называемой, графодинамической машиной [Голенков и др., 2012] – программной либо аппаратной системой, внутреннее состояние которой представлено графом. Процесс обработки информации в такой системе трактуется как графодинамический процесс, т.е. процесс изменения внутренней графовой структуры системы, который может включать в себя не только изменение внутреннего состояния элементов графа, но и изменение конфигурации графа (добавление либо удаление вершин и дуг в графе).

На практике при создании прикладной интеллектуальной системы, как правило, используется программная модель семантической сети, реализованная на универсальной вычислительной системе (персональном компьютере), что, очевидно, существенно снижает общую стоимость системы и нередко является подходящим решением. Однако использование универсальных процессоров имеет ряд ограничений по масштабируемости [Amdahl, 1967], [Gustafson, 1988], [Воеводин, 2010] и не подходит для реализации сверх сложных систем. В случае такого рода систем, предъявляющих особо высокие требования к производительности либо максимально возможному объему знаний, могут быть использованы спецпроцессоры (суперкомпьютеры) [Hillis, 1989], [Kitano et al., 1992], [Каляев и др., 2011], однако это зачастую невозможно из-за большой стоимости таких систем. К другим явным недостаткам можно отнести массогабаритные характеристики суперкомпьютеров, повышенные затраты на обслуживание (выделение специального помещения, охлаждение и т.п.) и обслуживающий персонал.

Другим хорошо известным решением является использование универсальных параллельных

систем, таких как GPU [Brodtkorb et al., 2013], кластеров, облачных вычислений, что позволяет достичь гораздо большей производительности по сравнению с обычными универсальными процессорами при сохранении относительно низкой стоимости всей системы в целом. Одним из недостатков такого рода систем является тот факт, что изначально они создавались для решения совершенно другого класса задач, соответственно в случае интеллектуальных систем они зачастую не могут быть использованы с максимальной эффективностью.

Альтернативное решение, предлагаемое авторами настоящей работы, заключается в разработке проблемно-ориентированного процессора [Байрак и др., 2012]. Из-за изначальной ориентации на решение задач семантической обработки является возможным достижение лучшей эффективности по сравнению с универсальными решениями (универсальными процессорами и универсальными параллельными процессорами) при сохранении низкой общей стоимости. В статье, в частности, приводится краткое описание разработанной архитектуры, системы команд и формата данных процессора, рассматриваются объективные закономерности и ограничения реализации процессора.

1. Архитектура процессора

В работе [Вереник и др., 2012] было показано, как задачи семантического анализа могут быть сведены к задачам из теории графов, в частности как произвольная семантическая сеть может быть представлена некоторым графом регулярной структуры. В свою очередь, задачи из теории графов в большинстве своем относятся к комбинаторным задачам, которые на практике могут быть решены только за счет введения ряда ограничений в условия исходной задачи. Подобные ограничения вводятся до тех пор, пока время решения задачи не достигнет необходимого допустимого значения. В результате полученное решение имеет смысл только в контексте прикладной задачи, для которой оно было разработано.

С другой стороны, стоит задача обеспечения рентабельности разработки проблемно-ориентированного процессора, которая предполагает определенные временные, а значит и материальные затраты. Такого рода проект никогда не достигнет рентабельности, если будет ориентирован только на одну частную задачу. Следовательно, с самого начала в разработку архитектуры процессора необходимо заложить определенную унификацию, т.е. ориентировать процессор на решение определенного класса схожих задач [Байрак и др., 2012].

Авторами предлагается реализовать аппаратную поддержку базовых операций, используемых в алгоритмах из теории графов, в частности операций

поиска элементов графа и операции над множествами элементов графа. Фактически, процессор предлагаемой архитектуры может рассматриваться как сложная ассоциативная память, где операции чтения соответствует выполнение некоторого поискового запроса по графу.

На рис. 1 изображена структурная схема процессора. В качестве базовой архитектуры предлагается использовать SIMD-архитектуру (по классификации Флинна) магистрального типа. Из очевидных преимуществ данной архитектуры можно назвать простоту связей между модулями, наименьшие аппаратные затраты, высокую степень модульности и, что важно, возможность наращивания. Известные недостатки, такие как большое количество связей в схеме и связанная с этим невысокая надежность (в случае повреждения одной из шин данных), не являются существенными при реализации схемы на ПЛИС. Наибольшую проблему составляет ограниченная пропускная способность шин данных, в частности при чтении данных из памяти.

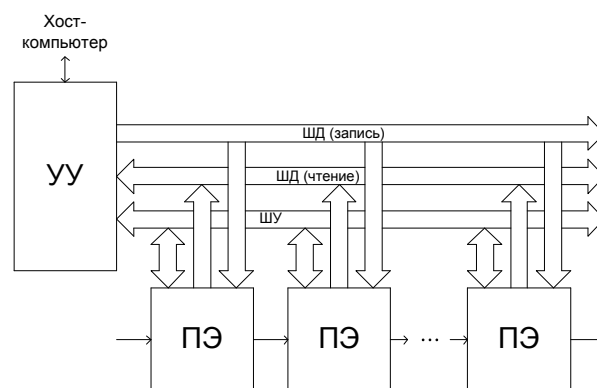


Рисунок 1 – Структурная схема процессора

На схеме процессор представлен одним общим устройством управления (УУ), которое считывает и декодирует последовательность команд исходной программы (одиночный поток команд), и множеством процессорных элементов (ПЭ), на которых в последствие параллельно исполняются эти команды. УУ взаимодействует с линейкой ПЭ через глобальные шины данных (ШД) трех видов:

- ШД записи – шина, по которой исполняемая команда параллельно транслируется на все ПЭ;
- ШД чтения – шина, по которой выполняется последовательное чтение данных с процессорных элементов (ШД с временным разделением);
- шина управления (ШУ) – множество управляющих сигналов.

Каждый ПЭ содержит свою собственную локальную память, за взаимодействие с которой отвечает только он один. ПЭ соединены между собой однонаправленными локальными связями, посредством которых осуществляется возможность приоритетного считывания данных из памяти.

Совокупность локальной памяти всех ПЭ составляет общую память процессора.

Рассмотрим общий механизм взаимодействия с памятью процессора. На данный момент система команд процессора представлена всего одной командой (см. рис. 2), используемой как для чтения, так и для записи данных в процессор.

одна ячейка	тип операции	M_A , маска-адрес	D_A , данные-адрес	M_D , маска-данные	D_D , данные-данные
-------------	--------------	---------------------	----------------------	----------------------	-----------------------

Рисунок 2 – Формат команды процессора

Для адресации (выбора одного или нескольких элементов памяти процессора, с которыми будет произведено действие) в состав команды входит битовая маска M_A и битовое поле D_A , размерность которых совпадает с разрядностью процессора. Если в ячейке находится значение D_0 , то она считается адресованной (выбранной, активной) в случае, когда все биты D_A и D_0 , отмеченные маской M_A , соответственно равны между собой, т.е. $D_{Ai} = D_{0i}$ для $\forall i, i = \overline{0, b-1}$ (где b – разрядность процессора), такого что $M_{Ai} = 1$.

Часто требуется (например, в случае операции чтения) гарантировано применить действие к не более чем одному элементу памяти. Для выбора этого режима используется бит “одна ячейка”. В этом случае, если была адресована более чем одна ячейка, выбор той, которая выполнит операцию, будет осуществлен по схеме приоритетов. Собственно приоритет определяется порядком следования в схеме. Для простоты можно считать, что сработает та ячейка, которая занимает наиболее левое положение в линейке ПЭ (см. рис. 1).

Операция записи данных в ячейку памяти также использует доступ через маску, в частности значение M_D определяет те биты ячейки, которые будут изменены в зависимости от текущего значения бита согласно некоторой функции $f(D_{0i}, D_{Di}) \Rightarrow D_{0i}$ для $\forall i, i = \overline{0, b-1}$, такого что $M_{Di} = 1$. Выбор функции осуществляется посредством поля “тип операции”. На текущий момент предполагается реализовать простое присваивание и, как минимум, основные функции бинарной логики (И, ИЛИ, НЕ).

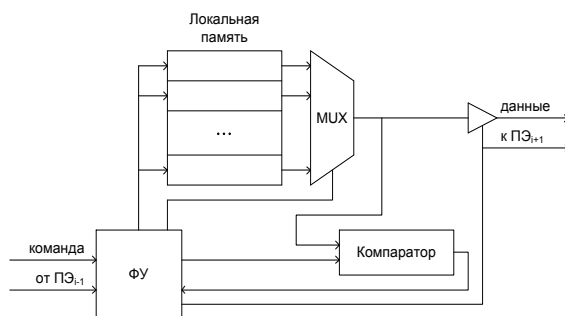


Рисунок 3 – Структурная схема ПЭ

На рисунке 3 представлена структурная схема ПЭ. В состав ПЭ входит локальная память и комбинационная схема, отвечающая за:

- последовательное сравнение ячеек памяти с входным значением (реализуется механизм адресации описанный выше);
- побитовую запись данных в ячейку (в соответствие со значением маски M_D);
- реализацию логической функции, вычисляющей значение для записи в ячейку;
- взаимодействие с соседними ПЭ (схема приоритетов);
- чтение данных.

2. Некоторые закономерности и объективные ограничения

Основная идея при построении процессора заключается в том, чтобы сохранить функциональную сложность ПЭ максимально простой, что в результате позволит реализовать огромное их количество на одном чипе. В то же время процессор должен быть в некотором роде универсальным (для обеспечения рентабельности проекта, как объяснялось ранее), что не может не повлиять на структуру ПЭ. В дополнение ко всему, требования предъявляемые конкретной прикладной системой, для которой изготавливается процессор, могут существенно отличаться от одной системы к другой. Некоторые системы требуют максимальной производительности (скорости выполнения), для других, в первую очередь, важен максимальный объем хранимой памяти, для третьих главным ограничивающим фактором будет являться стоимость.

Для решения данной проблемы предполагается поддержка процессором определенной настройки под конкретную прикладную систему (прикладную задачу), для которой этот процессор будет изготавливаться. Для предлагаемой архитектуры такого рода настройка может быть выполнена на нескольких концептуальных уровнях. В частности, можно выделить аппаратную настройку процессора (выбор разрядности данных процессора, количества ПЭ, объем локальной памяти) и программную настройку процессора (различная интерпретация формата данных процессора). Остановимся на аппаратной настройке процессора, рассмотрим некоторые закономерности и ограничения, присущие предлагаемой архитектуре.

1. Разрядность процессора b – в полной мере зависит от решаемой на системе задачи. Например, для хранения семантической сети в памяти процессора ячейка памяти должна иметь возможность вместить представление любого произвольного элемента сети. Так, для представления вершины графа необходимо хранить некоторый уникальный идентификатор вершины и ряд значений ее свойств (атрибутов). Для представления дуги графа – идентификаторы двух

инцидентных ей вершин и некоторые свойства дуги. Увеличение разрядности процессора b приводит к:

- увеличению объема локальной памяти ПЭ и, соответственно, увеличению схемы памяти ПЭ;
- увеличению комбинационной схемы управляющей логики ПЭ;
- уменьшению скорости исполнения управляющей логики ПЭ (дополнительные задержки на мультиплексорах и т.п.);
- увеличению размера команды (в 4 раза быстрее, чем разрядность b , при грубой оценке);
- увеличению количества шин данных и, вместе с тем, общей сложности трассировки схемы;
- росту стоимости одной ячейки памяти.

Стоимость одной ячейки памяти процессора можно принять равной стоимости изготовленного чипа деленной на общее количество ячеек памяти, которые и представляют собой основную ценность. Фактически, изменение стоимости (рост или падение) определяется отношением площади комбинационной схемы (КС) процессора (всей управляющей логики) к площади, занимаемой собственной элементами памяти (ЭП) процессора:

$$K = \frac{S_{КС}}{S_{ЭП}}.$$

Увеличение значения величины K приводит к росту стоимости одной ячейки памяти, уменьшение значение – к падению стоимости.

2. Количество элементов локальной памяти M , приходящееся на один ПЭ. Увеличение значения M приводит к:

- увеличению объема локальной памяти ПЭ и, соответственно, увеличению схемы памяти ПЭ;
- увеличению комбинационной схемы управляющей логики ПЭ;
- увеличению времени выполнения операции процессорным элементом (чем больше ячеек памяти содержится в ПЭ, тем больше времени требуется для их последовательной обработки);
- уменьшению производительности (количество операций выполняемых в единицу времени);
- падению стоимости одной ячейки памяти.

Очевидно, что для обеспечения максимальной производительности системы необходимо максимально уменьшить значение переменной M , т.е. уменьшить количество ячеек памяти, приходящееся на ПЭ и, соответственно, увеличить общее количество ПЭ на чипе. Однако это будет сопровождаться ростом стоимости одной ячейки памяти, что не всегда является допустимым.

3. Общее количество ПЭ на чипе N . В большинстве случаев является производной величиной от значения разрядности процессора b , размера локальной памяти M и максимальной допустимой площади схемы S_{MAX} , т.е. после выбора конкретного значения основных показателей

системы мы размещаем на чипе максимально возможное количество соответствующих ПЭ:

$$S = S_{КС} + S_{ЭП} \rightarrow S_{MAX}.$$

В общем случае увеличение количества ПЭ приводит к:

- увеличению общего объема памяти процессора;
- увеличению эффективности процессора (большой объем памяти обрабатывается за то же самое время);
- увеличению количества шин данных и, вместе с тем, общей сложности трассировки схемы.

Отметим, что максимальное количество ПЭ ограничено некоторым максимальным числом N_{MAX} , что обусловлено наличием локальных связей между ПЭ (задержками на последовательной схеме). Решение данной проблемы в настоящей статье не рассматривается.

3. Пример использования

Вкратце рассмотрим пример программной “настройки” процессора, основной смысл которой заключается в построении дополнительного уровня абстракции над системой команд процессора, который [уровень] жестко ориентирован на конкретную решаемую задачу. Для этого определяется свой собственный формат данных процессора (конкретные биты ячеек памяти наделяются нужным смыслом) и вводятся новые команды процессора, работающие в терминах решаемой задачи. Реализация такой расширенной системы команд может быть выполнена как прикладным программистом на уровне его среды разработки, так и системным программистом на уровне УУ процессора, который может представлять собой перепрограммируемый контроллер.

Used	Cell type	ID	Атрибуты вершины				
			v	n	f	distance _{min}	prevID
	0	k-1 0				m-1 0	k-1 0

Used	Cell type	ID1	ID2	Атрибуты дуги	
				c	cost
	1	k-1 0	k-1 0	m-1	0

Рисунок 4 – Формат данных процессора для решения задачи поиска кратчайшего пути в графе

В работах [Вереник и др., 2013], [Tatur et al., 2013] на базе программной модели предлагаемой архитектуры процессора была решена задача поиска кратчайшего пути в графе. Для решения задачи использовался параллельный алгоритм, разработанный на основе алгоритмы Дейкстры и волнового алгоритма. На рис. 4 показан выбранный формат данных процессора (ячейка памяти может содержать 2 различных типа данных – вершины и дуги), а ниже приведен расширенный список команд:

- инициализация (устанавливается начальное значение для всех ячеек памяти в соответствии с используемым алгоритмом)
- создать вершину с заданным идентификатором;
- создать дугу с заданным идентификатором и стоимостью перехода;
- установить для заданной вершины минимальное найденное расстояние и идентификатор предыдущей вершины (для возможности восстановить траекторию пути после);
- установить для заданной вершины флаг вхождения во фронт волны;
- считать вершину по ее идентификатору;
- считать следующую вершину из фронта волны;
- сдвинуть фронт волны;
- найти все дуги выходящие из заданной вершины;
- считать следующую дугу из найденных ранее.

Заключение

В настоящей работе была представлена архитектура процессора, предназначенная для решения задач на графах и, в частности, эффективного исполнения алгоритмов семантической обработки информации. Были рассмотрены основные принципы функционирования процессора, формат данных и система команд, структура процессорного элемента, некоторые закономерности и объективные ограничения реализации процессоров данной архитектуры. В частности, были описаны наиболее важные параметры архитектуры процессора, выбор значения которых и представляет собой настройку архитектуры для решения конкретной прикладной задачи.

В рамках разработки предложенной архитектуры следующим этапом исследования планируется реализовать архитектуру на существующих параллельных платформах, таких как GPU (технология CUDA), кластере и FPGA для оценки реального прироста производительности и эффективности при решении различных типовых задач из области семантического анализа по сравнению с активно используемыми на сегодня решениями.

Библиографический список

- [Байрак и др., 2012] Параллельные процессоры для построения интеллектуальных систем / С. А. Байрак, Д. Н. Одинец, М. М. Татур, Ф. Филипов, М. Мунос // Открытые семантические технологии проектирования интеллектуальных систем : материалы II Междунар. научн.-техн. конф. (Минск, 16-18 февраля 2012 г.) / редкол. : В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2012. – С. 135–140.
- [Вереник и др., 2012] Разработка проблемно-ориентированных процессоров семантической обработки информации / Н. Л. Вереник, Е. Н. Сейткулов, М. М. Татур // Электроника инфо. – 2012. – № 8. – С. 95–98.

- [Вереник и др., 2013] Имитационная модель векторного процессора на примере задачи поиска пути в графе / Н. Л. Вереник, А. И. Гирель, Е. Н. Сейткулов, М. М. Татур // Искусственный интеллект. – 2013. – №4. – С. 89–100.
- [Воеводин, 2010] Воеводин В. В. Вычислительная математика и структура алгоритмов. – М.: Издательство Московского университета, 2010. – 168 с.
- [Голенков и др., 2012] Графодинамические модели параллельной обработки знаний: принципы построения, реализации и проектирования / В. В. Голенков, Н. А. Гулякина // Открытые семантические технологии проектирования интеллектуальных систем : материалы II Междунар. научн.-техн. конф. (Минск, 16-18 февраля 2012 г.) / редкол. : В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2012. – С. 23–52.
- [Каляев и др., 2011] Высокопроизводительные реконфигурируемые вычислительные системы нового поколения / И. А. Каляев, А. И. Дордопуло, И. И. Левин, Е. А. Семерников // Вычислительные методы и программирование: новые вычислительные технологии. – 2011. – Т. 12. – №2. – С. 82 – 89.
- [Amdahl, 1967] Gene M. Amdahl. Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities // AFIPS Conference Proceedings. – Volume 30, April 18-20, 1967. – pp. 483–485
- [Brodtkorb et al., 2013] André R. Brodtkorb, Trond R. Hagen, Martin L. Sætra. Graphics processing unit (GPU) programming strategies and trends in GPU computing // Journal of Parallel and Distributed Computing. – Volume 73, Issue 1, January 2013. – pp. 4–13.
- [Gustafson, 1988] John L. Gustafson. Reevaluating Amdahl's Law // Communications of the ACM. – Volume 31, Issue 5, May 1988. – pp. 532–533.
- [Hillis, 1989] W. D. Hillis. The Connection Machine. – Cambridge : The MIT Press, 1989. – 208 p.
- [Kitano et al., 1992] Hiroaki Kitano, Dan Moldovan “Semantic Network Array Processor as a massively parallel computing platform for high performance and large-scale natural language processing”, Proc. Int'l Conf. on Computational Linguistics (COLING '92), vol. 2, pp. 813–819, 1992.
- [Tatur et al., 2013] M.M. Tatur, Y.N. Seitkulov, N.L. Verenik, A.I. Girel "Pathfinding on a Specialized Vector Processor", Proc. Int'l Conf. Parallel and Distributed Processing Techniques and Applications (PDPTA '2013), vol. 1, pp. 711–716, 2013.

SOME REGULARITIES AND OBJECTIVE LIMITATIONS WHEN IMPLEMENTING SEMANTIC PROCESSING ALGORITHMS ON COMPUTER SYSTEMS WITH MASSIVE PARALLELISM

Nick L. Verenik*, Yerzhan N. Seitkulov**,
Alexey I. Girel*, Mikhail M. Tatur*

* *Belarusian State University of Informatics and
Radioelectronics, Minsk, Republic of Belarus*

nick.verenik@gmail.com
tatur@i-proc.com

** *L.N. Gumilyov Eurasian National University,
Astana, Kazakhstan*
seitkulov_y@enu.kz

In the article description of semantic ASIP architecture which is in development is given, processor's data format and instruction set, basic principles of functioning are considered. Several architecture limitations and objective regularities are given. Usage example of processor for solution of the path in graph problem is briefly considered.

Key words: semantic information processing; semantic network; ASIP; massive parallel processing.

Introduction

Using of semantic networks is one of the most perspective approaches to formalize functioning of custom intelligent system. They can be used to represent both system's knowledge (knowledge database) and algorithms of intelligent information processing which are used in this system.

In practice when building applied intelligent systems then program model of semantic network is used and usually implemented on general-purpose computer system (PC). Obviously this approach significantly decreases the total cost of the system and frequently appears as a suitable solution. However using of general-purpose processors has number of scalability limitations. When implementing really complex intelligent systems, which requires especially high performance or huge value of knowledge database, supercomputers can be used. But this is often impossible because of high cost of such systems, weight and size characteristics or difficulties in maintenance.

Another well-known solution is using of general-purpose parallel systems such as GPU, clusters, and clouds computing. It allows reach much more performance in comparison with common general-purpose processors and keep relatively low cost of the entire system. The main disadvantage of this approach is the fact that such computer systems were developed to solve different kind of problems but semantic processing and as the result it's very difficult to achieve maximum of efficiency.

Authors have proposed alternative solution which is development of ASIP. Its problem orientation allows achieve better performance and efficiency in comparison with general-purpose processors (including parallel processors) while keeping low total cost. In the article brief description of developed architecture, its instruction set and data format are given, objective regularities and limitations of implementation are given.

Main Part

It's known that problems of semantic analysis can be reduced to problems in graph theory. Then it's known that problems in graph theory are mostly combinatorial problems and in practice they can be solved only after some limitations in original problem are introduced. Such limitations are introduced until the duration of problem solution reaches the required value. As the result the final solution has sense only in context of this specific applied problem.

On the other hand there's problem to ensure profitability of ASIP development which supposes some time and hence the material spending. Such project will never reach profitability if it's focused on only one specific applied problem. Therefore from the beginning the certain unification should be taken as basis while developing processor architecture, i.e. there's the

necessarily to orient it on the solution of some kind of similar problems.

Authors have proposed to implement hardware support of basic operations used in graph theory algorithms, in particular graph elements searching and working with set of graph elements. Practically, the processor with proposed architecture can be considered as a complex associative memory where read operation corresponds to execution of some search request within graph structure.

Schematic diagram of processor is presented in fig.1. The original architecture belongs to SIMD-kind where processor elements (PEs) are connected by three global buses (data bus for reading, data bus for writing and control bus) and locally connected with each other by unidirectional links (to perform mechanism of reading in order of priority). Each PE contains some value of local memory and is responsible for access to it. The approximate schematic diagram of PE is presented in fig.2.

The basic idea of processor is to provide bitwise access to memory cells which means to address memory by comparison of selected bits in cell and in input command. Selection mechanism is implemented via special mask field in command (see command format in fig.3). Writing operation is performed by bits as well.

The processor architecture is designed to support two-level customization. First of all number of low level parameters can be adjusted – such as processor capacity, number of memory cells per one PE, number of PEs. The regularities while changing these parameters are considered in the article. The next level of customization is performed by application or system engineers and includes the introduction of custom data format and high level instruction set which are completely determined by and formulated in terms of the problem to be solved. In fig.4 the example of such customization is presented where path finding problem is solved.

Conclusion

In the given paper the result of semantic processor development is presented. The description of proposed architecture, instruction-set and data format, basic principles of functioning are given. The solution of path finding problem is considered as the example of processor usage.

Further study and simulations are necessary for definition the problems to be solved on the proposed architecture. As the next step simulations on existing parallel systems are expected – such as GPU (CUDA technology), clusters and FPGA.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ УПРАВЛЕНИЯ НА БОРТУ ОТЕЧЕСТВЕННЫХ СПУТНИКОВ: ФАНТАСТИКА ИЛИ РЕАЛЬНОСТЬ

Димитров Д.М. *, Данилин Н.С. **, Сабиров И.Х. *

* ЗАО «Космос Комплект», г. Москва, Российская федерация

ilias@coscom.ru

office@dimitrov.ca

** ОАО «Российские космические системы», г. Москва, Российская федерация

N_dainilin@rambler.ru

Для обеспечения надежности и эффективности современных и будущих непилотируемых миссий в космическом пространстве необходимы новые технологии бортового управления и контроля. В первую очередь речь идет о внедрении интеллектуальных систем как на спутниках разного назначения, так и в соответствующей наземной инфраструктуре. В настоящей работе предлагается вариант обобщенной архитектуры автономной системы обнаружения и устранения нештатных ситуации (АСОУНС) для космических аппаратов (КА) с длительным сроком активного существования (САС).

Введение

В историческом плане как отечественные, так и миссии НАСА для беспилотных КА, были обусловлены разработкой спутника для одного типа полезной нагрузки или небольшого числа связанных между собой приборов. При этом из КА к наземной аппаратуре идет поток необработанных (или частично обработанных) данных, которые используются для управления миссией и для решения поставленных практических задач. Принятая в мировой практике методология анализа миссии с функциональной точки зрения выводит на первый план следующие ее характеристики, которые существенны для настоящей работы (1):

- уровень автономности управления (бортовая система управления по возмущающим воздействиям и по замкнутому циклу, бортовая система с встроенной программой, ситуационная система управления, интеллектуальное управление и др.);
- тип орбиты – геостационарная, низкая околоземная орбита, междупланетная орбита, эллиптическая орбита, L2 и др.;
- тип нагрузки – коммерческого, военного, научного, технологического и демонстрационного назначения;

- местоположение автономизации – на борту КА, в наземном оборудовании или в обоих местах;
- тип поколения КА – действующая разработка, текущая разработка, планируемый эксперимент, НИИР.

Основное внимание в работе уделено уровню автономности управления КА. В пионерских разработках В. фон Брауна бортовое радиоуправление реализовано как ручное с Земли. Позднее, в отечественных КА на достаточно раннем этапе развития космической промышленности, благодаря Н.А. Пилюгину была введена бортовая система управления по возмущающим воздействиям и по замкнутому циклу, что явилось важным шагом к автономизации проектируемых миссий. К настоящему времени все российские проекты беспилотных КА эксплуатируют с различной степенью технологической сложности базовую идею Н.А. Пилюгина, а в тоже время в США ведется работа по «автономным думающим КА» (Autonomous Thinking Spacecraft) (2).

Многие из настоящих и будущих отечественных миссий либо весьма сложные с технической точки зрения, либо требуют очень быстрой реакции с Земли, либо связаны с большими коммуникационными перерывами и т.д. Это неминуемо приводит к необходимости наполнения бортового управления КА интеллектом.

Важнейшая функция бортового управления это обеспечение надежного выполнения миссии как по времени, так и по функциональным требованиям. В связи с этим автономное (без или с минимальным участием экспертов) выявление и устранение нештатных ситуаций имеет стратегическое значение.

В настоящей работе предлагается вариант обобщенной архитектуры АСОУНС для КА с длительным САС.

Надежность КА и нештатные ситуации на борту

Вопросы надежности КА находятся в центре внимания специалистов и руководства космической отрасли на всех уровнях. Во всех ведущих фирмах организованы соответствующие структуры по надежности ЭКБ, испытательные центры и др. Помимо того, большое внимание этому вопросу уделено со стороны Роскосмоса, МО РФ и ряда федеральных целевых программ в области микроэлектроники и приборостроения. Тем не менее, несколько отечественных миссий, только в последнее время, перетерпели существенные неудачи. Одним из главных вопросов, возникающий при этом является поиск причин для такого рода проблем – то ли они связаны с эффективностью системы контроля качества и надежности от проекта КА до ЭКБ и комплектующих изделий, или это так называемый «человеческий фактор».

Подход к ответу на этот вопрос дают анализы, проведенные в NASA (JPL, Ames Research Center, Goddard Space Flight Center) после провала нескольких американских миссий – Geneis, Columbia, Mars Polar Lender, Mars Climate Orbiter, Mars Observer и др. Было выявлено, что даже при близкой к 100% надежности ЭКБ и комплектующих изделий, вместе с минимизацией «человеческого фактора» при реализации проекта, вероятность возникновения нештатной ситуации на борту КА, у которого САС более 7 лет превышает 20 %. С учетом факторов воздействия космического пространства, указанный риск может достигнуть в отдельных случаях и до 30-40%. В связи с этим требуется обнаружение нештатных ситуаций на борту в режиме реального времени с последующим устранением их возможных последствий в бортовой аппаратно-программной среде. Типичный пример применения АСОУНС это система диагностики Livingstone 2 (L2) для спутника КА EO-1 (Earth Observing One) в программе New Millenium Program (3).

Анализ интеллектуальных систем управления КА и вариант реализации АСОУНС

Настоящий анализ относится к разработкам ведущих мировых производителей беспилотных КА в США. Достигнутые ими результаты полностью

применимые в текущих и будущих отечественных проектах и безусловно окажется фактором повышения их эффективности и конкурентной способности в глобальном масштабе. Данные анализа получены от индустриального обзора SOSTC (Space Operation and Support Technical Committee) (4). Обзор охватывает 12 производителей КА за период 2006-2007 гг. и включает 88 проектов. Следует отметить, что межведомственный совет по передовым концепциям (Advanced Concepts Subcommittee) к SOSTC определил шесть уровней интеллектуализации на борту КА, где самый высокий уровень это автономно думающий КА. В анализе учтены проекты соответствующие уровням «самозащита от нештатных ситуаций», «самооптимизация бортовых ресурсов» и «самоконфигурация управляющих устройств» (5). Существенное значение для отечественной практики имеют указанные в обзоре проекты как New Millenium Program's Space Technology 5, Solar-Terrestrial Relation Observatory, Magnetospheric Multiscale Tracing and Data Relay Satellites, Geospace Electrodynamics Connections, Laser Interferometric Space Antenna, Constellation X, Magnetotrail Constellation и др.

В связи с АСОУНС можно выделить два методологических подхода при реализации уровня интеллигентности «самозащита от нештатных ситуаций» на борту КА – применение диагностических моделей (Model Based Diagnosis-MBD) (6) и применение ситуационно-определяемые модели (Case Based Reasoning Diagnosis - CBR) (7). Сравнительный анализ между MBD и CBR представлен в (8).

С учетом текущих исследований и разработок в области бортовых интеллектуальных систем управления компании ЗАО «Космос Комплект», для аппаратно-программной реализации выбрана CBR – модель. Для этой цели разработана дискретная модель, которая использует базу данных о сценариях функционирования программно-аппаратной среды КА в нештатных ситуациях и связанные с ними диагностические и коррекционные решения. Каждый компонент базы данных в своем описании может содержать вложенные компоненты более низкого иерархического уровня. Из за дискретного характера модели, ее переменные могут принимать значения из конечного множества значений качественных характеристик описания ситуаций, как например «низкое значение», «высокое значение», «допустимое значение» и т.п., а ее структурные компоненты могут находиться в нескольких состояниях – «включено», «выключено», «возник сбой», «сбой устранен» и т.д. Симптоматические признаки нештатного состояния КА определяются в режиме реального времени в терминах телеметрической информации и инициализируют поиск в ситуационной базе данных по нескольким возможным алгоритмам (процессор L2 на рис.1). В самом общем виде АСОУНС функционирует как последовательность нескольких процедур – синтез

телеметрического образа нештатной ситуации по симптоматическим признакам, поиск аналогичной ситуации в базе данных, верификация ее пригодности к текущему применению и последующие автономные действия по устранению нештатной ситуации. Существенно отметить, CBR - подход может проводить диагностирование как аппаратуры КА, так и его программного обеспечения. Обобщенная структурная схема АСОУНС представлена на рисунке 1.

В основе аппаратного решения для АСОУНС, заложен радиационно-стойкий компьютерный модуль формата PC104, известный как «Aeroflex

Lego» и разработанный в 2011 г. компанией ЗАО «Космос Комплект» совместно с корпорацией «Аэрофлекс» (США) для нужд отечественного космического приборостроения (9). Механическая конструкция модуля на 5 посадочных местах и возможности по питанию (до 40 Вт), позволяют реализовать двухпроцессорную систему на основе нового микропроцессора семейства Leon 3 FT UT 700 (с тактовой частотой 166 МГц и производительностью 1.2 DMIPS/MHz). Блок-схема аппаратного модуля АСОУВС показана на рис. 2.

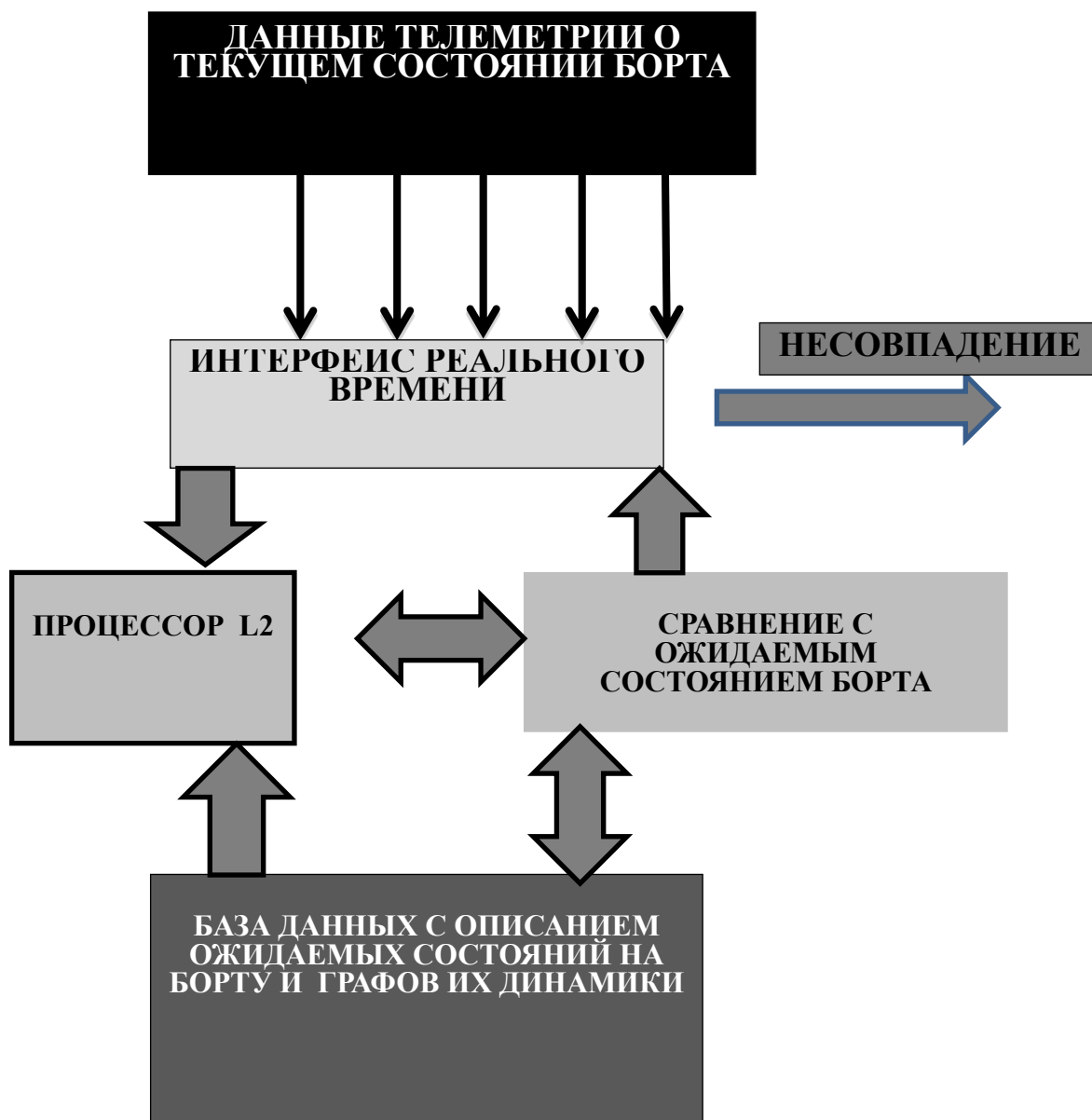


Рисунок 1 - Обобщенная структурная схема АСОУНС

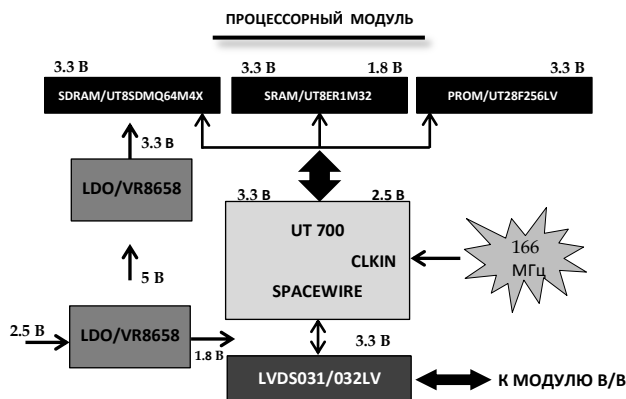


Рисунок 2 - Блок схема аппаратного модуля АСОУНС

Действующий макет АСОУНС в своем первом варианте предусматривает диагностику и устранение нештатных ситуаций для платформ отечественных микроспутников. Дополнительные усилия со стороны разработчиков потребуются при формализации бортовых сценариев нештатных ситуаций, а также при дальнейшей оптимизации системы правил и критериев для нахождения аналогичных ситуаций. ЗАО «Космос Комплект» планирует завершить технологическую версию проекта АСОУНС до конца 2014 г. и популяризировать его среди широкого круга специалистов заинтересованных в применении систем искусственного интеллекта на борту КА.

Заключение

Как показывает мировая практика, разработка и применение систем типа АСОУНС не фантастика, а реальность для любого проекта современного КА как с точки зрения выполнения его миссии на заданном функциональном уровне, так и с точки зрения сохранения инвестиций по его реализации.

В отечественной промышленности реально существует серьезный научно-технический потенциал по развитию тематики «интеллектуальные беспилотные КА». Активная работа в этом направлении является гарантией для высокой конкурентности нашей космической промышленности.

Библиографический список

- [Wartz J., Lorson W., 1999] Space Mission Analysis and Design. Springer.
- [Sherwood R., et.al., 2006] Enhancing Science and Automating operation using Onboard Autonomy.
- [Hayden S., Christa S., Shulman S., 2005] Advanced Diagnostic System on Earth Observing One, American Institute of Aeronautics and Astronautics, 2005.
- [LaVallee D., Jacobson J., Olsen C., 2006] Intelligent Control for Spacecraft Autonomy- An Industry Survey. AIAA 2006-6917.
- [Bujewski T., Turner S., Bush G.] Automation in Satellite TT&C Systems – A Survey of International Best Practices and Lessons Learned. IEEEAC, paper #1610.
- [Hayden S., Christa S., Livingstone, 2002] Model-Based Diagnosis of Earth Observing One, American Institute of Aeronautics and Astronautics.
- [Riesbeck C., Schank R., 1999] Inside Case-based Reasoning. Erlbaum Northvale.
- [Price C., 2002] Computer Based Diagnostic System. University of Wales.

[Димитров. Д.М., Данилин Н.С., Сабиров И.Х.] «Универсальная система модулей класса QML Q/V «Aeroflex Lego» для построения радиационно-стойких систем космической аппаратуры. Научно-техническая конференция «Высоконадежные системы управления», Москва, Измайлово, 2010 г.

ON-BOARD INTELLIGENT CONTROL SYSTEMS OF RUSSIAN SATELLITES: SCIENCE FICTION OR REALITY.

Dimitrov D.M. *, Danilin N.S. **, Sabirov I.Kh. *

* ZAO Cosmos Complect, Moscow Russian Federation

ilias@coscom.ru

office@dimitrov.ca

** JSC RSS, Moscow, Russian Federation

N_danilin@rambler.ru

Introduction

In order to address future unmanned space missions efficiency and support reliability of current designs new on-board control technologies are required. This is connected first of all with on-board and ground infrastructure intelligent control and management systems implementation.

Several aspects of the autonomic system for undefined on-board situations recovery (ASUSR) connected to the long life missions are discussed in the presented paper.

Main Part

The mission reliability and efficiency is a question of satellite design technology, parts quality, integration procedures, hardware and software testing etc. The analysis show that satellite reliability is also connected to so called “undefined on-board situations (UBS)” which can arise even in case of qualified parts use and minimum “human factor” influence. The probability of such situations comes to more than 20% for missions with more that 5 years of life in orbit. The existing satellite design practice does allow ASUSR implementation.

Case Based Reasoning Diagnosis methodology was chosen for ASUSR realization in the presented work. ASUSR prototype is a real time multiprocessor system (two radhard microprocessors UT 700 from Aeroflex Inc.). The application software includes data base with possible situations descriptions and real time anomaly detection and recovery modules.

ASURS is just one of the subsystems set which allows to organize intelligent on-board control and management. The future works in that direction will be defined after ASURS flight testing.

Conclusion

The world practice in robotic satellites design is the best approval for real needs in intelligent on-board control.

The current study of the Russian space industry and available technologies show that ASUSR application is not a science fiction but reality.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822

ОНТОЛОГИЯ РОССИЙСКОГО СЕГМЕНТА МЕЖДУНАРОДНОЙ КОСМИЧЕСКОЙ СТАНЦИИ И ЕЁ ПРАКТИЧЕСКОЕ ИСПОЛЬЗОВАНИЕ В ИНТЕЛЛЕКТУАЛЬНЫХ АЭРОКОСМИЧЕСКИХ ПРИЛОЖЕНИЯХ

Вакурина Т.Г. *, Котеля В.В. *, Лахин О.И. **, Матюшин М.М. *, Скобелев П.О. **

** ОАО Ракетно-космическая корпорация «Энергия» имени С.П. Королева,
г. Королёв, Россия*

*vlad@scsc.ru
tvakurina@yandex.ru
matushin@scsc.rus*

*** ООО «Научно-производственная компания «Разумные решения»
г. Самара, Россия*

*lakhin@smartsolutions-123.ru
skobelev@smartsolutions-123.ru*

Рассматриваются методы и средства описания концептуальной модели международной космической станции с помощью онтологий и ее практическое использование в интеллектуальных аэрокосмических приложениях.

Ключевые слова: российский сегмент международной космической станции; онтология; конструктор онтологий, моделей и сцен; мультиагентные технологии; адаптивное планирование; реальное время.

Введение

Международная космическая станция (МКС) представляет собой уникальный, сложный, инновационный научно-технический и инженерный орбитальный комплекс большой размерности. В настоящее время в связи с усложнением конструкции МКС и устойчивым возрастанием объемов информации, требуемой для управления полетом российского сегмента международной космической станции (РС МКС) со стороны центров управления полетом (ЦУП), необходимо развитие процессов управления полетом РС МКС в рамках единой наземно-космической системы, направленной на достижение комплекса целей полета:

- обеспечение в любых ситуациях спасения жизни экипажа;
- обеспечение условий жизнедеятельности экипажа, по возможности, приближенные к земным;
- обеспечение безопасной, надежной и эффективной эксплуатации МКС;
- обеспечение реализации программ научно-прикладных исследований, проводимых на борту РС МКС;

- выработка согласованного и оптимального решения, принимаемого ответственными специалистами группы управления полетом, выраженного в планах, методиках, указаниях экипажу, управляющих воздействиях на бортовые системы и т.д.

Так, для принятия решения и согласованных действий специалистов группы управления полетом и экипажа РС МКС требуется обработка и анализ возрастающего объема информации, поступающего с борта РС МКС, например, при управлении космическим кораблем «Восток» необходимо было анализировать около 400 телеметрических параметров, а при управлении полетом РС МКС необходимо анализировать уже более 80 000 телеметрических параметров.

Для решения этих задач в 2010 году была предложена концепция мультиагентного планирования, позволяющая учитывать специфику управления полетом РС МКС и повысить эффективность собственно процесса принятия решения за счет внесения элементов самоорганизации.

Инструментом информационной поддержки принятия решений в процессах управления полетом

РС МКС и описания контекста ситуации выступают онтологией, используемые для формальной спецификации понятий и отношений, характеризующих предметную область. Такой подход позволяет справляться с растущей сложностью данной задачи и высокой динамикой изменений, возникающей как в связи с непредсказуемым появлением новых срочных потребностей в доставке грузов на борт РС МКС, переносом стартов транспортных грузовых кораблей «Прогресс» или транспортных пилотируемых кораблей «Союз», расширением номенклатуры объектов грузов, так и в процессах оперативного управления полетом РС МКС, требующих незамедлительного принятия решений в случае возникновения различных событий при парировании нештатных и аварийных ситуаций на борту РС МКС.

Все это приводит к необходимости разработки качественно новых систем принятия решений, функционирование которых будет основано на накоплении и использовании изменяющихся знаний о РС МКС.

В данной статье мы рассмотрим механизм описания концептуальной модели РС МКС, приведем ее описание на основе онтологии, а также сделаем краткий обзор систем, применяемых для принятия решений в процессе управления полетом РС МКС, использующих онтологию РС МКС.

1. Онтология РС МКС

Онтология позволяет строить концептуальные модели РС МКС и формировать онтологические модели ситуаций, используемых для ситуационного управления полетом РС МКС.

Применение онтологий при разработке систем, используемых для управления полетом РС МКС, позволяет формализовать специфические предметные знания в виде, допускающем компьютерную обработку, и отделить эти знания от программного кода системы. Это позволяет в более широких пределах настраивать системы на возникающие новые для них ситуации, связанные с усложнением конструкции РС МКС и процессов управления полетом РС МКС, и далее поддерживать и развивать эти системы без перепрограммирования.

Кроме того, рассматриваемые знания, описанные в онтологии РС МКС, и полученные на их основе факты могут использоваться разрабатываемыми системами для выполнении рассуждений по формированию и обоснованию планов действий и других принимаемых решений в процессе управления полетом РС МКС.

Для формализованного представления знаний наибольшее распространение на данный момент получили подходы на основе фреймов, семантических сетей, логических и продукционных правил [Гаврилова, 2000].

Каждый подход обладает своими преимуществами и недостатками, однако на текущий момент семантические сети признаны наиболее универсальными и удобными формами концептуализации знаний, поэтому для построения онтологического базиса концептуальной модели РС МКС в дальнейшем будут использоваться семантические сети.

1.1. Онтологический базис

Семантические сети состоят из узлов и упорядоченных отношений (связей), соединяющих эти узлы. Узлы выражают понятия или предположения, а связи описывают взаимоотношения между этими узлами.

Для представления знаний обычно используют неоднородные семантические сети [Nuhns, 1997], в которых присутствуют разные типы отношений. Различают экстенциональные и интенциональные семантические сети. Интенциональные сети задаются посредством обобщающих концептов, экстенциональные сети – посредством перечисления всех экземпляров интенциональной сети.

Развитием подхода для представления знаний стали динамические семантические сети. В предлагаемом методе предоставляется возможность строить сетевые интеллектуальные системы или сети интеллектуальных систем. В основу концепции динамической семантической сети положены следующие принципы: интеграция процедурных и декларативных знаний; параллельное функционирование всех компонентов сети; эволюция сети в реальном времени. Каждый узел динамической семантической сети является выполняемым вычислительным процессом, обладающим алгоритмами обработки информации и выработки решений, средствами общения с другими узлами сети, определенным поведением. Таким образом, каждый узел графа представляет собой выполняемый процесс, характеризуемый набором атрибутов и присоединенных функций, а структура графа отражает текущую модель предметной области.

В развитие идеи универсального базиса для представления знаний, где важной частью является созидательная деятельность людей предметной области, которая изначально рассчитана на представление процессов и действий пользователей, нами предложена концептуальная модель (мета-онтология, или модель Аристотеля) [Скобелев, 2012].

Предлагаемая базовая модель позволяет описывать не только декларативные знания о предметной области, понятия и сущности «мира», но и процедурные знания, деятельную компоненту, представляющую сценарии действий над объектами, выражающими законы мира, свойства или функции объектов, или действия субъекта над объектами мира. Кроме того, модель мира всегда предполагает наличие некоторой модели

пространства и времени, в рамках которого существуют и взаимодействуют все объекты мира, над которыми можно выполнять действия.

Предлагаемая «мета-онтология Аристотеля» является общей для всех миров, как физических, так и абстрактных и включает следующие концепты: «объекты» - сущности, характеризующиеся свойствами, «свойства», отражающие способность объектов вступать во взаимодействия, «процессы» - цепочки действий по изменению состояний объектов, «отношения», позволяющие связывать объекты и конструировать сложные объекты из простых и «атрибуты», характеризующие состояния концептов. Мета-онтология Аристотеля предполагает следующие важные свойства:

- существуют объекты, которые обладают свойствами и характеризуются состояниями;
- с каждым объектом можно что-то делать, изменяя состояние, свойства или отношения между объектами;
- отношения между объектами могут отражать структурные, функциональные, временные или любые другие виды связей;
- чтобы выполнить действие над объектом, необходимо соблюдение определенных условий, которые задаются свойствами и отношениями;
- сложные объекты строятся из простых объектов путем выполнения действий (процессов) над ними для установления отношений и связывания простых объектов в сложные;
- свойства выражают способность объектов вступать в процессы взаимодействия на основе законов мира;
- события, действия (процессы) изменяют состояния объектов, их свойства и отношения и запускают новые процессы;
- процессы состоят из действий с объектами, так же как сложные объекты состоят из простых;
- с каждым объектом мира можно что-то делать в любой момент времени (нет тупиковых состояний), но чтобы выполнить действие над объектом необходимо выполнить определенные условия;
- объекты, свойства, отношения, процессы (действия) характеризуются атрибутами различных типов, которые имеют диапазоны значений и конкретные значения в заданный момент времени;
- атрибуты объекта/отношения являются качественной или количественной характеристикой понятия;
- правила являются обобщенными понятиями для формализованных условий вида «если-то» (предикатов) и высказываний (утверждений, аксиом, фактов).

Предлагаемая нами «мета-онтология

Аристотеля» позволяет описать концептуальную модель РС МКС.

1.2. Концептуальная модель РС МКС

На основе предлагаемого нами онтологического базиса можно описать концептуальную модель РС МКС, при этом описываются все возможные конфигурации РС МКС с состояниями объектов в заданный момент времени, что может давать для любой системы полный контекст состояния станции и складывающейся ситуации управления полетом РС МКС экипажем и специалистами оперативной смены ЦУП.

В качестве примера приведем некоторые понятия, представляющие концептуальную модель РС МКС:

- классы понятий: «РС МКС», «ТГК «Прогресс», «ТПК «Союз», «Модуль», «Отсек», «Стыковочный узел», «Люк», «Бортовое оборудование» (Пульты, Датчики, Клапаны и т.п.), «Бортовое средство» (Баллон наддува переносной, Секции грузового корабля, Мановакуумметр и т.п.), «ТМИ», «Экипаж», «Специалист ЦУП»;
- свойства: Отсек «имеет» Объем, Отсек «имеет» Длину, Объект «имеет» Горючесть, Средство наддува «имеет» Давление, Космонавт «имеет» Дееспособность, Люк «имеет» Текущее положение;
- процессы: «Падение давления» в Отсеке, «Перемещение» космонавта, «Открытие» баллона наддува, «Потребление кислорода» космонавтом, «Включение» устройства, «Открытие» люка;
- отношения: СОЮЗ «пристыкован к» МИМ2, Космонавт «держит» Огнетушитель, Средство наддува «находится в» Отсеке, Экипаж «использует» Средство наддува, Модуль «стыкуется» с Модулем;
- атрибуты: «Текущее давление», «Темп падения давления», «Резервное время», «Текущая масса».

Эти базовые понятия и отношения, представляющие собой семантическую сеть, вместе с встроенными сценариями и правилами их интерпретации определяют метаонтологию РС МКС. Используя указанные концепты метаонтологии можно сконструировать модель деятельности по управлению полетом РС МКС. Например, подключение к двум объектам, связанным отношением «контакта», свойств «иметь электрическую проводимость» и «иметь горючесть» позволяет активировать законы электричества и при определенных условиях проводник в концептуальной модели цепи, через который идет ток, может нагреться, а у рядом находящегося объекта со свойством «имеет горючесть» при достижении некоторой температуры может запуститься процесс возгорания.

Рассмотрим использование онтологии РС МКС на примере развития на борту РС МКС аварийной ситуации «Пожар».

1. Для объектов РС МКС вводится свойство «Может гореть», которому задаются параметры (температура загорания, теплоемкость и др.).

2. Свойство «Может гореть» связывается с процессом «Пожар», в котором пересчитывается температура соседних объектов.

3. При возникновении пожара автоматически запускается процесс «Пожар», в котором агент пожара находит ближайшие объекты в модуле и проверяет их на наличие свойства «Может гореть». Если температура выше заданной (загорания), происходит возгорание этих объектов и создаются новые агенты этих горящих объектов, которые сгорают за время, обусловленное их теплоемкостью.

4. Объект «огнетушитель», в режиме использования создает пенную массу, которая противостоит горению и останавливает горение при определенных параметрах;

5. В случае отсутствия кислорода, горючих объектов или при использовании средств тушения пожар завершается.

6. Напротив, от горящих объектов возгораются другие – пожар рекурсивно (вне зависимости от числа агентов, но в зависимости от конфигурации – концептуальной модели РС МКС) распространяется в заданном объеме, поднимая температуру, выжигая кислород и т.д.

Таким образом, с использованием онтологии можно специфицировать конкретные факты и строить модели описания ситуаций для работы агентов. В создаваемой концептуальной модели РС МКС «все связано со всем» и любой объект может запускать процессы, с ним связанные, которые меняют свойства или устанавливают отношения с другими объектами и т.д. В результате могут инициироваться такие процессы, как пожар, и уже на этой основе одновременно развиваются процессы планирования возникающей аварийной ситуации, которые могут быть легко визуализированы, использованы для прогнозирования развития ситуации и т.д.

1.3. Использование онтологии РС МКС в процессе принятия решений

При проектировании онтологии РС МКС все необходимые нам знания мы можем разделить на [Абрамов, 2005]:

- знания предметной области;
- знания, относящиеся к методу решения задачи или принятия решений.

Таким образом, онтология РС МКС представляет набор двух миров:

- мир первого рода – мир предметной

области, в какой-то мере внешнего по отношению к системе и определяющего объекты реального мира, которыми мы оперируем при принятии решения. Это пассивный мир природных и искусственных объектов РС МКС, который не «думает» и не «строит планы» по ситуации, и который характеризуется свойствами и физическими законами. Процессы, моделирующие физические законы, работают как «автоматы», создавая новые объекты, устанавливая отношения между ними, изменяя значения атрибутов, запуская новые процессы при возникновении необходимых ситуаций (событий). Приведенный в предыдущем разделе пример развития пожара описывает процессы пассивных объектов мира первого рода.

- мир второго рода – мир активных объектов, в котором отражены знания о том, каким именно образом мы принимаем решения. Это мир акторов (людей) и агентов (программ) – особый подкласс объектов, наделенных как свойствами, описанными выше, которым пассивно следует (человек так же, как огнетушитель, имеет свойство «вес» и вступает в процессы мира первого рода, связанные со свойством «вес»), так и целенаправленным поведением, способностью создавать сценарии (планы) и действовать исходя из собственных предпочтений и ограничений.

Два этих мира (физических объектов, подчиненных физическим законам, и людей или агентов, обладающих активным поведением) строятся на взаимодействиях: акторами или агентами можно «населить» любой мир первого рода, и тогда активное поведение агентов, будет влиять на изменение свойств объектов, и наоборот, объектами можно наполнить мир людей, тогда возникновение любых событий или явлений мира первого рода, будет вызывать активацию и ответную реакцию агентов.

При этом одним из достоинств такого построения является то, что для одних и тех же знаний предметной области мы можем применять различные знания о методе решения, изменяя тем самым поведение системы.

Ключевыми понятиями онтологии принятия решений являются: «цели», «критерии принятия решений», «предпочтения», «объекты» и «субъекты» деятельности, «сценарии действий», «стратегии», «результаты действий». В онтологии РС МКС мы можем определять конкретные цели, стоящие перед нами, декларировать возможные пути их достижения, критерии, которыми мы руководствуемся при принятии решения, взаимосвязи между различными понятиями. При этом определенные критерии будут связаны с целью, выражая тем самым то, что успешность достижения цели определяется данными критериями и т.п. Отметим, что в онтологии не задается четкий алгоритм решения поставленных задач, а определяются возможные пути получения

решения с указанием свойственных каждой альтернативе компромиссов.

Система поддержки принятия решения проектируется способной понимать и рассуждать в рамках описанной модели, другими словами, она, опираясь на знания, описанные в онтологии, осуществляет поиск решений в соответствии с нашими целями и предпочтениями, находя определенный баланс между различными компромиссами, такими, как например, резервное время, скорость падения давления или распространения пожара, риск и т.п.

2. Конструктор онтологий

Для создания онтологии РС МКС предлагается использовать конструктор онтологий (конструктор виртуальных миров) и набор сопутствующих инструментов, в рамках которого пользователь может построить модель предметной области и концептуальную модель РС МКС, описать исходную ситуацию, создать вручную или автоматически сценарий действий или рассуждений и исполнить его по шагам, моделируя поведение агентов для достижения результатов.

Конструктор онтологий представляет собой комплекс инструментальных средств для управления онтологиями, моделями, сценами и агентами, предназначен для создания интеллектуальных систем управления ресурсами на основе мультиагентных технологий.

В Конструкторе онтологий для удобства построения различных элементов концептуальной модели выделяются три уровня описания предметной области: «онтология» - «модель» - «ситуация (сцена)»:

- Онтология описывает понятия и отношения (подобно толковому словарю), необходимые для описания знаний в любой предметной области (аэрокосмические организации, биотехнологии, медицина, наносистемы, живые системы и т.д.); на основе онтологии в дальнейшем строится модель деятельности.

- Модель описывает устойчивые конфигурации объектов, упрощающие создание формализованных описаний сцен в конкретных ситуациях (например, не следует каждый раз описывать конфигурацию международной космической станции, достаточно сохранять эти сведения в модели для дальнейшего использования).

- Ситуация (сцена) описывает экземпляры понятий и отношений в заданный момент времени (набор фактов); сцена подобна мгновенной «фотографии» ситуации в заданный момент времени.

Конструктор онтологий предоставляет следующие основные функциональные возможности для работы с онтологией, моделью и сценой:

- проектирование онтологии в виде семантических сетей. Используя конструктор онтологий, разработчик и пользователь может создавать и редактировать онтологии, специфицируя свои концепты (классы) и устанавливая связи между ними.

- возможность описания сценариев действий помимо описания декларативной части;

- при выборе любого понятия в дереве понятий или на семантической сети получение его параметров и всех связей, добавление нового свойства или установление нового отношения;

- при выборе объекта пользователем получение всего списка действий над ним и атрибутов; при выборе некоторого действия получение всего списка объектов, над которым оно может выполняться, и набора атрибутов, которые действие может менять;

- задание в дескрипторе сценария действия цели, кто именно может являться субъектом действия, какие знания и инструменты применяются при его реализации, каковы условия этого применения и результаты действия;

- создание сцены, набора экземпляров классов и отношений, описывающих некоторую ситуацию в мире и работа с ними. Каждый из экземпляров набора связан с некоторым концептом онтологии отношением вид-род;

- осуществление навигации по семантическим сетям онтологий и сцен (получение связанных сущностей, проверка наличия пути и т.д.). При этом функции по навигации доступны как разработчикам с помощью программного интерфейса приложения (API), так и в методах концептов.

- хранение онтологии и сцены в реляционной базе данных, либо в XML хранилище, импорт/экспорт онтологий в XML.

Возможности такого конструктора онтологий позволяют описывать предметную область не только программисту или инженеру по знаниям, но и любому рядовому пользователю, не являющемуся профессиональным программистом.

3. Примеры применения онтологии РС МКС

Концептуальная модель РС МКС, построенная с помощью конструктора онтологий и постоянно развивающаяся, на данный момент используется в РКК «Энергия» в следующих интеллектуальных системах поддержки принятия решений при управлении полетом РС МКС:

- интерактивная мультиагентная система построения программы полета, грузопотока и расчета ресурсов МКС [Диязитдинова, 2011].

- интеллектуальная система поддержки принятия решений при использовании бортовых ресурсов в процессе парирования аварийных ситуаций [Матюшин, 2013];

- автоматизированная система поддержки принятия решений при организации работы оперативной смены Главной Оперативной Группы Управления (ГОГУ).

Онтология РС МКС на данный момент состоит из более чем 3000 объектов, отношений, атрибутов, свойств и процессов.

Специалистами «НПК «Разумные решения» совместно со специалистами РКК «Энергия» планируется дальнейшее развитие онтологии РС МКС и ее использование в новых интеллектуальных аэрокосмических приложениях, которые подробно будут представлены в следующих статьях: адаптивный тренажер для формирования и восстановления навыков ситуационной поддержки принятия решений специалистами ГОГУ; интеллектуальная автоматизированная система предупреждения нештатных ситуаций в бортовых системах ТГК «Прогресс»; мультиагентная система планирования ресурсов комплекса целевых нагрузок РС МКС; мультиагентная система формирования и выполнения программ научно-прикладных исследований на РС МКС и целый ряд других систем, решающих различные задачи поддержки принятия решений и поддержки коммуникативного взаимодействия различных специалистов при управлении полетом РС МКС.

Знания, используемые во всех приведенных интеллектуальных аэрокосмических приложениях, хранятся в виде онтологии РС МКС, созданной с помощью конструктора онтологий, позволяющего описывать предметную область, ограничения, существующие в реальном мире, а также бизнес-логику принятия решений.

Заключение

В работе кратко описана проблема и актуальность решения поставленных задач по оперативному управлению полетом РС МКС. Предложен подход к созданию концептуальной модели (онтологии) РС МКС и ее использования для ситуационного управления полетом РС МКС в реальном времени.

Представлен пример построения концептуальной модели РС МКС и возможностей ее использования для принятия решений. Описаны основные возможности конструктора онтологий для построения онтологии РС МКС. Кратко представлены системы поддержки принятия решения при управлении полетом РС МКС, в которых используется онтология РС МКС.

Сравнительный анализ ошибок, допущенных специалистами управления полетом РС МКС и станции «Мир», показывает, что комплекс мер, в

том числе, и мер, предлагаемых системами поддержки принятия решений при управлении полетом, позволил с началом эксплуатации РС МКС свести до минимума влияние различных факторов при управлении полетом РС МКС и повысил безошибочность работы специалистов оперативного управления полетом РС МКС.

Библиографический список

[Гаврилова, 2000] Гаврилова, Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В. Ф. Хорошевский // СПб. - 2000. - 384 с.

[Huhns, 1997] Huhns, M.N. Ontologies for Agents / M.N. Huhns, M.P. Singh // IEEE Internet Computing - 1997. - November - December. - P. 17-24.

[Скобелев, 2012] Скобелев, П.О. Онтологии деятельности для ситуационного управления предприятием в реальном времени / П.О. Скобелев // Онтология проектирования, 2012, №1(3), С. 26–48.

[Абрамов, 2005] Абрамов, Д.В. Разработка средств построения и использования онтологий для поддержки процессов принятия решений / Д.В. Абрамов, В.В. Андреев, Е.В. Симонова, П.О. Скобелев // Труды VII международной конференции «Проблемы управления и моделирования в сложных системах. Самара: СНЦ РАН. - 2005. - с. 435 - 440.

[Дязитдинова, 2011] Дязитдинова, А.Р. Обеспечение согласованного взаимодействия по планированию грузопотока Международной космической станции с помощью мультиагентных технологий / А.Р. Дязитдинова, А.В. Иващенко, И.И. Литвинов, А.Л. Новиков, П.О. Скобелев, М.В. Сычева, И.И. Хамиц // Труды XIII Международной конференции «Проблемы управления и моделирования в сложных системах», Самара: СНЦ РАН, 2011, С. 435–442.

[Матюшин, 2013] Матюшин, М.М. Автоматизированная система поддержки принятия решений в аварийных ситуациях / М.М. Матюшин, С.И. Потоцкий, П.О. Скобелев, В.И. Потапов, О.И. Лахин // Программные продукты и системы, 2013, №3. С. 62–69.

ONTOLOGY OF ISS RUSSIAN SEGMENT AND ITS PRACTICAL USE IN INTELLECTUAL AEROSPACE APPLICATIONS

Vakurina T. *, Kotelya V. *, Lakhin O. **,
Matushin M. *, Skobelev P. **

* *S.P. Korolev Rocket and Space Corporation
«Energia», Korolev, Russian Federation
vlad@scsc.ru
tvakurina@yandex.ru
matushin@scsc.ru*

** *Software Engineering Company Smart
Solutions, Samara, Russian Federation
lakhin@smartsolutions-123.ru
skobelev@smartsolutions-123.ru*

Methods and tools for describing International Space Station conceptual model by means of ontologies and their practical use in intellectual aerospace applications for situational flight management of ISS Russian segment in real-time are considered in this paper.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 519.859

МОДЕЛИ ОЦЕНКИ УБЕЖДЕННОСТИ ОБ АДЕКВАТНОСТИ ПРЕДСТАВЛЕНИЙ О ПРЕДМЕТНОЙ ОБЛАСТИ В ЗАДАЧАХ ВЫБОРА

Виноградов Г.П.^{*}, Филатова Н.Н.^{**}

**Тверской государственный технический университет,
г. Тверь, Россия
wgp272ng@mail.ru*

***Тверской государственный технический университет,
г. Тверь, Россия
nfilatova99@mail.ru*

Рассматривается проблема принятия решений, когда представления лица принимающего решения о ситуации выбора формируются в процессе построения ее модели. Для этого организуется сбор информации для «снятия» различного рода неопределенностей и формирования гипотетической модели ситуации выбора. Сбор информации прекращается после достижения у лица принимающего решения состояния убежденности в правильном понимании связей и отношений между объектами в предметной области. Предложены нечеткие меры для включения оценок этого состояния в модель выбора.

Ключевые слова: нечеткое множество; принятия решений; нечеткие предпочтения; убежденность; модели нечеткого выбора.

Введение

При решении прикладных задач выбора лицо, принимающее решения (ЛПР), использует модель ситуации выбора, отражающую его представления. В ситуации неполной информации, знания или дефицита времени ЛПР строит модель исходя из правдоподобных предпосылок. Комплекс таких предпосылок, идей, взглядов, объясняющих явления и процессы или связи между ними и образующих гипотетическую концепцию ЛПР о конкретной ситуации выбора, может рассматриваться как предположительное, субъективное знание. Таким образом, имея перед собой предмет познания, ЛПР задается целью изучить его, понять, что он собой представляет. Для этого он необходимо определяет, путь к цели, а также способ и средства ее достижения. Формирование представления о ситуации выбора – это процесс описания и объяснения, т.е. воспроизведение в мышлении познаваемого предмета.

Предпосылка выступает в форме убеждения, которое в свою очередь является мерой степени уверенности в не полностью определенном предположении.

Выводы, которые делаются на основе гипотетической концепции, определяют у ЛПР состояние убежденности, которое является субъективной мерой истинности предпосылок, гипотез, правил построения вывода (то есть гипотетической концепции). Необходимо отметить, что процесс принятия решений основывается на оценках фактов, которые получают с помощью субъективных шкал, формируемых ЛПР. Степень близости субъективных оценок реальным фактам зависит от параметров шкал, которые могут изменяться в зависимости от эмоционального и физиологического состояния ЛПР [Симонов, 1981].

Если результат, полученный от реализации решения, сформированного на основе субъективных представлений ЛПР, не соответствует его ожиданиям, то он реализует немонотонный процесс пересмотра убеждений, который предполагает изъятие ошибочной предпосылки и/или введения новой. Новые убеждения появляются в связи с поступлением новой информации, полученной как от системы вывода, так и от системы мониторинга решений, или в связи с переоценкой по новой шкале имевшихся фактов.

Для повышения степени убежденности в истинности предпосылок и уверенности в предполагаемых результатах ЛПР использует

различные процедуры повышения своей информированности. В этой связи представляет интерес разработка математических моделей, учитывающих такое поведение ЛПР на основе теории нечетких систем и теории отношений [Zadeh, 1965].

1. Модель нечеткого выбора

Пусть имеется шкала X , которая может быть конечной или бесконечной. Предполагается, что на множестве X задано бинарное отношение \succ , обладающее свойствами асимметричности, транзитивности и слабой связности. Такое отношение называется отношением строгого предпочтения на множестве значений критерия. Известно, что слабая связность отношения \succeq означает, что для любых двух элементов x_1 и $x_2 \in X$, $x_1 \succeq x_2$ выполняется либо соотношение $x_1 \succ x_2$, либо соотношение $x_2 \succ x_1$.

Пусть A произвольное непустое нечеткое множество на множестве X .

Определение 1. Нечеткой функцией выбора называется отображение C , заданное на множестве всех непустых подмножеств $2^X \setminus \{\emptyset\}$, которое ставит в соответствие каждому $A \subset X$ определенное нечеткое множество $\Sigma(A)$ с функцией принадлежности $\mu_A(x)$, обладающей свойствами

$$\mu_A(x) \in [0, 1] \quad \forall x \in A \subset X,$$

$$\mu_A(x) = 0 \quad \forall x \in X \setminus A.$$

Будем считать, что возможна ситуация, когда для некоторых x

$$x \in A \subset X, \quad \mu_A(x) = 0 \quad \forall x \in X.$$

Это означает, что выбор из множества A является пустым, то есть $\Sigma(A) = \emptyset$. Другими словами, при предъявлении некоторых A имеет место отказ от выбора.

Согласно этому определению исход или выигрыш от выбора определяется нечетким подмножеством на множестве исходов O . Это позволяет использовать представления о ситуации выбора человека, на основе которых он устанавливает соответствия между альтернативами и исходом, используя нечеткие действительные числа.

Под нечетким числом понимается нечеткое подмножество универсального множества действительных чисел, имеющих нормальную и выпуклую функцию принадлежности такую, что [Рутковская и др., 2008]:

- существует значение носителя, в котором функция принадлежности равна единице.
- при отступлении от единицы вправо или влево функция принадлежности не возрастает.

При сравнении исходов, представленных в виде нечетких действительных чисел, в соответствии с принципом обобщения Заде необходимо определить меры для выявления предпочтений при сравнении нечетких действительных чисел.

2. Оценки предпочтений на множестве нечетких действительных чисел

Формирование нечеткого предпочтения на базе использования операций отношения между нечеткими действительными числами состоит в выявлении следующих ситуаций предпочтения: 1) строгое предпочтение; 2) безразличие; 3) большая предпочтительность; 4) не сравнимость. Графически эти ситуации можно представить следующим образом (рис. 1):

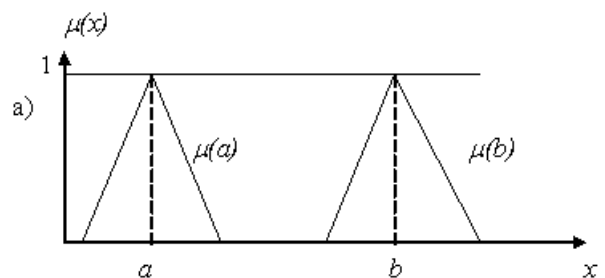


Рисунок 1а – Строгое предпочтение: ЛПР уверен, что b предпочтительнее a

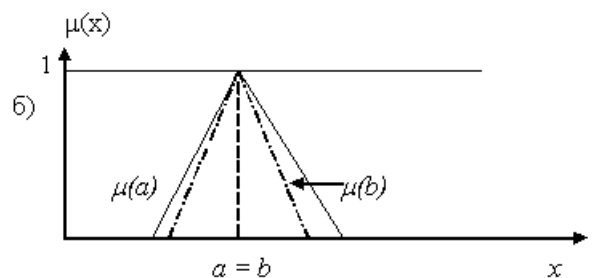


Рисунок 1б – Безразличие (равноценность a и b)

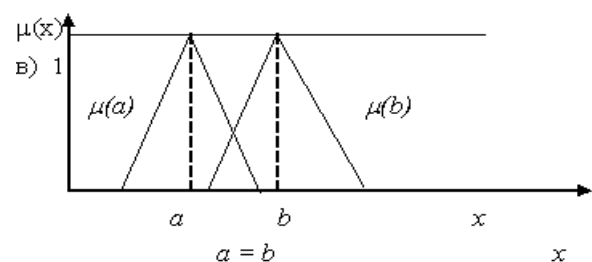


Рисунок 1в – Большая предпочтительность: ЛПР уверен, что a не предпочтительнее b

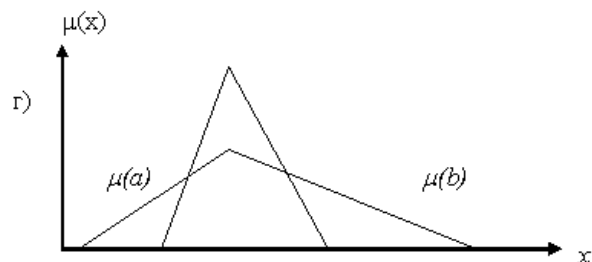


Рисунок 1г – Несравнимость: нельзя сказать ничего определенного о предпочтительности a перед b

Из теории нечетких множеств известно, что подмножество элементов множества X , для которых $\mu(x) > 0$, называется носителем (суппортом) нечеткого множества A :

$$A = \{(x, \mu_A(x)); x \in X\}$$

и обозначается $\text{supp } A$. Соответствующая формальная запись имеет вид

$$\text{supp } A = \{x \in X; \mu_A(x) > 0\}.$$

Тогда для случая а) $\text{supp } A \cap \text{supp } B = \emptyset$, то есть носители обоих нечетких множеств не имеют общих элементов.

Для случая б) нечеткое множество B содержится в нечетком множестве A или $\mu_B(x) \leq \mu_A(x)$, или $\text{supp } B \subset \text{supp } A$.

Случай б) предполагает две ситуации:

- нечеткое множество A равно нечеткому множеству B :

$$\mu_B(x) = \mu_A(x)$$

- нечеткое множество A почти равно нечеткому множеству B . Здесь можно ввести понятие степени равенства нечетких множеств A и B , например, в виде

$$E(A=B) = 1 - \max_{x \in T} |\mu_A(x) - \mu_B(x)|,$$

где $T = \{x \in X; \mu_A(x) \neq \mu_B(x)\}$.

Случай в) можно оценивать и с других позиций. Известно, что α -уровнем нечеткого множества $A \subseteq X$, обозначаемым, как A_α , называется четкое подмножество

$$A_\alpha = \{x \in X; \mu_A(x) \geq \alpha\}$$

то есть это подмножество определяется характеристической функцией

$$\psi_{A_\alpha}(x) = \begin{cases} 1 & \text{для } \mu_A(x) \geq \alpha \\ 0 & \text{для } \mu_A(x) < \alpha \end{cases}$$

Определение 2. Пусть нечеткие множества $A \subseteq X$ и $B \subseteq X$, где X – четкое множество. Пусть для каждого нечеткого множества определены множества α -уровня следующим образом

$$A_\alpha = \{x \in X; \mu_A(x) \geq \alpha\}$$

$$B_\alpha = \{x \in X; \mu_B(x) \geq \alpha\}$$

где $\mu_A(x)$ и $\mu_B(x)$ – функции принадлежности, значения которых выражают степень уверенности агента в принадлежности элемента x множествам A и B соответственно. Тогда альтернатива a будет предпочтительнее альтернативы b , тогда и только тогда, когда

$$x_a > x_b, \forall x_a \in A_\alpha(x), x_b \in B_\alpha(x),$$

то есть A больше B на уровне α .

Обозначим через $\underline{\alpha}$ минимальное значение α , при котором выполняется неравенство

$$x_a > x_b, \forall x_a \in A_\alpha(x), x_b \in B_\alpha(x).$$

Тогда $1-\underline{\alpha}$ будет степенью уверенности либо в предпочтительности a относительно b , либо в безразличии при выборе a или b .

По аналогии, если A_α содержится в B_α , то есть $A_\alpha \subseteq B_\alpha$, то говорят, что A содержится в B на уровне α . Так же, как и в предыдущем случае, можно ввести оценку степени уверенности $1-\underline{\alpha}$, где $\underline{\alpha}$ – это минимальное значение α , при котором будет справедливым $A_\alpha \subseteq B_\alpha$, то можно говорить, что $A_\alpha \subseteq B_\alpha$ со степенью уверенности равной $1-\underline{\alpha}$.

Величину $1-\underline{\alpha}$ можно считать мерой убежденности ЛПР в предпочтительности одной альтернативы над другой. Если величина $\rho=1-\underline{\alpha}$ возрастает (или $\underline{\alpha}$ уменьшается) утверждение A больше B (или A содержится в B) становится более ясным. При $\underline{\alpha}=0$ любой элемент, принадлежащий нечеткому множеству, будет для ЛПР достоверно принадлежать только этому множеству.

3. Убежденность и информация

Понятие степени убежденности при выборе альтернативы можно использовать как в классической, так и в поведенческой моделях выбора. Легко видеть, что величина $\rho=1-\underline{\alpha}$ зависит от вида функций принадлежности $\mu_A(x)$ и $\mu_B(x)$. Чем меньше размах $\text{supp } A$ и $\text{supp } B$ (интервал от минимального до максимального значения), тем более четко выражены представления ЛПР о ситуации выбора.

Введение меры степени убежденности при сравнении альтернатив позволяет:

- определить степень достаточности информации для принятия решения. При значении степени уверенности ниже некоторого порога принятие решения откладывается для сбора дополнительной информации или для коррекции;
- определить ценность для ЛПР собранной дополнительной информации. Она может быть равной нулю, если степень уверенности не изменится после ее получения. Если величина $\rho=1-\alpha$ выросла, то информация способствовала росту степени представления ЛПР о ситуации выбора. Если $\rho_i(\alpha_i) < \rho_i-1(\alpha_i-1)$, то либо имеет место дезинформация, либо полученные данные разрушают представление ЛПР о ситуации выбора и требуются новые данные.
- интерпретировать влияние эмоционального фона на результат оценки ситуации. Эмоциональное состояние ЛПР, изменяющееся под влиянием

интенсивной рабочей нагрузки и связанное с переживаниями, вызванными дефицитом времени или большой ответственностью, ведет к увеличению нечеткости оценок и к снижению величины ρ .

Значение пороговой величины степени уверенности зависит от индивидуальных характеристик ЛПР: более осторожный человек потребует, чтобы степень уверенности была бы высокой; решительный, привыкший рисковать – менее высокой. Это позволяет сформулировать меру для количественной оценки типа ЛПР.

Величина $\Delta\rho_i = \rho_i - \rho_{i-1} > 0$ (< 0) позволяет определить направление поиска информации. Пусть имеются два высказывания $p \approx X$ есть G и $q \approx X$ есть F , где F и G – предикаты, представленные в виде нечетких множеств. Тогда, если $G \subset F$, ($p \Rightarrow q$) (p влечет q). Это означает, что первое высказывание более информативно, чем второе.

Таким образом, степень убежденности при сравнении объектов для ЛПР описывает оценку степени разделения множеств, характеризующих каждый объект. Степень убежденности при поступлении более ценной информации не должна уменьшиться по сравнению со степенью убежденности, сформированной на основе данных прошлого опыта.

Таким образом, более информативное высказывание – это высказывание с меньшей нечеткостью, мешающей разделению объектов. Следовательно, изменение информированности ЛПР приводит к изменению его представлений и как следствие к изменению $\mu_d(x)$ и $\supp A$, и они могут быть использованы в качестве мер информированности ЛПР.

Значит, в теории принятия решений для более четкого различения альтернатив между собой, нужно уменьшить нечеткость в оценке каждого исхода и выигрыша при применении альтернативы путем уменьшения нечеткости функции исхода и функции выигрыша (модели объекта и оценок результатов).

Достижение эффекта $G \subset F$ требует увеличения числа учитываемых при описании свойств. При этом каждый добавляемый признак должен увеличивать степень убежденности в различении объектов.

Увеличение числа признаков может привести к двум ситуациям:

- Новая информация увеличивает степень убежденности в $G \subset F$, то есть утверждение с новым признаком является более информативным, чем такое же утверждение, но без него.
- Если сравниваются два объекта с одним и тем же количеством оцениваемых свойств, к которым добавляется еще одно свойство, но его значение у обоих объектов имеет трудно различимую величину, то добавочная информация

не повышает степень убежденности в различимости объектов, но и не уменьшает ее.

Третий момент связан с использованием либо редуцированной информации, либо косвенной информации при принятии решения. В этом случае уменьшение информации не оказывает положительного влияния на степень уверенности в правильном разделении объектов.

4. Представления о предметной области

Понятие представления ЛПР о предметной области в описанном выше смысле связано с такими понятиями как предмет и метод в процессе познания [Новиков и др., 2007]. Это означает, что для ЛПР ситуация выбора должна быть чувственно отражаемой, ощущаемой объективной реальностью.

Все свои ощущения ЛПР анализирует, обобщает и выделяет в них основное, повторяющееся. В результате этого, у ЛПР создается некое представление об изучаемой ситуации выбора. Он старается создать у себя состояние убежденности в адекватности своих представлений, организуя для этого процесс сбора и обработки информации. Этот процесс опирается на уже имеющиеся знания. ЛПР пытается выразить в терминах естественного языка то, что он ощутил, понял, узнал. В результате, у него формируется вербальное описание представления о ситуации выбора, и он вкладывает в него вполне определенное содержание, соотносящееся со всеми его знаниями.

Поскольку это выражение зависит от прошлых представлений субъекта, его знаний и от его способности ощущать, воспринимать, отражать реальность, постольку это представление содержит его субъективное влияние.

Следует отметить, что ЛПР изучает ситуацию выбора с какой-то целью. Следовательно, представления о ситуации выбора – это чувственно отражаемые стороны, части, свойства и отношения объектов, изучаемые с определенной целью, обобщенно выделенные в знаковом представлении, которое постоянно наполняется содержанием, соответствующим развивающемуся знанию как о непосредственно изучаемом, так и обо всей объективной реальности в целом.

Таким образом, представление о ситуации выбора в его символическом выражении – это форма описания представления ЛПР ситуации выбора для моделирования возможных исходов, которая и является основой процесса выбора.

5. Представления как субъективная модель, связывающая способы действия и результат

Принимаемое ЛПР решение имеет внутреннюю структуру, которая определяется его представлениями о принципах, законах, ограничениях, целях. Представление следует

рассматривать как связующее звено между внешней средой (окружением) и принимаемым решением по управлению процессами в предметной области. Представления, на основе которых в схожих состояниях окружения принимаются решения, позволяющие достигать желаемых целей, будем называть *знаниями*. Формирование знаний основывается на принципе рациональности, согласно которому ЛПР так организует свои представления, чтобы на их основе было возможно достижение желаемых состояний (целей) при определенном диапазоне изменения свойств внешней среды.

Структура представлений. Пусть Ξ – ситуация целеустремленного выбора. Она содержит в себе Γ – ситуацию целеустремленного состояния, которая, если ее рассматривать непосредственно в связи с действием, может включать в себя: Lim – ограничения, C – способы действия, i – нормативные или идеальные элементы, ie – символические выражения нормативных или идеальных элементов, Ω – внешнюю среду, o – результаты, которые ЛПР принимает во внимание в ситуации Γ .

Ситуация целеустремленного состояния в сознании ЛПР существует в форме представлений, элементами которых является T – научно обоснованное знание, которым обладает агент. Оно, в свою очередь, содержит: F – множество верифицируемых фактов, L – логически правильные дедукции из F , tr – элементы, которые в терминах знания, имеющегося у ЛПР, могут считаться им правильно научно определенными, но на самом деле являются отклонениями от научного стандарта, их можно назвать ненаучными элементами. К последним относятся: f – утверждения, ошибочно принимаемые за факты (ложные предположения), l – логически ошибочные выводы, g – невежественность (незнание), то есть элементы, существующие объективно, но субъективно не обнаруживаемые, r – элементы, варьирующиеся произвольно относительно элементов, сформулированных как T и t . Кроме T в субъективном представлении о целеустремленной ситуации существуют: G – цель и P – правило выбора, связывающее G и Γ .

Виды представлений. ЛПР формирует представления по следующим компонентам целеустремленной ситуации: множеству доступных для него способов действия; возможных результатов от реализации этих способов действия; возможных состояний окружения выбора; вероятностей того, что каждое возможное состояние окружения выбора окажется истинным; эффективностей каждого доступного способа действия по каждому возможному результату в каждом возможном состоянии окружения выбора; удельной ценности каждого возможного результата.

Представления как способ адаптации агента к окружению. Возможность самостоятельного выбора

ЛПР способа действия на основе собственных (субъективных) представлений о ситуации целеустремленного выбора для достижения цели (целей) интерпретируется как адаптивное, целенаправленное поведение. В этом смысле адаптация рассматривается как вид взаимодействия ЛПР со средой, в ходе которого он реализует свои требования и ожидания на основе воспринимаемой и осознаваемой им информации об изменяющихся свойствах и процессах ситуации целеустремленного выбора.

В процессе адаптации ЛПР: 1) изменяет свои представления о свойствах и закономерностях внешней среды и объекта управления; 2) изменяет свое поведение (способы действия на основе сформированных представлений), расширяет множество способов действия путем освоения новых; 3) организует целенаправленное изменение внешней среды для получения более выгодных ее состояний.

Будем считать, что ЛПР рационален: 1) его интересы выражаются в оценках ценности $\Phi_{ij}(O_i(C_j))$ ожидаемых им результатов $O_i, i=1, n$ от применения способов действия $C_j, (\Omega)=1, m$ на основе представлений о ситуации выбора Γ ; 2) рациональность поведения ЛПР состоит в стремлении максимизировать удельную ценность ожидаемых результатов. Выбор способа действия ЛПР выполняется в условиях неполной информации и неопределенности.

С ростом сложности и динамичности процессов в среде Ω ЛПР строит первоначально представление о ситуации выбора в форме гипотезы, придавая наблюдаемым параметрам разную степень значимости и классифицируя их как свидетельства по признаку «за» или «против». Затем его усилия направлены как на поиск данных, подтверждающих выдвинутую гипотезу, так и на поиск данных, ее отрицающих. При этом он использует только ту информацию, которую считает необходимой и достаточной для понимания процессов в предметной области. Факторы, опровергающие гипотезу, заставляют ЛПР ее либо модифицировать, либо пересмотреть, включив в нее позитивные моменты из старой. Анализ данных об исходах позволяет сформировать вопросы (запросы) для подтверждения предположений, поиск информации для ответа на которые и является основой для принятия или отбрасывания первоначальных представлений. Такая стратегия позволяет в условиях неполноты и недостоверности исходной информации сформировать непротиворечивые отношения между наблюдаемыми параметрами и представлениями у ЛПР.

Оценки адекватности представлений ЛПР характеризует терминами лингвистической переменной «уровень убежденности».

Определение 1. ЛПР убежден в адекватности своих представлений в ситуации типа Ξ

относительно цели G , если он считает, что выбор способа действия C на их основе позволит ее достичь. При этом: 1) он воспринимает часть характеристик X ситуации Ξ , 2) относительно другой части он делает предположения и проявляет намерение доказать (проверить) их правдоподобность, 3) в аналогичных ситуациях типа Ξ , в которых он воспринимал присутствие (отсутствие) X и стремился к G , выбирая C , он всегда достигал G , 4) когда он наблюдал отсутствие (присутствие) X в ситуации выбора типа Ξ , он никогда не выбирал C для достижения цели G на основе данного представления.

Под предположением будем понимать принимаемое по умолчанию значение наблюдаемой характеристики или описание причинно-следственной связи между наблюдаемыми характеристиками.

Определение 2. Уровень убежденности ЛПР в своих представлениях о ситуации выбора типа Ξ относительно цели G определяется частотой ее достижения при выборе способа действия C на их основе.

Оценка уровня убежденности изменяется в пределах от нуля до единицы. Если число неудачных попыток достичь цели G при выборе способа действия C на основе представлений ЛПР возрастает, то уровень его убежденности уменьшается (и наоборот), что становится для него стимулом для приложения усилий по их модификации или полной реконструкции вследствие возрастания сомнения в правдоподобности сделанных предположений.

Наличие стремления проверить правильность предположений является мерой сомнений ЛПР.

Определение 3. Усилия, которые ЛПР затрачивает для доказательства (опровержения) предположений, характеризуют степень его сомнений относительно представлений о ситуации выбора типа Ξ при стремлении к цели G .

Согласно положениям теории психологии поведения, если уровень убежденности, который по определению 3 зависит от количества подтверждений правильности выбора на основе представлений, возрастает, то стремление ЛПР к проверке падает, так как он не видит в этом смысла. С другой стороны, возрастание степени сомнения ЛПР является стимулом для поиска дополнительных аргументов (контраргументов).

Параметром, учитывающим эти две характеристики, является степень убежденности S_u , рассматриваемая как

$$S_u = U_u^\alpha * (1 - S_s)^\beta \in [0,1], \quad \alpha + \beta = 1,$$

где U_u – уровень убежденности (прошлый опыт); S_s – степень сомнения ЛПР в правильности своих представлений о ситуации выбора; α и β – коэффициенты значимости, которые ЛПР придает

своему опыту и необходимости поиска доказательств.

Предположение 1. ЛПР при формировании своих представлений в той или иной форме использует аппарат аргументации для построения последовательности гипотез, сходящихся к субъективно-истинной по убеждению.

Заключение

Предложенный подход может быть положен в основу построения интеллектуальных систем и предполагает:

- 1) разработку принципов построения эволюционных адаптивных баз данных и знаний;
- 2) разработку методов обработки данных на основе применения "активной" логической сети правил, управляемой потоком данных;
- 3) разработку методов быстрого поиска маршрута логического вывода на основе построения многополусной сети правил и поиска ее минимального разреза;
- 4) разработку методов распараллеливания потоковой обработки взаимозависимых данных путем построения виртуальных потоковых баз данных.

Библиографический список

- [Zadeh, 1965] Zadeh L. Fuzzy sets / L. Zadeh // Information and Control. - 1965 - №8, - P. 338-353.
- [Рутковская и др., 2008] Нейронные сети, генетические алгоритмы и нечеткие системы/ Рутковская Д. [и др.]; – М. Горячая линия – Телеком, 2006. – 452с.
- [Новиков и др., 2007] Новиков А.М. Методология./ А.М. Новиков, Д.А.Новиков – М.: Синтег, 2007. – 608 с.
- [Симонов, 1981] Симонов П.В. Эмоциональный мозг. М.: Наука, 1981. С. 140.

VALUATION MODELS CONVICTION ABOUT THE ADEQUACY OF THE PRESENTATION OF THE SUBJECT AREA IN THE PROBLEMS OF CHOICE

Vinogradov G.P. *, Filatov N.N. **

* Tver State Technical University,
Tver, Russia

wgp272ng@mail.ru-mail.address
nfilatova99@mail.ru-mail.address

The problem of decision-making when a person views the decision-making about the choice situation formed in the process of constructing its model. To do this, organized the collection of information for the "removal" of the various uncertainties and the formation of a hypothetical model of a situation of choice. Collection of information and stops after the person in the decision-making state conviction in the correct understanding of connections and relationships between objects in the domain. Fuzzy measures proposed for inclusion in the assessment of the state of the model of choice.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.81:159.942.52

ИССЛЕДОВАНИЕ НЕЙРОСЕТЕВЫХ АЛГОРИТМОВ ОБУЧЕНИЯ В ИНТЕЛЛЕКТУАЛЬНЫХ ЭВОЛЮЦИОННЫХ СИСТЕМАХ

Комарцова Л.Г., Лавренков Ю.Н., Антипова О.В.

*Московский государственный технический университет им. Н.Э.Баумана (Калужский филиал),
г. Калуга, Россия*

lkomartsova@yandex .ru

polikarp1@rambler.ru

В статье рассматриваются проблемы разработки и исследования нейросетевых и гибридных алгоритмов обучения для повышения эффективности функционирования интеллектуальных систем с целью поддержки принятия решений в сложных средах.

Ключевые слова: нейронная сеть (НС), алгоритм обучения, эволюционные и генетические алгоритмы обучения НС, обучение в реальном времени.

Введение

Развитие систем искусственного интеллекта (СИИ) в настоящее время связано с созданием новых методов и моделей, направленных на решение сложных проблем таких, как предсказание, планирование, распознавание образов и т. д. в различных прикладных областях [Ларичев и др., 1998]. Многие из публикаций по этим проблемам, несмотря на появление работ по созданию интеллектуальных систем реального времени, основываются на предположении неизменности во времени предъявляемых СИИ данных, статичности внешней среды. Хотя эти предположения в некоторых случаях являются допустимыми, используемые в СИИ статические методы не могут быть использованы при моделировании динамических процессов, а также при исследовании процессов, взаимодействующих с другими процессами, когда это взаимодействие априорно неизвестно. Другими словами, статические модели не могут быть применены к моделированию процессов, которые изменяются во времени, развиваются и эволюционируют. Для эволюционирующих процессов необходимым является отслеживание динамики их изменения во времени (т.е. функционирование в режиме on-line) и адаптации к внешней среде.

На основе анализа особенностей, которыми должна обладать эволюционная система, предложена многоуровневая и многомодульная открытая архитектура интеллектуальной эволюционной системы, основу которой составляет нейросетевая база знаний [Комарцова, 2003]. Эта

система может функционировать в двух режимах: on-line и off-line.

Обучение с учителем предназначено для организации пассивного («медленного») обучения в режиме off-line в отсутствии входной информации на входе системы. Накопленные в памяти примеры могут использоваться для обучения других нейросетевых модулей для поддержки «равновесия» с внешней средой. Основная нейросетевая модель в этом случае - многослойная нейронная сеть (МНС) с прямым распространением сигналов. Процедура обучения МНС может быть улучшена с помощью генетических алгоритмов (ГА) оптимизации, исследованных авторами [Комарцова, 2003, Комарцова, 2002]. Дальнейшее повышение эффективности использования ГА для обучения и адаптации нейросетевых модулей в составе эволюционной системы связывается с разработкой комбинированных алгоритмов. С этой целью предлагается алгоритм оптимизации, основанный на комбинации генетического алгоритма и алгоритма имитации отжига [Metropolis N. and al., 1998].

Развитие систем искусственного интеллекта (СИИ) в настоящее время связано с созданием новых методов и технологий, направленных на решение сложных проблем, таких, как предсказание, планирование, распознавание образов и т.д. в различных прикладных областях [Ларичев О.И. и др., 1998]. Многие из публикаций по этим проблемам, несмотря на появление работ по созданию систем реального времени [Еремеев А.П., 2004], основываются на предположении неизменности во времени предъявляемых СИИ данных, статичности внешней среды. Однако

статические модели не могут быть применены к моделированию процессов, которые изменяются во времени, развиваются и эволюционируют.

На основе анализа особенностей, которыми должна обладать эволюционная система, предложена многоуровневая и многомодульная открытая архитектура интеллектуальной эволюционной системы, основу которой составляет нейросетевая база знаний [Комарцова, 2003]. Эта система может функционировать в двух режимах: on-line и off-line.

Обучение с учителем предназначено для организации пассивного («медленного») обучения в режиме off-line в отсутствии входной информации на входе системы. Накопленные в памяти примеры могут использоваться для обучения других нейросетевых модулей для поддержки «равновесия» с внешней средой. Основная нейросетевая модель в этом случае - многослойная нейронная сеть (МНС) с прямым распространением сигналов. Процедура обучения МНС может быть улучшена с помощью генетических алгоритмов (ГА) оптимизации. Дальнейшее повышение эффективности использования ГА для обучения и адаптации нейросетевых модулей в составе эволюционной системы связывается с разработкой комбинированных алгоритмов. С этой целью предлагается алгоритм оптимизации, основанный на комбинации генетического алгоритма и алгоритма имитации отжига [Metropolis N. and al., 1998].

Этот алгоритм позволяет использовать алгоритм имитации отжига в ГА для целенаправленного отбора хромосом в новую популяцию в соответствии с приращением функции фитнеса в каждом поколении. Модификация комбинированного алгоритма заключается в изменении операторов генной мутации, операторов отбора и селекции, введенных авторами в [Комарцова, 2002]. Применение операторов случайной мутации в ГА фактически означает формирование новых генов, что, в конечном итоге, приводит к расширению области поиска и повышению вероятности нахождения оптимального решения. Однако случайные мутации с равной вероятностью могут привести как к увеличению значений функции фитнеса, так и к ее уменьшению. Таким образом, целесообразно динамически управлять вероятностью случайной мутации в процессе работы ГА: на начальном этапе поиска значение вероятности должно быть достаточно высоким (0,05..0,1), а на конечном – стремиться к нулю. Аналогичные рассуждения можно провести и в отношении операторов селекции и отбора.

Многие реально существующие информационные системы работают в реальном времени и используют для обучения данные только из входного потока (поток данных, передаваемый по сети Internet, телеметрическая информация о состоянии объектов и т.д.), которые не снабжены

метками класса, к которым их можно отнести. Поэтому в нейросетевых моделях, производящих обработку динамических данных, используют обучение без учителя в режиме on-line. В качестве основной нейросетевой модели, которую предполагается использовать в эволюционной системе для быстрого обучения в режиме on-line, является нейронная сеть адаптивного резонанса *ART* (*Adaptive Resonance Theory network*).

В классическом алгоритме построения сети адаптивного резонанса параметр сходства ρ фиксируется перед началом функционирования сети и в дальнейшем остается неизменным. При этом начальное значение ρ задается пользователем сети. Поскольку от параметра сходства в конечном итоге зависит качество функционирования всей сети, выбор правильного значения ρ имеет решающее значение для работы сети ART. Когда сеть ART работает в динамичной среде, на ее входы может подаваться множество различных векторов, значения которых трудно прогнозировать заранее. В этом случае ρ выбирает эксперт на основе своего опыта, поскольку из-за недостатка знаний о входных векторах невозможно прогнозировать величину параметра сходства, которая бы обеспечивала адекватное функционирование сети ART. В подобной ситуации желательно на первом этапе работы сети установить параметр ρ достаточно близким к 1, чтобы сеть могла создать достаточное число классов, а в дальнейшем уменьшить ρ , чтобы предотвратить чрезмерный рост сети.

При работе в динамичной среде невозможно проводить предварительное обучение НС, т.е. обучение отдельно от среды, в которой сеть будет функционировать. Поэтому ART-сеть должна работать в режиме on-line с момента запуска.

С этой целью в работе реализована возможность автоматического выбора параметра ρ на основе механизма нечеткого вывода с учетом специфики внешней среды. В рассматриваемом алгоритме fuzzy ART-1 [Комарцова, Максимов, 2004] методы нечеткой логики используются для управления параметром сходства ρ , при этом нечеткость вводится на уровне задания приращений параметра сходства, в зависимости от ситуации, возникающей в ART-1 при кластеризации.

Проведенное экспериментальное исследование алгоритма ART-1 с нечетким управлением параметром сходства подтвердили целесообразность его применения в тех случаях, когда желательно максимально автоматизировать процедуру обучения НС. Кроме того, подбирая соответствующие значения параметра сходства, можно существенно повысить помехоустойчивость сети, что является особенно важным для интеллектуальных систем, работающих в режиме on-line.

Выполненные эксперименты показали высокую эффективность нейросетевого классификатора на

основе модифицированного комбинированного алгоритма:

В связи с этим в работе предлагается реализовать возможность автоматического выбора параметра ρ на основе механизма нечеткого вывода с учетом специфики внешней среды. В рассматриваемом алгоритме fuzzy ART-1 [Комарцова, Максимов, 2004] методы нечеткой логики используются для управления параметром сходства ρ , при этом нечеткость вводится на уровне задания приращений этого параметра в зависимости от ситуации, возникающей в ART-1 при кластеризации.

Проведенное экспериментальное исследование алгоритма ART-1 с нечетким управлением параметром сходства подтвердили целесообразность его применения в тех случаях, когда желательно максимально автоматизировать процедуру обучения НС. Кроме того, подбирая соответствующие значения параметра сходства, можно существенно повысить помехоустойчивость сети, что является особенно важным для интеллектуальных систем, работающих в режиме on-line.

Выполненные эксперименты показали высокую эффективность нейросетевого классификатора на основе модифицированного комбинированного алгоритма:

1) этот классификатор быстрее обучается даже по сравнению с комбинированным алгоритмом на основе ГА и имитации отжига, при этом для обучения требует меньшей обучающей выборки (ошибка классификации уменьшается на 0,1 % при увеличении размера обучающей выборки с 200 до 500), что является особенно важным для построения эволюционных систем;

2) число ошибок классификации при той же обучающей выборке ниже, чем в других классификаторах.

1. решаемой задаче.
2. Выходной слой обеспечивает выдачу информации в заданном формате из слоя принятия решений во внешнюю среду.
3. Система адаптации, осуществляющая подстройку одного или нескольких модулей НС к решаемой задаче за счет их дообучения или включения новых нейронов в выходной слой (например, подобно тому, как это делается в сетях ART).
4. Знаниеориентированная подсистема, извлекающая информацию из НС и представляющая ее в виде правил IF-THEN. Наличие нечеткой базы правил обеспечивает объяснение результатов работы нейросетевых модулей.

Такую архитектуру можно рассматривать как теоретическую модель для построения реальных динамических систем. Несмотря на значительные достижения в области развития теории искусственного интеллекта, все еще не достигнут

тот уровень «интеллектуальности», который свойственен человеку. Поэтому большое значение для практики имеет исследование методов построения эволюционных систем, которые могли бы эволюционировать и адаптироваться к решаемым задачам, повышая тем самым уровень своего «интеллекта».

Функционирование ИС

Рассмотрим основные моменты функционирования эволюционной системы. Вначале эта система содержит небольшое число модулей, определяемых априорными знаниями специалистов в некоторой проблемно-предметной области, при этом некоторое множество правил может быть заранее встроено в систему. На основе процедуры самоорганизации и адаптации к внешней среде происходит постепенное накопление знаний системы подобно тому, как это делается в мозге человека, т.е. система становится все более интеллектуальной.

В ответ на входное воздействие активизируется определенный нейросетевой модуль на основе введенной меры сходства входного вектора с вектором весовых коэффициентов входных связей этого модуля. Если ни один из модулей не активизируется или эта активность находится ниже заданного порога, создается новый модуль (в простейшем случае нейронный узел или даже нейрон). Связи между модулями и нейронами определяются «возрастом» и средней активностью модулей на протяжении всего жизненного цикла существования эволюционной системы. Одни и те же нейроны могут быть объединены в классы, на основе их реакций на входные образы. Если, допустим, создается новый нейронный узел в ответ на некоторый входной вектор X , то этот узел на основе принципов самоорганизации начинает «сближаться» с нейронами, имеющими высокую активность на вектор X даже при недостаточном уровне сходства этого вектора с векторами входных связей нейронов. Таким образом, система эволюционирует во времени.

Реализация такого поведения системы осуществляется с использованием двух основных форм обучения: с учителем и без учителя (на основе самоорганизации). Основные требования, которые предъявляются к нейросетевым модулям в составе эволюционной системы, связаны с необходимостью обеспечения пластичности – способности воспринимать новые образы, и в то же время сохранению стабильности, гарантирующей не разрушение старых образов без потери ранее накопленной информации. К нейронным сетям, обладающим таким свойством, относятся сети адаптивного резонанса (Adaptive Resonance Theory - ART–сети). Для повышения эффективности функционирования нейросетевых модулей на основе обучения с учителем необходимо модифицировать существующие алгоритмы обучения с тем, чтобы адекватно реагировать на

изменения во внешней среде. Такие алгоритмы реализованы в [Комарцова Л.Г., 2002].

Наиболее важные достоинства нейронных сетей с адаптивным резонансом (ART-сетей) сводятся к следующему:

- при подаче на вход НС некоторого образа принимается «пластичное» решение о появлении нового образа и «стабильное» решение о совпадении со старым образом; это позволяет решить проблему стабильности-пластичности, которую очень трудно реализовать в сетях персептронного типа.
- НС типа ART (ART-1, ART-2, ART-3) обладают свойством самоорганизации, т.е. обучаются по алгоритму обучения без учителя, который основывается на использовании методов on-line кластеризации, квантизации и прототипного обучения [Metropolis N. and al.];
- в ART-сетях реализована возможность создания оптимальной топологии НС (с минимальным числом нейронов в выходном слое).

Однако эти достоинства влекут за собой необходимость управлять ростом сети, что в оригинальном алгоритме обучения ART осуществляется путем выбора пользователем желаемой величины параметра сходства $\rho \in [0,1]$. Чем выше значение этого параметра, тем большее число классов прототипов, и, соответственно, нейронов выходного слоя будет создано сетью. В предельном случае ($\rho=1$) происходит образование отдельного класса для каждого экземпляра обучающей выборки. Очевидно, что в этом случае существенно снижается способность нейронной сети к обобщению.

Чрезмерный рост числа распознаваемых классов приводит к неэффективному использованию компьютерных ресурсов и замедлению кластеризации. Слишком малое количество образованных классов прототипов, в свою очередь, может обусловить низкую точность распознавания.

Задача выбора числа необходимых классов, которые может распознать сеть, перекладывается на пользователя, который должен выбрать значение параметра ρ . Оптимальный выбор этого параметра трудоемок и требует проведения большого числа экспериментов. Поэтому первостепенное значение приобретает автоматизация процесса роста нейронной сети без задания пользователем конкретного значения параметра сходства. Один из подходов к решению этой проблемы для сети ART-1, работающей с двоичными входными образами и обучающейся без учителя, рассмотрен ниже.

Управление ростом сети ART-1 на основе методов нечеткой логики

В настоящее время известны два, наиболее часто применяемые на практике типа алгоритмов обучения нечеткой ART: 1) быстрого обучения и 2)

использующие нормализацию входных векторов с помощью комплементарного (дополнительного) кодирования [Kussul E.M. and al.]. Нечеткость рассматриваемой модели ART-1 определяется видом используемого логического оператора AND (вместо оператора пересечения \cap в ART-1 оператора \wedge (min) в fuzzy ART-1). Оператор min сводится к оператору пересечения в случае двоичных аргументов.

В алгоритмах быстрого обучения в формуле обновления векторов:

$$\mathbf{T}_j(t+1) = \beta(\mathbf{X} \wedge \mathbf{T}_j(t)) + (1-\beta)\mathbf{T}_j(t),$$

где β - коэффициент обучения, устанавливается $\beta=1$, посредством этого входной вектор \mathbf{X} быстро сходится к некоторому кластеру или создается новый нейрон, после чего β устанавливается <1 , когда определен кластер для входного вектора.

Для более сложного алгоритма в случае нормирования входных векторов в качестве меры нечеткости вводится функция выбора нейрона – победителя:

$$A_j(\mathbf{X}) = \frac{|\mathbf{X} \wedge \mathbf{T}_j|}{\varepsilon + |\mathbf{T}_j|},$$

где $\varepsilon > 0$, является константой, а нечеткий AND оператор \wedge определяется как оператор min, $|\cdot|$ - норма вектора. Функция выбора показывает степень, с которой весовой вектор \mathbf{T}_j является нечетким подмножеством входного вектора \mathbf{X} .

В предлагаемом алгоритме нечеткость вводится на уровне задания приращений параметра сходства ρ , в зависимости от ситуации, возникающей в ART-1 при кластеризации.

Общая схема подстройки различных параметров обучения нейронной сети (НС) с помощью нечеткой логики представлена на рис. 1. Нечеткая база правил, содержащая экспертные знания, используется для адаптивного изменения значений параметров обучения, в зависимости от состояния сети и реального значения параметров обучения.

Определим в качестве параметра управления величину ρ . Предположим, что априорно известно число кластеров N_a . Проведем первоначальную кластеризацию по приведенному выше алгоритму: пусть количество реальных кластеров равно N_r . Если $N_r < N_a$, то требуется подстройка параметра ρ с последующим новым предъявлением входных данных. Если $N_r < N_a$, ρ можно увеличивать, так как имеется число дополнительных кластеров $N_a - N_r$ (что приведет к более точной классификации); в противном случае, при $N_r > N_a$ - ρ должно уменьшаться, чтобы все входные векторы могли быть отнесены к одному из существующих кластеров.

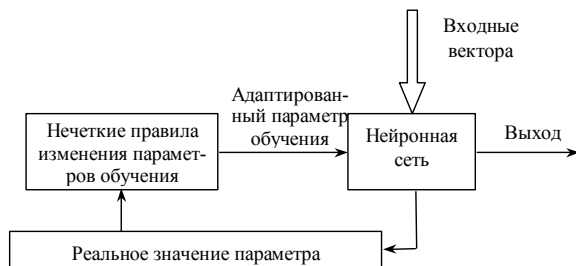


Рисунок 1 - Схема подстройки параметров обучения НС на основе методов нечеткой логики

Таким образом, в алгоритме обучения нечеткой ART-1 после проведения очередной кластеризации по алгоритму ART-1 необходимо определять величину изменения ρ , т.е. $\Delta\rho$ которое позволит N_r приблизиться к N_a . Схема вычисления искомого значения приращения параметра обучения $\Delta\rho$ (рис. 3) соответствует базовой схеме нечеткого контроллера [Kussul E.M. and al.].

Нечеткие правила, построенные на основе проведенного эмпирического анализа результатов работы нечеткой ART, позволили определить процедуру изменения ρ в соответствии с табл.2. Здесь $E = N_a - N_r$ и IE - изменение E . Определены следующие нечеткие множества: NB – отрицательное большое; NS – отрицательное малое; ZE – нуль; PS – положительное малое; PB – положительное большое.

Из табл.1, например, можно извлечь следующее правило: IF E есть положительное малое (PS) AND IE есть положительное большое (PB), THEN $\Delta\rho$ есть положительное малое (PS).

Далее по правилам нечеткой логики, зная вид функции принадлежности нечеткой переменной $\Delta\rho$ и используя дефазификацию, извлекается числовое значение приращения.

Таблица 1 - Фрагменты нечетких правил

$IE \backslash E$	NB	NS	ZE	PS	PB
NB	–	NS	ZE	PS	–
NS	NB	NS	ZE	PS	PB
ZE	NB	NS	ZE	PS	PB
PS	NB	NS	ZE	PS	PB
PB	–	NS	ZE	PS	–

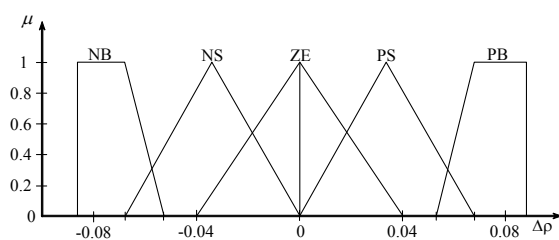


Рисунок 2 - Функции принадлежности для $\Delta\rho$

Определив значение $\Delta\rho$, сеть ART-1 выполняет повторную классификацию при новом значении $\Delta\rho$.

Алгоритм заканчивает свою работу (устанавливает соответствующее значение ρ), когда N_r сравняется с N_a .

Следует отметить, что на этапе распознавания в нечеткой ART-1 используется стандартная процедура поиска нейрона – победителя с $\rho = 0$, поскольку в общем случае образования новых классов не происходит.

При проведении экспериментов параметр сходства ρ в одном случае был выбран равным нулю для обеспечения минимального размера сети, а в другом - равным $\rho = 0.9$ для достижения более высокой точности. В табл. 3 представлены результаты классификации на выбранных наборах данных. В качестве результатов использовались ошибки классификации (даваемые в процентах от общего числа имеющихся примеров) и число образованных в результате функционирования ART-1 кластеров.

Таблица 2 - Характеристики тестовых наборов

Тесты	Кол-во классов	Размерность векторов	Кол-во Векторов
Распознавание сортов ирисов	3	4	150 (75, 5)
Генератор тестов	5	4	100-500

Результаты классификации для генератора тестов усреднены по 10 независимым компьютерным экспериментам, для распознавания сортов ирисов - по 100 экспериментам и представлены в табл.3. (ошибки распознавания образов даются в % от общего числа предъявляемых образов).

Таблица 3 - Результаты классификации

Тесты	ART-1 $\rho=0$		ART-1 $\rho=0.9$		ART-1 с неч. управлением ρ	
	Ошибки (%)	Кластер	Ошибки (%)	Кластер	Ошибки (%)	Кластер
Распознавание сортов ирисов	6,2	5	5,4	18	4,2	11
Генерат. тестов (200)	17,2	8	14,2	26	13,1	17
Генерат. тестов (500)	16,2	7	13,7	19	10,8	15

Анализ полученных результатов показывает, что ошибка распознавания при использовании сети ART-1 с нечетким управлением ρ значительно меньше по сравнению с другими алгоритмами.

Заключение

Проведенное экспериментальное исследование показало, что использование методов нечеткой логики для управления параметрами обучения нейронной сети адаптивного резонанса ART-1 позволяют автоматизировать процедуру настройки параметров сети в процессе ее функционирования, что позволяет сделать вывод о целесообразности использования этого типа сетей для построения ИС.

Точность распознавания (табл.3.) рассмотренного алгоритма практически не уступает точности стандартного алгоритма ART –1 при больших значениях параметра сходства, а число образуемых кластеров оказывается меньше. Кроме того, на основе предложенного алгоритма по желанию пользователя можно регулировать соотношение точность – размер нейронной сети.

Рассмотренные гибридные нечеткие нейросетевые модели на основе ART-1, в отличие от других подобных моделей, позволяют повысить качество работы динамических интеллектуальных систем за счет использования лучших свойств каждой из объединяемых технологий.

Библиографический список

- [Ларичев О.И. и др.], Ларичев О.И., Финн В.К. Теория и методы создания интеллектуальных компьютерных систем // Информационные технологии и вычислительные системы. - 1998.-№1.
- [Еремеев А.П.] Еремеев А.П. Концепции времени и их применение в интеллектуальных системах // Сб. научн. тр. «Интеллектуальные системы и технологии». –М.:МИФИ.-2004.
- [Grossberg S.] Grossberg S. Studies of Mind and Brain. –Reidel. –Boston.-1982.
- [Люгер Дж.] Люгер Дж. Искусственный интеллект. Стратегии и методы решения сложных проблем. - Пер. с англ. – М.: Изд дом «Вильямс». -2003.
- [Комарцова Л.Г.] Комарцова Л.Г. Исследование алгоритмов обучения многослойного персептрона // Нейрокомпьютеры. Разработка и применение. – М.: Радиотехника. 2002.-№12.
- [Kirkpatrick S. and al.] Kirkpatrick S, Gellat C.D. Vecchi M. Optimization by Simulated Annealing. Sciece. -1983.-vol. 220.
- [Metropolis N. and al.] Equation of calculation by fast computing machines. -J. of Chem. Phys. -21(6). – 1998.
- [Комарцова Л.Г. и др.] Комарцова Л.Г. Максимов А.В. Нейрокомпьютеры.-М.: МГТУ им. Н.Э. Баумана.-2004.
- [Kussul E.M. and al.], Kussul E.M., Baidyk T.N., Lukowich V.V., Rachkovskij D.A. Adapt. NN-classifier with multfloat input coding // Proc. of 6-th Int. Conf. "NeuroNimes-2008". -Nimes, France. -2008.
- [Chin-Teng Lin] Chin-Teng Lin. Neural Fuzzy Systems to Intelligent Systems//A.Simon and Schuster Company. Upper Saddle River, NJ 07458. -1996 by Prentice Hall.

RESEARCHING OF NEURAL LEARNING ALGORITHMS IN INTELLIGENT EVOLUTION SYSTEMS

Komartsova L.G., Lavrenkov Ju.N., Antipova
O.V.

*Moscow Bauman State Technical University,
Kaluga filial, Russia*

lkomartsova@yandex.ru

polikarp1@rambler.ru

The article is devoted to the problems for solving difficult problems, such as prediction, planning, pattern recognition and knowledge discovery in a number of application areas: bioinformatics, speech and language, image and video analysis, other engineering disciplines. Most of these publications deal with static process, assuming that the process is represented adequately by the data available at present and that it does not change over time.

When the process is evolving, the modelling system needs to be trace the dynamics of the process and to be adapt to changes in the process. Many real-world problems from engineering, economics, social sciences require continuously adapting models. The modelling system needs to be evolving.

Introduction

For the evolutionary process is necessary to monitor the dynamics of their changes over time (the operation mode of on-line) and adaptation to the environment (for example, the robot in an unfamiliar environment). In this paper we investigate some issues that need to be addressed for the creation of evolutionary systems that simulate human decision-making process in the complex is difficult to formalize problems in real time.

Main Part

Based on the analysis of the features that should have an evolutionary system architecture proposed by intellectual evolving system, which is a multi-module and multi-tier open architecture that can adapt to a particular domain based on the additional training of neural network modules in the operation of this system. Solved the problem of automating the process of growth of the neural network ART-1, without specifying the user-specific values for similarity based on the use of fuzzy logic methods, which significantly reduce training time.

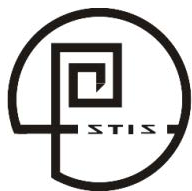
The fuzzy rule base that contains expertise has been used for adaptive changes in the values of learning parameters, depending on network conditions and real values of the parameters of learning.

At the stage of recognition in fuzzy ART-1 uses a standard search procedure of the neuron - the winner with $\rho = 0$, since in general the formation of new classes is not happening. To evaluate the effectiveness of the proposed algorithm was used a standard test set for recognizing irises and test generator.

Conclusion

Experimental tests showed that the use of fuzzy logic methods to control the parameters of neural network learning adaptive resonance ART-1 allow us to automate network configuration during its operation, which suggests the usefulness of this type of networks for dynamic intelligent systems.

The above hybrid fuzzy neural network model based on the ART-1, in contrast to other similar models can improve the dynamic performance of intelligent systems by using the best features of each of the merged technologies.



УДК 519.711.74

**СЕМАНТИЧЕСКИЕ АССОЦИАТИВНЫЕ РЕСУРСНЫЕ СЕТИ С
ПЛЮРАЛЬНЫМИ ВЕРШИНАМИ**

Жилякова Л.Ю. *

** Институт проблем управления им. В.А. Трапезникова РАН,
г. Москва, Россия***zhilyakova.ludmila@gmail.com**

В работе описывается ассоциативная модель памяти с переменной топологией. В ней изменения претерпевают не только пропускные способности ребер, но и вершины: в сети могут возникать новые вершины, образованные слиянием существующих. Такой прием позволяет моделировать ассоциации между множествами объектов разной природы, не распадающиеся на бинарные пары.

Ключевые слова: модель памяти, плюральная решетка, часть-целое, ресурсная сеть.

Введение

Работа продолжает серию исследований, посвященных построению ассоциативной памяти на основе неоднородной ресурсной сети. В основе ассоциативной сети лежит ресурсная сеть с переменной топологией. В процессе ее функционирования изменяются веса ребер, кроме того, могут возникать новые ребра. Принципы построения и функционирования таких сетей были описаны, например, в [Жилякова, 2009], [Жилякова, 2013].

В настоящей работе предлагается значительное расширение модели путем введения в нее плюральных конструкций, описанных для семантических сетей в [Жилякова, 2000] и берущих свое начало из работ по алгебраической семантике [Link, 1995]. Плюральные конструкции получаются путем слияния нескольких вершин в одну, причем, новая вершина добавляется в модель, а исходные вершины, также оставаясь в модели, связываются с ней двусторонними связями.

Математический аппарат, используемый при описании ассоциативной сети, опирается, с одной стороны, на теорию ресурсных сетей, – графовой динамической пороговой потоковой модели, предложенной О.П. Кузнецовым в [Кузнецов, 2009], с другой – на теорию плюральных решеток [Link, 1995], [Yi, 1999], [Yi, 2005], [Gélyb, 2009].

1. Ассоциативная ресурсная сеть**1.1. Основные понятия**

В основе *ассоциативной ресурсной сети*, предложенной для моделирования ассоциативной памяти в [Жилякова, 2009] лежит неоднородная ресурсная сеть.

Ресурсная сеть ([Кузнецов, 2009]) – графовая динамическая модель. Эта модель представляет собой ориентированный граф с постоянной топологией. Множеством вершин графа $V = \{v_i\}$, $|V| = n$; множество ребер: $E = \{e_{ij}\}$, $|E| = m$. Ребра e_{ij} имеют ограниченные пропускные способности r_{ij} . В сети могут иметься петли e_{ii} с пропускной способностью, равной r_{ii} .

Каждая вершина v_i в момент t содержит ресурс $q_i(t) \geq 0$. Вершины могут содержать произвольное количество ресурса. В каждый такт дискретного времени t между вершинами происходит обмен ресурсом. Ресурс, попавший в петлю на такте t , вернется в вершину на следующем такте.

В сети выполняется закон сохранения: суммарный ресурс $W = \sum_{i=1}^n q_i(t)$ не изменяется в процессе функционирования: $W = \text{const}$.

Ресурсная сеть является пороговой моделью: в каждый момент каждая вершина отдает ресурс во все исходящие ребра по одному из двух правил.

1) Если величина ресурса в вершине больше суммарной выходной пропускной способности всех ее исходящих ребер, она отдает по полной

пропускной способности в каждое ребро, оставляя себе излишки; если ресурс в вершине меньше этой величины, он распределяется пропорционально пропускным способностям во все исходящие ребра. Правила передачи ресурса и ряд основных свойств ресурсных сетей описаны в ряде работ, например, [Кузнецов и др., 2010].

Ассоциативная ресурсная сеть получается из обычной в результате двух существенных модификаций:

- 1) добавлением семантики;
- 2) введением переменной топологии.

1.2. Семантика

Семантика в сети означает, что вершины в ней соответствуют сущностям предметной области, а ребра – ассоциативным связям между ними. Переменная топология отвечает за то, что сила ассоциаций между сущностями изменяется в процессе функционирования сети. Чем больше сущности ассоциированы друг с другом, тем выше пропускная способность связывающих их ребер. Чем больше вершина значима, тем выше пропускная способность ее петли (отвечающая за автоассоциацию [Кохонен, 1980]), и тем больше суммарная пропускная способность входящих в нее ребер.

Отношение ассоциации будем считать обоюдным; поэтому каждая пара смежных вершин связана двумя противоположно ориентированными ребрами.

Количество ресурса в вершине соответствует яркости вершины. Чем вершина ярче, тем она виднее – доступнее при поиске.

1.3. Выполнение запросов и изменение пропускных способностей

В ассоциативной сети вводится два типа времени: *быстрое* и *медленное*.

Одному такту *медленного времени* соответствует исполнение одного запроса. Каждый запрос выполняется в *быстром времени*, которое соответствует времени ресурсной сети.

Запросы представляют собой распределение яркости по подмножеству вершин сети при помещении ее в некоторое начальное множество. Некоторые вершины имеют свойство *аттрактивности*: они притягивают яркость в процессе выполнения запроса. Эти вершины и попадут в результат выполнения запроса.

После выполнения запроса пропускные способности всех ребер, по которым тек ресурс, увеличиваются пропорционально суммарному пропущенному ресурсу. Если в начальном множестве запроса существуют несвязанные вершины, в сети создаются новые пары ребер, задающие новую ассоциацию.

Для того чтобы ресурс не растекался из начального множества вершин одинаково по всем направлениям, используются *рекурсивные запросы*. Это многократные запросы в один такт медленного времени, где распределение яркости происходит сначала на протяжении некоторого наперед заданного числа t' тактов быстрого времени, а затем формируется новое входное множество, в зависимости от распределения ресурса на такте t' . Эта процедура повторяется до стабилизации или останавливается по истечении заданного количества итераций k .

На каждом такте медленного времени происходит увеличение пропускных способностей выделенных ребер. Для того, чтобы суммарная пропускная способность сети оставалась ограниченной, вводится процедура нормировки. Когда суммарная пропускная способность достигает некоторой величины r_{sum_max} , вся сеть нормируется к заданной величине r_{sum_min} . Кроме того, с помощью нормировки происходит забывание: редко используемые ребра истончаются.

2. Плюральные конструкции в ассоциативной сети

2.1. Построение плюральных конструкций в семантических сетях

В [Жилякова, 2000] предложена семантическая сеть, в которой за счет введения плюральных конструкций оказалось возможным значительно расширить выразительную силу бинарных отношений. Обычная практика представления многоместных отношений – разложение их на конъюнкцию двухместных. Однако в некоторых случаях такое разложение невозможно произвести без искажения содержания. Одним из наиболее ярких примеров служат конструкции, представленные собирательными глагольными фразами, такими как (i) «Джон и Мэри встретились». Здесь «и» не является булевой операцией, поскольку иначе предложение (i) распалось бы на две части (ii): «Джон встретился и Мэри встретила». Но в отличие от (i), (ii) является бессмыслицей поскольку собирательные глаголы неприменимы к единичным объектам.

Существование не-булевого «и» уводит нас к мереологической парадигме, к рассмотрению сложных конструкций, состоящих из частей, которые существуют непосредственно рядом друг с другом и могут быть перемещены или удалены из целого без ущерба для остатка. Целое, все части которого состоят исключительно в таких отношениях, называется совокупностью или вполне суммируемым целым [Smith, 1995].

Операция, с помощью которой строятся такие конструкции, называется *слиянием*. Слияние двух объектов a и b означает « a и b , взятые вместе» [Link, 1995]. Это слияние (обозначим его $a[b]$), является еще одним объектом, связанным с a и b

посредством отношения часть-целое: « \mid ». Это отношение является частичным порядком. Операции \mid и \sqcup определяются друг через друга следующим образом:

$$a \mid b \Leftrightarrow a \sqcup b = b. \quad (1)$$

Введем в рассмотрение предметную область, которая содержит не только единичные сущности, но и их слияния. Причем слияние слияний не повышает уровня включения, как в теории множеств, а остается снова слиянием элементов. Единичные сущности и их слияния имеют одинаковый «онтологический статус».

В [Link, 1995] сформулирован ряд аксиом для доменов, которые содержат как единичные так и «плюральные» объекты, не делая между ними различия. Пусть $L \neq \emptyset$ – такой домен, и \mid – операция слияния на L , \mid – отношение часть-целое, определенное в (1) и A – множество \mid -минимальных элементов в L , называемых атомами в L . Определим для данного элемента a главный идеал a , как множество элементов «меньше» a :

$$a^\downarrow = \{x \in L \mid x \mid a\} \quad (2)$$

Следующие аксиомы дают характеристику плюральных структур:

A.1. $\forall x, y \{x \sqcup y = y \sqcup x\}$ (коммутативность)

A.2. $\forall x, y, z \{x \sqcup (y \sqcup z) = (x \sqcup y) \sqcup z\}$ (ассоциативность)

A.3. $\forall x \{x \sqcup x = x\}$ (идемпотентность)

A.4. $\neg \exists x \forall y x \mid y$ (отсутствие нуля)

A.5. $\forall X \subseteq L \{X \neq \emptyset \Rightarrow \exists x: x = \sup X\}$ (полнота)

A.6. $\forall x, y \{x \mid y \Rightarrow \exists u \in A: (u \mid x) \wedge (u \mid y)\}$ (отделимость)

A.7. $\forall x \exists u \in A: u \mid x$ (атомарность)

A.8. $\forall X \subseteq L, u \in A \{u \mid \sup X \Rightarrow \exists x \in X: u \mid x\}$ (\sup -атомарность)

Структура, удовлетворяющая аксиомам A1–A3 является верхней полурешеткой. Для любой пары элементов точная верхняя грань определяется как

$$\sup(x, y) = x \sqcup y.$$

Таким образом, плюральная структура является ограниченным типом полурешетки. Если L удовлетворяет аксиоме 4, будем говорить, что она не имеет дна.

Полную, отделимую, \sup -атомарную полурешетку без дна будем называть *плюральной решеткой* [Link, 1995].

2.2. Образование плюральных вершин в ассоциативной ресурсной сети

При построении ассоциативной ресурсной сети введение плюральных конструкций сильно упрощает представление ассоциативных связей, в том случае, когда отношение ассоциации не бинарно, и при этом не разлагается в композицию бинарных отношений. Так, в работе [Жиликова, 2000] рассматривается отношение «Джон, Джек и Джим владеют фермой». Для него в ассоциативной сети помимо вершин, соответствующих по отдельности Джону, Джеку и Джиму, будет введена еще одна вершина Джон \sqcup Джек \sqcup Джим, и именно она будет ассоциирована с вершиной «ферма».

Фрагмент сети, соответствующий описанной конструкции, представлен на рисунке 1.

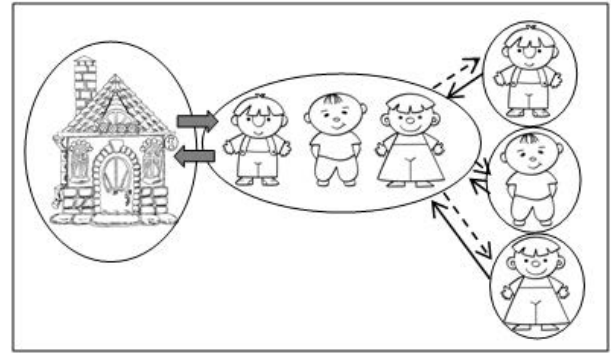


Рисунок 1 – Слияние вершин в ассоциативной сети

При построении запросов к ассоциативной сети создание плюральных вершин может сильно упростить процедуру поиска.

Пример. Пусть ассоциативная память хранит сведения о футбольном клубе «Молоток». Тогда в качестве плюральных вершин в этой памяти целесообразно хранить составы команды по годам; основные составы, игроков по специализации и т.п.

Тогда если x_N – состав всех игроков клуба в году N , y_N – основной состав, z_N – защитники, то справедливы следующие соотношения между ними:

$$y_N \in x_N^\downarrow, z_N \in x_N^\downarrow, y_N \mid x_N, z_N \mid x_N,$$

где x_N^\downarrow – главный идеал x_N (формула (2)).

При поиске информации о составе клуба в некотором году, о распределении игроков внутри команды и т.д. такое представление не только упростит работу с памятью, – без него хранение информации подобного рода невозможно.

При этом, конечно, увеличивается размер памяти, занимаемой сетью, но резко уменьшается время выполнения подобных запросов и оптимизируется структура представления информации.

Заключение

В настоящей работе предложен способ хранения в памяти и обработки «плюральных ассоциаций». Аппарат плюральных решеток позволяет, не усложняя структуру сети, строить ассоциации типов один-ко-многим, многие-к-одному и многие-ко-многим. Вершины, соответствующие плюральным конструкциям в сети, имеют тот же статус, что и вершины, описывающие единичные сущности.

Однако за рамками данного исследования остался важный вопрос: как в сети отразить не только ассоциативные связи, но и сохранить отношение часть-целое, которое формирует плюральные конструкции из единичных сущностей и других плюральных конструкций. По-видимому, придется вводить в сети отношения двух типов. Отношения первого типа соответствуют ассоциативным связям и описываются ребрами

ресурсной сети в том виде, в котором модель разработана к настоящему моменту. Второй тип отношений – антисимметричное отношение |. На рис. 1 эти отношения обозначены стрелками разных типов.

По мере разработки ассоциативная модель претерпевает ряд значительных изменений, и на данном этапе представляется, что ее с полным правом можно назвать семантической ассоциативной моделью памяти.

Работа выполнена при поддержке РФФИ (грант № 11-01-00771-а).

Библиографический список

- [Жилякова, 2013] Жилякова Л.Ю. Построение ассоциативной модели памяти. Когнитивный подход / Материалы III международной научно-технической конференции "Открытые семантические технологии проектирования интеллектуальных систем OSTIS-2013". Минск: БГУИР, 2013. С. 87-90.
- [Жилякова, 2009] Жилякова Л.Ю. Поиск в ассоциативной модели памяти. // IX международная конференция имени Т.А. Таран ИАИ-2009. Киев, «Просвіта», 2009. с. 124-130.
- [Жилякова, 2000] Жилякова Л.Ю. Мереологический подход в структурировании предметной области // Седьмая национальная конференция по искусственному интеллекту с международным участием КИИ' 2000. Труды конференции. М. Издательство физико-математической литературы, 2000. С. 247-254.
- [Кохонен, 1980] Кохонен Т. Ассоциативная память. – М.: Мир, 1980.
- [Кузнецов, 2009] Кузнецов О.П. Однородные ресурсные сети. I. Полные графы. // Автоматика и телемеханика, 2009, № 11, с.136-147.
- [Кузнецов и др., 2010] Кузнецов О.П., Жилякова Л.Ю. Двусторонние ресурсные сети – новая потоковая модель // Доклады АН, 2010, том 433, №5, с.609-612.
- [Gélyb, 2009] Gélyb, A., Medinaa, R., Nourine, L. Representing lattices using many-valued relations // Information Sciences. Volume 179, Issue 16, 20 July 2009, Pages 2729–2739.
- [Link, 1995] Link, G. Algebraic semantics for natural language: some philosophy, some applications. // International Journal of Human and Computer Studies, 1995.
- [Smith, 1995] Smith, B. Formal ontology, common sense and cognitive science. International Journal of Human and Computer Studies, special issue on The Role of Formal Ontology in the Information Technology edited by N. Guarino and R. Poli, vol 43 no. 5/6, 1995.
- [Yi, 1999] Yi, Byeong-Uk. Is mereology ontologically innocent? // Philosophical Studies 93: 141–160, 1999. Kluwer Academic Publishers, Netherlands.
- [Yi, 2005] Yi, Byeong-Uk. The Logic and Meaning of Plurals. Part I // Journal of Philosophical Logic. 2005. October-December, Volume 34, Issue 5-6, pp 459-506.

SEMANTIC ASSOCIATIVE RESOURCE NETWORKS WITH PLURAL VERTICES

Zhilyakova L.Yu.*

* Trapeznikov Institute of Control Sciences,
Russian Academy of Sciences, Moscow, Russia

zhilyakova.ludmila@gmail.com

In this work we study an associative network memory model with variable topology. Not only the edges can change their capacities, but also there is a possibility to create new (plural) vertices. Plural vertices are being built by means of fusion some existing single or plural ones. This technique allows to model the association between the sets of objects of different nature, which cannot be decomposed into the several binary pairs.

Keywords: **memory model, part-whole, plural lattice, resource network.**

Introduction

The paper continues a series of studies on the construction of an associative memory based on heterogeneous resource network. The basis of the associative network is a resource network with a variable topology. During network's functioning the edge weights are changed, furthermore, the new edges may appear. Principles of construction and operation of such networks have been described, for example, in [Zhilyakova, 2009] and [Zhilyakova, 2013].

In this article we propose a significant extension of the model by introducing plural constructions described for semantic networks in [Zhilyakova, 2000] and originating from works on algebraic semantics [Link, 1995]. Plural entities are constructed by merging several nodes in one new vertex, which is added to the model. The constituents also remain in the network and are associated with a new vertex by bilateral links.

The mathematical formalism used when describing the model is, on the one hand, a theory of network resource, – the model proposed by O. Kuznetsov in [Kuznetsov, 2009], and on the other hand – the theory of plural lattices [Link, 1995], [Gélyb, 2009].

Main Part

There are a lot of associative relationships among several objects which cannot be decomposed into sets of binary relations. But usually networks allow representing only binary relations. In order to model the n-ary associations in the associative network the plural vertices are suggested. They are being built of single vertices by means of operation called “fusion” [Link, 1995]. The new plural vertices have the same ontological status as single ones. The fusion of fusion has no differences from its components from and every vertex corresponding to a single entity. Such an operation specifies on the set of vertices the special semilattice, called plural lattice. The main principles of constructing such networks are described.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822

КОГНИТИВНОЕ МОДЕЛИРОВАНИЕ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ (СЕМИОТИЧЕСКИЙ ПОДХОД)

Кулинич А.А.

*Федеральное Государственное учреждение науки Институт проблем управления
им В.А. Трапезникова, Российской академии наук, г. Москва, Российская Федерация*

kulinich@ipu.ru

В работе рассмотрен новый подход к моделированию плохо определенных ситуаций на основе когнитивных карт семиотического типа. Определена фреймовая модель представления семиотических когнитивных карт.

Ключевые слова: когнитивное моделирование; плохо определенная предметная область; знаковые системы; прикладная семиотика.

Введение

Многочисленные исследования когнитивных карт и методов их анализа заложили теоретический фундамент для создания прикладных коммерческих систем моделирования динамических систем с помощью когнитивных карт. В работе [Кулинич, 2011a] предложена классификация существующих когнитивных карт, а также приведены методы их анализа. В обзорных работах [Кулинич, 2010, 2011b] описаны компьютерные системы поддержки принятия решений, основанные на моделировании ситуаций с помощью когнитивных карт.

Качество систем поддержки принятия решений, основанных на моделировании ситуаций на основе когнитивных карт, в значительной степени определяется не только удобством и развитостью интерфейсов программных систем, реализующих этот подход в управлении, но и качеством самой когнитивной карты - субъективной модели процессов реальности. Под качеством систем подразумевается их полезность при принятии решений в слабоструктурированных ситуациях, которая выражается в повышении обоснованности принимаемых решений.

Поддержка принятия решений основана на получении прогноза изменения значений всех факторов при изменении значения любого другого фактора, которая решается с помощью системы конечно-разностных уравнений вида: $X(t+1)=W X(t)$ где $X(t)$ - вектор начального приращения значений факторов, $t=0$; $X(t+1)$ - вектор изменений значений факторов в моменты времени $t=1, \dots, n$, W - матрица смежности орграфа.

Задача принятия решений заключается в анализе динамики изменения состояния системы, представленной векторами состояния в последовательные моменты времени $X(t)$ и их содержательной интерпретации.

Содержательная интерпретация заключается в представлении вектора состояния ситуации, полученного в процессе моделирования в некоторой другой системе $X(t) \rightarrow \mathcal{Q}$, где \mathcal{Q} - интерпретирующая система. В настоящее время в системах моделирования когнитивных карт интерпретирующая система это субъективная понятийная система лица, принимающего решение - его знания о предметной области и его интеллектуальные способности: рассуждения, обобщение, воображение, способные представить вектор состояния $X(t)$ в виде понятия предметной области \mathcal{Q} , в которой модель построена.

Процесс интерпретации достаточно грубой модели действительности в виде когнитивной в интерпретирующую систему (систему знаний субъекта) порождает содержательно более богатые когнитивные процессы в интеллекте человека, которые в этой работе называется процессом *когнитивного моделирования*.

Знания аналитика можно представить и формально в виде онтологии как множество понятий предметной области, упорядоченные и связанные множеством отношений, а его интеллектуальные способности как процедуры вывода на онтологиях. Однако когнитивные карты как инструмент анализа ситуаций представляют собой качественные (грубые) модели, ориентированные на быструю генерацию множества решений и их правдоподобное обоснование. В условиях дефицита времени на

принятие решений, построение онтологии предметной области для решения задачи интерпретации оказывается нецелесообразным.

Проблемой широкого применения технологии моделирования и принятия решений с помощью когнитивных карт остается проблема субъективности когнитивной карты. Неоднозначность когнитивной карты для описания одного и того же явления действительности и соответственно множественность когнитивных карт одной и той же действительности у разных экспертов порождает проблемы логического обоснования решения.

В этой работе для решения проблем применения когнитивных карт в процессах управления, предлагается изменить существующую парадигму представления знаний в когнитивных картах в виде множества концептов и причинно-следственных отношений на этом множестве, на ее представление в виде знаковой системы, т.е. семиотическое представление когнитивных карт.

1. Знаки, знаковые системы, прикладные семиотические системы

Исследованием знаков и знаковых систем занимались известные математики и логики при создании так называемой экстенциональной (объемной) логики [Бирюков, 1960]. Наибольший практический интерес представляют работы немецкого логика Г.Фреге. Его модель знаковой системы известна как «треугольник Фреге» включает три связанные составляющие: знак, смысл знака и значение знака. Знак – это символ, слово (изображение), обозначающее в сознании человека объект реального мира. Под смыслом знака Фреге понимает описание объекта реального мира, отличающего его от других объектов. И, наконец, под значением знака Фреге понимает реальный объект, который определяется смыслом знака.

Особенность представления знания в виде знака заключается в том, что знак связывает ментальные процессы, в которых принимает участие имя, смысл знака, определяемые субъектом и зависящие от уровня его знаний, с объектами реального мира (значением), обозначаемые именем и определяемые его смыслом. Это означает, что между реальным миром и знаками существуют отношения моделирования, позволяющее с помощью знаков осуществлять процесс мышления со знаками, а не реальными объектами. Так, основатель теоретической семиотики (науки о знаках и знаковых системах) Ч. Пирс считает, что: «У нас нет способности мыслить без посредства знаков».

Семиотика получила широкое распространение в гуманитарных науках, структурной лингвистики и т.д. В области искусственного интеллекта развивается прикладная семиотика,

ориентированная на использование знаков и знаковых систем в моделях представления знаний, принятия решений, реализованных в компьютерных системах [Поспелов, 1986, 1999], [Мартынов, 2001].

Семиотической системой W называется упорядоченная восьмерка множеств:

$$W = \langle T, R, A, P, \tau, \rho, \alpha, \pi \rangle, \text{ где}$$

T - множество основных символов;
 R - множество синтаксических правил;
 A - множество знаний о предметной области;
 P - множество правил вывода решений;
 τ - правила изменения множества T ;
 ρ - правила изменения множества R ;
 α - правила изменения множества A ;
 π - правила изменения множества P .

Первые четыре множества образуют формальную систему FS , элементы с пятого по восьмой правила изменения формальной системы обеспечивают изменения формальной системы, «подстраивая» ее для решения задач и проблем, которые в рамках системы FS решить не удается.

Иными словами, семиотическая система может быть определена как динамическая система: $W = \langle FS_b, M_{Fsi} \rangle$, где FS_i - можно назвать состоянием семиотической системы, а M_{Fsi} – правило смены ее состояния.

Важно отметить, что правила $M_{Fsi} = (\tau, \rho, \alpha, \pi)$, меняющие состояние формальной системы связаны зависимостью, аналогичной зависимости, существующей в элементах семиотического треугольника (треугольника Фреге) [Поспелов, 1999]. Это означает, что применение одного из правил из этой четверки приводит к применению оставшихся правил. Зависимости эти сложны, их аналитическое представление отсутствует, и это представляет трудность и является предметом исследования семиотических систем искусственного интеллекта.

Предложенные расширения формальных систем в виде их модификаторов M_{Fsi} обеспечивают свойства открытости систем, за счет возможности адаптации формальной системы к изменяющемуся внешним условиям, позволяют значительно расширить возможности поддержки принятия решений в условиях неопределенности, неполноты и противоречивости исходной информации [Поспелов, 1986, 1999].

Особенность применения семиотической системы в процессах моделирования процессов принятия решений и в моделировании динамических систем заключается в том, что исследование объектов осуществляется в трехмерном пространстве: синтаксис, семантика и прагматика. Для пояснения этого утверждения приведем цитату из работы [Тарасов, 2002] «Итак, основным исходным элементом семиотической системы является знак. Знак имеет три аспекта: синтаксис, семантику и прагматику; или в других

терминах: план выражения, план содержания и план значения [Поспелов, 1999]. Таким образом, любой язык как семиотический объект функционирует и эволюционирует в трехмерном пространстве (синтактика, семантика, прагматика). *Синтактика* — это часть семиотики, занимающаяся изучением внутренних отношений между знаками. *Семантика* охватывает сферу отношений между знаками и тем, что они обозначают, а *прагматика* — сферу отношений между знаками и теми, кто ими пользуется. Таким образом, прагматика учитывает требования конкретного адресата сообщения, в то время как у семантики его нет».

Исследованию адаптационных возможностей семиотических систем посвящено ряд работ. В работе [Эрлих, 1997] предложена архитектура системы управления, построенной на принципах семиотического подхода. В работе [Аверкин, 1999] предложена нечеткая семиотическая модель, состоящая из нечетких регуляторов и позволяющая их настройку на Т-нормы и модели конкретного пользователя. В работе [Еремеев, 2003] рассматриваются принципы построения семиотической системы поддержки принятия решений, в которой адаптация для решения проблемы осуществляется на основе учета прошлого опыта принятия решений в немарковской модели с обучением. В работе [Тарасов, 2002] основная идея автора заключается в объединении теоретической семиотики и эволюционного моделирования для моделирования процессов образования и эволюции знаков (*семиозиса*) в многоагентных системах.

2. Задачи разработки когнитивных карт семиотического типа

Основные отличия исследования, приводимого в этой работе от работ перечисленных выше следующие. Во-первых, исследуется применение семиотического подхода для моделирования динамических систем в условиях неопределенности, в том числе и неполноты их описания. Во-вторых, в качестве модели динамической системы используется модель знаний в виде когнитивной карты. Формально когнитивная карта — это модель экспертных знаний в виде однородной семантической сети с причинно-следственными отношениями. В-третьих, исследуется система поддержки принятия решений семиотического типа. Это означает, что аналитик, пользователь системы поддержки принятия решений включен в процесс принятия решений, через взаимодействие его естественной знаковой системы и формальных методов поддержки интерпретации и выработки решений через интерфейсы, компьютерной системы построенных на принципах семиотических систем.

Для применения в плохо определенных процессах управления, когнитивных карт семиотического типа необходимо разработать:

1. Методику построения когнитивных карт семиотического типа. При разработке методики рассматривался метод, основанный на структурно-функциональной декомпозиции рассматриваемой ситуации, позволяющий выделить основные элементы динамической системы и представить их в виде знаков.
2. Формальные модели когнитивной карты семиотического типа: а) в виде фрейма и множества связанных фреймов; в) в виде стратифицированного поля знаний, включающего понятийную, функциональную, оценочную подсистемы.
3. Методы поддержки принятия решений с использованием когнитивных карт семиотического типа.

В этой работе рассматривается формальная модель когнитивной карты семиотического типа основанная на ее фреймовом представлении.

3. Фреймовая модель когнитивных карт семиотического типа

Г. Фреге определил знак тройкой: имя, смысл (понятие) и значение знака. В системах прикладной семиотики [Поспелов, 1986] в качестве модели знака используют фреймы [Минский, 1979]. Фрейм — это минимальное описание некоторой сущности, позволяющее ее идентифицировать. Обычное представление фрейма в базе данных — это именованные поля свойств, описываемой сущности и значения этих свойств. Поля свойств называют слотами, а значения, соответственно, значениями слотов. В прикладной семиотике между слотами устанавливаются отношения такие же, какие существуют в модели знака Г. Фреге.

На рисунке показано соответствие между вершинами треугольника Г. Фреге и фреймового представления знака Д. Поспелова.

Имя знака ассоциировано с именем фрейма, понятие (смысл по Фреге) связано с протофреймом. Под протофреймом понимается описание класса (рода) и представляется множеством признаков. Представление — (значение по Фреге) связано с экзофреймом. Под экзофреймом понимаются конкретные экземпляры реального мира, принадлежащие классу, определенному протофреймом и обозначаемого именем фрейма.

Определена следующая расширенная структура и назначение слотов протофрейма знака [Поспелов, 1999]:

1. Признаки и значения признаков;
2. Условия целостности;
3. Связи наследования;
4. Статические связи;
5. Условия активности;
6. Представления.

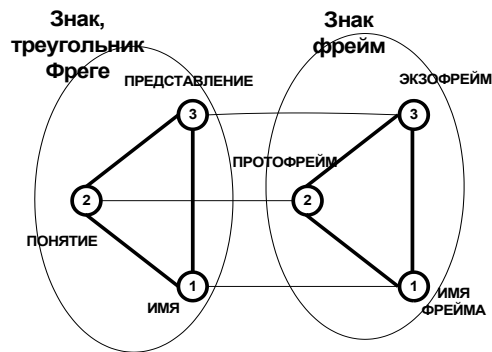


Рисунок 1 - Знак-фрейм

Далее подробно рассмотрим и прокомментируем все слоты фрейма-знака, но кроме этого будем описывать аналогичный слот фрейма семиотической когнитивной карты, его назначение, особенности и т.д. в сравнении с классическим фреймом знака.

3.1. Признаки и значения признаков

В классическом фрейме знака и во фрейме семиотической когнитивной карты для признаков и их значений отводятся Слоты 1,...,n.

В семиотической когнитивной карте в каждом слоте $i = 1, \dots, n$ определена тройка $\langle f_i, Z_i, \Delta_i \rangle$, где f_i - название признака понятия d ; Z_i - возможные значения признака; Δ_i - интервал значений $\Delta_i \subseteq Z_i$ признака, для которых имя понятия (фрейма) остается неизменным.

Значения признака могут быть определены на отрезке числовой оси $[a, b] \in \mathbb{R}^{\pm}$, либо в виде упорядоченного множества лингвистических значений, $Z_i = \{z_{i1}, \dots, z_{iq}\}$, $z_{iq+1} > z_{iq}$, $q = 0 \dots n-1$. В работе [Кулинич, 2013] определено семантическое пространство понятия d , $SS(d) = \times_i Z_i$, а интервалы значений признаков Δ_i названы интервалами толерантности понятия d по i -му признаку.

Базовым понятием в семантическом пространстве называется прямое произведение всех интервалов толерантности понятия d , т.е. $SS(d) = \times_i \Delta_i \subseteq SS(d)$.

Множество всех признаков и базовое понятие $SS(d)$ определим как содержание понятия d , $F(d) = (\{f_i\}, SS(d))$.

3.2. Условия целостности

Условия целостности задают закономерности строения объекта реального мира, определяемого знаком. В качестве примера условий целостности в работе [Поспелов, 1999] приведены условия целостности понятия треугольник, имеющего углы α, β, γ , тогда условием целостности будет равенство: сумма углов треугольника равна 180° градусов, т.е. $\alpha + \beta + \gamma = 180^\circ$.

В семиотических когнитивных картах представляются динамические ситуации и поэтому в слотах условий целостности определена

когнитивная карта (F, W) . Причинно-следственные отношения для значений признаков W могут быть заданы матрицей смежности ориентированного графа, где $w_{ij} \in W$ определяет силу связи факторов f_i и f_j . Сила связи факторов определяется на множестве их значений Z_i, Z_j . В общем случае такая зависимость задается отображением: $W: \times_i Z_i \rightarrow \times_i Z_i$

где W – множество правил отображения значений факторов.

В этом слоте задано уравнение динамики значений признаков, позволяющее моделировать исторический процесс изменения их значений – эволюцию. Уравнение динамики может быть задано системой конечно-разностных, логико-лингвистическими уравнений, которое в общем виде представлено как конечно-разностное уравнение вида: $Z(t+1) = W^Z(t)$, где $^\circ$ - правило вывода, $Z(t+1), Z(t) = (z_1, \dots, z_q)$ – векторы значений признаков в последовательные моменты времени.

3.3. Связи наследования

В классической фреймовой модели знака связи наследования представляются в слотах $1, \dots, q$ и определяются парой: имена понятий $1, \dots, q$ и тип связи (отношения) между понятиями. Обычными типами связи (отношениями) являются отношения Часть-Целое, Класс-Подкласс и др. Фактически задание связей наследования во фрейме знака позволяет определить его место в понятийной иерархии предметной области. Понятийная иерархия – это онтология предметной области, к которой этот знак принадлежит.

В семиотической когнитивной карте в этом слоте определяется связь наследования в виде так называемого концептуального каркаса [Кулинич, 2013]. Концептуальный каркас отражает возможную идеальную структуру знаний о предметной области. Применяются качественные концептуальные каркасы, структурирующие фазовое пространство динамической системы, заданной когнитивной картой. Подробно такие каркасы были описаны в работе [Кулинич, 2013].

3.4. Статические связи

В классической фреймовой модели статические связи также определяются парой: имена понятий $1, \dots, r$ и тип статической связи. Под статическими связями понимаются отношения между понятиями типа: омонимы, синонимы и др. Во фрейме семиотической когнитивной карты слотов статических связей – нет.

3.5. Условия активности

В знаке-фрейме условия активности рассматриваются в рамках т.н. метазнака. Под метазнаком понимается управляющая система, оценивающая текущее состояние знака-фрейма (значения признаков, условия целостности, и т.д.) и предлагающая действия, учитывающую оценку

этого состояния. Условия активности реализуются множеством правил вида [Поспелов, 1999]:

$$(C_i, A_i, H_i, B_i), \text{ где}$$

C_i – имя продукции;

A_i – условия активизации продукции;

H_i – последствия;

B_i – действия.

В качестве одного из возможных условий активности в работе [Поспелов, 1999] рассматривается наличие в знаке так называемого когнитивного диссонанса [Фестингер, 1999]. Фактически когнитивный диссонанс в знаке – это конфликт в моделируемой с помощью знака системе. Разрешение конфликта основано на оценке H_i его последствий. В зависимости от оценки последствий, конфликт разрешается действием B_i : либо путем изменения самого знака (изменение ситуации и, следовательно, изменение формальной системы), либо конфликт считается несущественным и знак (формальная система) остается неизменной до тех пор, пока оценка последствий конфликта не изменится до уровня, требующего изменения знака.

В семиотических когнитивных картах условия активности несколько иные. Основаны они на следующем рассуждении. Во-первых, считается, что когнитивные диссонансы, возможные в когнитивной карте отражают не только существующие в реальности конфликты, но могут отражать наше незнание, неполноту картины мира, отраженную в когнитивной карте. Конечно, в условиях неопределенности считать когнитивный диссонанс условием активности для принятия решения B_i , изменяющего знак нельзя. Здесь действие может означать уточнение (изучение) ситуации, и только потом принятие решения.

Поэтому, условие активности в семиотических когнитивных картах возникает в случаях смены состояния семиотической системы, основанном правиле смены состояния семиотической системы и модели оценивания ее настоящего и будущего состояния.

3.5.1. Правило смены состояния

Правило смены состояния основано на анализе состояния динамической системы, а именно векторов состояния $Z(t+1)=(z_1, \dots, z_q)$. Считается, что если в состоянии $Z(t+1)$ значение хотя бы одного признака вышло за пределы базового понятия $SS(d)=\times \Delta_i$, то происходит изменение состояния

семиотической системы, а именно, качественное обобщение по признаку или признакам, значения которых вышли за пределы базового понятия.

В работе [Кулинич, 2013] были определены положительное или отрицательное обобщение по одному признаку и качественное обобщение по нескольким признакам.

Качественное положительное обобщение:

$$\exists Z_q \in Z(t+1) \setminus Z_q \notin SS(d^0) \& Z_q > \sup \Delta q \rightarrow \\ SS(d^q) = \times \Delta_i \times [\inf \Delta q; \sup Z_q]$$

Качественное положительное обобщение базового понятия представлено на рисунке 3.

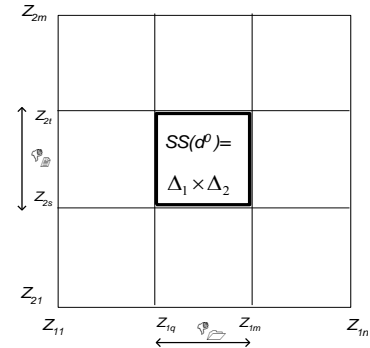


Рисунок 2 – Базовое понятие

На рисунке 2 представлено базовое понятие d^0 , содержание которого определено как подпространство $\Delta_1 \times \Delta_2$ семантического пространства $\times Z_i$. Если значение признака Z_l , $z_l(t+1)$ превысило значение верхней границы интервала толерантности $\Delta_l = [Z_{1q}; Z_{1m}]$, т.е. $z_l(t+1) > z_{1m}$, то согласно правилу смены семиотической системы происходит обобщение базового понятия по признаку Z_l , т.е. образуется новое понятие с именем d^l и содержанием $SS(d^l) = [Z_{1q}; Z_{1n}] \times \Delta_2$.

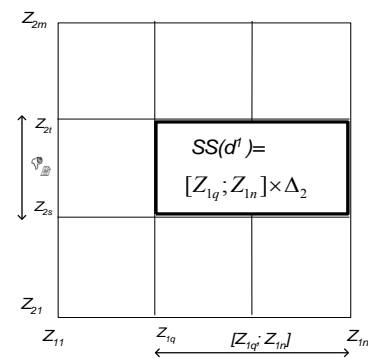


Рисунок 3 – Положительное обобщение базового понятия

Качественное отрицательное обобщение:

$$\exists Z_q \in Z(t+1) \setminus Z_q \notin SS(d) \& Z_q < \inf \Delta q \rightarrow \\ SS(d^q) = \times \Delta_i \times [\sup \Delta q; \inf Z_q]$$

Качественное отрицательное обобщение базового понятия представлено на рисунке 4.

Если значение признака Z_l , $z_l(t+1)$ стало меньше значения нижней границы интервала толерантности $\Delta_l = [Z_{1q}; Z_{1m}]$, т.е. $z_l(t+1) < z_{1q}$, то согласно правилу смены семиотической системы

происходит отрицательное обобщение базового понятия по признаку Z_l , т.е. образуется новое понятие с именем d^2 и содержанием $SS(d^2)=[Z_{11}; Z_{1m}] \times \Delta_2$.

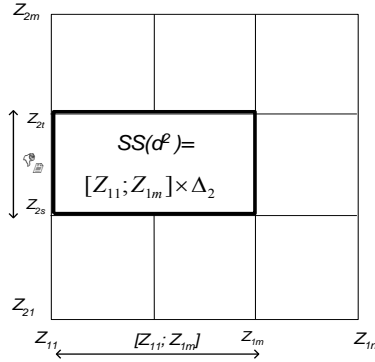


Рисунок 4 - Отрицательное обобщение базового понятия

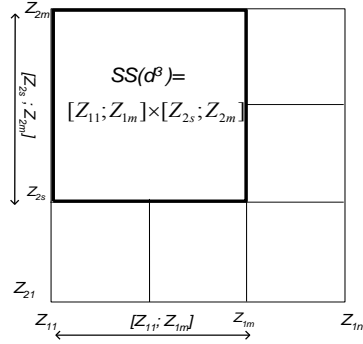


Рисунок 5 - Обобщение базового понятия по двум признакам

3.5.2. Качественное обобщение по нескольким признакам (общий случай).

Представим вектор состояния в следующем виде:

$$Z(t+1) = Z_A^*(t+1) \cup Z_B^+(t+1) \cup Z_C^-(t+1)$$

где $Z_A^*(t+1)$ – часть вектора состояния, содержащего элементы $z_i(t+1)$ значения которых не пересекали границ интервалов толерантности, т.е. $z_i(t+1) \in \Delta_i$, $i \in A$, где A – множество индексов элементов $Z_A^*(t+1)$;

$Z_B^+(t+1)$ – часть вектора состояния, содержащего элементы $z_k(t+1)$ значения которых больше верхней границы их интервалов толерантности, т.е. $z_k(t+1) > \sup \Delta_k$, $k \in B$, где B – множество индексов элементов $Z_B^+(t+1)$;

$Z_C^-(t+1)$ – часть вектора состояния, содержащего элементы $z_g(t+1)$ значения которых меньше нижней границы их интервалов толерантности, т.е. $z_g(t+1) < \inf \Delta_g$, $g \in C$, где C – множество индексов элементов $Z_C^-(t+1)$.

Правило смены состояния в общем виде запишется так:

$$\begin{aligned} Z_A^*(t+1) \cup Z_B^+(t+1) \cup Z_C^-(t+1) &\in Z(t+1) \\ \rightarrow SS(d^q) &= \times_{i \in A} \Delta_i \times_{k \in B} [\inf \Delta_k; \sup Z_k] \\ &\times_{g \in C} [\sup \Delta_g; \inf Z_g] \end{aligned}$$

Качественное обобщение базового понятия по двум признакам представлено на рисунке 5.

Здесь значение признака Z_l , $z_l(t+1)$ стало меньше значения нижней границы интервала толерантности $\Delta_l = [Z_{1q}; Z_{1m}]$, т.е. $z_l(t+1) < z_{1q}$, а значение признака Z_2 стало больше интервала толерантности $z_2(t+1) > z_{2t}$. Тогда, согласно правилу обобщения базового понятия по двум признакам получаем новое понятие с именем d^3 и с содержанием $SS(d^3) = [Z_{11}; Z_{1m}] \times [Z_{2s}; Z_{2m}]$.

3.5.3. Оценка состояния семиотической системы

Оценка состояния семиотической системы основана на определении относительной важности всех признаков и их текущих значений относительно экспертно заданной цели G_i . Формально оценка состояний может быть представлена в виде следующего отображения:

$$G_i: Z(t) \rightarrow (o_{ij}, \succ), \text{ где}$$

G_i – отображение текущих значений всех факторов для i -ой цели;

$Z(t)$ – текущие значения всех факторов;

(o_{ij}, \succ) – упорядоченные оценки важности всех признаков относительно i -ой цели.

В качестве метода построения этого отображения в работах [Аверкин, 2006; Кулинич, 2005] был рассмотрен метод анализа иерархий (МАИ) Т. Саати [Саати, 1993]. В этом случае все оценки o_{ij} получены методом парных оценок предпочтительности признаков и нормированы к единице, т.е.

$$\sum_j o_{ij} = 1.$$

Оценка состояния $O_i(t)$ представляется линейной сверткой:

$$O_i(t) = \sum_j o_{ij} x_j(t), j=1, \dots, n, \text{ где}$$

$x_j(t) \in [0, 1]$ – получено в результате отображения φ лингвистических значений признаков $z_j(t)$ на отрезок числовой оси $[0, 1]$, т.е. $\varphi: z_j(t) \rightarrow x_j(t) \in [0, 1]$.

В этом случае оценка $O_i(t)$ отражает степень достижения цели в состоянии $z_j(t)$.

В семиотических когнитивных картах важны не столько оценки достижения цели, а относительное изменение этой оценки в состоянии $z_j(t+1)$ и в текущем состоянии $z_j(t)$. Относительное изменение оценки γ_i определяется из следующего соотношения:

$$\gamma_i = \frac{O_i(t+1) - O_i(t)}{O_i(t)}.$$

Выбор действия B_i в семиотических когнитивных картах зависит от оценки γ_i и анализе нового состояния, представленного парой имя обобщенного состояния - d^{Ht} и его содержания $SS(d^{Ht})$.

3.7. Представление

В знаке-фрейме в слотах $1, \dots, t$ представлено множество экземпляров знака-фрейма. По сути это объем понятия $d - V(d)$. В семиотических когнитивных картах, в слот представления попадают состояния $Z(t+1)$, значения элементов вектора которых попадают в область толерантности соответствующего понятия, т.е. $Z(t+1) = (z_1, \dots, z_q) \in SS(d^0)$.

3.8. Выбор действия

Выбор действия основан на анализе относительного изменения оценки γ_i при изменении состояния семиотической системы и собственно самого изменения базового понятия $d^0 \rightarrow d^{Ht}$ и $SS(d^0) \rightarrow SS(d^{Ht})$.

Если оценка относительного изменения больше нуля, т.е. $\gamma_i > 0$, то динамика изменения (эволюция) семиотической системы происходит в направлении к цели. В этом случае, никаких действий можно и не принимать.

Если оценка относительного изменения меньше нуля, т.е. $\gamma_i < 0$, то динамика изменения (эволюция) семиотической системы происходит в направлении от цели. В этом случае, формулируются задачи параметрического и(или) структурного управления.

1. Параметрическое управление сводится к поиску управляющих воздействий $U(t)$, направленных на изменение значений признаков, способных изменить траекторию изменения значений признаков в когнитивной карте и, следовательно, изменить знак оценки относительного изменения состояния, т.е. $\gamma_i > 0$.

Формально поиск управляющих воздействий для заданной цели G_i сводится к решению уравнения $G_i = W^o U(t)$ относительно управляющих воздействий $U(t)$, т.е. к решению обратной задачи.

Решение запишется в следующем виде: $U^*(t) = G_i \cdot W^{-1}$ где $U^*(t) \in D$, $U^*(t)$ - решение обратной задачи – новые значения признаков, позволяющие достичь цели G_i , D – ограничения на возможные значения признаков, \cdot – процедура обратного вывода.

2. Структурное управление. Необходимость структурного управления возникает в случаях, когда не удастся найти управления, удовлетворяющих ограничениям, $U^*(t) \notin D$.

Суть структурного управления заключается в изменении когнитивной карты (матрицы смежности W), таким образом, чтобы ее

подстановка в уравнение обратной задачи $U^*(t) = G_i \cdot W^{-1}$ позволяло получать значения признаков, удовлетворяющие ограничениям $U^*(t) \in D$.

4. Целостность знака семиотической когнитивной карты.

В семиотике целостность знака, представленного тройкой: имя, смысл, значение, определяют семиотические операции между этими элементами. Это следующие базовые операции: *описания* (связывает имя и смысл (имя и содержание ($d \rightarrow F(d)$); *обозначения* (связывающие содержание и имя ($F(d) \rightarrow d$); *представления* (связывает имя и объем ($d \rightarrow V(d)$); *именования* (связывает объем и имя ($V(d) \rightarrow d$); *конкретизации* (связывает содержание и объем ($F(d) \rightarrow V(d)$); *обобщения* (связывает объем и содержание ($V(d) \rightarrow F(d)$).

Семиотические операции – это операции естественного интеллекта, связывающие три элемента знака Фреге в единое целое. Эти операции могут быть поддержаны специальными экспертными процедурами, позволяющие получить формализованное описание целостного знака. В предлагаемой модели поддержка базовых операций, обеспечивающих целостность знака когнитивной карты, основана на определении зависимостей слотов когнитивной карты семиотического типа и экспертных процедурах определении имени нового обобщенного понятия и его объема.

Зависимости слотов семиотической когнитивной карты, определяющие целостность когнитивной карты как знака:

1. Имя понятия определяет его содержание в виде множества признаков и их возможных значений. Содержание представляется в виде пары $F(d) = (\{f_{ij}\}, SS(d))$;
2. Множество признаков $\{f_{ij}\}$, включенные в содержание $F(d)$ является множеством факторов когнитивной карты, для которых определены причинно-следственные отношения между значениями признаков $Z_i - W: \times Z_i \rightarrow \times Z_i$. Матрица смежности орграфа W определяет коэффициенты в уравнении динамики: $Z(t+1) = W \cdot Z(t)$;
3. Область толерантности базового понятия $SS(d)$ определяет концептуальный каркас ($SS(d^{Ht}), \cup, \cap$) предметной области, представляющего собой идеализированную родовидовую структуру понятийной системы предметной области.
4. Изменение значений признаков $Z(t+1)$ и их выход за пределы области толерантности базового понятия приводит к смене состояния
5. Правило смены состояния семиотической системы основано на анализе относительной оценке изменения состояния семиотической

системы γ_i и экспертном определении имени d^H и объема нового состояния $V(d^H)$.

Таким образом, зависимости между слотами определяют динамическую систему. Это означает, что в семиотической когнитивной карте, для любого нового состояния $Z(t+1)$ должны быть определены семиотические операции: *обобщения* (новый объем изменяет содержание ($V(d) \rightarrow F(d)$); *обозначения* (новое содержание изменяет имя ($F(d) \rightarrow d$); *представления* (новое имя связывается с новым объемом ($d \rightarrow V(d)$)).

Формально когнитивную карту семиотического типа как динамическую систему будем представлять в виде следующего кортежа:

$$\langle d(t); F(d(t)); V(d(t)); (\{f_{ij}\}, W); W: Z(t) \rightarrow Z(t+1); \Omega \rangle,$$

где $d(t)$ – имя понятия; $F(d(t))$ – содержание понятия, $(\{f_{ij}\}, SS(d(t)))$; $V(d(t)) = (\{Z(t+1) = (z_1, \dots, z_q) \in SS(d^H)\})$ – объем понятия; $(\{f_{ij}\}, W)$ – когнитивная карта; $W: Z(t) \rightarrow Z(t+1)$ – уравнение динамики; $\Omega = (\gamma, d^H, V(d^H))$ – правило смены состояния когнитивной карты семиотического типа.

Заключение

Применение предложенной модели когнитивной карты семиотического типа позволит значительно расширить возможности систем поддержки принятия решений в динамических плохо определенных ситуациях.

Библиографический список

- [Аверкин и др., 2006] Аверкин А.Н., Кузнецов О.П., Кулинич А.А., Титова Н.В. Поддержка принятия решений в слабоструктурированных предметных областях. Анализ ситуаций и оценка альтернатив. Теория систем и управления. Вып. 3, 2006, стр. 139-149.
- [Аверкин, 1999] Аверкин А.Н., Головина Е.Ю. Нечеткая семиотическая система управления. Конференция «Интеллектуальное управление: новые интеллектуальные технологии в задачах управления (ISIT'99)» - Труды Международной конференции, Переславль-Залесский 6-9 декабря, 1999. – М.: Наука. Физматлит, 1999. стр. 141-145.
- [Бирюков, 1960] Бирюков Б. В. Теория смысла Готлоба Фреге. В кн.: Применение логики в науке и технике. М., 1960, с.502-555.
- [Еремеев, 2003] Еремеев А.П., Тихонов Д.А., Шутова П.В. Поддержка принятия решений в условиях неопределенности на основе немарковской модели // Изв. РАН: Теория и система управления, 2003. - № 5. - С. 75-88.
- [Кулинич, 2005] Кулинич А.А., Титова Н.В. Интеграция нечетких моделей динамики и оценивания ситуаций. Когнитивный анализ и управление развитием ситуаций (CASC 2005). Труды 5-й международной конференции. М.: Институт проблем управления РАН. 2005. стр.107-118.
- [Кулинич, 2006] Кулинич А.А. Моделирование динамических процессов в понятийной системе субъекта для генерации креативных решений. Когнитивные исследования: Сборник научных трудов: Вып. 1/ Под редакцией В.Д. Соловьева. – 2006. с. 94-123.
- [Кулинич, 2010] Кулинич А.А. Компьютерные системы моделирования когнитивных карт: подходы и методы. / Проблемы управления, 2010, № 3 С.2-16.
- [Кулинич, 2011a] Кулинич А.А. Классификация когнитивных карт и методы их анализа. (Пленарный доклад). Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов VI-й

международной научно-технической конференции (Коломна, 16-19 мая 2011 г.). – М.: Физматлит, 2011. – т.1. с. 124-136.

[Кулинич, 2011b] Кулинич А.А. Компьютерные системы анализа ситуаций и поддержки принятия решений на основе когнитивных карт: подходы и методы. / Проблемы управления, 2011, № 4 С.31-45.

[Кулинич, 2013] Кулинич А.А. Концептуальные «каркасы» плохо определенных предметных областей. Открытые семантические технологии проектирования интеллектуальных систем: материалы III международной научно-технической конференции (Минск, 21-23 февраля 2013 г.)/ под.ред. Голенкова В.В. – Минск: БГУИР, 2013, с. 135-142.

[Мартынов, 2001] Мартынов В.В. Основы семантического кодирования. Опыт представления и преобразования знаний. Минск, 2001.

[Минский, 1979] Минский М. Фреймы для представления знаний. М.: Мир, 1979.

[Осипов, 2002] Осипов Г.С. От ситуационного управления к прикладной семиотике. Новости искусственного интеллекта. 2002, № 6.

[Поспелов, 1986] Поспелов Д. И. Ситуационное управление: теория и практика. -М.: Наука, 1986.

[Поспелов, 1999] Поспелов Д.А., Осипов Г.С. Прикладная семиотика // Новости ИИ. - 1999. - №1.

[Саати, 1993] Саати Т. Принятие решений. Метод анализа иерархий. Пер. с англ. – М.: «Радио и связь», 1993. – 320 с.

[Тарасов, 2002] Тарасов В.Б. От многоагентных систем к интеллектуальным организациям. М.: Эдиториал УРСС, 2002. – 352 с.

[Фестингер, 2013] Фестингер Л. Теория когнитивного диссонанса. СПб.: Ювента, 1999.

[Эрлих, 1997] Эрлих А.И. Прикладная семиотика и управление сложными объектами. Программные продукты и системы. № 3, 1997.

COGNITIVE MODELLING IN THE CONDITIONS OF UNCERTAINTY (THE SEMIOTICS APPROACH)

A. A. Kulinich

*V.A. Trapeznikov Institute of Control Sciences,
Russian Academy of Sciences. Russia, Moscow.*

kulinich@ipu.ru

In work the new approach to modeling of ill-structured situations on a basis cognitive maps of semiotics type is considered. The frame representation model of semiotics cognitive maps is defined.

Introduction

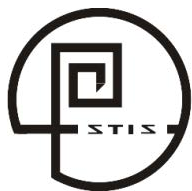
Problems of decision-making support in the dynamic systems in the conditions of uncertainty on a basis cognitive modeling are considered. The new paradigm of cognitive modeling on the basis of principles of applied semiotics is offered.

Main Part

The frame model of representation of semiotics type cognitive maps is considered.

Conclusion

Application of the offered cognitive maps model of semiotics type will allow to expand considerably possibilities of systems of decision-making support in dynamic ill-structured situations.



УДК 004.89

МАТРИЧНОЕ ПРЕДСТАВЛЕНИЕ КОНЕЧНЫХ ПРЕДИКАТОВ ДЛЯ АВТОМАТИЗАЦИИ ЛОГИКО-СЕМАНТИЧЕСКОГО АНАЛИЗА

Зуенко А.А.

Институт информатики и математического моделирования КНЦ РАН, Анапты, Россия

zuenko@iimm.kolasc.net.ru

В статье рассматривается программная система, основанная на матричном представлении конечных предикатов и теоретико-множественной интерпретации логических операций, принятой в алгебре кортежей. Система реализует оригинальные алгебраические методы решения таких задач логико-семантического анализа, как проверка правильности следования; вывод абдуктивных заключений; анализ семантических некорректностей. Также система может быть полезна при решении задач удовлетворения ограничений, которые сводятся к поиску выполняющих подстановок конъюнктивной нормальной формы конечного предиката.

Ключевые слова: алгебра кортежей, логико-семантический анализ, матричное представление конечных предикатов.

Введение

Данная работа продолжает цикл публикаций, посвященный методам алгебры кортежей (АК).

Ранее в [Кулик и др., 2010], [Зуенко и др., 2011], [Зуенко и др., 2013] рассматривались методы АК, предназначенные для решения различных задач логико-семантического анализа таких, как: проверка правильности логического следования; получение возможных следствий из системы посылок при заданных ограничениях; вывод абдуктивных заключений; анализ семантических некорректностей. Особенностью задач, решаемых средствами АК, является то, что они описываются на языке конечных предикатов.

В АК используется матричное представление конечных предикатов и теоретико-множественная интерпретация логических операций с ними. Близкий подход применяется также в [Zakrevskij, 2012] для решения задач распознавания образов.

В АК конечные предикаты можно сжато представить с помощью двух типов структур: *C*-систем и *D*-систем. Они формируются в виде матриц из подмножеств доменов атрибутов, называемых компонентами. В их число входят две фиктивные компоненты: полная компонента (обозначается «*») – это множество, равное домену соответствующего (по месту ее расположения в кортеже) атрибута; пустое множество – \emptyset . В виде *D*-систем удобно представлять конъюнктивные нормальные формы (КНФ) конечных предикатов, а

в виде *C*-систем – их дизъюнктивные нормальные формы (ДНФ). Так, например, в АК процесс поиска всех выполняющих подстановок некоторой КНФ сводится к преобразованию *D*-системы в *C*-систему.

В [Зуенко, 2013] показаны методы АК решения задач удовлетворения ограничений (constraint satisfaction problem, CSP). CSP характеризуются тем, что в них используются переменные, которые имеют конечные области определения. К числу таких задач относится задача раскраски карты (раскраски графа). Также к категории задач с конечными областями относятся булевы задачи CSP. Булевы задачи включают в качестве частных случаев некоторые *NP*-полные задачи, такие как 3SAT [Рассел и др., 2006]. Решение CSP в этом случае сводится к определению выполняющих подстановок (единственной подстановки) конъюнктивной нормальной формы конечного предиката.

Методы ускорения вычислительных процедур, основанные на матричном представлении конечных предикатов, позволяют на практике получать требуемые решения упомянутых задач за приемлемое время. Дополнительные возможности ускорения связаны с анализом семантических ограничений предметной области [Кулик и др., 2010].

Рассмотрим некоторые особенности программной реализации разработанной среды для автоматизации вычислений с АК-объектами.

1. Представление логических матриц и операций над ними с помощью булевых векторов

В основе всех операций с АК-объектами лежат операции с компонентами, из которых АК-объекты состоят. На уровне кода программы, на языке C++, компоненты представляются с помощью булевых векторов: каждая компонента моделируется последовательностью бит, чья размерность равна размерности соответствующего домена. Биту соответствует значение, принадлежащее определенному домену. Если некоторый бит установлен в "true", считается, что соответствующий ему элемент домена присутствует в компоненте (см. рисунок 1). Здесь и далее в рисунках, иллюстрирующих операции, будем слева показывать булевы векторы, а справа – соответствующие им компоненты АК-объектов или АК-объекты целиком.



Рисунок 1. Пример представления компоненты на основе булева вектора

Конечный пользователь формирует компоненты логических матриц привычным для АК способом – в виде множеств, а уже благодаря внутрипрограммному представлению, операции с компонентами сводятся к побитовым операциям с булевыми векторами.

Проиллюстрируем операцию пересечения компонент (см. рисунок 2):



Рисунок 2. Пример операции пересечения компонент

Также можно продемонстрировать операцию объединения компонент (см. рисунок 3):



Рисунок 3. Пример операции объединения компонент

Рассмотрим операцию дополнения компоненты (см. рисунок 4):

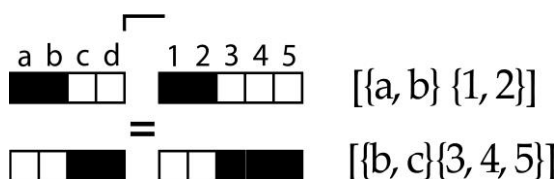


Рисунок 4. Пример операции дополнения компонент

Далее перейдем к описанию системы команд, с помощью которой конечный пользователь формирует алгоритмы (АК-программы) в рассматриваемой среде автоматизации преобразований логических матриц.

2. Описание системы команд программной среды

Система команд построена на присваиваниях, в результате которых создается новая переменная, и ей присписывается некоторое значение. Команда в общем виде выглядит следующим образом:

$\langle \text{NEWNAME} \rangle = \langle \text{DATA} \rangle$

Здесь $\langle \text{NEWNAME} \rangle$ – любое алфавитно-цифровое имя новой переменной, не совпадающее ни с одним из зарезервированных или уже использованных слов в документе, а $\langle \text{DATA} \rangle$ – любое из выражений $\langle \text{DOM} \rangle$, $\langle \text{ATT} \rangle$, $\langle \text{SCH} \rangle$, $\langle \text{EX} \rangle$.

Объявление домена с именем ($\langle \text{DOMNAME} \rangle$) и типом ($\langle \text{DOMTYPE} \rangle$), а также с указанием его содержимого производится в следующем виде:

$\langle \text{DOMNAME} \rangle = \text{domain}(\langle \text{DOMTYPE} \rangle; [\langle \text{EL} \rangle \{, \langle \text{EL} \rangle \}])$

Здесь тип домена $\langle \text{DOMTYPE} \rangle$ может принимать одно из следующих значений: «string», «int», «float», при этом домен будет иметь соответственно строковый, целочисленный или десятичный с плавающей запятой тип. Элементы домена $\langle \text{EL} \rangle$ указываются через запятую и должны соответствовать указанному типу. В числах с плавающей запятой в качестве разделителя используется точка. Приведем следующие примеры объявления домена:

$\text{Dom}_0 = \text{domain}(\text{string}; \text{mon}, \text{tue}, \text{wed}, \text{thu}, \text{fri}, \text{sat}, \text{sun});$

$\text{Dom}_1 = \text{domain}(\text{int}; 1, 2, 3, 4, 5, 6, 7, 8, 9);$

$\text{Dom}_2 = \text{domain}(\text{float}; 3.1415, -897.6, 0.14, -15.1345).$

Объявление атрибута с именем $\langle \text{ATTNAME} \rangle$, и указанием его домена выглядит следующим образом:

$\langle \text{ATTNAME} \rangle = \text{attribute}(\langle \text{DOM} \rangle).$

Примером объявления атрибута может служить следующая запись:

$A_1 = \text{attribute}(\text{Dom}_1).$

Объявление схемы $\langle \text{SCH} \rangle$ может быть как указанием на уже существующую схему по ее имени ($\langle \text{SCHNAME} \rangle$), так и объявлением новой схемы с указанием входящих в нее атрибутов:

$\langle \text{SCHNAME} \rangle = \text{scheme}([\langle \text{ATT} \rangle \{, \langle \text{ATT} \rangle \}]).$

Пример объявления схемы:

$S_1 = \text{scheme}(A_1, B_1).$

Выражение <EX> может быть как объявлением АК-объекта (<SYS>), так и объявлением массива АК-объектов (<ARR>). АК-объект <SYS> описывается путем поэлементного указания его компонент (<SYSDEF>) или путем объявлением операции <OP>, результатом вычисления которой этот АК-объект является.

Аналогично, объявление массива АК-объектов <ARR> может быть либо конкретным массивом (<ARRDEF>), либо объявлением операции <ARROP>, результатом вычисления которой служит массив АК-объектов.

Объявление конкретной системы (<SYSDEF>) может быть указанием на уже существующий АК-объект (<SYSNAME>) или объявлением новой системы с указанием ее вида <SYSTYPE>, который принимает значения «c-system» или «d-system». В качестве примера объявления <SYSDEF> приведем следующие команды:

$sys_1 = \text{c-system}(S_1; [\{3, 4, 5\}, \{a, b\}], [\{2\}, \{b\}]);$
 $sys_2 = \text{d-system}(S_1; [\#, \{b, c, d\}], [\{8\}, \{c, d\}], [\{7, 8, 9\}, \{d\}]).$

На языке АК эти структуры выглядят следующим образом:

$$Sys_1[A_1B_1] = \begin{bmatrix} \{3,4,5\} & \{a,b\} \\ \{2\} & \{b\} \end{bmatrix};$$

$$Sys_2[A_1B_1] = \begin{bmatrix} \emptyset & \{b,c,d\} \\ \{8\} & \{c,d\} \\ \{7,8,9\} & \{d\} \end{bmatrix}.$$

Формат задания массива АК-объектов следующий:

$ARRDEF = \text{array}([\langle SYS \rangle \{, \langle SYS \rangle \}]).$

Все входящие в массив АК-объекты должны принадлежать к одному типу (либо *C*, либо *D*). Приведем пример объявления массива:

$arr = \text{array}(sys_1, sys_2, sys_3).$

Далее опишем основные операции <OP> над *C*- и *D*-структурами, поддерживаемые рассматриваемой системой автоматизации матричных символьных преобразований:

$\sim \langle SYS \rangle$ – преобразование АК-объекта в альтернативный класс;

$! \langle SYS \rangle$ – дополнение;

$\langle SYS \rangle * \langle SYS \rangle$ – пересечение;

$\langle SYS \rangle + \langle SYS \rangle$ – объединение;

$\text{elimination}(\langle SYS \rangle; [\langle ATT \rangle \{, \langle ATT \rangle \}])$ – элиминация атрибутов, где через запятую в любом порядке указываются все атрибуты, подлежащие элиминации;

$\text{projection}(\langle SYS \rangle; [\langle ATT \rangle \{, \langle ATT \rangle \}])$ – проекция, где через запятую в любом порядке указываются все

атрибуты, которые должны остаться в результирующей системе.

Аналогичным образом определены операции преобразования в альтернативный класс, дополнения, пересечения, объединения, элиминации атрибутов, проекции для массивов АК-объектов:

$\langle ARR \rangle '[, \langle INDEX \rangle ']'$ – индексация массива;

$\sim \langle ARR \rangle$ – преобразование в альтернативный класс массива АК-объектов;

$! \langle ARR \rangle$ – вычисление дополнения для массива АК-объектов;

$\langle ARR \rangle * \langle SYS \rangle$ – пересечение каждого элемента массива АК-объектов <ARR> с АК-объектом <SYS>;

$\langle ARR \rangle + \langle SYS \rangle$ – объединение каждого элемента массива АК-объектов <ARR> с АК-объектом <SYS>;

$\text{elimination}'(, \langle ARR \rangle; [[\langle ATT \rangle \{, \langle ATT \rangle \}]]'$

$\text{projection}'(, \langle ARR \rangle; [[\langle ATT \rangle \{, \langle ATT \rangle \}]]'$ – соответственно операции элиминации и взятия проекции для каждого элемента массива в отдельности. Помимо этого, определена операция нахождения всех возможных проекций:

$\text{all_projections}'(, \text{full} | \text{not_full} | \text{all}; \langle SYS \rangle ')$.

В зависимости от заданного параметра (full, not_full, all) результатом операции будут соответственно полные, неполные, или все возможные проекции системы (массива).

Далее показаны примеры использования представленных ранее команд.

3. Примеры использования командного процессора

Пример 1. Проверить правильность логического вывода для следующего рассуждения: "Если Джонс не встречал этой ночью Смита, то либо Смит был убийцей, либо Джонс лжет. Если Смит не был убийцей, то Джонс не встречал Смита этой ночью, и убийство имело место после полуночи. Если убийство имело место после полуночи, то либо Смит был убийцей, либо Джонс лжет. Следовательно, Смит был убийцей".

Сначала выразим данное рассуждение на языке исчисления высказываний [Kulik et al., 2010]. Введем обозначения:

A – Джонс встречал этой ночью Смита;

B – Смит был убийцей;

C – Джонс лжет;

D – убийство имело место после полуночи.

Тогда заданное рассуждение можно сформулировать так:

первое предложение: $\neg A \supset (B \oplus C)$;

второе предложение: $\neg B \supset (\neg A \wedge D)$;

третье предложение: $D \supset (B \oplus C)$;

следствие: B .

Преобразуем первые три предложения для того, чтобы избавиться в них от импликации и строгой дизъюнкции. Получаем:

$$1) \neg A \supset (B \oplus C) = A \vee (B \wedge \neg C) \vee (\neg B \wedge C);$$

$$2) \neg B \supset (\neg A \wedge D) = B \vee (\neg A \wedge D);$$

$$3) D \supset (B \oplus C) = \neg D \vee (B \wedge \neg C) \vee (\neg B \wedge C).$$

Для представления этих формул АК-объектами используем универсум $X_1 \times X_2 \times X_3 \times X_4 = \{0, 1\}$, где $A = B = C = D = 1$ и $\neg A = \neg B = \neg C = \neg D = 0$. Тогда посылки, которые являются ДНФ, выражаются С-системами:

$$P_1 = \begin{bmatrix} \{1\} & * & * & * \\ * & \{1\} & \{0\} & * \\ * & \{0\} & \{1\} & * \end{bmatrix};$$

$$P_2 = \begin{bmatrix} * & \{1\} & * & * \\ \{0\} & * & * & \{1\} \end{bmatrix};$$

$$P_3 = \begin{bmatrix} * & * & * & \{0\} \\ * & \{1\} & \{0\} & * \\ * & \{0\} & \{1\} & * \end{bmatrix},$$

а следствие – С-кортежем $B[X_2] = [\{1\}]$.

Чтобы создать домен, пользователь вводит команду:

$dom = \text{domain}(\text{int}; 0, 1).$

Также необходимо определить атрибуты:

$x_1 = \text{attribute}(dom);$

$x_2 = \text{attribute}(dom);$

$x_3 = \text{attribute}(dom);$

$x_4 = \text{attribute}(dom).$

Далее задаются необходимые схемы:

$sch_1 = \text{scheme}(x_1, x_2, x_3, x_4);$

$sch_2 = \text{scheme}(x_2).$

Для создания АК-объектов, моделирующих посылки и заключение требуются команды:

$p_1 = \text{c-system}(sch_1; [\{1\}, *, *, *], [*, \{1\}, \{0\}, *], [*, \{0\}, \{1\}, *]);$

$p_2 = \text{c-system}(sch_1; [*, \{1\}, *, *], [\{0\}, *, *, \{1\}]);$

$p_3 = \text{c-system}(sch_1; [*, *, *, \{0\}], [*, \{1\}, \{0\}, *], [*, \{0\}, \{1\}, *]);$

$b = \text{c-system}(sch_2; [\{1\}]).$

Если вычислить АК-объект, соответствующий выражению $P_1 \cap_G P_2 \cap_G P_3$, получим:

$$P[X_1 X_2 X_3 X_4] = P_1 \cap_G P_2 \cap_G P_3 = \begin{bmatrix} \{1\} & \{1\} & * & \{0\} \\ * & \{1\} & \{0\} & * \\ \{0\} & \{0\} & \{1\} & \{1\} \end{bmatrix}.$$

Для этого введем соответствующую команду:

$p = \text{compression}(1; p_1 * p_2 * p_3).$

Здесь операция compression используется для устранения повторяющихся кортежей.

Решается задача путем определения пустоты АК-объекта $P_1 \cap_G P_2 \cap_G P_3 \cap_G B[X_2]$.

Можно использовать уже полученный ранее промежуточный результат $P[X_1 X_2 X_3 X_4]$, а можно ввести команду целиком, поскольку рассматриваемая система осуществляет разбор сложных выражений:

$check = p_1 * p_2 * p_3 * \sim!b.$

Проверка показывает, что вывод неверный. Результат работы системы с программой вычислений приведен на рисунке 5. Действительно:

$$\begin{bmatrix} \{1\} & \{1\} & * & \{0\} \\ * & \{1\} & \{0\} & * \\ \{0\} & \{0\} & \{1\} & \{1\} \end{bmatrix} \cap_G [*, \{0\}, *, *] = [\{0\} \{0\} \{1\} \{1\}] \neq \emptyset.$$

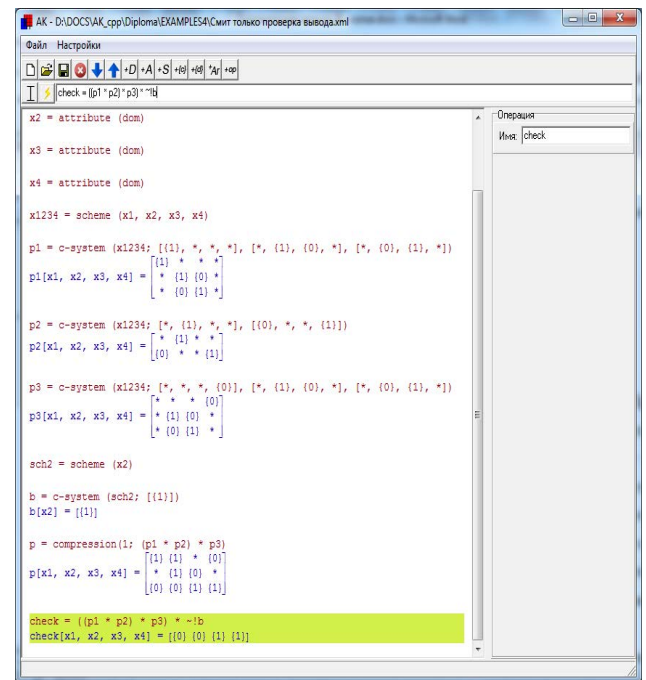


Рисунок.5. Решение задачи проверки правильности логического вывода

Пример 2. В предыдущем примере предполагаемое следствие (Смит был убийцей) не выводимо. Чтобы подтвердить или опровергнуть правильность вывода, требуется уточнить некоторые обстоятельства. Поиск таких обстоятельств можно сформулировать как задачу

поиска абдуктивного заключения [Kulik et al., 2012], [Вагин и др., 2008].

Предположим, что вывод правильный. Тогда необходимо найти подходящие гипотезы, которые можно использовать в качестве посылок. С помощью редактора можно сохранить программу вычислений предыдущего примера в файле с новым именем, удалить ненужные ветки вычислений и добавить новые. Отправной точкой будет следующий промежуточный результат:

$$P[X_1X_2X_3X_4] = \begin{bmatrix} \{1\} & \{1\} & * & \{0\} \\ * & \{1\} & \{0\} & * \\ \{0\} & \{0\} & \{1\} & \{1\} \end{bmatrix};$$

$$B[X_1X_2X_3X_4] = [* \{1\} **].$$

Далее используем алгоритм получения абдуктивных заключений, разработанный в АК [Kulik et al., 2012].

$$R = A \setminus_G B = \begin{bmatrix} \{1\} & \{1\} & * & \{0\} \\ * & \{1\} & \{0\} & * \\ \{0\} & \{0\} & \{1\} & \{1\} \end{bmatrix} \cap_G [* \{0\} **] \\ = [\{0\} \{0\} \{1\} \{1\}].$$

Для вычисления этого выражения введем команду:

$$R = p * \sim!b.$$

Здесь можно выбрать в качестве R_i любую проекцию R . Пусть это будет $R[X_4]$:

$$RX_4 = \text{projection}(R; x_4).$$

Для вычисления $H_i = \overline{R_i}$ введем следующую команду:

$$HX_4 = \sim!RX_4.$$

Поскольку коллизии нам не заданы, проверим, вырождается ли общая предпосылка P при полученной гипотезе:

$$P \cap_G H_i = \begin{bmatrix} \{1\} & \{1\} & * & \{0\} \\ * & \{1\} & \{0\} & * \\ \{0\} & \{0\} & \{1\} & \{1\} \end{bmatrix} \cap_G \\ [* ** \{0\}] = \begin{bmatrix} \{1\} & \{1\} & * & \{0\} \\ * & \{1\} & \{0\} & \{0\} \end{bmatrix}.$$

Введем соответствующую команду:

$$PH = p * HX_4.$$

Проверка подтверждает корректность гипотезы. На естественном языке данная гипотеза означает, что убийство произошло до полуночи. Отсюда следует, что вывод будет правильным, если уточнить время убийства.

Редактор поддерживает разбор сложных выражений, то есть все операции можно записать в одну строку и получить тот же результат:

$$res = \text{compression}(1; p_1 * p_2 * p_3 * \sim! \text{projection}(p_1 * p_2 * p_3 * \sim!b; x_4)).$$

Заключение

Матричное представление ДНФ и КНФ конечных предикатов, основанные на этом представлении методы решения задач логико-семантического анализа и методы ускорения вычислительных процедур позволяют организовывать вычисления с нечисловыми переменными подобно тому, как реализуются численные методы алгебры матриц в специализированных программных средах.

Разработанная система, позволяет решать следующие задачи логико-семантического анализа: проверка правильности логического следования; получение следствий, удовлетворяющих заданным условиям; получение абдуктивных заключений, анализ семантических некорректностей и т.п. Также система может быть полезна при решении задач удовлетворения ограничений, которые сводятся к поиску выполняющих подстановок КНФ конечного предиката.

Система команд описанного программного процессора реализована на основе операций с булевыми векторами с использованием библиотеки `boost::dynamic_bitset`. Разбор и вычисление сложных регулярных выражений осуществляется с использованием библиотеки `boost::regex`. Предусмотрено сохранение программы в формате XML.

Работа выполнена при поддержке РФФИ (проекты № 12-07-006689-а, № 12-07-000550-а, № 13-07-00318-а), Президиума РАН (проект 4.3 Программы № 15), ОНИТ РАН (проект 2.3 в рамках текущей Программы фундаментальных научных исследований).

Библиографический список

- [Вагин и др., 2008] Вагин, В.Н. Достоверный и правдоподобный вывод в интеллектуальных системах / Под ред. В.Н. Вагина, Д.А. Поспелова. – 2-е изд. испр. и доп. / В.Н. Вагин, Е.Ю. Головина, А.А. Загорянская, М.В. Фомина // – М.: ФИЗМАТЛИТ, 2008. – 712 с.
- [Зуенко и др., 2011] Зуенко, А.А. Интеграция баз данных и знаний интеллектуальных систем на основе алгебраического подхода / А.А. Зуенко, Б.А. Кулик, А.Я. Фридман // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2011): материалы Междунар. научн.-техн. конф. (Минск, 10-12 февраля 2011 г.) – Минск: БГУИР, 2011. – С.59-70.
- [Зуенко, 2013] Зуенко, А.А. Матрицеподобные вычисления в задачах удовлетворения ограничений / А.А. Зуенко // Шестая Всероссийская мультikonференция по проблемам управления (30 сентября – 5 октября 2013 г.). Материалы мультikonференции: в 4 т. – Ростов-на-Дону: Издательство Южного федерального университета, 2013. Т.1. – С. 30-34.
- [Зуенко и др., 2013] Зуенко, А.А. Интеллектуальные обучающие системы на основе алгебраического представления вопросно-ответных текстов / А.А. Зуенко, Б.А. Кулик, А.Я. Фридман // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2013): материалы III

Междунар. научн.-техн. конф. (Минск, 21-23 февраля 2013 г.) – Минск: БГУИР, 2013. – С.165-170.

[Кулик и др., 2010] Кулик, Б.А. Алгебраический подход к интеллектуальной обработке данных и знаний / Б.А. Кулик, А.А. Зуенко, А.Я. Фридман // – СПб.: Изд-во Политехнического ун-та, 2010. 235 с.

[Рассел и др., 2006] Рассел, С. Искусственный интеллект: современный подход. 2-е изд. / пер. с англ.; ред. К. А. Птицына./, С. Рассел, П. Норвиг // – М.: Изд. дом «Вильямс», 2006. 1408 с.

[Kulik et al., 2010] Kulik, B. Logical Analysis of Intelligence Systems by Algebraic Method / Boris Kulik, Alexander Fridman, Alexander Zuenko // Cybernetics and Systems 2010: Proceedings of Twentieth European Meeting on Cybernetics and Systems Research (EMCSR 2010), Vienna, Austria, 2010. – pp. 198-203.

[Kulik et al., 2012] Boris Kulik, Alexander Fridman, Alexander Zuenko. Logical Inference and Defeasible Reasoning in N-tuple Algebra. In: "Diagnostic Test Approaches to Machine Learning and Commonsense Reasoning Systems", IGI Global, P 102-128.

[Zakrevskij, 2012] Arkadij Zakrevskij. Integrated Model of Inductive-Deductive Inference Based on Finite Predicates and Implicative Regularities. In: "Diagnostic Test Approaches to Machine Learning and Commonsense Reasoning Systems", IGI Global, P 1-12.

MATRIX REPRESENTATION OF A FINITE PREDICATES FOR AUTOMATION OF LOGICAL AND SEMANTIC ANALYSIS

Zuenko A.A

*Institute for Informatics and Mathematical
Modelling, Kola Science Centre of RAS*

zuenko@iimm.kolasc.net.ru

The article discusses a program system based on a matrix representation of finite predicates and set-theoretic interpretation of logic operations used in the n-tuple algebra (NTA). The system realizes the original algebraic methods of solving such problems of semantic and logical analysis as: correctness check for a consequence; derivation of consequences from axioms; derivation of abductive conclusions; analysis of semantic inconsistencies. Also, the system can be useful for solving constraint satisfaction problems which are reduced to finding substitutions of conjunctive normal form of the finite predicate.

Introduction

The article continues a series of publications devoted to the NTA methods.

A matrix representation of finite predicates and the set-theoretic interpretation of logical operations are used in NTA.

The NTA methods developed to solve various problems of logical and semantic analysis were presented in our publications earlier. The distinguishing feature of interesting us problems is that they could be described in terms of finite predicates.

Also NTA methods were used to solve constraint satisfaction problems (constraint satisfaction problem, CSP) with finite domains of variables. Such problems include graph coloring problem, Boolean problems CSP and etc.

In this case solving of CSP reduced to the definition of satisfying substitutions (single substitution) conjunctive normal form of the finite predicate.

Methods for accelerating of computational procedures based on the matrix representation of finite predicates allow to practically obtain the desired solutions of the problems in a reasonable time. Additional acceleration capabilities associated with semantic constraints analysis of subject domain.

Main Part

The article presents a tools for development of NTA-algorithms. It allows to automate various procedures of logical-semantic analysis. The features of the software implementation developed environment for automation of calculations with NTA-objects are presented.

The basis of all operations with NTA-objects is the operations with components of NTA-objects. On the program level components are represented by Boolean vectors: each component is a sequence of bit, whose dimension is equal to the dimension of the corresponding domain. If a bit is set on "true" the corresponding element of the domain is present in certain component.

The user determines components of logic matrix in the form of sets as it is taken in the NTA. Through the matrix presentation operations with components could be reduced to bit-wise operations with Boolean vectors on the program level.

The article presents a system of commands of considered command processor, and also examples of using this system for solving some problems of logical analysis.

Conclusion

The matrix representation of DNF and CNF of finite predicates, logical-semantic analysis methods that use this representation, methods for accelerating computational procedures allow to organize computations with non-numeric variables similar to computations based on the methods of matrix algebra in specialized software.

The authors would like to thank the Russian Foundation for Basic Research (grants 12-07-00550, 12-07-00689, 13-07-00318), the Department for Nanotechnologies and Information Technologies of RAS (project 2.11 of the current Programme of Basic Scientific Researches), and the Chair of RAS (project 4.3 of the Program # 16) for their help in partial funding of this research.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 510.63

КОНСТРУКТИВНАЯ ПРОПОЗИЦИОНАЛЬНАЯ ЛОГИКА С НЕПАРАДОКСАЛЬНОЙ ИМПЛИКАЦИЕЙ

Сметанин Ю.М.

*ФГБОУ ВПО «Удмуртский государственный университет»,
г. Ижевск, Российская федерация*

Gms1234gms@rambler.ru

В статье рассматривается не классическая пропозициональная логика включающая традиционную силлогистику. Показана тесная связь продукционных систем и онтологии являющейся областью интерпретации правильных формул данной логики. Приведены примеры, имеющие прикладное значение.

Ключевые слова: ортогональный базис силлогистики; онтология; n -арные логические отношения; исчисление конститuent, логическое следование в семантическом смысле.

ВВЕДЕНИЕ

В статье рассматривается возможность и актуальность замены в классической логике и традиционной силлогистике многосмыслового базиса Аристотеля на односмысловой ортогональный базис изоморфный отношениям «равносильно», «влечет», «независимы» между терминами рассуждений и случайными событиями в теории вероятностей. Осуждаются теоретические результаты и приложения. В работах [Сметанин 2012], [Сметанин 2012b], [Сметанин 2013a] [Сметанин 2013b] [Сметанин 2014] выявляются недостатки математической модели, лежащей в основе классической логики, и предлагается ее улучшенный вариант логика S_{L_1} с областью интерпретации в алгебраической системе с невырожденной булевой алгеброй множеств. Описывается неклассическая интерпретация умозаключений и разработан алгоритм компьютерной проверки логического следования в семантическом смысле, основанный на новом способе решения логических уравнений [Сметанин 2010], [Сметанин 2013]. Показана связь онтологии как модели мира в которой производится интерпретация формул логики S_{L_1} . Выявляется связь онтологии и продукционных систем (в частности таблиц-решений). Приводятся содержательные примеры моделирования рассуждений

1. Моделирование рассуждений.

По необходимости люди осмысливая явления окружающего мира вынуждены, отраженные в мышлении, причинно-следственные связи суждения и, построенные на их основе, умозаключения выражать высказываниям естественного или формального языка. При этом в языке причинно-следственные связи выражаются в категориях истины и лжи посредством понятия логическое следование в семантическом смысле. При этом в отличие от формальной логики истина и ложь конкретны в рамках образов реальности в человеческом сознании. Средствами языка, логическое следование фиксируется путем сопоставления списку языковых конструкций, называемых посылками, некоторой конструкции, которая называется их следствием. Отношение логического следования должно не парадоксально выражаться в естественном языке или формальном языке. Если в языке существует такой конструкт, то, видимо, его и стоит называть **импликацией**. Ясно, что это не есть функция материальной импликации. Потому, что, образ отношения причинно-следственной связи принимаемый нашим мышлением, в языке классической логики, извращен в булеву функцию. В данной работе предлагается выразить импликацию в форме алгоритма вычисления логического следствия в семантическом смысле в специально созданной пропозициональной логике.

Рассмотрим универсум U и понятия. Понятия представлены своими именами (1) и объемами на

$$\sum_n = \{X_1, X_2, \dots, X_n\} \quad (1)$$

фоне универсума. Все термины из семейства (1) являются не пустыми и не универсальными.

Определение 1. Систему упорядоченных терминов и универсума (2) будем называть онтологией (моделью) мира.

$$I = \langle U, X_1, X_2, \dots, X_n \rangle \quad (2)$$

Эффективное использования этой модели возможно, если все её множества финитны.

Во многих случаях для логических вычислений достаточно знать лишь отношения между парами непустых терминов, число которых равно семи (смотри рис. 1).

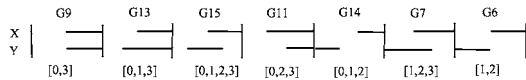


Рисунок 1 – Расширенные жергонновы отношения

Термины в онтологии (1) можно представлять конечными множествами и семь отношений выражать тремя логическими (совпадение, строгое включение и независимость). Они составляют ортогональный базис силлогистики (**OB**), предложенный Сметаниным Ю.М., который состоит из трех односмысловых суждений. $OB = \{A(X, Y), Eq(X, Y), IO(X, Y)\}$. Основные понятия и определения, позволяющие представить онтологию (2) конечными множествами приведены в работе [Сметанин 2014] настоящего сборника. В работах [Сметанин 2013], [Сметанин 2010] показана возможность моделирования рассуждений посредством онтологии (2).

2. Пропозициональная логика на основе ортогонального базиса.

На основе **OB** построена логика S_{L_1} с аксиомами алгебры множеств. Областью интерпретации формул этой логики является онтология (2) [Сметанин 2013]. Все правильно построенные формулы в нашей логике можно привести к виду который называется сингулярным дизъюнктом. **Определение 2.** Базовыми конъюнктами в логике S_{L_1} являются следующие высказывания (3)

$$\begin{aligned} &1) Eq(X', U); 2) Eq(X, U); 3) A(X, U); \\ &4) A(X', U); 5) A(X, Y), Eq(X, Y), IO(X, Y) \end{aligned} \quad (3)$$

Конъюнктами и ППФ также называются конъюнкции базовых конъюнктов и конъюнкции конъюнктов. Имеют место равносильности (4)

$$\begin{aligned} Eq(X', U) &\equiv A(X', U); Eq(X, U) \equiv A(X, U); \\ A(X', U)' &\equiv Eq(X, U); A(X', U)' \equiv Eq(X', U) \end{aligned} \quad (4)$$

Будем считать, что:

(1) конъюнктивные высказывания из определения 2 являются ППФ

(2) если Q есть ППФ, то $(Q)'$ (читается как «неверно, что Q » тоже ППФ;

(3) если Q_1 и Q_2 - ППФ, то $(Q_1 \cdot Q_2)$ и $(Q_1 + Q_2)$ тоже ППФ. Кроме того правильно построенными формулами являются сингулярные дизъюнкты.

Определение 3. Пусть A_1, A_2, \dots, A_n конъюнкты логики S_{L_1} . Сингулярным дизъюнктом, составленным из них называется совершенная дизъюнктивная нормальная форма (4)

$$\begin{aligned} \oplus_1(A_1, A_2, \dots, A_n) &\stackrel{df}{=} \\ A_1 \cdot A_2' \cdot \dots \cdot A_n' + A_1' \cdot A_2 \cdot \dots \cdot A_n' + \dots + A_1' \cdot A_2' \cdot \dots \cdot A_n \end{aligned} \quad (4)$$

Она выражает суждение о том, что одно и только одно из суждений A_1, A_2, \dots, A_n имеет значение истина.

$$\begin{aligned} &\forall i, j Eq((X_i + X_j), (X_j + X_i)), \\ &\forall i, j, k Eq((X_i + X_j) + X_k, (X_i + (X_j + X_k))), \\ &\forall i, j Eq((X_i' + X_j')' + (X_i' + X_j'), X_i), \\ &\forall i A(X_i, U) \cdot A(X_i', U) \end{aligned} \quad (5)$$

Выражение (5) есть аксиомы рассматриваемой логики. По сути это аксиомы алгебры множеств. Последняя утверждает не пустоту и не универсальность модельных множеств сопряженной с логикой онтологии. Логика является конструктивной, в том смысле, что сопоставляемая ей онтология (модель мира) меняется с изменением множества конструируемых ППФ. Онтология для пустого множества формул является канонической (см. определение 1) [Сметанин 2014]. В процессе применения правил построения ППФ из терминов соответствующих модельным множествам, множество формул пополняется и изменяется исходная онтология, причём так, что все сконструированные на данном этапе формулы интерпретируются в ней как истинные. Возможен случай, когда множество сконструированных формул одновременно не может быть истинным. Это означает, что соответствующая ему онтология имеет в качестве универсума пустое множество.

Рассмотрим смысловое содержание формул и их истинностную интерпретацию. Конъюнкт логики имеет много смысловое содержание, если его бинарный инвариант содержит несколько n -арных логических отношений (смотри статью [Сметанин 2014] в настоящем сборнике). Если его **BIN** (**nBIN**) состоит из одного логического отношений, то тогда конъюнкт одно смысловой. Можно условно считать, что конъюнкт имеет одно смысловое содержание, если за него принять n -арное логическое отношение соответствующее **максимальной онтологии** [Сметанин 2014] В этом случае смысл максимальной онтологии единственный. Сама максимальная онтология получается по алгоритму последовательного введения всех базовых конъюнктов, из которых состоит рассматриваемый

конъюнкт, в каноническую онтологию [Сметанин 2010]. Истинностное значение конъюнкта будет «истина», в данной онтологии, если в результате проверки (интерпретации) соответствующее смысловому содержанию конъюнкта многоместное отношение выполняется в данной онтологии, иначе истинностное значение конъюнкта есть «ложь». Смысловым содержанием сингулярного дизъюнкта является смысл в точности одного из смыслов выражаемых составляющими его конъюнктами, а именно того, которому в результате интерпретации приписывается истинностное значение «истина». Если все составляющие его конъюнкты имеют истинностное значение «ложь», то ему приписывается истинностное значение «ложь» и смысловое содержание, выраженное в языке теории множеств, равное объединению *BSN* отрицаний составляющих его конъюнктов. Представление суждения в форме сингулярного дизъюнкта позволяет рассматривать многосмысловые умозаключения типа силлогизмов и мультисиллогизмов и проверять, является ли их заключение логическим следствием посылок. Для этого разработана компьютерная программа. Рассмотрим процесс проверки в данной логике модуса (*Bramantip*) *AAI* четвертой фигуры. Требуется проверить логическое следование в семантическом смысле (*AMS*) & (*APM*) ^с *ISP*. В ортогональном базисе модус имеет вид (5).

$$\begin{aligned} & [Eq(P, M) + A(P, M)] \cdot [Eq(M, S) + A(M, S)]^c \\ & [A(S', P) + Eq(S, P) + A(S, P) + A(S', P') + IO(S, P)] \end{aligned} \quad (5)$$

Запись модуса в форме сингулярного дизъюнкта имеет вид (6)

$$\begin{aligned}
& \bigoplus_1 \left(\underbrace{Eq(P, M) \cdot Eq(M, S)}_{1p}; \underbrace{A(P, M) \cdot Eq(M, S)}_{2p}; \right. \\
& \left. \underbrace{Eq(P, M') \cdot A(M, S')}_{{3p}}; \underbrace{A(P, M) \cdot A(M, S')}_{{4p}} \right), \quad (6) \\
& \bigoplus_1 \left(\underbrace{A(S', P)}_1; \underbrace{Eq(S, P)}_2; \underbrace{A(S, P)}_3; \underbrace{A(S', P')}_4; \underbrace{IO(S, P)}_5 \right)
\end{aligned}$$

Размер модуса неправильный. Посылка имеет один из четырех смыслов из них не может следовать каждый из пяти смыслов заключения. Рисунок 2 показывает, как его можно уточнить.

$$\begin{array}{lcl}
 U = & 0 & 7 \\
 X1 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & M \\
 X2 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & P \\
 X3 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & S \\
 \text{Eq}(M, P) & \text{Eq}(M, S) & \text{Eq}(P, S)
 \end{array}
 \quad 2.$$

$$\begin{array}{lcl}
 U = & 0 & 1 & 7 \\
 X1 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & M \\
 X2 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & P \\
 X3 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & S \\
 \text{Eq}(M, P) & \text{A}(M, S) & \text{A}(P, S)
 \end{array}
 \quad 4.$$

$$\begin{array}{lcl}
 U = & 0 & 5 & 7 \\
 X1 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & M \\
 X2 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & P \\
 X3 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & S \\
 \text{A}(M', P') & \text{Eq}(M, S) & \text{A}(P, S)
 \end{array}
 \quad 4.$$

$$\begin{array}{lcl}
 U = & 0 & 1 & 5 & 7 \\
 X1 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & M \\
 X2 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & P \\
 X3 = & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} & S \\
 \text{A}(M', P') & \text{A}(M, S) & \text{A}(P, S)
 \end{array}
 \quad 4.$$

Рисунок 2 Соответствие сочетаний смыслов посылок и смыслов (таблицы - решений) для следствий модуса *AAI* четвёртой фигуры (*Bramantip*)

Уточнение имеет вид (7)

$$(APM) \cdot (AMS) \leq (AS'P') \quad (7)$$

В этом качестве модус может служить заменой модусу *AAI* четвёртой фигуры. Таким образом при проверке логического следования в семантическом смысле мы стоим таблицу - решений, в которой каждому из смыслов сингулярного дизъюнкта посылок ставится в соответствие один из смыслов заключения. После выражения каждого модуса в ортогональном базисе и построения его в виде таблицы-решений (рисунка который её заменяет), оказалось, что 13 оказались лишними в списке правильных. Они дублируют (совпадают по смыслу) с семью из одиннадцати оставшихся. Таким образом, набор модусов сокращен до 11 и уточнен в форме таблиц – решений.

Признаком по которому требуется уточнение является признак неправильности размера модуса.

Имеет место теорема 1 Любая ППФ логики S_{L_1} может быть приведена к виду сингулярного дизъюнкта.

3. Алгоритм вычисления логического следования в семантическом смысле.

Дано Сингулярный дизъюнкт посылки **G** и
сингулярный дизъюнкт заключения **B**.

Требуется установить наличие либо отсутствие логического следования (8) в данной онтологии *I*.

$$G \stackrel{\epsilon}{\rightarrow}_{S_{I_1}} B \quad (8)$$

Алгоритм путем регулярной процедуры сопоставления каждому смыслу посылки каждого смысла заключения проверяет, для каждой выбранной пары, имеет ли место логическое следование из конъюнкта посылки в конъюнкт следствия. Если следование есть, то формируется очередное правило в таблице-решений, которая сопоставляет, данному смыслу посылки ***G*** данный смысл заключения ***B***.

0. Номер очередного правила $t:=I$; установить флажок S_L - (индикатор наличия следования) в значение «истина»;

1. Для каждого i -го конъюкта посылки
выполнить (1):

для каждого j -го конъюнкта заключения

выполнить (2):

1.1. проверить его истинностное значение в онтологии, с введенным i -м конъюнктом посылки;\

1.2. **если** истинностное значение будет «истина», **то** сформировать очередное правило (9) с номером t

$$R_t: K_i^p \rightarrow K_i^z \quad (9)$$

конец (1) цикла по j ;

если в цикле по j не было сформировано ни одного правила, то установить флажок S_L в значение «ложь»;

конец (2) цикла по i ;

2. Сформировать из правил R_i в порядке их получения совершенную таблицу-решений;

3. Объединить правила, в продукции которых находятся одинаковые правые части;

4. Если S_L = «истина», то выдать сообщение о том, что логическое следствие имеет место, иначе выдать сообщение о том, что логическое следование не имеет места. **Конец алгоритма.**

3.1. Пример применения алгоритма.

Рассмотрим решение задачи Порецкого о птицах. В качестве примера рассмотрим задачу Порецкого о птицах в зоосаде. Для этого построим **BSN** четырехарного отношения между терминами которое определяется конъюнкцией посылок. Пусть X – множество птиц качества X , Y – множество птиц качества Y , S – множество певчих птиц, G – множество крупных птиц, то есть имеем четыре термина X , Y , G , S . При этом посылки сформулированы Порецким следующим образом:

(1) птицы певчие (S) суть или крупные или обладающие качеством Y ;

(2) птицы не имеющие качества Y , или не крупные, или не имеют качества X ;

(3) птицы певчие, вместе с крупными (G), объединяют всех птиц с качеством X ;

(4) каждая не крупная птица есть или певчая, или обладающая качеством X ;

(5) между птицами с качеством X совсем нет таких птиц с качеством Y , которые, не будучи певчими, были бы крупными.

Требуется:

(а) не зная качеств X и Y определить, были ли птицы качества X певчими или нет, крупные они или нет.

(б) узнать, то же в отношении птиц качества Y

(с) найти, были ли между птицами качества X птицы качества Y , и обратно.

Посылки задачи можно выразить в ортогональном базисе следующим образом:

$$\begin{aligned} P1. & A(S, Y + G) \oplus Eq(S, Y + G); \\ P2. & A(Y', X' + G') \oplus Eq(Y', X' + G'); \\ P3. & A(X, S + G) \oplus Eq(X, S + G); \\ P4. & A(G', S + X) \oplus Eq(G', S + X); \\ P5. & A(X, (Y \cdot S' \cdot G)') \oplus Eq(X, (Y \cdot S' \cdot G)') \end{aligned} \quad (10)$$

Для ответа на поставленные вопросы нужно рассмотреть 32 варианта возможных односмысловых сочетаний этих посылок и построить их **BSN**; после этого проверить, выполняются ли отношения, о которых говорится в требованиях задачи. Варианты имеют различные **BIN**, что сразу указывает на их смысловое различие.

Кроме того из 32 вариантов сочетаний смыслов посылок 18 оказались противоречивыми и 12 содержат пустые или универсальные термины. Поэтому заключения, которые нужно проверить, выраженные в требовании задачи и формализованные в (12) не имеют места. Поэтому логического следования нет. Удовлетворяющие требованиям традиционной силлогистики 2 варианта сочетания смыслов посылок в задаче о птицах изображены на рис. 3. Для этих двух вариантов имеет место логическое следование для вопросов а) и с) и не имеет места для вопроса б).

$$\begin{aligned} (a) & [A(X, S) + Eq(X, S)] \cdot [A(X, G) + Eq(X, G)] \\ (b) & [A(Y, S) + Eq(Y, S)] \cdot [A(Y, G) + Eq(Y, G)] \\ (c) & IO(X, Y) \end{aligned} \quad (11)$$

$$\begin{aligned} a) & \oplus_1 [A(X, S) \cdot A(X, G); A(X, S) \cdot Eq(X, G); \\ & Eq(X, S) \cdot A(X, G); Eq(X, S) \cdot Eq(X, G)] \\ b) & \oplus_1 [A(Y, S) \cdot A(Y, G); A(Y, S) \cdot Eq(Y, G); \\ & Eq(Y, S) \cdot A(Y, G); Eq(Y, S) \cdot Eq(Y, G)] \\ c) & \oplus_1 [A(X', Y); IO(X, Y); A(X, Y); \\ & A(Y, X); Eq(X, Y)] \end{aligned} \quad (12)$$

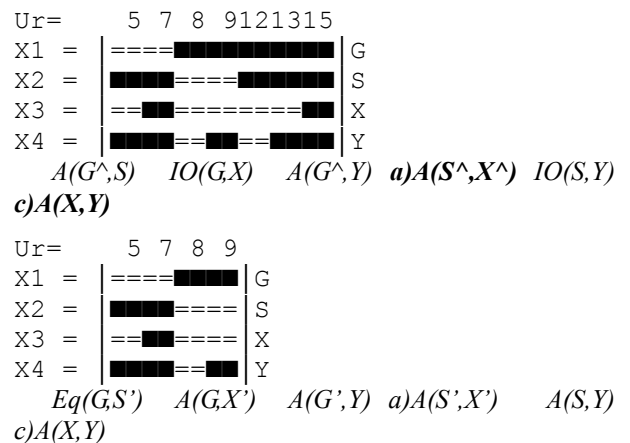


Рисунок 3 Два допустимых варианта для задачи Порецкого.

Приведенные на рисунках варианты имеют максимальные онтологии, различные **BIN**, в каждом из которых один элемент. Для вариантов, имеющих смысл в традиционной силлогистике, ответы на вопросы примера непосредственно считываются с диаграмм рисунков. При этом ответы совпадают.

4. Интерпретация онтологии как таблицы -решений

Проведём анализ инструкции по выбору технологии.

Имеется заготовки из трёх видов материалов: латунь хрупкая (LH), латунь вязкая (LV) и алюминий (AL) и два вида резцов – быстрорежущий (BR) и нормальный (NR) и три технологии $T10$, $T16$, $T25$. Пусть $O = \{BR, NR\}$, $M = \{AL, MLH, MLV\}$, $T = \{T10, T16, T25\}$.

Инструкция состоит из четырех предписаний.

1. Можно использовать технологию T25, если при этом материал заготовки не определяется как LH.

2. Для состояния LH при использовании оборудования BR необходимо использовать технологию T16.

3. Технологию T16 также необходимо применять при использовании оборудования типа NR и состояниях объекта AL либо LH.

4. В случае состояния LV необходимо использовать только оборудование типа NR и выполнять действия по технологии T10.

Очевидно, что здесь мы имеем систему продукций (12)

$$\begin{aligned} 1. (O = BR) \cdot (M = LH') &\Rightarrow T := T25 \equiv A(BR \cdot LH', T25) \\ 2. (O = BR) \cdot (M = LH) &\Rightarrow (T := T16) \equiv A(BR \cdot LH, T16) \\ 3. (O = RN) \cdot (M = AL) &\Rightarrow (T := T16) \equiv A(AL \cdot RN, T16) \\ 4. (O = RN) \cdot (M = LH) &\Rightarrow (T := T16) \equiv A(LH \cdot RN, T16) \\ 5. (O = RN) \cdot (M = LV) &\Rightarrow (T := T10) \equiv A(RN \cdot LV, T10) \end{aligned} \quad (13)$$

Введём обозначения. Универсум состоит из объектов, обрабатываемых по технологиям T10, T16, T25. Обозначим множества этих объектов соответственно как X_6, X_7, X_8 . В универсуме выделяются объекты, для обработки которых используется оборудование BR и NR. Обозначим множества этих объектов соответственно как X_1, X_2 . В универсуме выделяются объекты из материала AL, LH, LV. Обозначим множества этих объектов соответственно как X_3, X_4, X_5 . Далее считаем, что имя множества X_i означает также свойство элементов этого множества. Поэтому мы имеем универсум и семейство образующих $X_1 - X_8$.

Если считать, что образующие модель мира термины независимы в совокупности, то онтология соответствующая данному фрагменту техносферы является канонической. И, следовательно, состоит из 256 непустых конституент. Однако это не так. Зависимости между терминами выражены посредством конъюнкции бинарных логических отношений (13). Введение в данную онтологию фактов и зависимостей (13) приводит к такому её сужению, в котором каждое бинарное логическое отношение (13) является истинным (выполняется). Эта онтология состоит из 147 непустых конституент и не отражает все особенности моделируемого фрагмента мира. В ней не учтены следующие соотношения инструкции, которые в ней присутствуют по умолчанию. Вот они.

Каждый объект обладает только одним из свойств x_3, x_4, x_5 , то есть имеет место отношение

$$Eq(X_3 \cdot X_4' \cdot X_5' + X_3' \cdot X_4 \cdot X_5 + X_3' \cdot X_4' \cdot X_5, U) \quad (14)$$

Все объекты универсума разбиваются на два класса, каждый из которых обрабатывается своим оборудованием, следовательно имеет место (15)

$$Eq(X_1 \cdot X_2' + X_1' \cdot X_2, U) \quad (15)$$

Каждый объект обрабатывается только по одной технологии следовательно имеет место (16).

$$Eq(X_6 \cdot X_7' \cdot X_8' + X_6' \cdot X_7 \cdot X_8 + X_6' \cdot X_7' \cdot X_8, U) \quad (16)$$

Онтология соответствующая этим ограничениям изображена на рис. 4А. В ней конституента с номером 137 выделена серым цветом. Эта конституента не удовлетворяет еще одному ограничению четвертой инструкции, которое гласит «В случае состояния LV необходимо использовать только оборудование типа NR» Введя его в виде

$$Eq(X_1 \cdot X_5, U') \quad (17)$$

мы получим онтологию изображенную на рис. 4Б).

4А)

X1	X2	X3	X4	X5	X6	X7	X8	U
:	:	:	:	:	:	:	:	:76
:	:	:	:	:	:	:	:	:82
:	:	:	:	:	:	:	:	:98
:	:	:	:	:	:	:	:	:137
:	:	:	:	:	:	:	:	:146
:	:	:	:	:	:	:	:	:161
BR	NR	AL	LH	LV	T10	T16	T25	

4Б)

X1	X2	X3	X4	X5	X6	X7	X8	U
:	:	:	:	:	:	:	:	:76
:	:	:	:	:	:	:	:	:82
:	:	:	:	:	:	:	:	:98
:	:	:	:	:	:	:	:	:146
:	:	:	:	:	:	:	:	:161
BR	NR	AL	LH	LV	T10	T16	T25	

Рисунок 4 Модель технологической инструкции

Эта онтология по сути является совершенной таблицей – решений. Совершенная таблица обладает важным свойством её правила можно упорядочить произвольным образом, другими словами они перестановочны. Правила таблицы по соответствию продукциям (13). Онтология 4Б) позволяет переформулировать инструкцию в виде трех правил. Правило 1 нельзя переставлять с правилом 2 так как оно является исключением из него. Исключение всегда выполняется вперед общего правила.

1. Если материал алюминий можно использовать резец BR и технологию T25

2. Если материал алюминий либо латунь хрупкая можно использовать резцы BR либо NR и технологию T16.

3. Если материал латунь хрупкая и резец NR, то применяй технологию T10. Иначе ошибка в исходных данных для принятия решения и нужно дорабатывать исходную инструкцию.

Отметим, что, хотя трудоёмкость построения продукций (12) и онтологии рис. 4Б) различны, тем не менее формализация в виде онтологии предпочтительнее с точки зрения автоматического синтеза программы и с точки зрения автоматизации процедур модификации. Например, требуется

вести в технологию и выявить влияние нового технологического фактора X_9 . Первоначально появление нового технологического фактора не отменяет ограничения (13), (14), (15), (16), (17), они в новой онтологии с модельными множествами X_1 - X_9 показаны на рис.5

В данной онтологии можно планировать эксперименты с технологическим фактором X_9 и оставлять в ней только те конститuentы, которые соответствуют реальным причинно - следственным связям.

Существует тесная связь между таблицами-решений и автоматными спецификациями (моделями) предметных областей деятельности смотри [Сметанин и др. 2009], в частности там

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	\cup
:	:	:	:	:	:	:	:	:	:152
:	:	:	:	:	:	:	:	:	:153
:	:	:	:	:	:	:	:	:	:164
:	:	:	:	:	:	:	:	:	:165
:	:	:	:	:	:	:	:	:	:196
:	:	:	:	:	:	:	:	:	:197
:	:	:	:	:	:	:	:	:	:274
:	:	:	:	:	:	:	:	:	:275
:	:	:	:	:	:	:	:	:	:292
:	:	:	:	:	:	:	:	:	:293
:	:	:	:	:	:	:	:	:	:322
:	:	:	:	:	:	:	:	:	:323

Рисунок 5 Модель технологической инструкции с новым фактором X_9

показано, что таблица-решений ассоциируется с состоянием конечного автомата. Рассмотрение этого вопроса выходит за рамки настоящей статьи.

Заключение

В статье описывается пропозициональная логика с непарадоксальной импликацией в виде алгоритма проверки логического следования в семантическом смысле. Эта логика включает традиционную силлогистику. Показана тесная связь онтологии, в которой проводится интерпретация формул логики, с продукционными системами и их формализацией. Можно сказать, что предлагаемый подход является развитием основополагающих идей П.С. Порецкого в части решения логических уравнений и уточнением понятий возможных миров, модельных множеств и дистрибутивных нормальных форм, предложенным профессором Я. Хинтикка. Приводятся содержательные примеры.

Библиографический список

[Порецкий П.С., 1884] Порецкий П. С. О способах решения логических равенств и об одном обратном способе математической логики.// Собрание протоколов заседаний секции физико-математических наук общества естествоиспытателей при Казанском университете, т. 2, Каз.,1884

[Сметанин, 2014] Сметанин Ю. М. Смысловое содержание простых субъектно-предикатных суждений. В настоящем сборнике.

[Сметанин, 2013] Сметанин Ю.М. Логика высказываний на основе алгебраической системы, включающая традиционную силлогистику. // Вестник Удмуртского университета. Математика. Механика. Компьютерные науки. 2013. Вып. 2. С.127-146.

[Сметанин, 2010] Сметанин Ю.М. Алгоритм решения полисиллогизмов в ортогональном базисе посредством исчисления конститuentных множеств // Вестник Удмуртского университета. Математика. Механика. Компьютерные науки. 2010. Вып. 4. С. 172--185.

[Сметанин Ю.М. и др. 2009] Сметанин Ю.М., Сметанина Е.Ю., Бусоргин А.В. Таблицы решений и автоматное моделирование бизнес-процессов // Вестн. Удм. ун-та. Сер. Экономика и право. 2009. Вып. 2. С.126-142

CONSTRUCTIVE PROPOSITIONAL LOGIC WITH IMPLICATION UNPARADOXICAL

Smetanin Yu. M.

Udmutr State University

gms1234gms@rambler.ru

Introduction

The article explains the reasons to replace the multi-semantic basis of Aristotle in classical logic and traditional syllogistic. In return it is suggested looking at mono-semantic basis, isomorphic to relationship "equivalent", "entailing", "independent", which happen between random events in probability theory.

Key words: ontology modeling, syllogistics, syllogistic orthogonal basis, Boolean algebra, constituents' calculus, isomorphism and homomorphism of algebraic systems, logical implication in semantics sense, logical equations.

Conclusion

The author identifies the drawbacks of mathematical model in classical logics. More advanced version of propositional logic S11, based on non-degenerative Boolean algebra, is proposed. The article considers non-classical interpretation of judgments in the orthogonal basis of syllogistics; it also describes the opportunities of effective computer validation of logical implication in semantics. It may be said that the suggested approach is more precise definition of hypothetical worlds and model sets and distribution of normal forms proposed by Professor Ya.Hintikka. The samples of solutions are presented. The results of Aristotle model validity are discussed.



УДК 510.63

СМЫСЛОВОЕ СОДЕРЖАНИЕ ПОСЛЕДОВАТЕЛЬНОСТИ ПРОСТЫХ СУБЪЕКТНО-ПРЕДИКАТНЫХ СУЖДЕНИЙ

Сметанин Ю.М.

ФГБОУ ВПО «Удмуртский государственный университет»,
г. Ижевск, Российская федерация

Gms1234gms@rambler.ru

В работе рассматривается интерпретация суждений на основе ортогонального базиса силлогистики в модели мира (онтологии) изоморфной алгебраической системе, с основой определяемой конститuentами из объемов терминов онтологии. Показано, что предлагаемая система для интерпретации суждений позволяет осуществлять проверку логического следования в семантическом смысле

Ключевые слова: ортогональный базис силлогистики; онтология; n - арные логические отношения; исчисление конститuent.

Введение

В аксиоматике и в структурах формул математической логики достаточно неестественно отражается структура многих предложений естественного языка. В большей степени этой структуре соответствует структура суждений Аристотелевой силлогистики. Многосмысловость базиса Аристотеля значительно огрубляет модели получаемые рассуждений [Сметанин 2013b]. Абстрактная истина и ложь, используемая в «деревянном» логическом выводе, вкупе с неудовлетворительной (парадоксальной) моделью причинно-следственных отношений в виде материальной и других импликаций приводит к неадекватности модели. Естественный язык предпочитает сложные n - арные отношения выражать простыми суждениями о бинарных отношениях объемов используемых терминов., в отличие от матлогики, в которой они обманчиво просто задаются их индикаторам – предикатами. Это приводит к тому, что «многие методы рассуждений, которые используются в естественном языке, часто весьма трудно однозначно отобразить на языке математической логики. В некоторых случаях такое отображение приводит к существенному искажению сути естественного рассуждения.» [Кулик 2010]. То, что математическая логика во времена естествоиспытателей вместе со схоластической водой выплеснула ребенка (простые суждения и здравый смысл), представляется всё более вероятным. В связи с этим актуальной представляется попытка построить логическую модель рассуждений и её интерпретацию без

вышеуказанных недостатков. Сохранив при этом такие привлекательные черты силлогистики как **наглядность**, **логическое следование** в семантическом смысле, не допускающее кривых толкований, **непарадоксальность** и **конкретность истины** и **лжи** в терминах отношений объемов используемых терминов.

1. Смысловое содержание n -арного логического отношения.

Область интерпретации в силлогистике выражена пятнадцатью модельными схемами, задающими отношения между объемами двух терминов [Бочаров и др. 2010]. В работе [Сметанин] они разделены на две группы – расширенные жергонновы отношения (не допускающие пустоту и универсальность терминов) - см. рис. 1 и

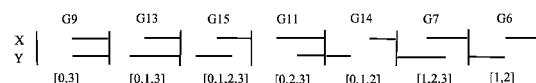


Рисунок 1 – Расширенные жергонновы отношения

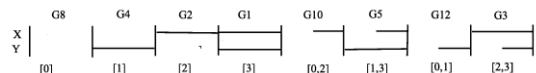


Рисунок 2 – Вырожденные жергонновы отношения

вырожденные жергонновы отношения см. рис.2 (допускающие пустоту или полноту терминов). В работах [Сметанин 2010], [Сметанин 2012a], [Сметанин 2013b] введено понятие ортогонального базиса силлогистики. Ортогональный базис сводит все 15 бинарных отношений между множествами к

трем логическим отношениям (равенства – $Eq(X, Y)$, строгого включения – $A(X, Y)$ и независимости $IO(X, Y)$). Областью истинности суждений ортогонального базиса являясь следующие модельные отношения смотри (1):

$$\begin{aligned} A(X, Y) &\equiv G_4 \oplus G_5 \oplus G_{12} \oplus G_{13} \\ Eq(X, Y) &\equiv G_1 \oplus G_8 \oplus G_9 \\ IO(X, Y) &= G_{15} \end{aligned} \quad (1)$$

При исследовании вопроса о том, какие множества можно построить из заданных n множеств посредством операций (объединения, пересечения и дополнения до универсума) было введено понятие конституенты.

$$\begin{aligned} X_1^{\sigma_1} \cdot X_2^{\sigma_2} \cdot \dots \cdot X_n^{\sigma_n} &\leftrightarrow \langle \sigma_1 \sigma_2 \dots \sigma_n \rangle, \\ \sigma_i &\in \{0, 1\} \\ X_i^{\sigma_i} &= \begin{cases} X_i, & \sigma_i = 1, \\ X_i', & \sigma_i = 0. \end{cases} \end{aligned} \quad (2)$$

Если порождающие множества упорядочить по номерам, то с каждой конституентой однозначно можно соотнести десятичное число (индикатор), являющееся эквивалентом двоичного представления конституенты. Например, конституенте (3) соответствует индикатор равный 5.

$$X_1^1 \cdot X_2^0 \cdot X_3^1 = X_1 \cdot X_2' \cdot X_3 \quad (3)$$

Далее будем рассматривать частично упорядоченную отношением строгого порядка систему подмножеств множества U , называемого далее универсумом. При этом отдельные элементы U в случае надобности будем трактовать как его одноэлементные подмножества. Пусть (4) есть система из всех непустых подмножеств универсума, которые могут быть построены из конечной системы образующих (5)

$$B(\Sigma_i) \quad (4)$$

являющихся собственными, не пустыми

$$\Sigma_n = \{X_1, X_2, \dots, X_n\} \quad (5)$$

подмножествами U , посредством операций объединения, пересечения и дополнения до U . Множество (4) включает также универсум пустое множество. Всего можно составить не более чем 2^n непустых конституент, являющихся элементарными «кирпичиками», из которых можно конструировать указанные выше множества системы (4). Рассмотрим алгебраическую систему (6)

$$\langle B(\Sigma_i), W_F, W_R \rangle, W_F = \langle +, \cdot, ' \rangle, W_R = \langle =, \subset \rangle \quad (6)$$

в которой множество функций есть объединение, пересечение множеств и дополнение множества до универсума, а множество отношений есть равенство и строгое включение множеств. Зафиксируем порядок образующих множеств алгебраической системы (6), например,

$$\pi = \langle i_1, i_2, \dots, i_n \rangle X_{i_1}, X_{i_2}, \dots, X_{i_n} \quad (7)$$

Такую алгебраическую систему будем называть системой с заданным порядком образующих или просто заданной и будем ее обозначать как

$$\langle B(\Sigma_n, \pi), W_F, W_R \rangle \quad (8)$$

Определение 1. Единицей – M заданной алгебраической системы (AC) (8) или её характеристическим множеством будем называть семейство непустых конституент данной системы. Если все конституенты заданной AC (8) являются непустыми множествами, то такую AC будем называть **канонической**.

Определение 2. Базовой системой номеров $BSN(U)$ универсума будем называть множество номеров (индикаторов) конституент составляющих характеристическое множество (единицу) заданной алгебраической системы.

Определение 3. Базовой системой номеров $BSN(X)$ любого множества X из семейства (8) будем называть множество номеров (индикаторов) непустых конституент из характеристического множества, объединение которых совпадает с данным множеством.

Характеристическое множество M (единица) и его дополнение N в заданной (AC) полностью определяются своими BSN . Поэтому для AC с упорядоченной системой образующих $BSN(M)$, $BSN(N)$ и их характеристические множества можно отождествлять. Далее будем считать, что в заданной системе (8) все множества и универсум заданы в виде BSN . По аналогии с бинарными логическими отношениями $A(X, Y)$; $Eq(X, Y)$; $IO(X, Y)$. Введем понятие n -арного логического отношения. Так же как бинарные логические отношения задаются единицей, BSN , так и произвольное n -арное логическое отношение задается с помощью алгебраической системы, путем указания перечня её непустых конституент, или равнозначно, единицей. Эту единицу также будем называть онтологией. Алгебраическую систему (8) удобно представить кортежем множеств (9)

$$I = \langle U, X_1, X_2, \dots, X_n \rangle \quad (9)$$

Назовём логическое n -арное отношение, между образующими алгебраической системы отношением независимости образующих множеств (9) или будем говорить, что множества (и соответствующие им термины либо события) **независимы в совокупности**, тогда и только тогда когда все конституенты составленные из множеств X_i являются не пустыми. Соответствующая этому отношению заданная алгебраическая система названа в **определении 1** канонической. Основным препятствием для эффективного использования модели мира (9) является трудность задания модельных множеств и универсума. В компьютерных приложениях, например, очень важно иметь возможность оперировать конечными множествами. Мы используем конечное семейство

множеств из (4) и (8), каждое из которых представлено конечным множеством индикаторов. Нагляднее всего графически можно проиллюстрировать теоретико – множественную интерпретацию (онтологию) в виде линейной диаграммы смотри рис. 1, 2, 3. Характеристическое множество заданной АС (онтологии), выражаемая **BNS**, можно представить в виде формулы --- объединения непустых конституент. Нормальной формой Кантора (**НФК**) называется представление множества в виде объединения некоторого количества пересечений. Если в качестве пересечений множеств используются конституенты, то такую **НФК** будем называть совершенной (**СНФК**). Кроме того, универсум можно выразить любой другой правильно построенной из образующих формулой алгебры множеств, равносильной его **СНФК**. Таким образом, каждой онтологии соответствует её **СНФК** и множество ППФ алгебры множеств тождественных этой **СНФК**. И обратно классу равносильных ППФ, включающему **СНФК** однозначно соответствует единственная онтология. Нас будет интересовать возможность и однозначность задания *n*-арного логического отношения посредством последовательности отношений включения, равенства и независимости между образующими терминами. Этот интерес обусловлен особенностями выражения результатов мышления на естественном языке и возможностью разработки алгоритмов вычисления логического следования в семантическом смысле. Из полученных теоретических результатов [Сметанин 2010], [Сметанин 2012a], [Сметанин 2012b], [Сметанин 2013a], [Сметанин 2013b] вытекает, что введение отношений $A(X,Y)$ и $Eq(X,Y)$ в данную онтологию приводит к сужению её универсума из него убираются номера *тех и только тех* конституент которые должны быть пустыми при выполнении вводимого отношения. Для построения алгебраической системы на базе заданной с введением в неё отношения независимости между X и Y необходимо, в случае отсутствия в ней такового, дополнить универсум исходной онтологии номерами *тех* конституент в которые входят пересечения X и Y , X' и Y , X и Y' , X' и Y' . Эти действия не сужают, а могут только расширить исходный универсум. Доказано, что модифицированная онтология, получаемая в результате удаления необходимого и достаточного числа конституент, которые должны быть пустыми при выполнении вводимого логического отношения, является **максимальной** из возможных по числу непустых конституент.

2. Многосмысловость бинарных инвариантов

Выше мы отметили, что естественный язык предпочитает сложные *n*- арные отношения, выражать простыми суждениями о бинарных отношениях объемов используемых терминов. Рассмотрим, какую неточность (вариативность)

смыслов) дает этот способ выражения мышления как отражения объективной реальности в психике человека. **Рассмотрим пример 1.** Пусть даны три суждения.

1. Все друзья Сидорова любят прихвастнуть.
2. Тот, кто хвастается, неуверен в себе.
3. Уверенные в себе люди - не скандалисты.

Если ввести обозначения:

X_1 - друзья Сидорова;

X_2 - все, кто хвастается;

X_3 - все уверенные в себе;

X_4 - потенциальные скандалисты;

U - универсум с семейством образующих из вышеобозначенных терминов, то в ортогональном базисе силлогистики [Сметанин 2010] суждения представляются следующим образом, смотри (2).

$$\begin{aligned} P1. A(X_1, X_2) \oplus Eq(X_1, X_2); \\ P2. A(X_2, X_3') \oplus Eq(X_2, X_3'); \\ P3. A(X_3, X_4') \oplus Eq(X_3, X_4') \end{aligned} \quad (10)$$

Список (10) имеет в традиционной силлогистике восемь смысловых содержаний. Рассмотрим первое смысловое содержание (11)

$$P1. A(X_1, X_2); P2. A(X_2, X_3'); P3. A(X_3, X_4') \quad (11)$$

Количество терминов в суждениях определяет минимальную общую арность *n*-местного логического отношения между ними. Например, отношение, задаваемое списком логических бинарных отношений (2), является четырех-арным логическим отношением, но может быть представлено 5-арным и более логическим отношением.

$$P1. A(X_1, X_2); P2. A(X_2, X_3'); \quad (12)$$

В работах [Сметанин 2010], [Сметанин 2013a] описан программно-реализованный алгоритм построения максимальной онтологии, с соответствующей данному списку бинарных логических отношений между терминами. Для четырёх-арного отношения онтология показана а на рис. 3 а и б. Для каждой онтологии (9), являющейся смыслом некоторого логического *n*-арного отношения можно указать **полный набор** из C_n^2 бинарных логических отношений между терминами.

Определение 3. Полный набор определяемого этой онтологией *n*-местного логического отношения для заданной онтологии **I** будем называть бинарным инвариантом сокращенно **BIN(I)**. Если логические бинарные отношения заданы не для всех возможных сочетаний модельных множеств, то такой набор называется неполным бинарным инвариантом **nBIN(I)**. Рисунок 3 в частях А) и Б) изображает онтологию для двух неполных бинарных инвариантов А) $nBIN = [A(X_1, X_2), A(X_2, X_3'), A(X_3, X_4')]$, и Б) $nBIN = [A(X_1, X_2), A(X_2, X_3')]$. Например, из смысла отношения Б) не следует, что $A(X_3, X_4')$, то есть «все уверенные в себе люди не скандалисты». Если бы вторую онтологию изменить

за счет использования следующего $nBIN=[A(X_1, X_2), A(X_2, X_3'), Eq(X_1'X_2'X_3'X_4, X_0')]$, то получится онтология А).

X1	X2	X3	X4	X0	X1	X2	X3	X4	X0
А) -----	Б) -----								
:	:	:	:	:0	:	:	:	:	:0
:	:	:	■	:1	:	:	:	■	:1
:	:	■	:	:2	:	:	■	:	:2
:	■	:	:	:4	:	:	■	■	:3
:	■	:	■	:5	:	■	:	:	:4
■	■	:	:	:12	:	■	:	■	:5
■	■	:	■	:13	■	■	:	:	:12
					■	■	:	■	:13

Рисунок 3 – Максимальные онтологии, соответствующие двум неполным бинарным инвариантам.

Таким образом, из последней цепочки суждений также следует, что «все уверенные в себе люди не скандалисты». Одинаковые онтологии имеют совпадающие BIN . Зададимся вопросом, а сколько ещё имеется онтологий с $nBIN=[A(X_1, X_2), A(X_2, X_3') A(X_3, X_4')]$? Для данного неполного $nBIN$ установлено выражающее его семейство из 25 тетрадных логических отношений. Это семейство разбивается на шесть подсемейств, каждое из которых имеет одинаковый полный инвариант. Ниже эти инварианты перечислены

$BIN_1=A(X_1, X_2); A(X_1, X_3'); IO(X_1, X_4); A(X_2, X_3'); IO(X_2, X_4); A(X_3, X_4');$ - 6 онтологий

$BIN_2=A(X_1, X_2); A(X_1, X_3'); A(X_1, X_4); A(X_2, X_3'); IO(X_2, X_4); A(X_3, X_4');$ - 8 онтологий

$BIN_3=A(X_1, X_2); A(X_1, X_3'); A(X_1, X_4'); A(X_2, X_3'); A(X_2, X_4'); A(X_3, X_4');$ - 7 онтологий

$BIN_4=A(X_1, X_2); A(X_1, X_3'); IO(X_1, X_4); A(X_2, X_3'); A(X_2'; X_4'); A(X_3, X_4');$ - 2 онтологии

$BIN_5=A(X_1, X_2); A(X_1, X_3'); Eq(X_1, X_4); A(X_2, X_3'); A(X_2'; X_4'); Eq(X_3, X_4');$ - 1 онтология

$BIN_6=A(X_1, X_2); A(X_1, X_3'); A(X_1, X_4); A(X_2, X_3'); Eq(X_2, X_4); A(X_3, X_4')$ - 1 онтология

3. Заключение

Таким образом, фиксированному множеству терминов, данному, полному или неполному, бинарному инварианту может быть сопоставлено не единственное смысловое содержание в виде несущих данный инвариант онтологий. Каждой онтологии может быть сопоставлена СНФК и равносильные ей ППФ алгебры множеств. Каждая ППФ выражающая характеристическое множество онтологии в конечном итоге может быть выражена в ортогональном базисе. Следовательно, возможно различным способом выражать смысл в терминах структур этих формул. Поэтому можно различными способами выражать этот смысл в русском языке. Например, имеют одинаковое смысловое содержание 3 суждения из примера 1 и следующие три сужения, так как, их характеристические множества совпадают, смотри рисунок 3 А).

1. Все друзья Сидорова любят прихвастнуть.
2. Тот, кто хвастается, неуверен в себе.
3. Множество людей, не являющихся

друзьями Сидорова и не являющихся хвастунами, притом уверенных в себе и не скандалистов пусто.

Библиографический список

- [Бочаров и др. 2010] Бочаров~В.А., Маркин~В.И. Силлогистические теории. М.: Прогресс-Традиция, 2010, 336-с.
- [Кулик 2010] Кулик Б.А., С чем идет современная логика в XXI век? // Интернет ресурс: http://philosophy.ru/library/kulilk/new_log.htm
- [Сметанин, 2010] Сметанин~ Ю.М. Алгоритм решения полисиллогизмов в ортогональном базисе посредством исчисления конституентных множеств // Вестник Удмуртского университета. Математика. Механика. Компьютерные науки. 2010. Вып. 4. С. 172--185.
- [Сметанин, 2012а] Сметанин~ Ю.М. Вероятностная логика и ортогональный базис силлогистики. OSTIS 2012, материалы второй международной конференции «Открытые семантические технологии проектирования интеллектуальных систем», Минск, 2012. С. 479--488
- [Сметанин, 2012б] Сметанин~ Ю.М. Формальная логика на основе ортогонального базиса силлогистики // Логика, язык и формальные модели: Сб. статей // Под ред. Е.Н. Лисанюк, И.Б. Микиртумова, Ю.Ю. Черноскутова СПб.: Изд-во С.-Петерб. ун-та, 2012. С. 264-270
- [Сметанин, 2013а] Сметанин~Ю.М. Логика высказываний на основе алгебраической системы, включающая традиционную силлогистику. // Вестник Удмуртского университета. Математика. Механика. Компьютерные науки. 2013. Вып. 2. С. 127-146.
- [Сметанин 2013б] Сметанин~Ю.М. Ортогональный базис силлогистики как основа пропозициональной логики. // Восьмые Смирновские чтения по логике: материалы Международной научной конференции., Москва, 19-21 июня 2013 г. [редкол.: И.А. Герасимова, Д.В. Зайцев, А.С. Карпенко, О.М. Григорьев, Н.Е. Томова; отв. ред. В.И. Маркин]---- М.: Современные тетради, 2013.---- 160с.---- ISBN 978-5-88289-414-5. С. 73-76

SEMANTIC CONTENT OF A SEQUENCE OF SUBJECTPREDICATE JUDGMENTS

Smetanin Yu. M.

Udmutr State University

gms1234gms@rambler.ru

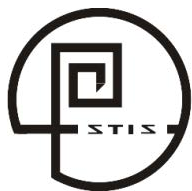
Introduction

This paper considers the interpretation of the judgment on the basis of an orthogonal basis in syllogistic model of the world (ontology) is isomorphic to an algebraic system with a basis determined by the constituents of the scope of the term ontology. It is shown that the proposed system allows for the interpretation of judgments to verify logical inference in the semantic sense

Keywords: orthogonal basis of syllogistic ontology; n - ary logical relations; calculation of constituents.

Conclusion

Thus, a fixed set of terms, this complete or incomplete binary invariant, can be compared not only the semantic content of a given bearing invariant anthologies. Each ontology can be mapped formulas sets algebra. Each formula expressing characteristic set ontology, can ultimately be expressed in an orthogonal basis. [Smetanin 2010]. Consequently, there are various ways to express the meaning in terms of the structures of these formulas.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.82

ЛОГИКО-ЛИНГВИСТИЧЕСКАЯ МОДЕЛЬ ИДЕНТИФИКАЦИИ СЕМАНТИЧЕСКИХ ОТНОШЕНИЙ СУЩНОСТЕЙ СРЕДСТВАМИ АЛГЕБРЫ КОНЕЧНЫХ ПРЕДИКАТОВ

Хайрова Н.Ф., Узлов Д.Ю., Шаронова Н.В.

*Национальный технический университет “Харьковский политехнический институт”,
г. Харьков, Украина*

nina_khajrova@yahoo.com

popucik@mail.ru

nvsharonova@mail.ru

В работе предлагается логико-лингвистическая модель извлечения слабоструктурированных фактов из естественно языковых текстов. Для идентификации факта в тексте определяются некоторые сущности, выраженные лексическими единицами, и семантические связи между ними. Семантические связи определяются семантическими функциями партиципентов предложения, которые описаны предикатами алгебры конечных предикатов. Модель применяется на семантическом этапе лингвистического процессора информационной подсистемы идентификации криминалистически значимых фактов в слабоструктурированных текстах.

Ключевые слова: слабоструктурированные факты; семантические функции; алгебра конечных предикатов; лингвистический процессор.

Введение

Знания о некоторой предметной области (ПО) представляют собой совокупность сведений об объектах/субъектах данной ПО, их существенных свойствах и связывающих отношениях, а также о фактах, семантически объединяющих партиципенты ПО и их отношения в триаду субъект – атрибут – значение (или субъект – отношение – объект).

Для извлечения фактов, представленных в хорошо структурированных текстовых документах, существуют достаточно надежные алгоритмы [Baeza-Yates, 1999].

К плохо структурированной фактографической информации относятся сведения, представленные различными нерегламентированными словесными конструкциями на естественном языке. Задача извлечения плохо структурированных фактов из произвольных текстов до сих пор не имеет сколь угодно общего решения [Ландэ, 2009].

Для идентификации некоторого знания, представленного в форме слабоструктурированного факта, необходимо извлечь из текстовой информации некоторые объекты/субъекты,

выраженные лексическими единицами, и определить семантические отношения между ними.

Так как факт выражается на естественном языке в форме законченного высказывания, то необходимо построить некий шаблон, отображающий семантические (или понятийные) связи партиципентов предложения (участников действия, выраженных существительными).

Для задания таких смысловых связей предлагается использовать семантические функции, выражаемые через отношения морфологических и семантических категорий партиципентов предложения средствами алгебры конечных предикатов (АКП).

1. Описание используемой модели.

Введем конечное множество грамматических и семантических характеристик партиципентов предложения $M = \{m_1, \dots, m_n\}$, где n количество указанных характеристик. Отношения между характеристиками можно представить в виде $m_i * m_j * \dots * m_k$, где $m_i, m_j, \dots, m_k \in M$, а знак $*$ – обозначает, что данные характеристики соответствуют существительному, выполняющему некоторую семантическую функцию.

На множестве M введем систему предикатов S так, чтобы любой предикат $P(q_m) \in S$, обращался в 1 на множестве существительных с грамматико-семантической информацией, соответствующей определенной семантической функции, и был равен 0 в противном случае. Таким образом, множество предикатов S можно сопоставить с множеством семантико-грамматических характеристик приписанных партиципantu предложения.

Для формализации семантических функций партиципantов предложения русского языка и их явного представления средствами поверхностной структуры были выделены и описаны предметными переменными морфологические (грамматический падеж) и семантические категории существительных [Бондаренко, 2007]. Рассматривались семантические категории: живой / неживой, инструмент, часть тела, объемное пространство, пункт назначения, место отправления, плоскость/точка, механизм, определенный час, период, месяц/сезон.

Область изменения введенных переменных формально задается следующим образом:

$$\begin{aligned} x^o \vee x^h &= 1, \\ z^h \vee z^p \vee z^l \vee z^b \vee z^t \vee z^n &= 1, \\ y^m \vee y^c \vee y^i \vee y^u \vee y^t \vee y^o \vee y^b \vee y^n \vee y^n &= 1, \end{aligned}$$

где x^i , z^j , y^k – предметные переменные, характеризующие: x^o – категорию живого, x^h – категорию неживого; y^m – наличие признака «механизм», y^n – наличие семантического признака «инструмент», y^u – наличие семантического признака «часть тела», y^t – наличие семантического признака «плоскость/точка», y^o – семантического признака «объемное пространство», y^b – семантического признака «определенное время», y^n – наличие семантического признака «период», y^n – наличие семантического признака «пункт назначения»; z^h , z^p , z^l , z^b , z^t , z^n — грамматическая категория падежа.

Семантическая функция существительного — партиципantа предложения описывается предикатом $P(x, y, z) = 1$, связывающим элементы семантического значения существительного x и y с его грамматическими значениями z [Хайрова, 2012]. Тогда, используя конъюнкцию предикатов, можно записать:

$$P(x, y, z) \rightarrow P(x) \bullet P(y) \bullet P(z), \quad (1)$$

где \bullet — операция конъюнкции.

Таким образом, отношения между морфологическими и семантическими признаками существительных предложения, выражающие семантические функции, можно записать логическим произведением:

$$\begin{aligned} P(x_n) * P(y_n) * P(z_n) &= \\ = \gamma_k(x_n, y_n, z_n) \bullet P(x_n) \bullet P(y_n) \bullet P(z_n), \end{aligned} \quad (2)$$

Логическое произведение предикатов $P(x_n)$, $P(y_n)$ и $P(z_n)$, описывает всевозможные отношения

морфологических и семантических характеристик, а предикат $\gamma_k(x_n, y_n, z_n)$ исключает часть связей, которые не реализуются в данной семантической функции. Предикат γ_k принимает значение 1, если морфосемантическая информация словоформы n формирует некоторую семантическую функцию лексемы, и значение 0 в противном случае.

Множество рассмотренных в системе семантических категорий значения существительного задается предикатом:

$$P(y_n) = y_n^m \vee y_n^c \vee y_n^i \vee y_n^u \vee y_n^t \vee y_n^o \vee y_n^b \vee y_n^n \vee y_n^n. \quad (3)$$

Множество значений грамматических категорий, определяющих грамматические падежи существительного, выражается предикатом:

$$P(z_n) = z_n^h \vee z_n^p \vee z_n^l \vee z_n^b \vee z_n^t \vee z_n^n. \quad (4)$$

Признак одушевленности выражается предикатом:

$$P(x_n) = x_n^o \vee x_n^h. \quad (5)$$

Семантическая функция агенса, представляющая субъект действия, обычно выступающего инициатором действия, лицо или предмет, имеющее потенцию на осуществление действий, выражается предикатом:

$$\gamma_A(x_n, y_n, z_n) = x_n^o \vee z_n^h \vee z^n x_n^h y_n^m \vee z^n x_n^o y_n^c. \quad (6)$$

Семантическая функция инструменталиса, определяющая непосредственную причину действия, играющую определенную роль в совершении процесса, выражается предикатом:

$$\gamma_I(x_n, y_n, z_n) = z_n^t x_n^h y_n^u \vee z_n^t x_n^h y_n^u. \quad (7)$$

Семантическая функция локатива, выражающая характеристики месторасположения, пространственной ориентации действия или состояния, выражается предикатом:

$$\gamma_L(x_n, y_n, z_n) = z^n x_n^h y_n^t \vee z^n x_n^h y_n^m \vee z^n x_n^h y_n^u \vee z^n x_n^h y_n^o. \quad (8)$$

Семантическая функция объектив, определяющая объект, над которым непосредственно осуществляется действие, выражается предикатом:

$$\gamma_O(x_n, y_n, z_n) = z^b x_n^h \vee z^b x_n^o. \quad (9)$$

Семантическая функция темпоралиса, выражающая временную характеристику действия, выражается предикатом:

$$\gamma_T(x_n, y_n, z_n) = z^b x_n^h y_n^b \vee z^n x_n^h y_n^n. \quad (10)$$

Семантические функции в различных естественных языках имеют разные формы формального выражения и соответственно определяются предикатами АПК различного вида. Например, в русском, украинском и белорусском языках они выражаются грамматической формой и семантическими категориями существительного,

стоящего после или перед определенным глаголом, как представлено в нашей модели.

Тогда как в английском языке семантические функции во многом определяются синтаксическими категориями, в частности отношением с предлогом, стоящим после глагола. Такие категории могут быть как уникальными для определенных глаголов, так и общими, как, например, признак направления движения, определяемый предлогом (в случае его наличия) после глаголов go, run, drive, ride, transport, ship и т.д.

2. Идентификация криминалистически значимых фактов из текстов

Рассмотрим применение данной модели для идентификации криминалистически значимых фактов в источниках полнотекстовой информации. Из огромных информационных потоков текстовой информации, обрабатываемой в процессе оперативно-служебной деятельности (сводки, объяснительные/служебные записки, отчеты, газетные и интернет публикации, словесные портреты фигурантов и т. п.), и сопровождающего ее шума следователь (или иное процессуально-должностное лицо) должен извлечь факты конкретного уголовного или иного дела. В подавляющем числе случаев к таким фактам относятся: сведения о фигурантах, сведения об объекте посягательства, сведения о механизме и способе совершения преступления.

Для извлечения из неструктурированной текстовой информации фактов о дате и месте рождения некоторых персоналий (фигурантов дела), с целью исключения или привлечения потенциального кандидата, а так же о дате, месте, субъекте и объекте некоторого противоправного действия использовались семантические функции, выражающие информацию, соответствующую требованиям:

- темпоралис – временная характеристика события, позволяющая определить дату (в нашем случае: дату рождения человека или некоторого противоправного деяния);
- локатив – функция, характеризующая местонахождение, положение или состояние объекта, определяя место (в нашем примере, место рождения человека или некоторого преступного события);
- объектив – функция, определяющая сферы и продукты деятельности человека (в данном случае: сведения об объекте посягательства);
- агенс – семантическая функция, представляющая субъект действия, обычно выступающего инициатором действия (в нашем случае: лицо/субъект противоправного действия).

Были выбраны наиболее распространенные глаголы, соответствующие фактам преступного действия и идентифицирующие личность, а также определены семантические функции, являющиеся

центральной частью триплета субъект – атрибут – значение и предикаты (формулы 6-9), описывающие отношения морфологических и семантических категорий существительных, соответствующих внешним элементам данного триплета (табл. 1).

Таблица 1 – Формализм модели идентификации семантических отношений

Глагол	Падеж	Предикат семантической функции
родиться, похищать, похитить, убивать, убить, красть, выкрасть, украсть, обмануть, обманывать, грабить, ограбить и др.	Темпоралис whenacted	10
	Локатив whereacted	8
	Объектив toactsmth	9
	Агенс tobeactedbysmth	6

3. Этапы работы лингвистического процессора рассмотренной задачи

Предложенная модель используется на семантическом этапе лингвистического процессора, включающего также графемный, морфологический, контекстный и семантический этапы анализа (рис. 1).



Рис. 1. Структурная схема лингвистического процессора подсистемы идентификации криминалистически значимых фактов в слабоструктурированных текстах.

Подавляющую часть субъектов криминалистически значимых фактов, составляют персоналии, компании и организации, кроме того, часто необходимо определять факт местоположения того или иного субъекта. Специальное графическое оформление таких сущностей позволяет для их выделения использовать формализмы графемного анализа. Графемный этап используется для выявления антропонимов, топонимов (включающих названия населенных мест, внутригородских объектов и др.), названий организаций, предприятий

и учреждений и базируется на базе имен собственных и специальных алгоритмах.

На этапе морфологического анализа осуществляется обработка словоизменительных и словообразовательных форм, позволяющая, например, отнести глаголы ~~красть~~ и ~~украсть~~ к одному и тому же факту.

Качественное построение шаблона факта предполагает определение возможных имен сущностей, которые могут встречаться в текстах под разными знаками. Для повышения точности определения факта на этапе синтаксического анализа необходимо решить проблему кореферентности, в частности, ассоциировать местоимения со своими антецедентами, соотнося их с именуемой сущностью.

Задача построения связей шаблона факта представляет собой одну из центральных задач извлечения знаний из текстов, она направлена на распознавание в тексте набора возможных отношений между элементами шаблона, разработанными на предыдущем этапе. Для решения данной задачи на этапе семантического анализа лингвистического процессора используется разработанная логико-лингвистическая модель идентификации отношений между сущностями.

4. Практическая реализация разработанного метода

Задача извлечения криминалистически значимых фактов из полнотекстовой информации, обрабатываемой в процессе оперативно-служебной деятельности, была реализована для извлечения даты, места рождения персоналии, а также сведений о дате, месте, субъекте и объекте некоторого противоправного действия.

Программа представляет собой веб-приложение, анализирующее текст или список рассматриваемых текстовых файлов. Определенная фактографическая информация представляется пользователю в форме диалогового окна, в котором отображается извлеченный факт и предложение или несколько предложений, представляющие описание данного факта.

В разработанном приложении определяются семантические функции, выражающие информацию, соответствующую поставленной задаче: темпоралис - временная характеристика события, локатив - характеристика местонахождения, положения или состояния; агенс и объектив - участники заданного действия. В процессе реализации модели был определен набор глаголов, требующих выполнения данных семантических функций участников предложения (табл. 1).

Тестовая проверка разработанного приложения показала приемлемость использования предложенной модели.

Заключение

Таким образом, разработанная логико-лингвистическая модель идентификации семантических отношений сущностей позволила извлекать факты из слабоструктурированной текстовой информации виде триплета субъект – отношение – объект. Отношения между сущностями в данном триplete описывается предикатом взаимосвязи морфологических и семантических категорий партиципанта того или иного действия. Модель применяется на семантическом этапе лингвистического процессора информационной подсистемы идентификации криминалистически значимых фактов в слабоструктурированных текстах для чего к настоящему моменту описаны функции: темпоралис, локатив, объектив и агенс.

Библиографический список

- [Baeza-Yates, 1999] Baeza-Yates R., Ribeiro-Neto B. Modern Information Retrieval. — Addison-Wesley, 1999. — 340 p.
- [Ландэ, 2009] Ландэ Д. В. Интернетика: Навигация в сложных сетях: модели и алгоритмы: моногр. / Д. В. Ландэ, А. А. Снарский, И. В. Безсуднов — М.: Либроком (Editorial URSS), 2009. 264 с.
- [Бондаренко, 2007] Бондаренко М. Ф. Теория интеллекта: учебник/ Бондаренко М. Ф., Шабанов-Кушнарченко Ю. П. Харьков: Комп. СМІТ, 2007. — 576 с.
- [Хайрова, 2012] Хайрова Н. Ф. Використання логіко-алгебраїчної моделі семантичних відмінків для семантичного аналізу речення/ Н. Ф. Хайрова // 36. наук. пр. Військового ін-ту Київ. нац. ун-ту. — К.: ВІКНУ, 2012. — Вип. № 38. — С. 239—245.

LOGICAL-LINGUISTIC MODEL FOR IDENTIFICATION OF SEMANTIC RELATIONSHIPS BETWEEN ENTITIES ON BASE OF ALGEBRA OF FINITE PREDICATES

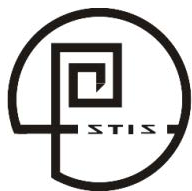
Khairova N., Uzlov D., Sharonova N.

*National Technical University
“Kharkiv Polytechnic Institute”, Kharkiv, Ukraine*

nina_khajrova@yahoo.com
poputcik@mail.ru
nvsharonova@mail.ru

This paper proposes a logical-linguistic model extracting semi-structured facts of natural language texts. To identify the fact some entities expressed by lexical units as well as semantic relations between them are defined in the text. The semantic relations are expressed by semantic functions of sentence participants. The functions are described by predicates of algebra of finite predicates. The model is applied to the semantic stage of linguistic processor of information subsystem for facts identification, which are essential for criminalistics, in the framework a semi-structured texts.

Key words: semi-structured facts; semantic functions; algebra of finite predicates; linguistic processor.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004:007:51-7

СПИРАЛЬНАЯ КОГНИТИВНАЯ МЕТАДИНАМИКА: ИССЛЕДОВАНИЕ МАСШТАБИРУЕМОСТИ ДИНАМИЧЕСКИХ ПРОЦЕССОВ

Прокопчук Ю.А.

*Институт технической механики НАНУ и ГКАУ,
Украинский государственный химико-технологический университет
г. Днепродзержинск, Украина
Itk3@ukr.net*

Развита концепция спиральной когнитивной метадинамики. Исследование показало наличие четко выраженной масштабируемой когнитивной динамики от элементарного когнитивного осциллятора на базе орграфа значений до динамического ядра в рамках орграфов значений, набросков и слоев познания. Предложенная модель удовлетворяет концепции нейродарвинизма и критерию целостности, так как все структуры переходят друг в друга при обобщении/детализации.

Ключевые слова: фрактальные когнитивные архитектуры и процессы; парадигма предельных обобщений; спиральная когнитивная метадинамика.

Введение

В связи с неуклонным возрастанием интереса к когнитивным технологиям большую значимость приобретают вопросы изучения и моделирования семантической структуры глобальной базы знаний [Голенков В.В. и др., 2001; Дубровский, 2007], –Global Work Space” [Seth, Baars, 2005], динамического ядра [Edelman, 2007], адаптивного резонанса [Grossberg, 2012], динамически интегрирующих различные накапливаемые знания.

Настоящая работа посвящена рассмотрению данных вопросов с точки зрения новой посткибернетической парадигмы системологии — парадигмы предельных обобщений (ППО), которая предложила новый подход к решению проблем познания, научного понимания и рационального объяснения феномена сложности открытых когнитивных систем (естественных и гибридных) [Прокопчук, 2012, 2013а, б, в]. Центральными объектами изучения данной работы являются эволюционный процесс возникновения когнитивной сложности, выразителем которого служит спиральная когнитивная метадинамика (СКД), явление масштабируемости когнитивных (рефлексивных) динамических процессов, а также пути реализации рассмотренных архитектур и процессов в информационных технологиях. Впервые концепция спиральной когнитивной метадинамики была предложена автором на конференции «Biologically Inspired Cognitive Archi-

tectures – ВІСА (Киев, сентябрь 2013 г.)» [Прокопчук, 2013в].

Цели исследования предполагают решение следующих задач:

- построение набора взаимосвязанных масштабируемых моделей репрезентативных структур; изучение особенностей конкуренции и селекции структур;

- изучение коллективных процессов активности, синхронизации, десинхронизации и управления в нейроморфных динамических сетях; построение масштабируемой модели динамического ядра, процессов "повторного ввода" (reentry) - динамического рекурсивного обмена сигналами;

- построение модели спиральной когнитивной метадинамики на наивысшем масштабном уровне, сравнение модели с теорией нейродарвинизма Дж. Эдельмана, глобального рабочего пространства Баарса и теорией адаптивного резонанса Гроссберга.

Именно популяционное мышление, по-видимому, оказывается наиболее сложным для восприятия [Edelman, 2007]: поведение (сначала в основном случайные реакции, а по мере формирования категорий – все более и более упорядоченное) приводит к опробыванию среды; адаптивные реакции, повторяемые чаще других, сопровождаются сохранением и усилением синапсов тех «нервных групп», которые их обеспечивают; итог - предельные модели знаний и функциональные системы [Прокопчук, 2013а, б].

Предложенные масштабируемые структуры на формальном языке позволяют объяснить, каким образом нервная система обеспечивает категоризацию предварительно неупорядоченного мира, без предположений о том, что мир исходно каким-либо образом размечен. Модель раскрывает также сложную динамику активности структур в ходе решения когнитивно-поведенческих задач.

1. Набор репрезентативных структур

Для решения поставленных задач в рамках ППО сформирован целостный набор репрезентативных структур (сущностей). К числу базовых и ключевых структур-сущностей относятся:

- элементарный тест или модальность τ ;
- Z-задача различения, где z/Z – это разновидность теста (Z-Difference, Z-Distinguishing, Z-Differentiation, Z-Control); поток Z-задач (Z-stream); поток результатов Z-задач (ReZ-stream); внутренний поток Z-задач (int Z-stream); внешний поток Z-задач (ext Z-stream);
- орграф значений теста $Gv(\tau)$, орграф доменов теста $G(\tau)$, орграф набросков образа $Gs(W)$, критические наброски; конус и окрестность обобщения, конус и окрестность детализации;
- динамический системопаттерн f/μ , идеальная V и вероятностная R закономерности, синдром S , предельный синдром S^* , радикал, системоквант;
- структурная энергия e (активность); импульс-солитон, элементарный (рефлексивный) осциллятор (когнитивный осциллятор); масштабируемое динамическое ядро, аттракторы;
- фрактальное время, дерево ритмокаскадов активности в рамках системокванта и модели деятельности;
- модель знаний $\{V\}$, среда радикалов, функциональная система, «собственное поведение»;
- модуль компетентности, слой познания, спираль усложнения, виртуальная сплошная среда (BCC), модель деятельности;
- Sketch of 'Thinking' = Z-stream; Sketch of 'Consciousness' = ReZ-stream; набросок «Переживания/воображения» = int Z-stream (привлекается энергия для роста орграфов, пересмотра эмоциональных индексов событий, формирования предельных моделей знаний, функциональных систем и т.д.); синдромный принцип управления.

Системопаттерны образуют каналы движения структурной энергии. Структурная энергия превращает любой объект ментальной сферы в "когнитивную структуру под напряжением".

Орграфы представляют собой специфический набор пространственно-временных структур фрактальной природы. В контексте данной работы рассмотрим некоторые структуры более подробно.

1.1. Орграфы значений, набросков, доменов

Элементарным тестом называются произвольные модальности, характеристики, категории, свойства, качества, радикал, квалиа (qualia), признаки явлений действительности и субъективной реальности. Тесты отвечают за первичное различение (коды-различения). Любой тест может принимать определенные значения. Под значениями понимаются как собственно значения, а также коды, метки, пейсмейкеры, ссылки, указатели на комплексные структуры и процессы. Множество значений теста τ обозначим $\{\underline{\tau}\}$. Множество значений теста τ с обобщающими связями в декогерентном представлении называется *орграфом значений* и обозначается

$$Gv(\tau) = \{a \rightarrow_e b\}_\tau, \quad (1)$$

где a, b – значения теста (b обобщает a ; a детализирует b); e – структурная энергия. В рамках орграфа имеет место суперпозиция всех значений теста τ . Фундаментальная триада ($a \rightarrow_e b$), реализуя сильные связи, является простейшим системопаттерном и базовым конструктором смысла (различием).

Орграфы значений дуалистичны – будучи материальны, они же работают как источники физических полей. Так, если триада ($a \rightarrow_e b$) проводит активность только в прямом направлении, то генерируется *импульс-солитон* – одиночный «нервный» импульс. Активный орграф значений (и набросков), следовательно, есть многоликое семейство солитонов (метафора – «орграф-оркестр»). Функции таких солитонов могут выступать как способы регуляции знаковых взаимодействий в рамках ВСС. Этот механизм «запоминает» последовательность значений тестов (как например, нуклеотидов в гене) и может, вероятно, передавать информацию об этом дистантно. Без дальней (волновой) миграции сигнала невозможна реализация ассоциативно-контекстных регуляций синтеза набросков образов. Здесь необходима волновая континуальность.

Триада ($a \rightarrow_e b$) может быть также элементарным когнитивным (рефлексивным) осциллятором – ЭКО (рис. 1).

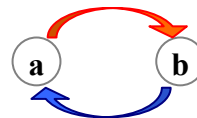


Рисунок 1 – Элементарный когнитивный осциллятор

Верхняя стрелка символизирует обобщение, а обратная нижняя – детализацию. Верхняя стрелка

есть всегда, а нижняя стрелка может отсутствовать, тогда ЭКО исчезает. Собственно солитоны и ЭКО являются теми кирпичиками (атомами), которые лежат в основе формирования многих когнитивных структур и процессов. ЭКО является примером нового типа обратной связи в когнитивных системах. На рис. 2. показано возможное склеивание (синхронизация) двух ЭКО. Подобное слияние образует *цепочки значений* (цепочки знаков, символов), которые играют самостоятельную роль, обнаруживая сходство с фрактальной структурой ДНК. Такие знаковые структуры превращаются в *солитонные модулированные поля*.

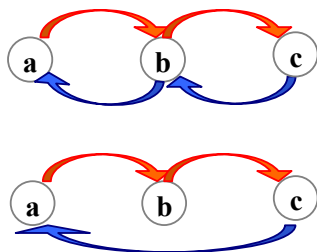


Рисунок 2 – Склеивание двух последовательных ЭКО

В общем случае на базе орграфа значений путем синхронизации активности могут возникать сколь угодно сложные аттракторы, группы солитонов и ЭКО (рис. 3 и рис. 4).



Рисунок 3 – Возбуждение солитонов и ЭКО на орграфе

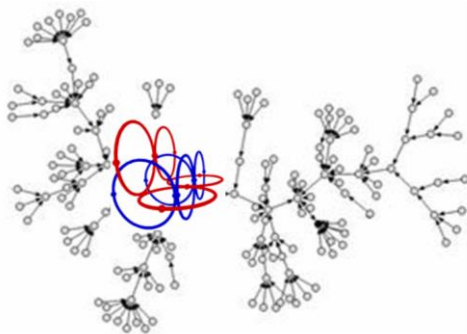


Рисунок 4 – Множественные контуры активности на базе орграфа значений

На рисунке 3 показаны как солитоны (фронты и солитонные пакеты, «бегущий фронт», «бегущие импульсы»), так и ЭКО. Некоторый ЭКО может выступать при восприятии (моделировании) конкретной ситуации в качестве фиксированного базового ритма-водителя. В отличие от солитонов активность, показанная на рис. 4, может существовать относительно долго, что важно для синхронизации с ЭКО других орграфов.

В рекогерентном представлении [Прокопчук, 2012, 2013a] каждое значение можно рассматривать как кубит (квантовое обобщение бита). Связи также могут иметь квантовое представление (мерцающие связи).

Если на базовые значения одного теста наложить ограничение целостности и проследить эволюцию данной целостности (домена) в процессе обобщения, то получим *орграф доменов теста* τ :

$$G(\tau) = \{T \rightarrow_e T'\}_{\tau}, \quad (2)$$

где e – структурная энергия (проводимость связи). Совокупная смысловая область элементов домена–потомка полностью совпадает с совокупной смысловой областью элементов домена–предка. Для фиксации того, что в качестве множества результатов теста τ используется домен T , будем использовать нотацию: τ / T .

Если ограничение целостности наложить на группу значений разных тестов и проследить эволюцию данной целостности (образа, наброска) в процессе обобщения, то в декогерентном представлении получим *орграф набросков*:

$$Gs(W) = \{P \rightarrow_e P'\}_W, \quad (3)$$

где W – произвольное явление действительности, образ; P, P' – наброски; e – структурная энергия (проводимость связи). Причем $||I(P')| < |I(P)|$, где I – оператор вычисления информации (происходит диссипация информации). Для одного и того же образа W могут быть построены орграфы набросков разных типов. Каждому типу соответствует свой оператор I . При рекогерентном рассмотрении существует квантово-семантическое представление орграфов доменов и набросков [Прокопчук, 2012, 2013a]. Орграф набросков можно рассматривать как действие *принципа дополненности* в ментальной области.

Орграфы набросков являются примером когнитивной категоризации. Переход от одного наброска к другому происходит в результате действия множественных элементарных *актов означивания* (свертки, символизации). Наряду с фундаментальной триадой $(a \rightarrow_e b)$ элементарные акты означивания $(\{a\} \rightarrow_e \{b\})$, $(P \rightarrow_e p)$ или $(P \rightarrow_e P')$ – состояния знака одновременно как модели и структуры – являются базовым конструктом смыслопорождения и текстопорождения.

Любые преобразования, движения, вывод, импульсы, вычислительные модели в системе координат $\{Gv(\tau)\}$ или $\{G(\tau)\}$ можно описать с помощью *динамических системопаттернов* (или просто системопаттернов) вида:

$$f / \mu : \{a / A\}, e / E \rightarrow \{b / B\}, \mu \in \{\mu\}_j, \quad (4)$$

где $\{a/A\}$ – входные тесты; $\{b/B\}$ – выходные тесты; e/E – требуемая структурная энергия, ресурсы; μ – механизм реализации. Важно отметить, что системопаттерн может быть как активным, так и реактивным.

Любая триада $(a \rightarrow_e b)$, $(P \rightarrow_e P')$, $(T \rightarrow_e T')$, являясь частным случаем системопаттерна f/μ , может генерировать как солитон, так и быть осцилляторным рефлексивным звеном – рефлексивной петлей, индукционным контуром на соответствующем уровне общности. В этом проявляется интегральный принцип рекурсивности – соотносимость знания с самим собой (знание начинает взаимодействовать само с собой). На рис. 5 показаны масштабируемые осцилляторы, на базе которых могут возникать произвольные динамические структуры в рамках орграфов значений, набросков, доменов и в целом – в рамках ВСС.

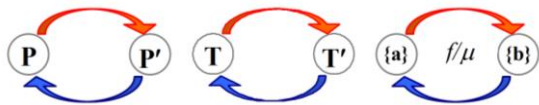


Рисунок 5 – Масштабируемые осцилляторы активности

На каждом иерархическом, квазизамкнутом уровне осцилляторы могут задавать автоколебательные эндогенные ритмо-вводители (говоря синергетическим языком – параметры порядка), которые универсальным образом порождают богатый спектр эндогенных внутренних ритмов системы. Таким образом, законы холизма, законы самосборки реальности не локальны ни в пространстве, ни во времени, но функционально самоподобны на разных масштабах.

Осцилляторная модуляция облегчает установление синхронности в пределах больших расстояний между когнитивными структурами. Орграф набросков – это сильные дискретные связи в орграфах значений и динамическая синхронизация между разными сетями – орграфами значений («склеивание» признаков стимула). Энергетический ландшафт на основе орграфа набросков показывает сосуществование осцилляторных мод (набросков, паттернов) и конкуренцию между ними. Он позволяет исследовать фазовые (качественные) переходы между различными типами осцилляторной активности. Показано [Прокопчук, 2012], что энергия распределяется по осцилляторным модам согласно степенному закону (имеет место самоорганизующаяся критичность): максимум приходится на критические наброски (наброски, которые нельзя обобщить без потери однозначности заключения). Нейродинамические феномены синхронизации и групповой осцилляторной активности являются инструментальным средством мозга в решении задач восприятия и сенсомоторной координации.

Концепция орграфов значений, доменов и

набросков, системопаттернов указывает на то, что в рамках ВСС действуют глобальные холистические пространственно-временные законы, в частности два фундаментальных холистических механизма связности Универсума. Первый механизм опосредован динамическим хаосом в нелинейных системах и заключается в возможности синергетической синхронизации слабо связанных, удаленных нелинейных систем (в нашем случае – когнитивных осцилляторов разного уровня общности). Второй механизм основан на существовании макроквантовых корреляций (значений-кубитов, орграфов и т.д.). Паттерны в орграфах задают третий механизм (сильной) связи.

1.2. Модель динамического ядра

Чтобы ментальный образ оставался стабильным и не угасал, нужно как-то поддерживать работу нейронов, которые возбуждаются при осознании, но без поддержки мгновенно гаснут. За это отвечает обратная связь между набросками (рекурсивные контуры с массовым параллелизмом), которая заставляет стимулы циркулировать по петлям обратной связи, поддерживая стабильность нейронной цепи. Благодаря обратным связям – процессу посылания стимула из коры в зоны перцепции и обратно (повторный вход – re-entry), мозг может «говорить сам с собой», что необходимо для возникновения памяти и мыслей – феномена «внутреннего видения». По аналогии с моделью сознания Дж. Эдельмана [Edelman, 2007] подобную активность назовем *динамическим ядром* (ДЯ), порождаемому доминантными, но постоянно изменяющимися циклическими цепями связей между элементами активности. ДЯ является результатом фазового перехода в осцилляторной активности (от бессознательного к сознательному) и позволяет экономно расходовать энергию, концентрируя ее на относительно небольшом множестве набросков (осцилляторных мод). ДЯ связывает между собой наброски-аттракторы разного уровня обобщенности, обеспечивая масштабируемость. Схематично ДЯ с петлями обратной связи в рамках орграфа набросков показано на рис. 6

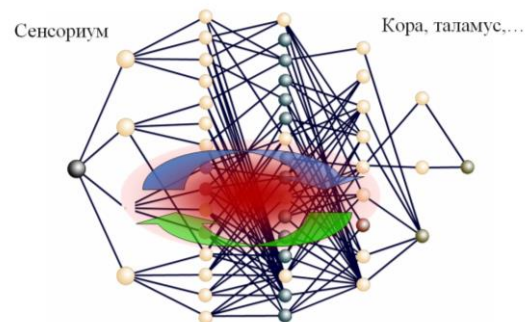


Рисунок 6 – Динамическое ядро в рамках орграфа набросков

Сравнение ДЯ в рамках орграфа набросков с активностью в рамках орграфа значений (рис. 4) и ЭКО позволяет проследить масштабируемый характер такой активности.

Для существует также в среде радикалов/функциональной системе, представляя собой текущее состояние системокванта – множества активных в текущий момент системопаттернов. Движение энергии (активности) по системопаттерну (развертывание системопаттерна) создает «структуру под напряжением».

2. Спиральная когнитивная метадинамика

2.1. Слой познания

Для решения той или иной когнитивной задачи (Z-задачи) формируется множество прецедентов с известными исходами $\Omega = \{\alpha(\{\tau/T\}, \underline{z}/Z)\}$, где $Z = \{1, \dots, N\}$ – множество заключений (диагнозов, прогнозов, управлений); $\{\tau/T\}$ – множество значений тестов. Без потери общности примем, что каждый тест входит в описание прецедента (ситуации действительности) один раз. Кроме того, будем рассматривать описания прецедентов с полной информацией (имеются значения всех тестов из $\{G(\tau)\}$). Через $\Omega(\{\tau/T_0\})$ обозначим априорные описания прецедентов. Множество всех описаний базы прецедентов образует оргграф набросков $\Omega(Z)$, в котором $\Omega(\{\tau/T\}, Z)$ – отдельный набросок. Можно также рассмотреть оргграф набросков каждого прецедента α , тогда описание $\alpha(\{\tau/T\}, \underline{z}/Z)$ – это отдельный набросок [Прокопчук, 2012, 2013б].

Контекстом Z-задачи называется кортеж $K = \langle \Omega(Z), \{G(\tau)\}, G(z) \rangle$. Идеальной закономерностью V в рамках контекста K называется произвольная совокупность значений тестов, позволяющая однозначно установить заключение:

$$V = (\{\underline{a}/A\} \rightarrow \underline{z}/Z), \quad (5)$$

$$\exists \alpha(\{\tau/T\}_\alpha, \underline{z}/Z) \in \Omega(Z) : \{\underline{a}/A\} \subseteq \{\tau/T\}_\alpha.$$

Компактная запись: $V(\{\underline{a}/A\}, \underline{z}/Z)$. Закономерность является разновидностью системопаттерна. Формальным синдромом S (или просто синдромом) в рамках контекста K называется избыточная идеальная закономерность. Другими словами, ни один тест из описания $S(\{\underline{a}/A\}, \underline{z}/Z)$ убрать нельзя без потери однозначности заключения. Закономерность $V'(\{\underline{a}/A'\}, \underline{z}/Z)$ доминирует закономерность $V(\{\underline{a}/A\}, \underline{z}/Z)$, если $\{a\}' \subseteq \{a\}$, $\forall a \in \{a\}' : A \leq A'$ и выполняется хотя бы одно из условий: а) $|\{a\}'| < |\{a\}|$; б) $\exists a \in \{a\}' : A < A'$. Предельным синдромом S^* в рамках контекста K называется синдром, у которого отсутствуют доминирующие закономерности.

Множество всех закономерностей в рамках контекста K обозначим через $\{V\}_{Full}$, множество всех синдромов через $\{S\}_{Full}$, а множество всех предельных синдромов через $\{S^*\}_{Full}$. Ясно, что $\{S^*\}_{Full} \subseteq \{S\}_{Full} \subseteq \{V\}_{Full}$.

Моделью знаний называется произвольное множество закономерностей $\{V\}$, которое позволяет установить заключение как минимум для каждого прецедента из $\Omega(\{\tau/T_0\})$. Модель знаний $\{V'\}$ доминирует модель знаний $\{V\}$, если она применима к большему числу набросков прецедентов из $\Omega(Z)$, включая и те наброски, к которым применима $\{V\}$. Модели знаний $\{V\}_{Full}$, $\{S\}_{Full}$ и $\{S^*\}_{Full}$ недоминируемы и эквивалентны в плане доминирования. Большой практический интерес представляет поиск всех минимальных (неизбыточных) по составу моделей знаний, которые принадлежат $\{S^*\}_{Full}$ и эквивалентны по доминированию $\{S^*\}_{Full}$. Такие модели знаний обозначим $\{S^*\}_{Full-Min}$. «Материализация» минимальных моделей в виде среды радикалов (СР) приводит к функциональной системе (ФС) когнитивно-поведенческого уровня - « $\{S^*\}_{Full-Min}$ - СР» и описывает процесс формирования имплицитного опыта. Спонтанное формирование предельных моделей знаний рассмотрено в [Прокопчук, 2013а, б].

Модуль компетентности (МК) представляет собой совокупность всех структур в рамках Z-задачи, а именно: $МК_Z = \langle K \vee \{Gs(W)\}, \{V\}, \{S\}, \{S^*\}, \{СР/ФС\} \rangle_Z$. Оргграф МК – совокупность взаимосвязанных МК, отвечающих оргграфу $G(z)$. Оргграф МК является формализацией слоя познания в рамках произвольной Z-задачи. Каждый модуль компетентности отражает результаты всех этапов когнитивной самоорганизации в слое познания - интенциональную структуру актов познания:

$$\Omega(\{\tau/T_0\}, Z), \{G(\tau)\} \vee \{G\nu(\tau)\} \rightarrow \{Gs(\alpha)\} \rightarrow \{V\} \rightarrow \{S\} \rightarrow \{S^*\} \rightarrow \{S^*\}_{Min} \rightarrow \{ФС\}. \quad (6)$$

Модуль компетентности получает активацию от цели, если он дает эффект, который удовлетворяет цель. Модуль может быть также заторможен целью, если эффект от его действий будет мешать удовлетворению цели. В рамках синдромного принципа управления [Прокопчук, 2012] целями-потребностями служат активированные предельные синдромы.

2.2. Модель СКД

Спиральная когнитивная метадинамика (СКД) представляет собой путь направленного

морфогенеза или спонтанного нарастания сложности (рис. 7). СКД опирается на процесс самоорганизации (6) и свойство орграфов набросков переходить в орграфы значений на более высоком системном уровне (акт означивания или "движение означивания").



Рисунок 7 – Спиральная когнитивная метадинамика

В рамках каждого слоя познания имеет место самоорганизующаяся критичность, в то время как количество метапереходов (усложнений) не лимитировано. Более высокие уровни познания (управления) рассматриваются как ингибиторы активности более низких уровней. Однако при оптимальном регулировании управление происходит с минимальным участием высших уровней (действуют автоматизмы). СКД не предполагает вербализацию, поэтому характерна для всех живых существ, но у человека эволюция, как в филогенезе, так и в онтогенезе достигла максимального уровня сложности. На рис. 8 показан самый крупный масштаб динамического ядра, соединяющего разные слои познания.

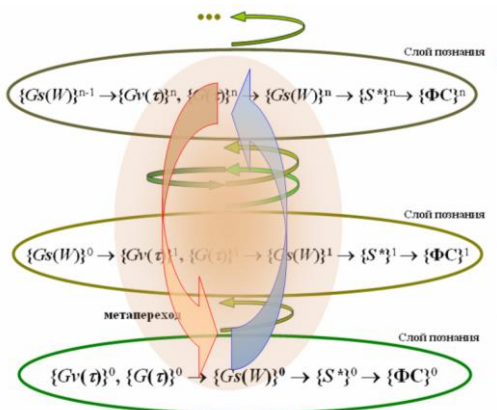


Рисунок 8 – Динамическое ядро в рамках СКД

Для СКД характерны самоподобие, саморазвитие, самодвижение. С возрастанием уровня сложности возрастает управляемость фазовыми переходами в энергетическом поле слоя (осцилляторной активностью) и управляемость перемещением динамического ядра и, как следствие, возрастает степень осознанности (наблюдаемости) когнитивных процессов. Пределом

управляемости и наблюдаемости является логическое мышление [Прокопчук, 2013в].

Заключение

Проведенное исследование показало наличие четко выраженной масштабируемой когнитивной динамики от элементарного когнитивного осциллятора на базе орграфа значений до динамического ядра в рамках орграфов значений, набросков и слоев познания. Предложенная модель удовлетворяет концепции нейродарвинизма и критерию целостности, так как все структуры переходят друг в друга при обобщении/детализации.

Библиографический список

- [Голенков В.В. и др., 2001] Голенков В.В. Представление и обработка знаний в графодинамических ассоциативных машинах / В.В.Голенков, О.Е.Елисеева, В.П.Ивашенко и др.: Под ред. В.В.Голенкова. – Мн.: БГУИР, 2001. – 412 с.
- [Дубровский, 2007] Дубровский Д.И. Сознание, мозг, искусственный интеллект. – М.: Стратегия-Центр, 2007. – 272 с.
- [Edelman, 2007] Edelman G. Second Nature: Brain Science and Human Knowledge. Yale University Press, 2007. – 224 p.
- [Seth, Baars, 2005] Anil K. Seth, Bernard J. Baars. Neural Darwinism and Consciousness // Consciousness and Cognition. – Vol. 14. – 2005. – Pp. 140-168.
- [Grossberg, 2012] Grossberg, S. (2012). Adaptive Resonance Theory: How a brain learns to consciously attend, learn, and recognize a changing world. Neural Networks, 37, 1-47.
- [Прокопчук, 2012] Прокопчук Ю. А. Принцип предельных обобщений: методология, задачи, приложения. Монография. – Дн-вск: ИТМ НАНУ и НКАУ, 2012.- 384 с.
- [Прокопчук, 2013а] Прокопчук Ю.А. Модели спонтанной когнитивной динамики // Материалы III Международной научной конференции «Открытые семантические технологии проектирования интеллектуальных систем – OSTIS-2013» (Минск, 21-23 февраля 2013 г.). – Минск: БГУИР, 2013. – С. 251 – 256..
- [Прокопчук, 2013б] Прокопчук Ю. А. Модели когнитивных архитектур и процессов на основе парадигмы предельных обобщений / Ю.А. Прокопчук // Кибернетика и вычисл. техника. – 2013. - Вып. 171. - С. 37-51
- [Прокопчук, 2013в] Prokopchuk Y. Spiral Cognitive Metadynamics // Annual International Conference on Biologically Inspired Cognitive Architectures (BICA) (Kiev, Ukraine, September 14-15). - Fairfax, USA, The BICA Society Publ, 2013. – P. 25.

SPIRAL COGNITIVE METADYNAMICS: DYNAMIC PROCESS SCALABILITY ANALYSIS

Prokopchuk I.A.

*Institute of Technical Mechanics of the NASU,
Dnepropetrovsk, Ukraine*

Itk3@ukr.net

A concept of spiral cognitive metadynamics is developed. The study has shown the presence of distinct scalable cognitive dynamics from an elementary cognitive oscillator on the basis of a value digraph to a dynamic core in the framework of value digraphs and cognition sketches and layers. The proposed model meets the neural Darwinism concept and the integrity criterion because all structures turn into one another on generalization/detailing.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8+620

СИТУАЦИОННЫЙ ПОЛИГОН КАК ИНСТРУМЕНТ СИТУАЦИОННОГО УПРАВЛЕНИЯ В ЭНЕРГЕТИКЕ

Массель А.Г., Иванов Р.А.

*Институт систем энергетики им. Л.А. Мелентьева СО РАН,
г. Иркутск, Россия*

amassel@gmail.com

crowndriver@gmail.com

В статье рассмотрен подход к разработке программной системы, получившей название Ситуационного полигона. Разработка базируется на методах семантического моделирования (когнитивного, событийного и БСД-моделирования) и научных прототипах интеллектуальной ИТ-среды, разработанных ранее при участии авторов для целей поддержки принятия решений в исследованиях и обеспечении энергетической безопасности России и ее регионов. Предусмотрено расширение интеллектуальной ИТ-среды за счет использования принципов сетцентричности и ситуационной осведомленности, а также авторского геокомпонента для 3D-визуализации рассматриваемых ситуаций. В дальнейшем предполагается интеграция Ситуационного полигона, ориентированного на поддержку стратегических и тактических решений и разрабатываемой одновременно распределенной интеллектуальной интерактивной советующей системы, предназначенной для поддержки оперативных решений.

Ключевые слова: ситуационное управление, семантические технологии; когнитивное, событийное и БСД-моделирование, ситуационная осведомленность.

Введение

В связи с развитием в России концепции интеллектуальных энергетических систем становится актуальным использование методов искусственного интеллекта при управлении энергетическими системами [Кобец, 2010, Воропай, 2011]. Представляется, что своевременно вспомнить предложенный и развиваемый в 1970-х -1980-х годах Д.А. Поспеловым подход к ситуационному управлению, основанный на применении достижений искусственного интеллекта [Поспелов, 1986]. В свое время этот подход не был реализован в полном объеме, в первую очередь по причинам несовершенства технических средств.

Современный уровень, как вычислительной техники, так и достижений в области разработки интеллектуальных технологий, позволяет вернуться к концепции ситуационного управления. Актуальность этого обусловлена также тем, что появилось направление, называемое «ситуационной осведомленностью», связанное с возможностью 3D-визуализации моделируемых ситуаций.

Авторами предложена интеграция этих подходов в рамках так называемого «Ситуационного полигона», предназначенного как для

моделирования (проигрывания) возможных ситуаций (в этом случае он может рассматриваться как платформа для ситуационного моделирования), так и для поддержки принятия стратегических и тактических решений. Для поддержки оперативных решений реализуется, при участии авторов, версия «Ситуационного полигона», получившая название распределенной интерактивной интеллектуальной советующей системы (РИИСС) [Massel A., 2013].

Обе разработки базируются на методах семантического моделирования и научных прототипах интеллектуальной ИТ-среды, разработанной ранее при участии авторов для целей поддержки принятия решений в исследованиях и обеспечении энергетической безопасности России и ее регионов.

Ситуационный полигон реализуется в виде программной разработки, построенной по принципам интеллектуальной инструментальной среды. В основе разработки лежат методы ситуационного управления, семантического моделирования и принципы сетцентричности и ситуационной осведомленности.

1. Ситуационное управление, семантическое моделирование и

принципы сетцентричности и ситуационной осведомленности.

Суть ситуационного управления заключается в выборе управленческих решений с учетом сложившейся ситуации из некоторого набора допустимых (типовых, стандартных) управляющих воздействий. При ситуационном управлении проблема выбора управляющих воздействий сводится к адекватной оценке состояния объекта и среды (что усложняется при наличии факторов неопределенности), отнесению соответствующей текущей ситуации к одному из типовых классов и выбору такого управления (из определенного набора альтернатив), которое приводит к достижению поставленной цели управления (целевой ситуации) [Поспелов, 1986], [Васильев, 2012].

Учитывая наличие факторов неопределенности, при ситуационном управлении, как правило, не удастся построить и использовать математические модели, поэтому предлагается использовать методы семантического моделирования, к которым автор относит когнитивное и событийное моделирование и моделирование на основе байесовских сетей доверия (БСД-моделирование). Модели представляются, как правило, в графическом виде.

Когнитивные модели отображают основные понятия и причинно-следственные отношения между ними в графическом виде. Событийные модели, основанные на алгебраических сетях (Joiner-net – отечественная разработка [Столяров, 2010]), используются для моделирования развития ситуаций. БСД-моделирование предлагается применять для оценки рисков тех или иных ситуаций с использованием экспертных оценок.

Сетцентричность – принцип организации систем управления, позволяющий реализовать режим ситуационной осведомленности благодаря формированию и поддержанию единой для всех ярусов управления, целостной, контекстной информационной среды и включения в процесс её непрерывной актуализации возможно большего числа источников первичной информации. Важнейшим условием реализуемости концепции сетцентричности на практике является использование одного и того же, не фрагментированного по масштабному признаку информационного образа реальной ситуации всеми ярусами системы управления [Ерёмченко, 2009], что достигается созданием единой информационной среды, которая позволяет добиться высокоточной актуализации данных.

Неотъемлемой частью успешного принятия управленческих решений в любой области является владение текущей обстановкой или ситуационная осведомленность (англ. Situational Awareness). Исходя из определения, ситуационная осведомленность – принцип комплексного, в минимальной степени опосредованного картографическими, модельными, либо иными условиями представления разнородной (общегеографической, навигационной, тактической

и т.д.) информации в единой глобальной геоцентрической системе координат. Смысл понятия сводится к управлению ситуацией посредством документально точной, ориентированной на чувственное восприятие информационной среды, не фрагментированной по пространственному, масштабному, тематическому, ведомственному, либо иным признакам.

Сочетание документально точных изображений, полученных разными средствами с разных ракурсов (космического и аэроснимков) и не опосредованных картографическими условностями, позволяет обеспечить принципиально новое качество восприятия ситуации.

Ситуационная осведомленность имеет прямое отношение к процессу принятия решений. Рис. 1 иллюстрирует связь между ситуационной осведомленностью и поддержкой принятия решений. Ситуационная осведомленность предшествует принятию решений, потому что ЛПР должен воспринимать ситуацию, чтобы ставить перед собой цель [Endsley, 1997].



Рисунок 1 – Место ситуационной осведомленности в процессе принятия решения

2. Ситуационный полигон.

Ситуационный полигон – совокупность данных, знаний и инструментальных средств, которые предназначены для рассмотрения (моделирования) разных ситуаций, направления их развития и анализа их последствий.

Предполагается два режима использования Ситуационного полигона:

- 1) в стабильной обстановке, когда могут описываться или моделироваться возможные ситуации и предлагаемые решения;
- 2) в стрессовой обстановке (условия чрезвычайной ситуации), когда необходимо оперативно принимать решения.

В стабильной ситуации Ситуационный полигон может использоваться как тренажер, искусственно формирующий образ реальной ситуации с заданными параметрами, моделирующий развитие этой ситуации и совмещающий: генерирование, сценариев описания и развития ситуации, визуализацию геопространственной информации, возможность внесения управляющих воздействий на разных стадиях, различное отображение результатов моделирования.

В стрессовой ситуации Ситуационный полигон может использоваться как для поддержки принятия стратегических и тактических решений, так и, при развитии ситуационного полигона (интеграции его с РИИСС), для поддержки принятия оперативных решений в реальном времени.

Разработка Ситуационного полигона выполняется на основе интеллектуальной инструментальной ИТ-среды [Массель А., 2010]. Предполагается реализация Ситуационного полигона в виде многоагентной системы с реализацией интеллектуальных агентов как сервисов.

Одним из основных компонентов Ситуационного полигона является авторский геокомпонент, направленный на визуализацию результатов энергетических исследований. В связи с тем, что геосервисы, подобные GoogleEarth, обладают ограниченным набором объектов и процессов, было предложено использовать программный компонент (Геокомпонент) для формирования KML-файла из результатов исследований.

В качестве баз знаний описаний и реализаций ЭкС предложено использовать базы знаний интеллектуальной ИТ-среды и РИИСС [Massel A., 2013].

Лицу, принимающему решения (ЛПР), предлагается для оценки различных вариантов развития ситуации воспользоваться инструментальными средствами когнитивного, событийного моделирования и БСД-моделирования. Все результаты ситуационного моделирования предполагается визуализировать.

На рис. 2 представлена схема Ситуационного полигона, которая включает в себя:

- Базу знаний описаний сценариев ЭкС.
- Базу знаний прецедентов ЭкС.
- Авторский геокомпонент [Иванов, 2013].
- Авторские инструментальные средства когнитивного, событийного и БСД-моделирования [Массель А., 2011, Аршинский, 2010, Пяткова, 2013].

3. Пример применения Ситуационного полигона.

Рассмотрим ситуацию, связанную, например, с разрушением Иркутской ГЭС в результате террористического акта или природной катастрофы

(землетрясения). Будем считать, что реализуется второй режим. В базе знаний уже хранится

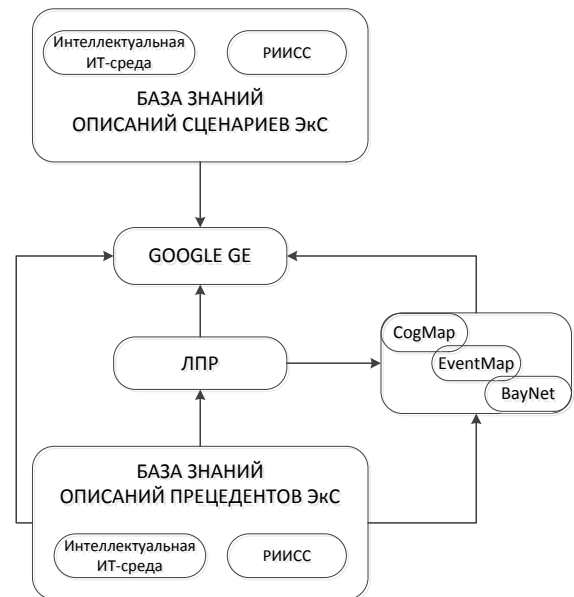


Рис. 2. Схема Ситуационного полигона.

когнитивная модель, описывающая эту ситуацию, которая отражает основные понятия и связи между ними. В качестве понятий могут выступать такие события, как: «угроза разрушения ГЭС»; возможные последствия: «отключение энергоресурсов», «угроза затопления», «разрушение объектов инфраструктуры» и т.п.; ликвидационные мероприятия: «мобилизация сил МЧС и военных частей», «подключение резервных источников питания», «эвакуация населения» и др. Событийная модель (также может извлекаться из базы знаний) позволяет моделировать развитие ситуации: например, распространение паводка, каскадные аварии либо устранение последствий чрезвычайной ситуации. Для 3D-визуализации результатов моделирования используется авторский геокомпонент. БСД-модель позволит оценить риски принятия тех или иных решений, а также затраты на мероприятия. Предполагается, что в состав Ситуационного полигона входит экспертная система, которая при наличии заполненных баз знаний позволит сформировать список рекомендуемых решений. Также Ситуационный полигон, как платформа для ситуационного моделирования, позволит «проигрывать» всевозможные стратегии по типу: «что будет, если...».

Заключение

Предлагается, на основе современной трактовки ситуационного управления [Массель Л., 2014], разработка Ситуационного полигона, основанного на использовании методов семантического моделирования и принципов сетцентричности и ситуационной осведомленности.

Ситуационный полигон, как платформа для ситуационного моделирования, является развитием разработанной ранее при участии авторов

интеллектуальной ИТ-среды, предназначенной для поддержки принятия решений в исследованиях и обеспечении энергетической безопасности России и ее регионов.

Впоследствии предполагается интеграция Ситуационного полигона, ориентированного на поддержку стратегических и тактических решений и разрабатываемой одновременно РИИСС, предназначенной для поддержки оперативных решений.

Результаты, представленные в статье, получены при частичной финансовой поддержке грантов РФФИ № 12-07-00359, № 13-07-00140, № 14-07-00116, гранта Программы Президиума РАН №229 и гранта на выполнение интеграционного проекта СО РАН и НАН Беларуси №18 (2012-2014).

Библиографический список

- [Кобец, 2010] Кобец Б.Б., Волкова И.О. Инновационное развитие электроэнергетики на базе концепции Smart Grid / М.: ИАЦ Энергия, 2010. – 208 с.
- [Воропай, 2011] Воропай Н.И. Интеллектуальные электроэнергетические системы: концепция, состояние, перспективы // Автоматизация и ИТ в энергетике. – №3. – 2011. – С. 11-16.
- [Поспелов, 1986] Поспелов Д.А. Ситуационное управление. Теория и практика. – М.: Наука, 1986. – 284 с.
- [Васильев, 2012] Васильев В.И. Интеллектуальные системы защиты информации – М.: Машиностроение, 2012. – 171 с.
- [Столяров, 2010] Столяров Л.Н. Философия событийного моделирования на примере сценария энергетической катастрофы // Труды Международной конференции «Информационные технологии в науке, образовании, телекоммуникации и бизнесе». – Украина, Гурзуф, 2010. – С. 197-200.
- [Massel A., 2013] Massel A.G. Distributed intelligent interactive advising system for decision support in extreme situation / The 15 International Workshop on Computer Science and Information Technologies (CSIT'2013). – Ufa: UGATU, 2013. – Vol. 1. - P. 1 - 6.
- [Ерёмченко, 2009] Ерёмченко Е. Н. Неогеография и Situational Awareness. Материалы конференции «Неогеография XXI-2009» X Международного Форума «Высокие технологии XXI века, Москва, 2009. - С. 434-436.
- [Endsley, 1997] Endsley, M. R. (1997) The Role of Situation Awareness in Naturalistic Decision Making. In C. Zambok & G. Klein (Eds.), Naturalistic Decision Making (pp. 269-284). Mahwah, NJ: LEA
- [Массель А., 2010] Массель А.Г. Методологический подход к организации интеллектуальной поддержки исследований проблемы энергетической безопасности / «Информационные технологии». – №9. – 2010. – С. 32-36.
- [Иванов, 2013] Иванов Р.А. Методика 3D-визуализации для поддержки принятия решений в энергетических исследованиях // Современные технологии. Системный анализ. Моделирование. - 2013. №1(37). – С. 116-121.
- [Массель А., 2011] Массель А.Г. Когнитивный подход в исследованиях проблем энергетической безопасности России / Когнитивный анализ и управление развитием ситуации (CASC'2011): Труды IX Международной конференции (14-16 ноября 2011 г., Москва). – М.: ИПУ РАН, 2011. – С. 224-228.
- [Аршинский, 2010] Аршинский В.Л. Событийное моделирование чрезвычайных ситуаций в энергетике // Информационные технологии в науке, образовании, телекоммуникации и бизнесе: Труды Международной конференции – Украина, Гурзуф, 2010. – С. 299-301.
- [Пяткова, 2013] Пяткова Е.В. Методика моделирования угроз энергетической безопасности с помощью байесовских сетей доверия // Современные технологии. Системный анализ. Моделирование. – Иркутск: ИрГУПС. – № 3(39). – 2013. – С. 133-140.
- [Массель Л., 2014] Массель Л.В., Массель А.Г. Ситуационное управление и семантическое моделирование в энергетике / Материалы IV международной научно-технической

конференции «OSTIS-2014» – Беларусь, Минск: БГУИР, 2014 (в наст. сборнике).

SITUATIONAL MODELING PLATFORM AS A TOOL OF CONTINGENCY MANAGEMENT IN ENERGY SECTOR

Massel A.G., Ivanov R.A.

** Melentiev Energy Systems Institute of Siberian
Branch of the Russian Academy of Sciences
Irkutsk, Russia*

amassel@gmail.com

crowndriver@gmail.com

The article describes the approach to the development of a software system, called Situational polygon, what means Situational modeling platform (SMP) for contingency management. The SMP development is based on the semantic modeling methods (cognitive, event and Bayesian believe nets (BBN) simulation) and on the research prototypes of the intelligent IT-environments, previously developed with the participation of the authors for the purposes of decision support in research and providing of energy security of Russia and its regions. SMP assumes an extension of intelligent IT- environment through the use of network-centric and situational awareness principles, as well as author geocomponent for 3D- visualization of the considered situations.

Introduction

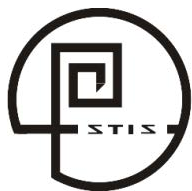
Modern level as computer technology and advances in the development of intelligent technologies allows returning to the concept of situational or contingency management. The relevance of this is due to the fact that there was a direction called "situational awareness", associated with the possibility of 3D-visualization of simulated situations. The authors propose an integration of these approaches in the so-called SMP, intended for modeling (playback) of possible situations and for supporting of strategic and tactical decisions.

Main Part

The concepts of situational management, semantic modeling and principles of network-centric and situational awareness are described. The role and place of situational awareness in decision-making is illustrated. The SMP scheme and list of their main components are considered. The possibility of using the SMP is shown on the example of the situation associated with the destruction of the Irkutsk hydroelectric station.

Conclusion

The results presented in this paper were obtained with the partial financial support by RFBR grants № 12-07-00359, № 13-07-00140, № 14-07-00116, by grant of RAS Presidium Program № 229 and by grant for the implementation of the integration project of SB RAS and the National Academy of Sciences of Belarus № 18 (2012-2014).



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ЭКСПЕРТНЫЕ СИСТЕМЫ БЕЗ ЭКСПЕРТОВ

Клещев А.С., Смагин С.В.

*Институт автоматизации и процессов управления Дальневосточного отделения
Российской академии наук, г. Владивосток, Россия*

kleschev@iacp.dvo.ru

smagin@iacp.dvo.ru

В работе проведен анализ традиционных способов формирования баз знаний для экспертных систем, выделены их ключевые достоинства и недостатки. Рассмотрен подход, при котором в качестве модели зависимости между знаниями и данными используются онтологии предметных областей, имеющие вид небогатых систем логических соотношений с параметрами. Предложена схема индуктивного формирования и сопровождения хорошо интерпретируемых баз знаний для таких моделей, которая сохраняет все достоинства традиционных подходов и исключает их недостатки.

Ключевые слова: индуктивное формирование баз знаний; экспертная система; хорошая интерпретируемость; модель зависимости с параметрами; онтология предметной области.

Введение

Традиционно базы знаний для интеллектуальных (экспертных) систем формируются экспертами предметных областей вручную – при помощи соответствующих редакторов знаний [Protégé, 2013] [Клещев и др., 2006]. Как показали многолетние исследования, именно такое, экспертное, формирование баз знаний является наиболее «узким местом» в эффективном использовании подобных систем на практике [Гаврилова и др., 2004]. При своих очевидных преимуществах, такой подход имеет ряд серьезных недостатков, которые во многом являются причинами сложности разработки и сопровождения, а также высокой стоимости, как самих баз знаний, так и экспертных систем в целом.

Альтернативным способом создания баз знаний является интеллектуальный анализ данных (Data Mining), в частности автоматическое (индуктивное) формирование баз знаний на основе эмпирических данных (обучающих выборок) [Вагин и др., 2008]. Индуктивное формирование знаний представляет собой извлечение и переработку семантики из одной формы в другую, а именно из обучающей выборки – в базу знаний. Однако, все известные на сегодняшний день из литературы попытки двигаться в данном направлении не привели к прорывным результатам, потому что, отчасти снижая сложность и стоимость разработки и сопровождения, такой подход неизбежно приводил к формированию плохо интерпретируемых баз знаний [Витяев, 2006].

Причиной этого является то, что в традиционных постановках задач этого и смежных направлений (таких как машинное обучение (Machine Learning), обнаружение знаний в базах данных (Knowledge Discovery in Databases) и др.) считается, что модель зависимости между знаниями и данными (классами (кластерами) и объектами предметной области) – не известна. В этом предположении в качестве таких моделей зависимости выбираются те или иные проблемно-независимые математические модели [Загоруйко, 1999]. В результате формируемая при таком подходе база знаний получается формально пригодной для использования ее в экспертных системах, но при этом сама остается непонятной экспертам предметной области. При этом залогом эффективного использования экспертных систем является хорошая интерпретируемость (в системе понятий предметной области) их результата пользователями, а также понимание ими того, почему был получен именно этот результат. Автоматически (индуктивно) сформированные базы знаний (т.е. описания классов или кластеров) могут быть использованы в интеллектуальных системах только в том случае, если они понятны экспертам соответствующих предметных областей [Michie, 1980]. В этом случае специалисты предметных областей смогут не только сами пользоваться такими базами знаний в своей профессиональной деятельности, но и будут доверять экспертным системам, использующим модели этих знаний, а также смогут проверять выводы таких систем, сформированные подсистемами объяснений.

В работе [Клещев и др., 2012] рассматривается подход, при котором в качестве модели зависимости между знаниями и данными используются онтологии предметных областей, имеющие вид небогатых систем логических соотношений с параметрами. В таких моделях описанием классов (кластеров) является набор значений параметров, названный базой знаний, который является хорошо интерпретируемым по построению. Цель данной работы состоит в том, чтобы предложить схему индуктивного формирования и сопровождения хорошо интерпретируемых баз знаний для таких моделей, которая сохраняет все достоинства традиционных подходов и исключает их недостатки.

1. Экспертное формирование баз знаний

На Рисунке 1 представлена схема формирования и сопровождения баз знаний с участием эксперта предметной области. Эксперт предметной области при помощи редактора знаний формирует начальное состояние базы знаний. Поток объектов названо множество описаний объектов, поступающих на вход экспертной системе при ее практическом использовании. Решатель задач, получая на вход описание некоторого объекта и используя в своей работе базу знаний, формирует результат, а также его объяснение, которые передаются пользователю. Далее результат поступает в систему верификации, где верифицированные результаты накапливаются, чтобы затем быть переданными эксперту, который, анализируя их, осуществляет сопровождение базы знаний (формируя ее текущее состояние).

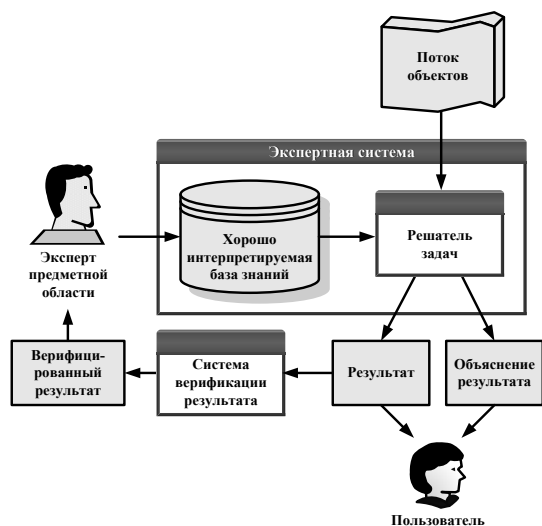


Рисунок 1 – Схема формирования и сопровождения баз знаний с участием эксперта предметной области

Перечислим наиболее важные преимущества и недостатки подобного подхода.

1.1. Преимущества экспертного формирования баз знаний

- База знаний хорошо интерпретируема, т.е. понятна специалистам предметной области.
- Уровень доверия к базе знаний зависит от квалификации и авторитета эксперта – чем выше авторитет, тем выше уровень доверия.

1.2. Недостатки экспертного формирования баз знаний

- Формирование и сопровождение базы знаний – процесс трудоемкий и по ряду причин (таких как неудобство, а также непривычность использования, низкая адаптивность программных средств) вызывает затруднения у экспертов.
- База знаний является субъективной, т.к. мнения групп экспертов одной и той же предметной области могут различаться.
- База знаний является неполной, т.к. опыт любого, даже первоклассного эксперта, всегда уже совокупности всех знаний о предметной области.
- Стоимость базы знаний (а, следовательно, и стоимость всей экспертной системы) высока, т.к. работа высококлассного специалиста стоит дорого – как на этапе формирования начального состояния базы знаний, так и на этапе ее сопровождения.
- Формирование и сопровождение базы знаний осуществляется вручную и требует участия того эксперта, который ее формировал, в течение всего периода использования экспертной системы.
- Замена одного эксперта на другого может привести к формированию новой базы знаний.

2. Индуктивное формирование и сопровождение хорошо интерпретируемых баз знаний

На Рисунке 2 представлена схема формирования и сопровождения баз знаний без участия эксперта предметной области.

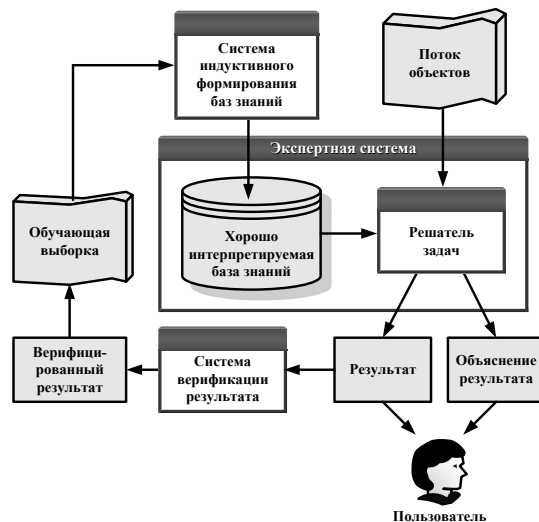


Рисунок 2 – Схема формирования и сопровождения баз знаний без участия эксперта предметной области

Принципиальным отличием данной схемы от предыдущей, является замена эксперта на систему индуктивного формирования баз знаний, на вход которой поступает обучающая выборка, а на выходе получается хорошо интерпретируемая база знаний. Тем не менее, на начальном этапе такая выборка, состоящая из максимально возможного числа наиболее полно описанных объектов предметной области, должна быть отобрана специалистом или экспертом. Перечислим преимущества этой схемы.

2.1. Преимущества предлагаемой схемы индуктивного формирования и сопровождения баз знаний

- База знаний так же понятна специалистами предметной области, как и в случае экспертного ее формирования.
- Формирование базы знаний происходит автоматически – по обучающей выборке.
- Уровень доверия к базе знаний зависит от объема и репрезентативности обучающей выборки.
- База знаний объективна, если обучающая выборка репрезентативна и имеет большой объем.
- База знаний полна настолько, насколько велик объем обучающей выборки, при условии, что обучающая выборка репрезентативна.
- Оценку полноты описаний объектов обучающей выборки может осуществить не только эксперт, но и достаточно опытный специалист предметной области, что снижает соответствующие расходы на формирование и сопровождение.
- В процессе практического использования экспертной системы качество базы знаний растет, т.к. результаты, которые прошли этап верификации, автоматически пополняют обучающую выборку.
- Ввиду особенностей системы индуктивного формирования (для указанных выше моделей), получаемая база знаний является абсолютно правильной и максимально точной относительно обучающей выборки, на основе которой она была сформирована, и при этом может обладать формой представления, принятой в предметной области.
- Алгоритмы, лежащие в основе системы индуктивного формирования знаний, независимы от экспертной системы и могут сопровождаться без необходимости в новой обучающей выборке.

3. Пример применения предлагаемой схемы

Рассмотрим в качестве адекватной и хорошо интерпретируемой модели зависимости онтологию медицинской диагностики, приближенную к реальной, которая представлена в виде небогатой системы логических соотношений с параметрами. Эта модель является важным частным случаем онтологии, опубликованной в работе [Клещев и др., 2005]. В работе [Клещев и др., 2012] предложены новые постановки основных задач индуктивного формирования баз знаний для таких моделей, а также представлен алгоритм обучения, решающий указанные задачи в этих постановках.

В работе [Смагин, 2013] представлено описание комплекса программ InForMedKB v1.1 (INductive FORmation of MEDical Knowledge Bases), в котором реализован указанный выше алгоритм обучения. Комплекс позволяет создавать обучающие выборки (состоящие из историй болезни (ИБ) различных разделов медицины) и на их основе индуктивно формировать базы медицинских знаний (в форме, принятой в медицинской литературе) хорошего уровня интерпретируемости и такого уровня

качества, который эксперты предметной области медицинской диагностики оценивают как достаточный для решения в ней практических задач. Также комплекс позволяет формировать объяснения баз знаний на основе описаний ИБ из используемых обучающих выборок. В разработанном комплексе нет формальной разницы между экспертными и индуктивно сформированными знаниями, т.к. форма представления одинакова – основана на онтологии и ориентирована на медицинские представления.

Описания заболеваний, входящих в базу медицинских знаний, состоят из описаний клинических проявлений признаков, входящих в клинические картины этих заболеваний. В таком описании для каждого признака указывается количество ИБ, в которых он наблюдался, число вариантов его клинического проявления (ВКП), а также описание этих вариантов. ВКП соответствует типу реакций организма на данное заболевание по данному признаку. Описание ВКП содержит информацию о числе периодов динамики в нем, о значениях признака в этих периодах и о границах длительности этих периодов. При этом для каждого ВКП формируется его объяснение – указывается количество ИБ, которые его поддерживают (в которых этот ВКП проявился), а также номера ИБ, которые входят в определяющее подмножество ИБ данного ВКП (исключение этих ИБ из обучающей выборки не позволит сформировать данный ВКП). Совокупность объяснений всех сформированных ВКП образует объяснение заболевания, а совокупность объяснений всех заболеваний образует объяснение базы медицинских знаний.

При помощи данного комплекса, на основе обучающей выборки реальных данных, содержащей истории болезни из раздела медицины «острый живот», была индуктивно сформирована база медицинских знаний, получившая в работе экспертную оценку. Ниже представлен фрагмент описания клинического проявления признака «Боли в животе (Локализация)» при аппендиците.

Признак «Боли в животе (Локализация)»

Количество историй болезни, в которых наблюдался признак, равно 22. Число вариантов динамики равно 9.

1 вариант: число периодов динамики = 2.

- *эпигастральная область*, затем через 5-7 часов

- *правая подвздошная область*

Вариант поддерживают истории болезни количеством 6.

2 вариант: число периодов динамики = 1.

- *правая подвздошная область*

Вариант поддерживают истории болезни количеством 3.

3 вариант: число периодов динамики = 2.

- *весь живот*, затем через 3-12 часов

- *правая подвздошная область*

Вариант поддерживают истории болезни количеством 6.

По мнению эксперта, полученная база знаний обладает высоким уровнем интерпретируемости для практикующего врача, входящие в нее описания заболеваний соответствуют знаниям, имеющимся в научной и учебной медицинской литературе, а в ряде случаев дополняют их описанием динамики клинических проявлений.

Заключение

Ключевым недостатком традиционного экспертного формирования баз знаний является сложность их сопровождения, в случае, когда на вход экспертной системе поступает поток объектов. Полученные и верифицированные результаты работы неизбежно выявляют дефекты в базе знаний, и чем таких объектов больше, тем сложнее эксперту ее модифицировать. Ключевым недостатком традиционного индуктивного формирования и сопровождения баз знаний является их плохая интерпретируемость. В аналогичных предыдущему случаю ситуациях, невозможность формирования понятного объяснения становится причиной того, что реально на практике экспертные системы не используются. Это, в свою очередь, влечет невозможность улучшения используемых баз знаний, т.к. поток объектов через экспертные системы не движется, а никакая обучающая выборка не способна адекватно заменить поток объектов реальной предметной области.

В работе представлена схема индуктивного формирования хорошо интерпретируемых баз знаний (для моделей зависимости с параметрами, являющихся онтологиями предметных областей), которая сохраняет все достоинства традиционных подходов, исключает их недостатки. Использование таких моделей зависимости и систем индуктивного формирования баз знаний, разработанных для них, позволяет формировать хорошо интерпретируемые базы знаний без участия экспертов предметной области. Работа выполнена при финансовой поддержке РФФИ (проекты № 12-07-00179-а и 14-07-00270-а) и ДВО РАН (проект № 12-I-П15-03).

Библиографический список

- [Вагин и др., 2008] Достоверный и правдоподобный вывод в интеллектуальных системах / Вагин В.Н. [и др.]; – М.: ФИЗМАТЛИТ. – 2008. – 704 с.
- [Витяев, 2006] Витяев Е.Е. Извлечение знаний из данных. Компьютерное познание. Модели когнитивных процессов. – Новосибирск: НГУ. – 2006. – 293 с.
- [Гаврилова и др., 2004] Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский – СПб: Питер. – 2000. – 384 с.
- [Загоруйко, 1999] Загоруйко Н.Г. Прикладные методы анализа данных и знаний. Новосибирск: Изд-во Ин-та математики. – 1999. – 270 с.
- [Клещев и др., 2005] Клещев А.С., Москаленко Ф.М., Черняховская М.Ю. Онтология и модель онтологии предметной области «Медицинская диагностика». – Владивосток: ИАПУ ДВО РАН. – 2005. – 44 с.
- [Клещев и др., 2006] Клещев А.С., Орлов В.А. Компьютерные банки знаний. Универсальный подход к решению проблемы редактирования информации // Информационные технологии. – М: Новые технологии. – 2006. – №5. – С. 25-31.
- [Клещев и др., 2012] Клещев А.С., Смагин С.В. Задачи индуктивного формирования знаний для онтологии медицинской диагностики // Научно-техническая информация. Серия 2. – М.: ВИНТИ РАН. – 2012. – №1. – С. 9-21.
- [Смагин, 2013] Смагин С.В. Комплекс программ для индуктивного формирования баз медицинских знаний в форме, принятой в медицинской литературе // Ломоносов-2013: Материалы XX Международной научной конференции студентов, аспирантов и молодых ученых. – М.: Издательский отдел факультета ВМК МГУ. – 2013. – С. 36-37.

[Michie, 1980] Michie D. Expert Systems // The Computer Journal. – 1980. – Vol. 23. № 4. – pp. 369-376.

[Protégé, 2013] Protégé – a free, open source ontology editor and knowledge-base framework [http://protege.stanford.edu]

EXPERT SYSTEMS WITHOUT EXPERTS

Kleschev A.S., Smagin S.V.

*Institute of Automation and Control Processes,
Far Eastern Branch of the Russian Academy of
Sciences, Vladivostok, Russia*

kleschev@iacp.dvo.ru

smagin@iacp.dvo.ru

Introduction

The article provides the analysis of traditional ways of knowledge bases formation for expert systems and highlights their key advantages and disadvantages. An approach is explored in which domain ontologies, having a form of unenriched systems of logical relationships with parameters, are used as a knowledge-data dependence model. A framework is suggested for inductive formation and maintenance of easily interpretable knowledge bases for such models, which retains all the advantages of traditional approaches and avoids their disadvantages.

Main Part

The key disadvantage of the traditional knowledge bases formation by experts is the complexity of maintaining them, when a stream of objects comes as input data into the expert system. Received and verified results inevitably reveal defects in the knowledge base, and the more such objects there are, the more difficult it becomes for the expert to modify the knowledge base.

The key disadvantage of traditional inductive formation and maintenance of knowledge bases is their poor interpretability. In such cases the inability of the expert system to form an understandable explanation, precludes the system from being used in actual practice. This, in turn, entails the impossibility to improve the knowledge base, as due to the fact that such expert systems are not being used, the stream of objects is not getting formed, and no learning sample is able to adequately replace the real stream of objects.

Conclusion

In the dependence models which are considered in this article the description of the classes and clusters of a domain is a set of parameter values (called a knowledge base), which is easily interpretable by way of construction. Use of such models and program systems for inductive formation of knowledge bases, developed using the models, will allow to form easily interpretable knowledge bases without the participation of domain experts.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

СОДЕРЖАНИЕ СИСТЕМНОГО АНАЛИЗА ПРИ АВТОМАТИЗАЦИИ ИНТЕЛЛЕКТУАЛЬНОЙ ДЕЯТЕЛЬНОСТИ НА УРОВНЕ ОТРАСЛИ

Клещев А.С., Шалфеева Е.А.

*Институт автоматизации и процессов управления,
г. Владивосток, Россия*

kleschev@iacp.dvo.ru

shalfe@iacp.dvo.ru

Работа посвящена особенностям выполнения, содержанию и результатам системного анализа в составе комплексной технологии автоматизации интеллектуальной деятельности на уровне отрасли. Выполнение всех представленных этапов и получение соответствующих результатов позволяют осуществить поддержку принятия решений на основе знаний и систематическое управление качеством этих знаний. Выполнение такого системного анализа гарантирует семантическую совместимость создаваемых интеллектуальных систем.

Ключевые слова: предметная область; интеллектуальные задачи; системный анализ; управление знаниями.

Введение

Анализ организации повседневной интеллектуальной деятельности и управления ее качеством показал необходимость перехода к новой *парадигме автоматизации* - автоматизации отрасли в целом, т.е. разработке единой для отрасли «облачной» интеллектуальной программно-информационной системы для выполнения всех задач принятия решений на основе знаний и всех задач управления качеством знаний [Клещев и др., 2013].

Ключевыми аспектами этой парадигмы являются: *понятные* специалистам *базы знаний* (БЗ); *полезные* БЗ в течение всего времени их эксплуатации; *возможность включения новых научных результатов* в БЗ; *автоматическое накопление результатов верификации* решений в базе прецедентов для индуктивного формирования и отладки БЗ.

Разработано множество экспертных систем (ЭС) и систем поддержки интеллектуальной деятельности в разных предметных областях. Выполняемый при их создании *системный анализ* (СА) позволяет формировать в таких системах базы декларативных знаний как самостоятельный компонент. Признается необходимость их *развития* в процессе эксплуатации ЭС, в том числе *возможность включения новых научных результатов* в БЗ. Иногда осуществляется автоматическое накопление результатов верификации решений в

базе прецедентов, выявление в них закономерностей и индуктивное обобщение примеров [Witten, 2005; Поспелов, 1987; Черняховская, 2011].

Однако за прошедшие десятилетия, несмотря на значительные успехи в области теории и технологии создания, ЭС так и не начали внедряться в повседневную практику на уровне целой отрасли.

Альтернативой «классическим» экспертным системам являются системы поддержки интеллектуальной деятельности, основанные на «owl-ориентированных» базах знаний. Такие базы знаний формируются в терминах онтологий соответствующих предметных областей, что позволяет *знаниям быть понятными* специалистам и экспертам предметных областей. Наиболее популярным инструментом их построения является система Protégé. В качестве примеров применения таких знаний в литературе удается найти, в основном, использование их для решения задач классификации (они и признаются наиболее распространенными [Бениаминов, 2008]). Не удается найти информацию о накоплении верифицированных решений задач и методах их использования для обновления БЗ Protégé.

Отсутствует технология такой автоматизации поддержки принятия решений в отрасли, которая бы конструктивно решала *проблему правильности применения знаний* при решении задач интеллектуальной деятельности. Не предложены

механизмы непрерывного усовершенствования декларативных баз знаний.

Новый подход к автоматизации повседневной интеллектуальной деятельности и управления ее качеством возможен *с использованием онтологий предметных областей* [Клещев и др., 2013]. Подход требует детальной разработки всех тех этапов, от которых зависит решение проблемы *поддержки интеллектуальной деятельности и правильности применения знаний* при этом. Первый из них - этап *системного анализа*.

Цель настоящей работы – обсудить особенности выполнения и содержание результатов *системного анализа*, позволяющего осуществить автоматизацию повседневной интеллектуальной деятельности и управления качеством знаний на уровне отрасли.

1. Начальные этапы автоматизации интеллектуальной деятельности отрасли

Деятельность в предметной области может быть рассмотрена как [Клещев и др., 2013]:

- *повседневная*, в том числе интеллектуальная (т.е. применение знаний при решении интеллектуальных задач);
- *контроль принимаемых решений* (электронный документооборот, включающий принятые специалистами решения и результаты верификации этих решений);
- *деятельность по управлению знаниями*, т.е. меры по уточнению, расширению и отладке знаний.

Системный анализ должен обеспечивать поддержку всех этих трех видов деятельности. Вопросы автоматизации *повседневной* неинтеллектуальной деятельности в этом исследовании не рассматриваются, а будут рассмотрены этапы, связанные с *базами знаний*.

Поскольку роль *баз знаний* и способ их организации отличаются от других хранилищ данных, они создаются на базе *онтологии предметной области*, которая должна быть построена на этом «раннем этапе». Назовем здесь *инженерией баз знаний* этап, состоящий из *формирования* каждой базы знаний и управления качеством (сопровождения) каждой базы.

Целесообразно различать роль и результаты системного анализа от концептуального проектирования системы и инженерии ее баз знаний.

Системный анализ состоит из нескольких шагов:

- идентификация интеллектуальных задач в предметной области;
- разработка онтологий для каждой такой задачи;
- постановка каждой интеллектуальной задачи;

- разработка или выбор методов решения и соответствующих им алгоритмов для интеллектуальных задач.

По завершении этих работ возможен переход, во-первых, к *концептуальному проектированию системы автоматизации*, во-вторых, к *инженерии баз знаний*. Следует отметить, что переход к *инженерии баз знаний* возможен только после реализации ряда подсистем для инженерии знаний, выделенных при *концептуальном проектировании*.

Концептуальное проектирование системы состоит из шагов:

- идентификация (сколько и каких потребуется) программных подсистем для автоматизации повседневной «обычной» и для повседневной интеллектуальной деятельности;
- идентификация баз знаний;
- идентификация программных подсистем (подсистем-редакторов, подсистем-индуктивного формирования знаний, подсистем проверки соответствия знаний базе прецедентов) для *инженерии баз знаний*;
- идентификация программных подсистем для обучения (и оценивания) принятия решений;
- построение схемы взаимосвязи всех подсистем в единой системе.

Инженерия базы знаний состоит из шагов:

- формирование и проверка качества первого варианта базы знаний;
- расширение базы знаний и проверка качества нового варианта;
- слежение за появлением в повседневной деятельности такого прецедента, для которого база знаний оказалась недостаточна;
- приведение базы знаний в соответствие имеющимся прецедентам.

2. Этап онтологического инжиниринга

При автоматизации интеллектуальной деятельности на уровне отрасли в качестве *системы* обсуждается *предметная область*, в которой имеются сложные *реальные объекты, процессы и явления*, и над этими объектами специалистами решаются (и требуют поддержки) *интеллектуальные задачи*. Вариант *системного анализа* для разработки интеллектуальных программных систем часто называют *онтологическим инжинирингом*.

2.1. Идентификация интеллектуальных задач

В результате проведения типичного анализа повседневной профессиональной деятельности, подлежащей автоматизации, системные аналитики выявляют виды работ, связанных с обработкой и преобразованием информации, последовательность их выполнения, связь работ по информации и, возможно, зависимость их от хранимой информации.

В случае анализа повседневной интеллектуальной деятельности в такой «схеме задач» обязательно появятся интеллектуальные задачи, их зависимость от типов профессиональных знаний, связи задач по данным (как в примере на рис. 1).

Такая модель может быть представлена на языке графов, что способствует и пониманию, и дальнейшему планированию СА.

2.2. Разработка онтологий

Интеллектуальная задача решается специалистом по отношению к (сложно устроенному) объекту действительности, и при ее решении необходимы профессиональные знания. Поэтому типовая схема моделирования одной произвольной интеллектуальной задачи начинается с шага построения *онтологии предметной области*, охватывающей все профессиональные понятия, используемые при решении повседневных задач [Клещев, 2008].

Онтология предметной области включает структуру вербального описания исследуемого и/или управляемого объекта (например, пациента в медицине) и знаний для принятия решений. Другими словами, строятся *онтология действительности* и *онтология знаний*, связываемые *онтологическими соглашениями* между ними [Клещев, 2008].

Творческому процессу выявления онтологий, методологии которого посвящено немало работ [Клещев, 2008; Загоруйко, 2008; Gavrilova, 2007], сопутствует рутинный («технический») процесс их явного формализованного представления. На сегодняшний день для этого имеются универсальные редакторы, такие как Protégé или двухуровневый редактор *иерархических семантических сетей* (ИСС) [Клещев, 2006]. Следует отметить, что представление структуры профессиональных понятий «сверху вниз» через операции соединения, повторения и выбора, как это традиционно делается при создании словарей данных предметной области в технологии разработки программного обеспечения [Pressman, 2001], представляется наиболее естественным (по сравнению с построением сущностей предметной

области из элементов и отношений, как это делается на owl-языке). Формирование информации для медицинских задач в виде иерархий используется уже давно (the hierarchy of the patient model) [Mittal et al, 1979].

Применение редактора ИСС позволяет отдельно создавать и сохранять все онтологии действительности и онтологии знаний, необходимые для рассматриваемой интеллектуальной задачи. В частности, для задачи медицинской диагностики потребуется создать одну онтологию действительности (структуру истории болезни пациента) и более одной онтологии знаний (структура описания клинических проявлений заболеваний, структура описания знаний о воздействующих факторах).

2.3. Постановка задач

В типовой схеме моделирования одной интеллектуальной задачи после построения онтологии предметной области переходят к *постановке задачи* в терминах онтологии. В случае формирования *постановки интеллектуальной задачи* имеются две отличительные особенности.

Одна - явное указание используемых знаний. При этом в постановке *задачи*, скорее, будут указаны не сами базы, а онтологии этих баз знаний, поскольку профессиональные знания в отрасли с интеллектуальной деятельностью постоянно совершенствуются.

Вторая особенность – формат выходных данных. Поскольку при автоматизации интеллектуальной деятельности не требуется (как правило) получение единственного решения, но требуется ограниченное множество гипотез о решении вместе с объяснением каждой, то формат выходных данных формируется на основе онтологий действительности, и знаний.

Пример структуры постановки задачи медицинской диагностики:

входные данные – заполненная частично *история болезни* (формат - структура ИБ для заполнения паспортных данных, анатомо-физиологических особенностей пациента, жалоб и ведения дневника наблюдений и обследований);

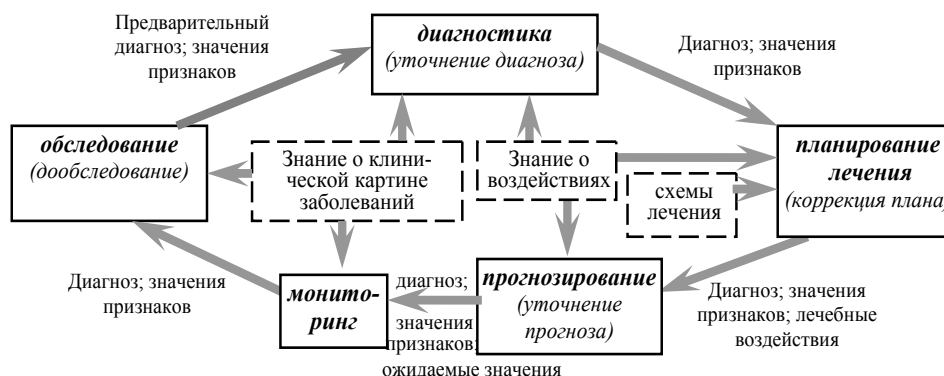


Рисунок 1 – Пример схемы задач для медицинской повседневной деятельности

входные знания - знания о наблюдениях, их возможных и нормальных значениях (формат - структура описания значений признаков при отсутствии заболеваний), знания о клинических проявлениях заболеваний (формат - структура описания вариантов проявлений и периодов развития), знания о воздействующих факторах (формат - структура описания изменения значений признаков при воздействии фактора);

результат - альтернативные диагнозы пациента с их объяснениями (формат каждого объяснения включает название заболевания и объяснение всех наблюдаемых значений каждого признака, подтверждающего клиническую картину заболевания).

Применение редактора ИСС позволяет создать и сохранить онтологию результата, предлагающую необходимый уровень детализации объяснения для понимания специалистом выбранной гипотезы.

2.4. Разработка или выбор методов решения и алгоритмов

Традиционно после того, как выполнена постановка интеллектуальной задачи, осуществляется разработка метода ее решения и доказательство его корректности.

Существенное отличие СА при автоматизации интеллектуальной деятельности состоит в обязательности этого шага, поскольку без гарантий существования реализуемого метода автоматизация решения интеллектуальной задачи нецелесообразна. (В сравнении с этим подбор методов для решения не-интеллектуальной задачи менее сложен и может быть выполнен на более позднем этапе).

В частном случае может быть создан уникальный алгоритм, входные и выходные данные которого задаются в терминах *онтологии действительности*, а составляющие шаги могут использовать термины *онтологии знаний* и проверять выполнение *онтологических соглашений*. Тем не менее, в свое время было предложено несколько общих методов для задач (Taxonomic Classification, Hypothesis Matching и др.) и общий подход декомпозиции задачи на подзадачи, для которых применимы известные методы или дальнейшая декомпозиция [Chandrasekaran, 1992; Gennari et al, 2003]).

Другой опыт дает основания предполагать, что алгоритм решения задачи, для которого доказана корректность, (например, алгоритм медицинской диагностики [Москаленко, 2006]) может быть обобщен до целого класса подобных задач, не зависимо от предметной области.

3. Концептуальное проектирование подсистем для инженерии знаний

При автоматизации деятельности (с использованием знаний) важно обеспечить необходимые средства поддержки для тех, кто

знаниями управляет. Из вышеперечисленных шести этапов концептуального проектирования системы непосредственно к этому имеют отношение два: идентификация баз знаний и идентификация программных подсистем для инженерии знаний.

3.1. Идентификация баз знаний

После того, как в рамках онтологического инжиниринга построены онтология всех знаний для всех задач, можно провести анализ их зависимостей и предложить оптимальную модуляризацию онтологий и баз знаний. Например, в медицине может быть принято решение о соединении (в одной базе) знаний о нормальных признаках пациента со знаниями о клинических картинах заболеваний или, наоборот, об их разделении; может быть принято решение об отделении знаний о возможных значениях всех наблюдений, а также и о возможных названиях самих этих наблюдений в самостоятельную мета-базу, используемую при формировании целевых баз.

Результатом становится решение о том, сколько и каких БЗ должно быть создано в системе, чтобы запланировать для них средства создания, использования и управления.

3.2. Идентификация подсистем-редакторов для формирования каждой базы знаний

Формирование первого варианта базы знаний обычно осуществляется с помощью тех же средств, что и последующее расширение БЗ, в частности, добавление *новых научных* результатов.

Современные инструменты позволяют строить сложные композиции понятий, соответствующие сложно устроенным сущностям и обеспечивать при этом «читабельные» знания [Rogers, 2004]. Распространены универсальные инструменты, например, Protégé, пригодные для формирования любых знаний, представленных как представители классов с конкретными слотами. В их методологии после построения *domain-онтологии* предусмотрена генерация по ней инструмента приобретения знаний, далее – построение с его помощью БЗ и затем – интеграция всех этих компонентов в *knowledge-based system* [Gennari et al, 2003].

Для приближенного к реальности вышеупомянутого представления знаний «сверху вниз» удобен редактор ИСС, который (как и Protégé) автоматически формирует интерфейс редактора знаний после того, как зафиксирована онтология этих знаний с помощью этого же редактора.

Выбор в качестве формы представления знаний ИСС, а в качестве инструмента – соответствующего редактора, позволяет описывать простые и составные признаки, связи между неисправностями (для диагностики), изменчивость признаков во времени и другие варианты сложно устроенных знаний предметной области.

3.3. Идентификация средств программного доступа к содержимому баз знаний

От представления баз знаний зависит и способ доступа к ним, поэтому в концептуальном проекте системы автоматизации уместен «слой» доступа к знаниям, состоящий из операций или запросов достаточно абстрактных. Для доступа к знаниям, хранимым в Protégé, например, импортируют их из Protégé в системы Drools, Eclipse и создают продукционные правила, работающие с «экземплярами как с фактами» [Глибовец, 2012]. Для доступа к каждой БЗ в виде ИСС удобнее всего была бы своя оболочка, реализующая все типы запросов из всех приложений или подсистем, обращающихся к БЗ. Уже на основе анализа разработанных алгоритмов такой набор запросов может быть составлен.

3.4. Идентификация программных подсистем для управления качеством знаний

Построение первого варианта БЗ, предназначенного для использования в реальной повседневной деятельности специалистов, должно быть основано на методах обеспечения качества, и не может (в частности, в таких областях, как медицина) быть проверенным только вручную.

Дальнейшее *управление знаниями* можно представить так. Обнаруживается задача, для правильной поддержки решения которой знаний в системе оказалось недостаточно. Найденное другим способом решение становится *прецедентом для обучения БЗ* и дополнительным тестом для проверки ее качества.

Но для формирования обучающей выборки может быть недостаточно одного прецедента, поэтому в системе *управления знаниями* может быть запланирована подсистема – «рабочее место» эксперта для поддержки анализа верифицированных решений и отбора прецедентов. Требуется и подсистема для проверки соответствия очередного варианта базы знаний - базе прецедентов; такая подсистема нужна и для контроля добавленных вручную знаний.

Для получения нового варианта базы знаний по обучающей выборке разработан метод индуктивного формирования знаний на основе анализа отдельных представителей, описанных единообразно в терминах онтологии [Kleschev et al, 2012] и опытный инструментарий [Смагин, 2013]. Предшественниками таких средств можно считать систему ФИАКР [Соловьев, 1996], интеллектуальные системы типа ДСМ с анализом данных (knowledge discovery) и автоматическим расширением БЗ [Финн, 1999]. Но эти методы относятся к созданию систем, основанных на правилах или на прецедентах.

4. О концептуальном проектировании подсистем поддержки принятия решений

При автоматизации профессиональной деятельности важно обеспечить необходимые средства поддержки специалистов, принимающих решения на основе *управляемых* знаний. Каждая идентифицированная программная подсистема для поддержки интеллектуальной деятельности соответствует ранее созданной постановке задачи, реализует выбранный алгоритм и обращается при этом к подсистемам («слою» или оболочке) доступа к необходимым базам знаний.

Наличие разработанного метода и алгоритма решения задачи обычно означает реализуемость подсистемы поддержки принятия решений. В процессе концептуального проектирования подсистем проявляются их внешние интерфейсы и параллельно могут обсуждаться и формироваться все виды требований к каждой подсистеме, на основании которых начнется разработка.

Поддержка повседневной деятельности соединена с документооборотом. Этапы идентификации программных подсистем для автоматизации повседневной «обычной» и интеллектуальной деятельности могут проводиться независимо, но очевидно, что в общей модели системы они будут связаны информационными потоками или хранилищами, отличными от БЗ.

Заключение

Возможность автоматизации интеллектуальной деятельности на уровне отрасли зависит от наличия методов и инструментов управления качеством знаний. Решающая роль принадлежит результатам *онтологического инжиниринга*, на основе которых осуществляется проектирование семантически совместимых подсистем решения задач и управления качеством баз знаний и их интеграция.

Предлагаемые для представления знаний и их онтологий иерархические семантические сети не только наиболее удобны для работы специалистов с информацией, но и перспективны с точки зрения управления качеством таких знаний и доступа к ним в процессе поддержки решения задач.

Этот подход позволяет внедрить новые механизмы управления интеллектуальной деятельностью, согласованные с существующими в отраслях механизмами. Такое более эффективное управление возможно только при разработке единого комплекса интеллектуальных программ и сопряженных с ними программ для управления качеством знаний.

Работа выполнена при финансовой поддержке грантов РФФИ № 12-07-00179-а «Интернет-сервисы для преобразования семантических сетей» и ДВО РАН № 12-I-П15-03 «Управление интеллектуальными системами».

Библиографический список

- [Бениаминов, 2008] Бениаминов Е.М. Некоторые проблемы широкого внедрения онтологий в ИТ и направления их решений. // Труды Симпозиума "онтологическое моделирование". М.: ИПИ РАН, 2008, с.71-82.
- [Глибовец, 2012] Глибовец Н.Н., Красиков Д.С. Интеллектуализация экспертных систем с помощью онтологий // Information science and computing. Book 28. "Problems of Computer Intellectualization", 2012. 84-90. http://www.foibg.com/ibs_isc/ibs-28/ibs-28-p10.pdf.
- [Загоруйко, 2008] Загоруйко Ю.А. Методы и методологии разработки, сопровождения и реинжиниринга онтологий: труды симпозиума «Онтологическое моделирование». М.: ИПИ РАН, 2008.
- [Клещев, 2006] Клещев А.С., Орлов В.А. Компьютерные банки знаний. Универсальный подход к решению проблемы редактирования информации // Информационные технологии. - М.: Новые технологии. - 2006. - №5. - С. 25-31.
- [Клещев, 2008] Клещев А.С. Роль онтологий в программировании. Часть 1. Аналитика // Информационные технологии. - М.: Новые технологии. - 2008. - №10. - С. 42-46.
- [Клещев и др., 2013] Клещев А.С., Черняховская М.Ю., Шалфеева Е.А. Парадигма автоматизации интеллектуальной профессиональной деятельности. Часть 1. Особенности интеллектуальной профессиональной деятельности // Онтология проектирования. Самара: "Новая техника", 2013. № 3(9). 53-69.
- [Москаленко, 2006] Москаленко Ф.М. Алгоритм диагностики, основанный на реальной онтологии медицины, для многопроцессорной ЭВМ // Доклад III Междун. конф. "Расо 2006". М.: Институт проблем управления им. В.А. Трапезникова РАН, 2006.
- [Поспелов, 1987] Поспелов Д.А. Послесловие / Горелов И.Н. Разговор с компьютером. Психолингвистический аспект проблемы. М.: Наука, 1987.
- [Смагин, 2013] Клещев А.С., Смагин С. В. Комплекс программ для индуктивного формирования баз медицинских знаний в форме, принятой в медицинской литературе // Материалы XX Международной научной конференции студентов, аспирантов и молодых ученых: секция «Вычислительная математика и кибернетика»; МГУ, 9-12 апреля. М.: Издательский отдел факультета ВМК МГУ, 2013. С. 36-37.
- [Соловьев, 1996] Соловьев С. Ю. Методы отладки баз знаний в системе ФИАКР. 1996. http://www.park.glossary.ru/serios/read_053.php#p35;
- [Финн, 1999] Финн В.К. Синтез познавательных процедур и проблема индукции // НТИ. Сер.2.-1999.- №1-2 С.8-44.
- [Черняховская, 2011] Черняховская Л. Р., Федорова Н. И., Низамудинова Р. И. // Интеллектуальная поддержка принятия решений в оперативном управлении деловыми процессами предприятия / Вестник УГАТУ: научн. журнал Уфимск. гос. авиацион. техн. ун-та, 2011. Т. 15, № 2 (42). С. 172–176.
- [Chandrasekaran, 1992] B. Chandrasekaran, T. Johnson, and J.W. Smith, "Task Structure Analysis for Knowledge Modeling," Communications of the ACM, Vol. 33, No. 9, September 1992, pp. 124-136; anthologized in Knowledge Oriented Software Design, ed., J. Cuenca, North-Holland (Amsterdam), 1993, pp. 1-22.
- [Gavrilova, 2007] Gavrilova T. Ontological Engineering for Practical Knowledge Work // Lecture Notes in Artificial Intelligence 4693, Proc. of 11th Int. Conf. Knowledge-Based Intelligent Information and Engineering Systems (KES 2007), Italy, Vietri sul Mare, Springer, 2007. pp. 1154–1162.
- [Gennari et al, 2003] Gennari J.H., Musen M.A., Fergerson R.W., Grossod W.E., Crubezy M., Eriksson H., Noy N. F., Samson W. The evolution of Protégé: an environment for knowledge-based systems development, International Journal of Human-Computer Studies, 2003. 89-123.
- [Kleschev et al, 2012] Kleschev A. S., Smagin S. V. Problems of inductive formation of knowledge in the ontology of medical diagnosis // Automatic Documentation and Mathematical Linguistics. 2012. Vol. 46. Number 1. P. 8-21.
- [Mittal et al, 1979] Mittal S., Chandrasekaran B. Smith J. "Overview of MDX - a medical diagnosis system," Proc. III Annual Symposium on Computer Applications in Medical Care, Washington, D.C., October 1979.
- [Pressman, 2001] Pressman R.S. Software Engineering: Practitioner's Approach. Fifth edition. McGraw-Hill Inc., 2001. 860 p.
- [Rogers, 2004] Rogers J. E. Development of a methodology and an ontological schema for medical terminology. 2004

[http://www.opengalen.org/download/2004-Rogers-Methodology-and-ontological-schema-for-medical-terminology-\(MD-Thesis\).pdf](http://www.opengalen.org/download/2004-Rogers-Methodology-and-ontological-schema-for-medical-terminology-(MD-Thesis).pdf)

[Witten, 2005] Witten, I. H. & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*, second edition, Elsevier: San Francisco, 525 p. <http://home.etf.rs/~vm/os/dmsw/Morgan.Kaufman.Publishers.Weka.2nd.Edition.2005.Elsevier.pdf>.

SYSTEM ANALYSIS CONTENTS FOR INTELLIGENT ACTIVITY AUTOMATION AT BRANCH LEVEL

Kleschev A. , Shalfeeva E.

The Institute of Automation and Control Processes, Vladivostok, Russia

kleschev@iacp.dvo.ru

shalf@iacp.dvo.ru

This article is devoted to features of execution, contents and results of system analysis as a part of complex technology of automation of intelligent activity at the branch level. Performance of all presented stages and receiving the corresponding results allow to implement knowledge-based decision-making support and systematic control of the knowledge quality. Implementation of such system analysis guarantees semantic compatibility of created intelligent systems.

Introduction

The analysis of daily intellectual activity's organization and management of its quality revealed the need of transition to a new paradigm of automation – «branch automation», i.e. development of uniform "cloudy" intelligent system for whole branch, its decision-making and knowledge quality control support.

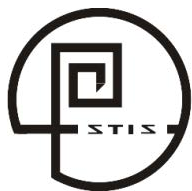
The purpose of this work – to discuss features of execution, contents and results of system analysis, allowing to carry out automation of daily intelligent activity and control of knowledge quality at the branch level.

Main Part

In the main part of the article a system analysis stages' features are presented: identification of tasks in domain, development of ontologies, tasks setting, development of decision methods and algorithms. Also conceptual design of subsystems for knowledge engineering stages are described.

Conclusion

The presented stages of the system analysis are typical for process of automation of intellectual activity. But possibility of such automation at the branch level depends on existence of methods and instruments of quality management of knowledge. The crucial role belongs to results of ontologic engineering on the basis of which all semantic compatible subsystems and instruments are being developed and integrated.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822

ИНТЕРНЕТ-КОМПЛЕКС ДЛЯ ПРЕОБРАЗОВАНИЙ КЛАССОВ СЕМАНТИЧЕСКИХ СЕТЕЙ

Тимченко В.А.

*Федеральное государственное бюджетное учреждение науки Институт автоматики и процессов
управления Дальневосточного отделения Российской академии наук,
г. Владивосток, Россия*

vadim@dvo.ru

В работе представлены архитектура и методы реализации комплекса средств для преобразований классов семантических сетей, включающих эндогенные и экзогенные преобразования, а также преобразования в разных технологических пространствах и между ними. Информационные компоненты программного комплекса хранятся в фонде платформы облачных вычислений IACPaaS. Каждый информационный компонент формируется и поддерживается в актуальном состоянии с помощью соответствующего специализированного редактора, функционирующего как сервис платформы IACPaaS.

Ключевые слова: семантические сети, средства преобразования семантических сетей, представление информации семантическими сетями, структурные проекции.

Введение

В настоящее время обработка информации, включающая в себя задачи по хранению, инженерии и обмену различными видами разнородных данных и знаний между программными сущностями и их компонентами, является одним из важнейших видов деятельности в большинстве теоретических исследований и прикладных областей. Одним из ключевых аспектов обработки информации является преобразование обрабатываемой программными сущностями информации из одной формы (языка, формализма) представления в другую. Различные графовые структуры и семантические сети успешно зарекомендовали себя в качестве модели представления разных видов данных, знаний, онтологий и т.п. и широко используются во многих предметных областях. Более того, несмотря на совершенно разный конкретный синтаксис форматов представления различных видов информации, в абстрактном синтаксисе они могут быть представлены в общем случае типизированными (мульти-)графами [Agrawal A. et al., 2005] [Ehrig K. et al., 2005].

В силу вышесказанного многие содержательные задачи преобразования семантической информации, возникающие в различных технологических пространствах [Kurtev, 2002] [Bezivin, 2005], а также на их стыке, можно сформулировать в терминах преобразования семантических сетей

следующим образом: по исходной семантической сети и спецификации преобразования сформировать новую семантическую сеть, удовлетворяющую этой спецификации.

К настоящему времени разработано множество прикладных и инструментальных средств поддержки, позволяющих эффективно специфицировать и выполнять преобразования информации из одной формы представления в другую при решении различных классов задач в разных технологических пространствах.

Однако большинство разработанных систем рассчитано, как правило, на выполнение только одно вида преобразований – эндогенных или экзогенных или на работу в рамках одного технологического пространства: в них используются специфические для технологического пространства способы и формализмы представления информации, метainформации, а также описания преобразований (либо используются специализированные адаптеры для импорта/экспорта информации в представление, необходимое для использования технологии и методов другого технологического пространства) [Czarnecki, 2006]. Так, например, средства, позволяющие одновременно выполнять преобразования вида “текст – семантическая сеть” и “семантическая сеть – текст”, можно найти только в технологическом пространстве *grammarware* (они предназначены для преобразования программ – *TXL*, *Stratego/XT*). В области преобразования

графов (graphware) графовыми структурами представляют разного вида информацию, однако разработанные программные средства (например, PROGRES, AGG, GrGen.NET) позволяют выполнять только эндогенные преобразования. Помимо этого, большинство средств могут работать только в однопользовательском режиме, либо в локальной сети компании разработчиков, но не могут функционировать в сети Интернет, а значит, информация, обрабатываемая этими системами, хранится на локальных компьютерах и не доступна за рамками этой сети. Это сокращает доступность и масштаб практического применения таких средств, а также замедляет накопление опыта их практического использования.

Таким образом, разные задачи преобразования информации решаются различными методами, единый подход к преобразованию разных видов информации, представленных семантическими сетями, отсутствует. При возникновении новой задачи приходится искать адекватный метод решения и программное средство его поддержки. Поэтому если задача преобразования возникает на пересечении разных технологических пространств, и данный класс задач не охвачен ни одной программной системой, то под этот случай необходимо разрабатывать новое специализированное средство преобразования, либо адаптировать одно из существующих.

В данной работе представлены архитектура и методы реализации комплекса средств для преобразований классов семантических сетей, реализующего метод преобразования классов семантических сетей [Тимченко, 2013], основанный на модели классов семантических сетей и их преобразований, которые являются инвариантными по отношению к технологическим пространствам и позволяют определять как структурные преобразования, так и преобразования из текстового представления информации в структурное (в виде семантической сети) и наоборот [Тимченко, 2012].

1. Функциональность комплекса средств для преобразований классов семантических сетей

Этапы решения задачи преобразования информации, представленной семантическими сетями, предполагают выполнение следующих основных видов деятельности:

- описание класса семантических сетей информации (S) и формирование представления информации в виде семантической сети (S_{inst});
- описание спецификации преобразования класса семантических сетей – структурной проекции (M);
- преобразование информации.

Функциональность по описанию S , M , а также формированию S_{inst} реализуется с использованием соответствующих средств редактирования.

Функциональность по преобразованию информации реализуется с помощью Преобразователя классов семантических сетей, предоставляющего следующие возможности:

- возможность выполнять преобразование как структурного, так и текстового представления одной и той же информации, получать на выходе и сохранять как структурное, так и текстовое представление. В этом случае формирование S_{inst} выполняется Преобразователем классов семантических сетей по текстовому представлению этой информации (а не с помощью средства редатирования);
- возможность работы как в “сетевом” (через Интернет/Инtranет), так и в “локальном” режиме. Последний позволяет пользователям работать с программным средством при отсутствии доступа к серверу, на котором развернута платформа облачных вычислений IACPaaS [Грибова и др., 2011];
- возможность формировать и сохранять графические представления семантических сетей (S и S_{inst});
- возможность протоколирования процесса преобразования, результатом которого является протокол, содержащий результаты, количественные и временные характеристики, полученные в ходе преобразования.

2. Архитектура комплекса средств для преобразований классов семантических сетей

В данном разделе описывается архитектура программного комплекса для преобразования классов семантических сетей на основе описания структурных проекций.

Пользователей программного комплекса можно разделить на две группы:

- разработчики спецификации преобразования;

- прикладные пользователи.

В задачи разработчиков спецификации преобразования входит:

- формирование Базы описаний классов семантических сетей (доступно только в “сетевом” режиме работы);
- формирование Базы структурных проекций (доступно только в “сетевом” режиме работы);
- формирование файлов с описанием классов семантических сетей;
- формирование файлов с описанием структурных проекций.

Для решения этих задач разработчик спецификации преобразования должен знать перечень метасимволов, используемых при описании синтаксических ограничений (S_{sr}), язык для записи выражений в описании структурных проекций, а также уметь работать с редактором семантических сетей и редактором структурных

проекций. Редактор семантических сетей используется для формирования Базы описаний классов семантических сетей в соответствии с моделью для описания классов семантических сетей, а редактор структурных проекций – для формирования Базы структурных проекций в соответствии с моделью для описания структурных проекций.

При формировании файлов – знать формат, в котором эти описания представляются в файлах. Для редактирования этих файлов может использоваться любой стандартный (например, Notepad, WordPad) или специализированный текстовый редактор.

В задачи **прикладных пользователей** входит:

1. формирование Базы семантических сетей – формирование представлений информации в виде семантических сетей и занесение их в базу (доступно только в “сетевом” режиме работы);
2. формирование файлов с текстовым представлением информации;
3. выполнение процесса преобразования семантических сетей представления информации.

Для решения **1-ой** задачи **прикладные пользователи** должны уметь работать с редактором семантических сетей, который в данном случае используется для редактирования Базы семантических сетей. С его помощью пользователь заносит в базу структурное представление информации, формируемое в соответствии с описанием структуры этой информации, которое затем при работе с Преобразователем классов семантических сетей можно выбрать для преобразования.

Для решения **2-ой** задачи достаточно воспользоваться любым стандартным (например,

Notepad, WordPad) или специализированным текстовым редактором для формирования текста и сохранения его в файл.

Для решения **3-ей** задачи используется Преобразователь классов семантических сетей.

Разработчик и сопровождающий программного средства формирует, а также расширяет (при необходимости) модель описания классов семантических сетей и модель описания структурных проекций с помощью соответствующих редакторов. Занимается сопровождением программного средства, если расширение данных моделей влечет за собой необходимость внесения изменений в его исходный код.

На рис. 1 представлена архитектура средства преобразования классов семантических сетей на основе описания структурных проекций как совокупность программных и информационных компонентов. На данной схеме отражены потоки данных между программными компонентами, а Преобразователь классов семантических сетей представлен в виде совокупности своих подсистем.

В архитектуре можно выделить следующие информационные компоненты:

- **модель описания классов семантических сетей** – определяет структуру и формализм для формирования описаний классов семантических сетей информации (S);
- **модель описания структурных проекций** – определяет структуру и формализм для описания структурных проекций (M);
- **база описаний классов семантических сетей** – содержит множество описаний классов семантических сетей информации (S);

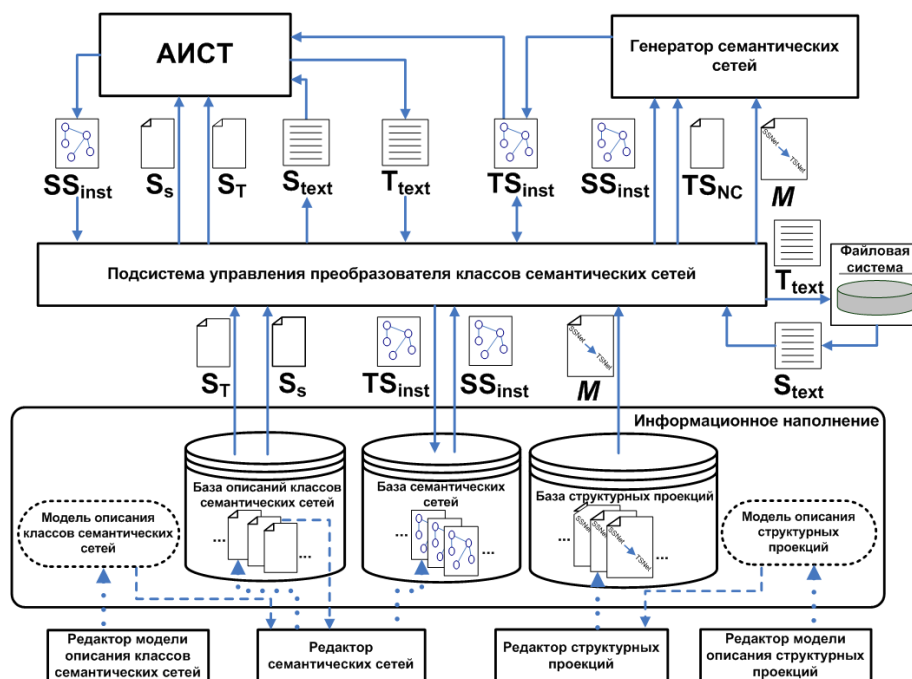


Рисунок 1 – Архитектура программного комплекса

- **база семантических сетей** – содержит множество представлений информации в виде семантических сетей (S_{inst});
- **база структурных проекций** – содержит множество описаний структурных проекций (M);
- под **файловой системой** обобщенно понимаются любые постоянные запоминающие устройства для хранения файлов, доступные на локальном компьютере пользователя, либо на удаленном компьютере, доступном через сетевой интерфейс.

Программными компонентами комплекса средств для преобразований классов семантических сетей являются:

- **редактор модели описания классов семантических сетей** – средство формирования и редактирования модели описания классов семантических сетей;
- **редактор модели описания структурных проекций** – средство формирования и редактирования модели описания структурных проекций;
- **редактор семантических сетей** – средство формирования и редактирования базы описаний классов семантических сетей и базы семантических сетей;
- **редактор структурных проекций** – средство формирования и редактирования базы структурных проекций;
- **преобразователь классов семантических сетей**, состоящий из трех модулей (подсистем):

1. **подсистема анализа и синтеза текстов (АИСТ)** – выполняет синтаксический анализ тестового представления информации ($Text$) и формирует S_{inst} , а также обратную процедуру – синтез $Text$ по S_{inst} ;

2. **генератор семантических сетей** – выполняет собственно преобразование S_{inst} : генерацию представления целевой информации в виде семантической сети (TS_{inst}) на основе представления исходной информации в виде семантической сети (SS_{inst}) по описанию класса сетей понятий целевой информации (TS_{nc}) и описанию структурной проекции (M) описания класса сетей понятий исходной информации (SS_{nc}) на TS_{nc} ;

3. **подсистема управления преобразователя классов семантических сетей** – организует внешний интерфейс с пользователем и внутренние интерфейсы с остальными подсистемами преобразователя, осуществляет загрузку входной информации с внешних источников данных и из информационных баз, а также сохранение выходной информации на внешних носителях.

3. Методы реализации комплекса средств для преобразований классов семантических сетей

В данном разделе описываются методы реализации комплекса средств для преобразования классов семантических сетей на основе описания

структурных проекций: описана реализация программного комплекса как клиент-серверного приложения, методы реализации средств редактирования и подсистем преобразователя классов семантических сетей.

3.1. Клиент-серверная реализация

Для обеспечения возможности работы с программным комплексом через Интернет/Инtranет он реализован как клиент-серверное приложение. На рис 2 представлена его клиент-серверная архитектура.

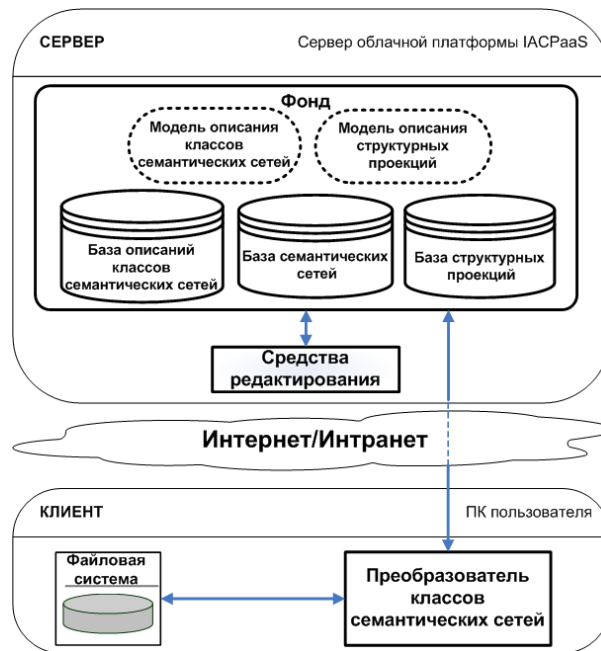


Рисунок 2 – Клиент-серверная архитектура программного комплекса

Все информационные компоненты программного комплекса расположены на **сервере**, на котором платформа облачных вычислений *IACPaaS*, в *фонде* платформы. Фонд представляет собой совокупность единиц хранения – программных и информационных ресурсов различных типов.

Весь фонд разделен на предметные области, а те, в свою очередь, на разделы. Для каждого информационного компонента комплекса средств для преобразований классов семантических сетей на основе описания структурных проекций через административную систему платформы *IACPaaS* зарегистрирован и создан соответствующий ему *информационный ресурс* в фонде.

Все средства редактирования представляют собой облачные сервисы платформы *IACPaaS*, разработанные в соответствии с методологией и технологией проектирования интеллектуальных многоагентных сервисов с использованием облачной платформы *IACPaaS*.

На **клиентском компьютере** пользователя находится Преобразователь классов семантических сетей, который для доступа к информационным компонентам обращается к функциям программного

интерфейса (API) для (удаленного) доступа сервисов к фонду платформы IACPaaS.

3.2. Методы реализации средств редактирования

Функциональность редактора модели описания классов семантических сетей и редактора модели описания структурных проекций реализуется интеллектуальным сервисом редактирования программных и информационных ресурсов различного уровня общности (редактором *IWE*), который позволяет создавать и модифицировать информацию в соответствии с моделью структурированного повторно используемого представления единиц хранения фонда платформы IACPaaS [Орлов, 2006].

В фонде IACPaaS основной формой представления любого программного и информационного ресурса является семантическая сеть понятий. Таким образом, каждый информационный компонент комплекса средств для преобразования классов семантических сетей на основе описания структурных проекций хранится в информационном наполнении в виде сети или совокупности сетей понятий.

Редактор *IWE* используется, также, как инструментальное средство для создания специализированных (прикладных) средств редактирования информационных ресурсов фонда IACPaaS. Чтобы получить специализированный (прикладной) редактор “объектной информации”, необходимо специфицировать метаинформацию и занести ее в фонд IACPaaS. Требуемый специализированный редактор получается подключением такой метаинформации к редактору *IWE* в качестве управляющей информации. Таким образом, управляемый в процессе редактирования метаинформацией, редактор *IWE* становится по существу специализированным (прикладным) редактором “объектной информации” в терминах, специфицированных в метаинформации (рис. 3).



Рисунок 3 – Редактирование структурных проекций с помощью редактора *IWE*

Так, специализированные редакторы для

формирования описаний классов семантических сетей (*S*) и структурных проекций (*M*), ориентированные на разработчика спецификации преобразования, получаются в результате подключения к редактору *IWE* модели описания классов семантических сетей и модели описания структурных проекций соответственно. Специализированный редактор для формирования представлений информации в виде семантических сетей (*S_{inst}*), ориентированный на прикладного пользователя, получается в результате подключения к редактору *IWE* описания класса сетей понятий (*S_{nc}*) в качестве метаинформации. После этого редактор используется для формирования множества *S_{inst}*, причём для каждого элемента этого множества *S_{nc}* является метаинформацией.

3.3. Методы реализации преобразователя классов семантических сетей

Подсистема анализа и синтеза текстового представления информации (АИСТ) преобразователя классов семантических сетей состоит из двух компонентов – модуля анализа и модуля синтеза, реализующих базовые методы преобразования “текст – семантическая сеть”, “семантическая сеть – текст” соответственно [Тимченко, 2013].

Генератор семантических сетей реализует базовый метод преобразования “семантическая сеть – семантическая сеть” [Тимченко, 2013]. Генерация может быть завершена успешно, если каждое правило в *M* определяет описание устройства подсети *TS_{inst}*, которую возможно породить на основе подсети из описания класса сетей понятий *TS_{nc}*. Если же в *M* обнаружено правило, которое определяет описание устройства подсети *TS_{inst}*, которая не может быть порождена на основе подсети из описания класса сетей понятий *TS_{nc}*, то процесс генерации завершается с ошибкой.

Подсистема управления преобразователя классов семантических сетей состоит из трёх компонентов – *Интерфейса пользователя*, *Организатора* и *Отображателя*.

Интерфейс пользователя инкапсулирует реализацию классического WIMP-интерфейса, характерного для оконных операционных систем.

Организатор инкапсулирует основную функциональность подсистемы управления. Он организует внутренние интерфейсы с остальными компонентами и подсистемами программного средства и управляется пользователем через интерфейс. Организатор осуществляет кэширование информации в оперативной памяти, ведет протокол процесса преобразования, а также загружает входные данные и сохраняет выходные данные в файловой системе компьютера пользователя, запоминает и восстанавливает настройки программного средства между сеансами работы.

Отображатель инкапсулирует функциональность по загрузке входных данных из фонда платформы IACPaaS во внутренние структуры данных

преобразователя в оперативной памяти, а также сохранения выходных данных в фонде. Взаимодействие Отображателя с фондом IACPaasS осуществляется через API доступа к фонду. Чтобы добавить поддержку отображения входных и выходных данных в любое другое нужное представление, понадобится расширить только этот компонент подсистемы управления, остальные подсистемы эти изменения не затронут.

В качестве языка программирования всех компонентов Преобразователя классов семантических сетей использовался язык Java. Общий объем исходного кода преобразователя (без учета подключаемых библиотек) ~ 27000 строк.

Заключение

Разработаны методы реализации программного комплекса для преобразования информации, представленной семантическими сетями, на основе описания структурных проекций. Для обеспечения процесса раздельного, независимого (от программных компонентов) сопровождения информации соответствующими специалистами программный комплекс спроектирован с соблюдением принципа разделения программных компонентов и обрабатываемой ими информации, которая должна храниться отдельно и редактироваться независимо от средств ее обработки. Все информационные компоненты выделены в отдельные информационные ресурсы фонда облачной платформы IACPaasS. Для каждого компонента разработан свой специализированный редактор, позволяющий удаленно формировать и поддерживать в актуальном состоянии соответствующий информационный ресурс. Преобразователь классов семантических сетей обеспечивает возможность работы как в “сетевом” (через Интернет/Интранет), так и в “локальном” режиме, а также позволяет формировать и сохранять графические представления семантических сетей.

Показана практическая применимость разработанного комплекса программных средств при решении задач преобразования информации, возникающих в разных технологических пространствах. Для этого была подготовлена выборка примеров задач преобразования информации, которая покрывала бы весь спектр видов преобразований: эндогенных и экзогенных, преобразований внутри технологических пространств и между ними. При этом применяются все базовые типы преобразований: “семантическая сеть – семантическая сеть”, “текст – семантическая сеть” и “семантическая сеть – текст”. Помимо этого, задействуются все три комбинации базовых типов преобразований: “семантическая сеть – семантическая сеть – текст”, “текст – семантическая сеть – семантическая сеть” и “текст – семантическая сеть – семантическая сеть – текст”. Результаты практического применения показали, что разработанный программный комплекс позволяет успешно решать практические задачи.

Работа выполнена при финансовой поддержке РФФИ, проект 12-07-00179-а, проект 13-07-00024-а и ДВО РАН по Программе ОНИТ, проект 12-I-ОНИТ-04.

Библиографический список

- [Agrawal et al., 2005] Reusable Idioms and Patterns in Graph Transformation Languages / Agrawal, A. [et. al.]; // Journal Electronic Notes in Theoretical Computer Science (ENTCS). – Vol. 127. – Issue 1. – 2005. – Pp. 181 – 192.
- [Ehrig et al., 2005]. Model Transformation by Graph Transformation: A Comparative Study / Ehrig, K. [et al.]; // Proceedings of Model Transformations in Practice Workshop, MoDELS Conference. – Jamaica. – 2005. [Electronic resource]. URL: <http://www.inf.mit.bme.hu/FTSRG/Publications/varro/2005/mtip05.pdf> (дата обращения: 26.11.2012).
- [Kurtev, 2002] Kurtev, I. Technological Spaces: An Initial Appraisal / I. Kurtev, J. Bezivin, M. Aksit // Int. Federated Conf. (DOA, ODBASE, CoopIS), Industrial track, Irvine. – 2002.
- [Bezivin, 2005] Bezivin, J. Model-based Technology Integration with the Technical Space Concept / J. Bezivin, I. Kurtev // In Proceedings of the Metainformatics Symposium, Springer-Verlag. – 2005.
- [Czarnecki, 2006] Czarnecki, K. Feature-based survey of model transformation approaches / K. Czarnecki, S. Helsen // IBM Systems Journal, special issue on Model-Driven Software Development. 45(3). – 2006. – Pp. 621 – 645.
- [Тимченко, 2013] Тимченко, В. А. Метод преобразования классов семантических сетей / В. А. Тимченко // Материалы международной научно-технической конференции OSTIS-2013. – Минск БГУИР. – 2013. – С. 81 – 86.
- [Тимченко, 2012] Тимченко, В. А. Модель классов семантических сетей и их преобразований / В. А. Тимченко // Материалы международной научно-технической конференции OSTIS-2012. – Минск БГУИР. – 2012. – С. 63 – 70.
- [Грибова и др., 2011] Облачная платформа для разработки и управления интеллектуальными системами / Грибова В.В. [и др.]; // Материалы международной научно-технической конференции OSTIS-2011. – Минск БГУИР. – 2011. – С. 5 – 14.
- [Орлов, 2006] Орлов, В. А. Компьютерные банки знаний. Универсальный подход к решению проблемы редактирования информации / В. А. Орлов, А. С. Клещёв // Информационные технологии. – 2006. – №5. – С. 25 – 31.

THE INTERNET SOFTWARE TOOL FOR SEMANTIC NETWORK CLASSES TRANSFORMATION

Timchenko V.A.

*Federal State Budget Institution of Science
Institute of Automation and Control Processes Far
Eastern Branch of the Russian Academy of
Sciences, Vladivostok, Russia*

vadim@dvo.ru

The paper presents the architecture and implementation methods of the toolset for semantic network classes transformations. These transformations include endogenous and exogenous transformations as well as transformations in different technological spaces and between technological spaces. Information components of the toolset are stored in the data warehouse of cloud-computing software platform IACPaaS. Each information component is created and supported in an actual state by the appropriate specialized editor, which is running as an internet-service of the software platform IACPaaS.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822

ИЕРАРХИЯ НЕЧЕТКИХ ПОНЯТИЙ ДЛЯ ОПИСАНИЯ ИСТОРИЧЕСКОЙ ЛИЧНОСТИ

Глоба Л.С. *, Островский С.А. *, Попова М.А. **, Стрижак А.Е. **, Терновой М.Ю. *

** Национальный технический университет Украины «Киевский политехнический институт»,
г. Киев, Украина*

lgloba@its.kpi.ua

ternovoy@its.kpi.ua

ostazzz92@gmail.com

*** Институт телекоммуникаций и глобального информационного пространства НАН Украины,
г. Киев, Украина*

sae953@gmail.com

pma1701@gmail.com

В работе представлен подход к построению иерархии нечетких понятий для описания исторической личности. Построение такой иерархии состоит из нескольких этапов, включающих формирование нечеткого контекста на основе литературных биографических источников, последующего применения анализа нечетких формальных понятий, кластерного анализа и затем создания иерархии понятий. Представлен фрагмент иерархии понятий, описывающей украинского поэта Шевченко Т.Г., полученной с использованием описанного подхода. Показан пример использования данного подхода в рамках тематического портала знаний.

Ключевые слова: иерархия понятий, нечеткий контекст, формальный анализ нечетких понятий.

Введение

Как правило, жизнь и деятельность известных исторических личностей описывается в различных биографических книгах, статьях и т.п. Эти источники могут, как пересекаться по содержанию, так и иметь свое особое видение данной личности. С другой стороны, специализированные порталы знаний должны иметь возможность представлять информацию о конкретной личности в структурированном виде [Borovikova, 2012], а также давать возможность описать историческую личность несколькими наиболее характерными ключевыми словами. Одним из возможных вариантов представления такого описания является представление в виде иерархии понятий [Ganter, 1999].

Отметим, что некоторые слова могут в большей степени описывать данную историческую личность, а другие – в меньшей степени. Такого рода неопределенность хорошо описывается при помощи математического аппарата нечетких множеств и нечеткой логики [Заде, 1976], [Новак, 2006]. В данной работе будет рассматриваться процесс

построения описания исторической личности в виде иерархии нечетких понятий.

1. Постановка задачи

Анализ формальных понятий [Ganter, 1999] является методом анализа данных и представления знаний. В работе [Quan Thanh Tho, 2006] данный метод расширен за счет внедрения математического аппарата нечеткой логики и сформулирован анализ нечетких формальных понятий. Это позволило добавить возможность описания информации, обладающей неопределенностью. Анализ нечетких формальных понятий положен в основу описываемого в работе подхода.

Постановку задачи для построения описания исторической личности можно сформулировать следующим образом.

Дано: $D = \{d_i \mid i = \overline{1, n}\}$ – множество книг, статей и т.п., описывающих историческую личность.

Найти: (H, \angle) – иерархия понятий, где H – конечное множество понятий, \angle – отношение частичного порядка на множестве H .

Решение данной задачи состоит в последовательном решении нескольких подзадач:

- построение нечеткого формального контекста исходя из множества документов D ;
- определение множества нечетких понятий;
- построение решетки нечетких понятий;
- выделение кластеров в решетке нечетких понятий;
- формирование иерархии понятий.

Ниже рассмотрено последовательное решение этих задач на примере построения иерархии понятий (таксономий) жизнеописания для украинского поэта и общественного деятеля Шевченко Т.Г.

2. Построение нечеткого контекста для описания исторической личности

Под нечетким формальным контекстом будем понимать тройку $K = (G, M, I = \varphi(G \times M))$, где G - множество объектов, M - множество атрибутов, I - нечеткое множество на домене $G \times M$. Каждому отношению $(g, m) \in I$ соответствует значение функции принадлежности $\mu(g, m) \in [0; 1]$.

В рассматриваемом случае множество объектов совпадает с множеством документов $G = D$. В качестве множества атрибутов будет выступать множество входящих в документы наиболее часто встречающихся слов $M = W = \{word_{ji}\}$, таким образом, что всем словоформам ставится в соответствие одно слово. Тогда нечеткий формальный контекст будет записываться так $K = (D, W, I = \varphi(D \times W))$.

Решение задачи нахождения нечеткого формального контекста состоит из нескольких этапов. Вначале все документы проходят предварительную обработку, и каждому из них ставится в соответствие множество слов с количеством вхождения данного слова в данный документ. Таким образом, для каждого документа d_j мы получим $\{(word_{ji}, number_{ji}), i = \overline{1, m_j}\}$, где $word_{ji}$ - слово, $number_{ji}$ - количество вхождения всех словоформ слова $word_{ji}$ в документ d_j , m_j - количество разных слов с учетом всех словоформ в документе d_j . Данная обработка текста проводилась с использованием системы КОНСПЕКТ [Величко, 2009].

Для построения нечеткого формального контекста для каждого документа выбирается m наиболее часто встречающихся слов. После этого проводится редактирование списка слов, таким образом, что каждому документу d_j вместо начального списка слов $WORD_j$ ставится в соответствие новое множество

$WORD_j^* = \{word_{ji} \mid i = \overline{1, m_j^*}\}$, для которого выполняется условие:

$(\forall l \in L : word_{ji} \in WORD_l) \wedge (|L| \geq t) \wedge (j \in L)$, где

L - множество индексов $WORD_l$, в которые входит $word_{ji}$,

t - заданный порог, определяющий минимальное количество множеств, в которые входит $word_{ji}$.

Поскольку размер разных документов различен, то и количество наиболее употребляемых слов может отличаться в несколько раз. Этот факт необходимо учесть при определении значений функций принадлежности в нечетком формальном контексте. В данной работе предлагается определять значения функции принадлежности в нечетком формальном контексте следующим образом:

$$\mu(d_j, word_{ji}) = \frac{\sum_i number_{ji}}{\max_{i,j} \sum_i number_{ji}} \quad (1)$$

При формировании нечеткого формального контекста в некоторых случаях имеет смысл задать порог на значение функций принадлежности и таким образом удалить не сильно влияющие связи.

Таким образом $W = \bigcup_j WORD_j^*$, $D = \bigcup_j d_j$, а отображение I определяется при помощи определенной в (1) функции принадлежности.

Нечеткий формальный контекст $K = (D, W, I)$, определенный для Т.Г. Шевченко показан в таблице 1. Имена строк соответствуют названиям документов: [Хлыпенко, 2008], [Рыльский, 1964], [Коваленко, 2000], [Меньшиков, 2008], [Бородин, 1984], обозначенным d1, d2, d3, d4, d5 соответственно. Имена столбцов соответствуют наиболее часто встречающимся словам. На пересечении стоит значение функции принадлежности, определяемой при помощи формулы (1).

Таблица 1 – Нечеткий формальный контекст

	Тарас	Кобзарь	Поэт	Стих	Человек
d1	0,72	0,5	0,35	0,35	0
d2	0	0	0,7	0,21	0,28
d3	0,52	0	1	0	0
d4	0	0,31	0	0	0,51
d5	0,35	0	0,89	0	0,29

3. Определение нечетких понятий и построение решетки

Для заданного нечеткого формального контекста $K = (D, W, I)$ и порога доверия T запишем несколько дополнительных понятий. Определим множества:

$$A^* = \{word \in W \mid \forall d \in A: \mu(d, word) \geq T\}$$

для $A \subseteq D$ и

$$B^* = \{d \in D \mid \forall word \in B: \mu(d, word) \geq T\}$$

для $B \subseteq W$.

Тогда нечеткое формальное понятие или просто нечеткое понятие для нечеткого формального контекста K и порога доверия T - это пара $(K_f = \varphi(A), B)$, где $A \subseteq D$, $B \subseteq W$, $A^* = B$ и $B^* = A$. Каждый документ $d \in \varphi(A)$ имеет функцию принадлежности, которая определяется формулой (2):

$$\mu_d = \min_{word \in B} \mu(d, word) \quad (2)$$

Если $B = \emptyset$, то $\mu_d = 1$ для всех документов. При таком определении формального понятия $(\varphi(A), B)$, множества A и B являются объемом и содержанием соответственно.

Таким образом, задача определения нечетких формальных понятий решается путем нахождения множеств A^* и B^* , проверки что $A^* = B$ и

$B^* = A$, и последующего формирования нечетких понятий $(\varphi(A), B)$.

Для формирования решетки нечетких формальных понятий воспользуемся понятием похожести нечетких формальных понятий и алгоритмом построения их решетки, введенным в работе [Quan Thanh Tho, 2006].

Для определенного на предыдущих шагах нечеткого формального контекста и нечетких формальных понятий была получена решетка, которая использована для решения задачи описания исторической личности, показанная на рисунке 1.

4. Выделение кластеров в решетке и построение иерархии концептов

Под кластером понятий для решетки нечетких понятий L и порога подобия T_s будем понимать подрешетку S_L в L , которая имеет следующие свойства:

1. Наибольшее понятие C_S , которое входит в S_L , не совпадает ни с одним из своих суперпонятий.
2. Любое понятие $C \neq C_S$ в S_L должно иметь хотя бы одно суперпонятие $C' \in S_L$, такое, что мера похожести $E(C, C') > T_s$.

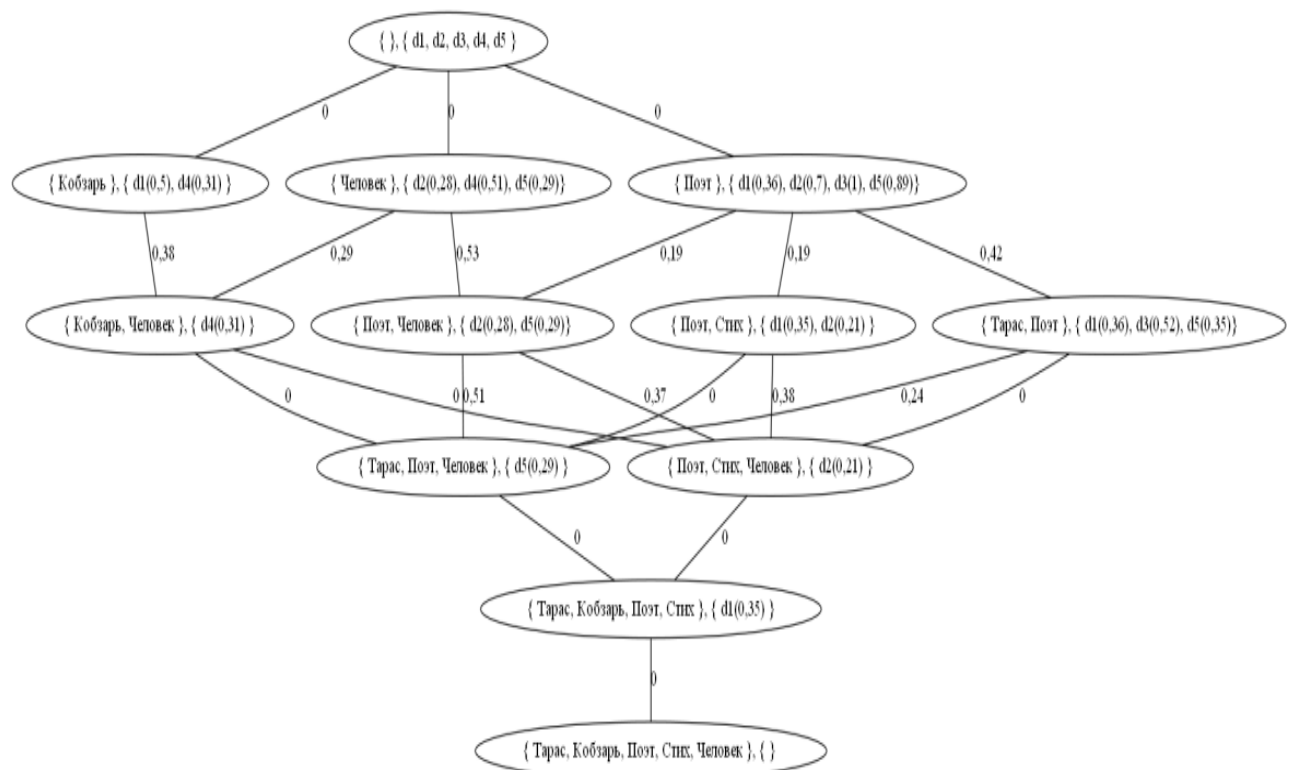


Рисунок 1 – Решетка нечетких формальных понятий

Под суперпонятием C' для конкретного понятия C здесь понимается более общее понятие, находящееся выше в решетке формальных понятий и связанное ребром с C .

Решетка с выделенными на ней кластерами показана на рисунке 2.

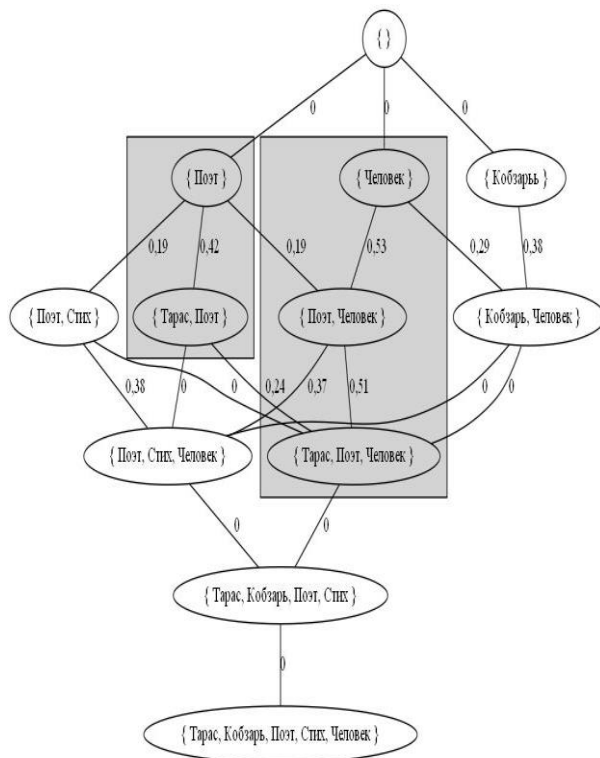


Рисунок 2 – Выделение кластеров в решетке нечетких формальных понятий

На основе полученной после кластеризации решетки нечетких формальных понятий формируется иерархия понятий, показанная на рисунке 3.

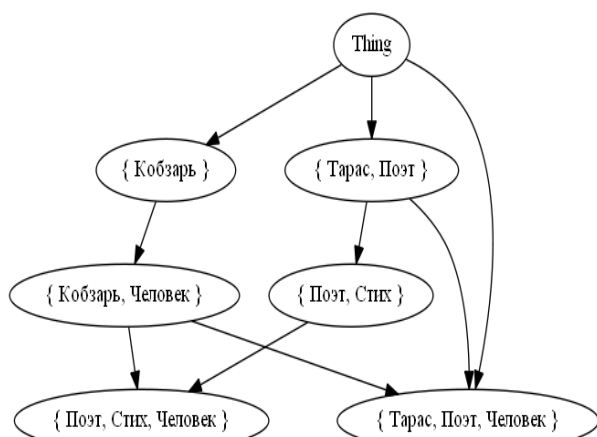


Рисунок 3 – Иерархия понятий

Полученная иерархия, по сути, является кратким описанием исторической личности, которое также несет дополнительную информацию о близости слов, используемых для описания.

5. Использование в порталах знаний

Рассмотрим расширение полученной выше таксономии для портала знаний, связанного с исследованием и анализом фактов жизни и деятельности Т. Г. Шевченко, с осмыслением и выделением главных факторов и причин тех или иных событий, связанных с биографией Кобзаря, а также их возможных последствий, влияние творчества на современников и будущие поколения.

Таксономии в портале знаний, обеспечивают группировку классов объектов предметной области. Таксономии формируются на основе установления отношений между понятиями и классами «часть – целое». Рассмотрим таксономии, вершины которых образуются понятиями, из которых в рамках ограничений решетки нечетких понятий можно сформулировать тавтологию [Мендельсон, 1971]. Для этого на основе определения кластера понятий для решетки нечетких понятий сформулируем следующее правило формирования тавтологий из понятий исследуемых предметных областей: $\langle \text{понятие } S_L \rangle$ включает в себя $\langle \text{понятие } C_S \rangle$. Данное правило обеспечивает формирование таксономий.

Применим функцию принадлежности как образующую для формирования тавтологий из понятий таксономий. Тогда множество всех допустимых тавтологий состоит из утверждений о принадлежности понятий к определенной таксономии.

При обработке информационных источников на основе введенного правила были выделены понятия, объединённые в классы таксономий рассматриваемой предметной области, при чем из понятий сформированных таксономий могут быть сформулированы тавтологии в виде утверждений о событиях жизнедеятельности. Из множества всех таксономий выделены основные, которые приведены ниже.

Тарас Григорьевич Шевченко (места пребывания в ипостаси):

- «Человек»;
- «Поэт»;
- «Писатель»;
- «Творец»;
- «Общественный/Политический деятель».

Маршруты путешествий:

- «Первое путешествие по Украине»;
- «Второе путешествие по Украине»;
- «Ссылка»;
- «Третье путешествие по Украине».

Почтение памяти:

- «Музеи»;
- «Памятники»;
- «Премии».

При этом отдельно взятое понятие, в свою очередь, может являться суперпонятием для других понятий. Например, понятие «Человек» является суперпонятием для понятий «Детство и юность», «Любовь Тараса Шевченко», «Смерть и похороны», а понятие «Писатель» - суперпонятие для понятий «Драматические произведения», «Повести», «Археологические заметки», «Записи народного творчества».

Для улучшения существующих таксономий на портале знаний предоставляется возможность дополнительного поиска по ключевым словам при помощи поисковых машин Интернет. Найденные таким образом электронные информационные источники могут быть обработаны аналогично для построения новых таксономий, которые в свою очередь, могут доопределить или расширить существующие таким образом, чтобы конструкция, начиная с понятия нижнего уровня и заканчивая понятием верхнего уровня таксономии, образовывала логическую цепочку.

Отображение понятий таксономии в портале знаний позволяет объединить понятия места и времени с понятиями фактов и событий в неизвестной до этого комбинации, под новым углом зрения.

Благодаря объединению различных типов баз данных в рамках портала знаний, в таксономии атрибуты объектов могут быть представлены не только в табличном виде, но и в текстовом, а также в виде тематических гиперссылок на распределенные в сети информационные ресурсы.

Заключение

Рассмотренный в работе подход к построению описаний исторических личностей позволяет учесть информацию из различных источников. Описания (иерархии нечетких понятий), полученные в результате, могут использоваться при построении и наполнении специализированных порталов знаний, а также формирования глобальной онтологии исторических личностей.

Данный подход не ограничен применением для описания исторических личностей, а может использоваться, например, для построения описания исторических событий, определения основного направления работы ученого и т.д. Основным условием является семантически схожая тематическая направленность исследуемых информационных ресурсов. Как видно из примера, именно тематическая направленность и влияет на значения функции принадлежности в нечетком формальном контексте, исследуемых распределенных информационных ресурсов.

Среди вопросов, которые необходимо дополнительно исследовать, можно выделить исследование зависимости построенных иерархий понятий от различных вариантов определения функции принадлежности в нечетком формальном контексте, а также развитие рассмотренного

подхода для построения онтологий исторических личностей.

Библиографический список

- [Бородин, 1984] Бородин, В. С. Т.Г. Шевченко. Биография (на украинском языке) / В. С. Бородин, Е. П. Кирилук, В. Л. Смилянская, Е. С. Шаблювский, В. Е. Шубравский // К., Наукова думка. – 588 с.
- [Величко, 2009] Величко, В. Автоматизированное создание тезауруса терминов предметной области для локальных поисковых систем / В. Величко, П. Волошин, С. Свитла // “Knowledge – Dialogue – Solution” International Book Series “INFORMATION SCIENCE & COMPUTING”, Number 15. – FOI ITHEA Sofia, Bulgaria. – 2009. – pp.24-31.
- [Заде, 1976] Заде, Л. Понятие лингвистической переменной и его применение к принятию приближенных решений / Л. Заде // М.: Мир, 1976. 166с.
- [Коваленко, 2000] Коваленко, П. П. Сердце мое трудное, что у тебя болит?..(на украинском языке) / П.П. Коваленко // Черновцы, 2000. – 64с.
- [Мендельсон, 1971] Мендельсон, Э. Введение в математическую логику /Э. Мендельсон // М. Наука, 1971. – 320 с.
- [Меньшиков, 2008] Меньшиков, И.И. Поэтическое слово Кобзаря: Словарь лексических компонентов атрибутивных конструкций (на украинском языке) / И. И. Меньшиков, Н. В. Подмогиляная // К., «Дирекция ФВД». – 212 с.
- [Новак, 2006] Новак, В. Математические принципы нечеткой логики = Mathematical Principles of Fuzzy Logic. / В. Новак, И. Перфильева, И. Мочкрож // Физматлит, 2006. — 352 с.
- [Рыльский, 1964] Рыльский, М. Тарас Шевченко. Биографический очерк (на украинском языке) / М. Ф. Рыльский, А. И. Дейч // К., Держлітвидав України. 1964. – 89 с.
- [Хлыпенко, 2008] Тарас Шевченко в Кыргызстане: Научные, публицистические и художественные материалы на русском, киргизском и украинском языках / Ред-сост. Г.Н. Хлыпенко // Бишкек: Учкун, 2008. – 194 с.
- [Borovikova, 2012] Borovikova, O. I. Methodology for knowledge portals development: background, foundations, experience of application, problems and prospects / O. I. Borovikova, L. S. Globa, R. L. Novogrudska, M.Y. Ternovoy, G. B. Zagorulko, Yu. A. Zagorulko // Bulletin of the Novosibirsk Computing Center, Issue: 34 (2012), pp. 73-92.
- [Ganter, 1999] Ganter, B. Formal Concept Analysis: Mathematical Foundations / B. Ganter, R.Wille // Berlin-Heidelberg, Germany: Springer-Verlag, 1999.
- [Quan Thanh Tho, 2006] Quan Thanh Tho. Automatic Fuzzy Ontology Generation for Semantic Web. / Quan Thanh Tho, Siu Cheung Hui, Tru Hoang Cao // IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 18, NO. 6, JUNE 2006.

FUZZY CONCEPT HIERARCHY DEVELOPMENT FOR HISTORICAL PERSONALITY DESCRIPTION

Globa L.S.^{*}, Ostrovskiy S.A.^{*}, Popova M.A.^{**},
Stryzhak O.Y.^{**}, Ternovoy M.Y.^{*}

**National Technical University of Ukraine 'Kyiv
Polytechnic Institute',
Kyiv, Ukraine*

lgloba@its.kpi.ua
ternovoy@its.kpi.ua
ostazzz92@gmail.com

***The Institute of Telecommunications and Global
Information, National Academy of Science of
Ukraine, Kyiv, Ukraine*

sae953@gmail.com
pma1701@gmail.com

An approach for fuzzy concept hierarchy development for historical personality description is described in the paper. This approach consists of following steps: based on literary biographic sources determination of the fuzzy formal context, fuzzy formal concept analysis, fuzzy conceptual clustering and concept hierarchy generation. A part of generated with the using of the approach fuzzy concept hierarchy for Ukrainian poet T.G. Shevchenko are given. An example of the approach usage in knowledge portal development is described.

Introduction

Problem-oriented knowledge portals must have a possibility to present information about personality in structured manner [Borovikova, 2012] and give an opportunity to describe historic personality in a few key words.

One of the possible alternatives of such description is concept hierarchy [Ganter, 1999].

This paper describes an approach for historical personality description as fuzzy concept hierarchy.

Main Part

Formal concept analysis [Ganter, 1999] is a formal technique for data analysis and knowledge presentation. Quan Thanh Tho and others [Quan Thanh Tho, 2006] proposed a new technique called fuzzy formal concept analysis that combines fuzzy logic and formal concept analysis, in which the uncertainty information is directly represented by a real number of membership value in the range of [0,1]. Fuzzy formal concept analysis is the basis of the approach described in this paper.

Problem formalization for historical personality description follows.

Given:

$D = \{D_i \mid i = \overline{1, n}\}$ – a set of books, papers, another documents about historical personality.

Determine:

(H, \angle) – concept hierarchy, where H - finite set of concepts, \angle - is a partial order on H .

To solve this problem we need to solve several subproblems:

- to determine fuzzy formal context from the literary biographic sources D ;
- to determine fuzzy formal concepts;
- to generate fuzzy formal concept lattice;
- to determine clusters in fuzzy formal concept lattice;
- to generate concept hierarchy.

Generated hierarchy is the brief description of the historical personality. It also has additional information about contextual similarity of words using for description.

Conclusion

Described approach for historical personality description allows one to take into account information from different sources. Obtained descriptions (fuzzy concept hierarchies) can be used in problem-oriented knowledge portal development. They also can be used for generation of global ontology for historic personality.

The application of this approach is not limited only for historical personality description. For example, it can be used for historical events description, for scientist's major research areas determination and so on. The main requirement for using the approach is the thematic similarity of the given information sources.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 007.52:004.81

СРЕДСТВА КОГНИТИВНОЙ ГРАФИКИ В ИНТЕЛЛЕКТУАЛЬНЫХ ОБУЧАЮЩЕ-ТЕСТИРУЮЩИХ СИСТЕМАХ

Янковская А.Е. *, Ямшанов А.В. **, Кривдюк Н.М. **

** Томский государственный архитектурно-строительный университет,
Томский государственный университет,
Томский государственный университет систем управления и радиоэлектроники,
г. Томск, Россия
ayyankov@gmail.com*

*** Томский государственный университет систем управления и радиоэлектроники,
г. Томск, Россия
yav@keva.tusur.ru
knm@kcup.tusur.ru*

В работе приводятся предложенный подход к оценке качества обучения респондентов, математические основы отображения исследуемого объекта в 3-симплексе, архитектура обучающе-тестирующей системы, рассматривается отображение результатов тестирования и динамики обучения в 3-симплексе, приводятся иллюстрирующие примеры. Предлагается осуществлять разбиение респондентов на группы по степени обучаемости. Описывается программная реализация модуля визуализации 3-симплекса. Приводятся преимущества применения 3-симплекса при создании интеллектуальных обучающе-тестирующих систем и перспективы дальнейшего развития модуля визуализации.

Ключевые слова: средства когнитивной графики, интеллектуальные обучающе-тестирующие системы, п-симплекс, программный модуль.

Введение

Когнитивная графика, как научное направление, связанное с принятием и обоснованием решений в интеллектуальных системах (ИС), начала развиваться с 60-х годов XX столетия. Значительный вклад в развитие когнитивной науки внесли R. Axelrod [Axelrod, 1976], R.G. Basaker [Basaker, 1965] Д.А. Поспелов [Pospelov, 1991, 1992, Поспелов, 1996], А.А. Зенкин [Zenkin, 1991, Зенкин, 1996], В.Ф. Хорошевский [Albu, 1990], Б.А. Кобринский [Kobriniski, 1995], А.Е. Янковская [Янковская, 1997, Yankovskaya, 2009, 2010, 2013-2].

Принятие и обоснование решений в ИС с использованием разнообразных когнитивных средств, инвариантных к проблемным областям, ориентированных на пользователей различной квалификации и обладающих различным потенциалом интеллектуальной энергии, представлены в статьях [Янковская, 1994, 1996, 1997, 2000-1, 2000-2, 2000-3, 2011-1, Yankovskaya, 2009, 2010]. Идея сочетания в ИС разнообразных, имеющих и не имеющих отображение в обычной

реальности когнитивных средств при представлении информационных структур, выявлении закономерностей, а также принятии и обосновании решений, а также в целях тестирования и обучения частично отражена в статьях [Янковская, 1994, 1996, 2000-1, 2000-2, 2000-3, Yankovskaya, 2013-1].

Актуальность проводимых нами исследований связана с созданием систем управления качеством подготовки специалистов, осуществляемой как в ВУЗах, так и в других учебных заведениях, включая учреждения по дополнительному образованию и для переподготовки кадров.

В настоящее время, насколько нам известно, тестирование знаний студентов, называемых далее респондентами, проводится на основе безусловных диагностических тестов с использованием среды Интернет в режиме off-line или on-line. Как правило, в обучающе-тестирующих системах весьма слабо развита объяснительная компонента результатов тестирования и организация диалога с респондентом. Кроме того проверка тестирования по конечному результату (ответу), тем более на основе меню и с использованием безусловных

диагностических тестов, не всегда возможна, но всегда примитивна [Янковская, 2011-2].

В связи с вышеуказанным мы предлагаем построение обучающе-тестирующих систем проводить на основе смешанных диагностических тестов, представляющих собой оптимальное сочетание безусловных и условных составляющих [Yankovskaya, 1996], а обоснование результатов обучения и тестирования осуществлять с применением когнитивных средств на основе 2- и 3-симплексов.

О применении основанных на n-симплексе средств когнитивной графики для принятия и обоснования решений, и используемых для интерпретации и обоснования результатов обучения и тестирования, а также для анализа динамики результатов обучения и тестирования в доступной нам литературе не известно.

В данной статье приводится предложенный подход к оценке качества обучения респондентов, математические основы отображения исследуемого объекта в 3-симплексе, рассматривается отображение результатов обучения и тестирования в 3-симплексе, а также визуализация 3-симплекса.

1. Основы оценки качества обучения

Для оценки качества обучения предлагается использовать средства когнитивной графики, основанные на 2- и 3-симплексе в обучающе-тестирующих системах, являющиеся развитием когнитивных средств приведенных в статьях [Янковская, 1996, Yankovskaya, 2013-2].

Далее в процессе изложения будем использовать термин дидактическая единица [ФЦОЗ, 2013]. "Дидактическая единица (ДЕ) - элемент содержания учебного материала, изложенного в виде утвержденной в установленном порядке программы обучения в рамках определенной профессиональной дисциплины или общеобразовательного предмета. Дидактическая единица - одна из предметных тем, подлежащих обязательному освещению в процессе подготовки специалистов, обучающихся по данной дисциплине (предмету)."

Для проверки качества обучения (например, по ЕГЭ и ФЭПО), используется следующий подход:

1. Каждому тестовому вопросу сопоставляется выраженный в баллах вес, зависящий от сложности рассматриваемого вопроса и вклад в конечную оценку респондента. Вопросы для каждой ДЕ разбиваются на группы схожих вопросов. Все вопросы в группе имеют одинаковый вес и любое сочетание вопросов, полученное выбором одного из них по каждой группе, полностью покрывает ДЕ. Тест строится по группе ДЕ. Для каждого респондента генерируется псевдослучайным образом тест, составленный из вопросов из каждой группы. Так как для закрытых вопросов теста с одним правильным ответом существует большая вероятность угадать

правильный ответ, часть вопросов представляется в другой форме, более сложной для случайного угадывания. Примерами таких вопросов могут быть вопросы: с возможностью указать несколько правильных ответов; с полем для ввода ответа; на сопоставление пар, порядка и т.д.

2. Вычисляется суммарное количество баллов для всех правильно отвеченных вопросов, а также количество баллов для частично-правильных ответов, причем количество баллов для таких ответов может быть вычислено по-разному: либо часть баллов, либо ноль. На основе заранее заданных порогов тех или иных ответов суммарное количество баллов переводится в одну из оценок: неудовлетворительно (2), удовлетворительно (3), хорошо (4), отлично (5).

3. Результаты прохождения теста оцениваются как отношение количества баллов, полученных в результате правильного ответа на вопросы теста, к максимально возможному количеству баллов, которое можно набрать, правильно ответив на все вопросы. Кроме того каждое ДЕ, входящее в тест, оценивается по вышеописанной схеме, но учитываются только вопросы, относящиеся к конкретной ДЕ.

4. Динамика результатов обучения (тестирования) отображается на плоскости (рис. 1): по оси X откладывается дата прохождения теста, а по оси Y – оценка результатов прохождения теста (ДЕ). Такое отображение позволяет с достаточной степенью наглядности представить успешность обучения респондента. Отметим, что используется $m+1$ график, где m – количество ДЕ. На первом графике отображается динамика по всем вопросам теста, а на каждом последующем - по вопросам, относящимся к конкретной ДЕ. На рис. 1 величина m равна 3.

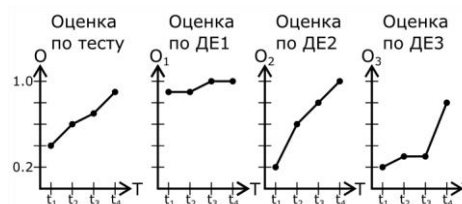


Рисунок 1. Результаты прохождения теста и трех дидактических единиц

Применение средств когнитивной графики, а именно 2- и 3-симплексов, позволяет повысить интерпретируемость и обоснованность результатов тестирования качества обучения.

Оценка результатов обучения и тестирования может быть осуществлена на основе различных подходов к распознаванию образов, включая предложенный в [Yankovskaya, 2013-1] подход, основанный на сочетании нечеткой и пороговой логики. При этом в качестве исследуемого объекта выступает результат тестирования респондента.

2. Математические основы отображения исследуемого объекта в n-симплексе

В основе принятия и обоснования решений в обучающе-тестирующих системах лежит следующая теорема, предложенная в публикациях [Янковская, 2000-3, 2004].

Теорема: для любого набора одновременно не равных нулю чисел a_1, a_2, \dots, a_{n+1} , где n – размерность правильного симплекса, можно найти одну и только одну такую точку, что $h_1:h_2:\dots:h_{n+1} = a_1:a_2:\dots:a_{n+1}$, где h_i ($i \in \{1, 2, \dots, n+1\}$) – расстояние этой точки до i -ой грани [Янковская, 1991, Кондратенко, 1992].

При $n=3$ коэффициент a_i ($i \in \{1, 2, 3, 4\}$) представляет собой степень условной близости исследуемого объекта к i -му образу.

Поскольку 3-симплекс обладает свойством постоянства суммы расстояний (h) из любой точки до его граней и свойством сохранения отношений $h_1/a_1=h_2/a_2=h_3/a_3=h_4/a_4$, то расстояния h_1, h_2, h_3, h_4 вычисляются на основе коэффициентов a_i ($i \in \{1, 2, 3, 4\}$) и операции нормализации из следующих соотношений

$$\left\{ \begin{array}{l} h = \sum_{i=1}^4 h_i \\ h = \alpha \sum_{i=1}^4 a_i \\ \frac{h_1}{a_1} = \frac{h_2}{a_2} = \frac{h_3}{a_3} = \frac{h_4}{a_4} \end{array} \right. , \quad (2)$$

по формуле

$$h_i = \frac{h \cdot a_i}{\sum_{i=1}^4 a_i}, \text{ при } i \in \{1, 2, \dots, 4\}, \quad (3)$$

где α – коэффициент масштабирования.

Данная теорема использовалась в более чем тридцати прикладных интеллектуальных системах и в трех инструментальных средствах выявления различного рода закономерностей и принятия диагностических, организационно-управленческих и классификационных решений в целях принятия и обоснования принимаемых решений.

Далее при изложении будем использовать термин n-симплекс, опустив слово правильный.

3. Архитектура обучающе-тестирующей системы на основе 3-симплекса

Поскольку рамки доклада ограничены, то опустим создание математических моделей для обучающе-тестирующих систем и описание самой системы, а приведем только основные составляющие модули системы и её архитектуру (рис. 2).

Представим последовательность обучения (тестирования) респондента по очередной ДЕ:

1. Производится обучение респондента.
2. Псевдослучайно генерируется тест из набора вопросов ДЕ.
3. Результаты пройденного теста передаются в модуль распознавания образов, который, основываясь на дополнительной информации о вопросах, невидимой для респондента, получает численные значения о степени близости к одному или иному образу. Для интерпретации и обоснования динамики обучения респондента эта информация также сохраняется в базе результатов.
4. Коэффициенты близости к образам являются исходными данными для модуля интерпретации результатов. Они визуализируются с использованием графика и n-симплекса. Преподаватель наблюдает на графиках и 3-симплексе интерпретацию результатов прохождения теста и динамику обучения респондентов, видит закономерности при обучении группы респондентов со сходной динамикой обучения и т.д. Восприятие информации по степени обучаемости респондентов повышается путем применения автоматической кластеризации респондентов по степени обучаемости, что отображено далее на рис. 5.
5. Определяется, имеются ли еще такие ДЕ, по которым не проведено обучение. Если да, то осуществляем переход к пункту 1, иначе аналогичным образом осуществляется тестирование.

6. Если обучение и тестирование пройдено для всех респондентов, то считается, что процедура обучения и тестирования закончена.

Для визуализации и обоснования динамики результатов прохождения теста предлагается использовать 3-симплекс, отображающий четыре образа (результата тестирования): неудовлетворительно (2), удовлетворительно (3), хорошо (4), отлично (5).



Рисунок 2. Архитектура системы тестирования респондентов, интерпретации и обоснования результатов

4. Иллюстрирующие примеры

Пусть в результате обработки теста получены следующие коэффициенты близости к образам: $a_1=0.6$, $a_2=0.6$, $a_3=0.3$, $a_4=0.2$, где a_i - коэффициент близости к i -му образу, $i \in \{1, 2, 3, 4\}$.

Вычисляем четыре коэффициента h_i ($i \in \{1, 2, 3, 4\}$) по формуле 3 для отображения на 3-симплексе, причем, нормирование коэффициентов осуществляется по формуле (3), где величина h - определяющая высоту визуализируемого 3-симплекса, задается пользователем. При $h = 200$ имеем,

$$\alpha = \frac{h}{\sum_{i=1}^4 a_i} = \frac{200}{0.6 + 0.6 + 0.3 + 0.2} = 117.64, \quad (4)$$

$$h_1 = 70.58, \quad h_2 = 70.58, \quad h_3 = 35.29, \quad h_4 = 23.52, \quad (5)$$

Результат тестирования, представленный зеленым кругом, и элементы обучающей выборки, представленные разноцветными точками, соответствующими цветам образов, изображены в 3-симплексе, представленном на рисунке 3.

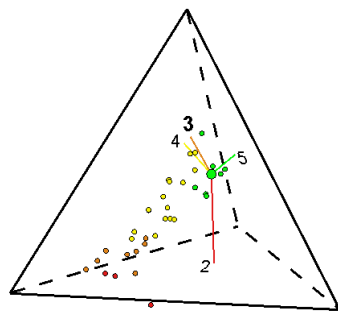


Рисунок 3. Отображение результата тестирования и элементов обучающей выборки в 3-симплексе

На рисунке 4 изображена динамика для трех респондентов. Первый результат (нижняя красная точка) соответствует первому проведению тестирования в начале курса обучения, где все респонденты показали неудовлетворительный результат. Второй результат соответствует проведению тестирования в конце курса обучения, без уведомления респондентов о проведении тестирования. На нем отчетливо отображается дифференциация респондентов по скорости обучения: респонденты а и б продемонстрировали примерно одинаковую скорость обучения, в то время как третий респондент демонстрирует большую скорость обучения (стремление к отличной оценке, способность к обучению). На основе этой информации можно:

1. Прогнозировать дальнейшее повышение уровня знаний респондентов.
2. Разбивать респондентов на различные группы (рис. 5), соответствующие различной степени обучаемости, и предъявлять им задания соответствующей сложности. Такой подход позволит повысить уровень знаний у респондентов

в каждой группе, чем обучение всех респондентов в одной группе. С другой стороны, разбиение респондентов на группы экономичней по стоимостным и временным затратам, чем индивидуальное обучение каждого респондента.

Третий результат соответствует проведению тестирования после уведомления респондентов о проведении тестирования. На рис. 4 видно, что второй и третий респондент получили отличную оценку, но третий респондент показал более твердые знания и более высокий потенциал к развитию, чем второй. С большой достоверностью, этот факт мог быть правильно определен из динамики прохождения двух первых тестирований.

Первый респондент прошел тест только на 4, и ему мало помогла повторная сдача, хотя наблюдается небольшой прогресс по продвижению к более высокой оценке - пятерке. Этот факт также с большой достоверностью, мог быть определен из динамики прохождения предыдущих тестирований.

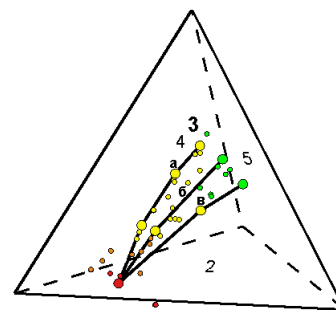


Рисунок 4. Изображение динамики обучения для трех респондентов в 3-симплексе

Существенным преимуществом применения 3-симплекса для обучающе-тестирующих систем является:

1. Большая наглядность отображения динамики результатов обучения и возможность сравнения динамики обучения для разных респондентов
2. Учет весовых коэффициентов для разных категорий вопросов, позволяет достоверно оценивать результаты тестирования.
3. Выявление «анормальных» результатов, то есть результатов со списанными или угаданными вопросами.
4. Выявление и отображение степени обучаемости респондента.
5. Возможность разбиения респондентов на группы по степени обучаемости и его отображение, что наглядно видно на рис. 5, на котором изображен результат разбиения шести респондентов на 2 группы: с хорошей и средней динамикой обучения.

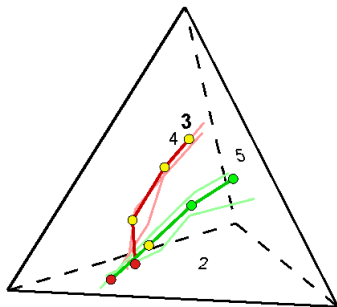


Рисунок 5. Изображение разбиения респондентов по степени их обучаемости на две группы в 3-симплексе

5. Программная реализация модуля визуализации 3-симплекса

В данный момент реализованы прототип модуля визуализации 3-симплекса. Для реализации прототипа был выбран язык C#.

На вход модуля визуализации подается текстовая строка, содержащая описание 3-симплекса и представленная либо файлом на жестком диске, либо областью в оперативной памяти. 3-симплекса описывается на прототипе простого декларативного языка, разработанного нами специально для описания n-симплексов. Данный язык наиболее прост для кодогенерации (для легкого внедрения в программные комплексы) и при этом является достаточно простым и наглядным для понимания и редактирования описания n-симплекса пользователем.

Результатом работы модуля является изображение, отрендеренное с применением GDI+ на Windows Forms. Для 3-симплекса доступна интерактивная смена ракурса обзора, включая задание углов обзора в строке описания n-симплекса.

Алгоритм построения изображения 3-симплекса включает в себя следующие шаги:

1. Грамматический разбор описания 3-симплекса.
2. Создание внутреннего списка примитивов.
3. Преобразование коэффициентов a_i ($i \in \{1, 2, 3, 4\}$) в коэффициенты h_i ($i \in \{1, 2, 3, 4\}$) в координаты x, y, z декартового пространства.
4. Преобразование коэффициентов h_i ($i \in \{1, 2, 3, 4\}$) в координаты x, y, z декартового пространства.
5. Преобразование смещения и поворота координат в декартовом пространстве координат с применением матриц 3D преобразований.
6. Рендеринг 3-симплекса на Windows Forms.

Заключение

Описан предлагаемый нами подход для интерпретации и обоснования результатов обучения (тестирования), основанный на применении 3-симплекса. Даны математические основы отображения исследуемого объекта в n-симплексе. Предложена разработанная нами архитектура обучающе-тестирующей системы. Приведены иллюстрирующие примеры отображения

результатов тестирования, динамики обучения, степени обучаемости респондентов и разбиение их на группы по степени их обучаемости на основе 3-симплексов. Создан алгоритм визуализации изображения 3-симплекса и отображения в нем результатов тестирования, динамики степени обучения. Дано описание программной реализации 3-симплекса с описанием входа и выхода модулей.

Показано существенные преимущества применения 3-симплекса при создании интеллектуальных обучающе-тестирующих систем.

Далее планируется развитие других интерактивных возможностей, облегчающих анализ данных с применением n-симплекса и переписывании кода на язык C++ в целях увеличения производительности и облегчения встраивания разработанного модуля в другие приложения, в том числе в интеллектуальное-инструментальное средство ИМСЛОГ [Yankovskaya, 2003], предназначенное для выявления различного рода закономерностей в данных и знаниях, принятия и обоснования принятия диагностических, организационно-управленческих и классификационных решений с использованием графических, включая когнитивные, средств.

Работа выполнена при частичной финансовой поддержке грантов РФФИ (проекты № 13-07-00373а "Основанное на конвергенции нескольких наук и научных направлений построение отказоустойчивых диагностических тестов и принятие решений в интеллектуальных системах" и № 13-07-98037p_сибирь-а "Создание гибридной интеллектуальной обучающе-тестирующей системы, ориентированной на смешанное образование и обучение") и гранта РГНФ (проект № 13-06-00709 "Структуризация данных и знаний, создание интеллектуальной системы биопсихосоциальной экспресс-диагностики и профилактики депрессии у женщин в репродуктивный период").

Библиографический список

- [Журавлев, 1990] Распознавание образов и анализ изображений / Ю.И. Журавлев, И.Б. Гуревич // Искусственный интеллект в 3-х книгах. Книга 2. Модели и методы: Справочник / Под ред. Д.А. Поспелова. М: Радио и связь. – 1990. – С. 149-191
- [Зенкин, 1996] Знание - порождающее технологии когнитивной реальности / Зенкин А.А. // Новости искусственного интеллекта. – 1996. - № 2. – С. 72-78.
- [Кондратенко, 1992] Система визуализации TRIANG для обоснования принятия решений с использованием когнитивной графики / С.В. Кондратенко, А.Е. Янковская // Тезисы докладов III конференции по Искусственному интеллекту. Том, Тверь, 1992, 152-155
- [Кренделев, 1967] Сравнение геологического строения зарубежных месторождений докембрийских конгломератов с помощью дискретной математики / Ф.П. Кренделев, А.Н. Дмитриев, Ю.И. Журавлев // Доклады АН СССР. – Т. 173, №5, 1967, 1149-1152.
- [Поспелов, 1996] Десять "горячих точек" в исследованиях по искусственному интеллекту / Поспелов Д.А. // Интеллектуальные системы (МГУ). - Т.1, вып.1-4., 1996, с.47-56
- [ФЦОЗ, 2013] Федеральный центр образовательного законодательства // [Электр. ресурс]. Режим доступа <http://lexed.ru/faq/index.phtml?cat=38>, свободный, (15.11.2013).

[ФЭПО, 2010] Федеральный экзамен в сфере высшего профессионального образования // [Электр. ресурс]. Режим доступа <http://www.fepo.ru>, свободный, (05.03.2010).

[Янковская, 1991] Преобразование пространства признаков в пространство образов на базе логико-комбинаторных методов и свойств некоторых геометрических фигур / А.Е. Янковская // Распознавание образов и анализ изображений: новые информационные технологии. Тез. докладов I Всесоюзной конференции. Часть II. – Минск, 1991. – С. 178-181.

[Янковская, 1994] Тестовые распознающие медицинские экспертные системы с элементами когнитивной графики / Янковская А.Е. // Компьютерная хроника. – 1994. – № 8/9. – С. 61-83.

[Янковская, 1996] Графические средства в интеллектуальных обучающих распознающих системах / Янковская А.Е. // Искусственный интеллект в образовании: Труды Междунар. семинара. – Ч. 2. – Казань, 1996. – С. 101-106.

[Янковская, 1997] Принятие и обоснование решений с использованием методов когнитивной графики на основе знаний экспертов различной квалификации / А.Е. Янковская // Известия РАН. Теория и система управления, № 5, 1997, 125-126.

[Янковская, 2000-1] Интеллектуальная подсистема когнитивной графики обоснования принятия решений / А.Е. Янковская, Р.В. Аметов // Информационные системы и технологии (ИСТ-2000). Труды Международной конференции. Том 3. – Новосибирск: изд-во НГТУ, 2000. – С. 542-547.

[Янковская, 2000-2] Графическая визуализация данных, знаний и закономерностей в прикладных интеллектуальных информационных системах / Янковская А.Е., Аметов Р.В., Черногорюк Г.Э. // Искусственный интеллект. Науч.-теорет. журнал. – Донецк. – 2000. – № 2. – С. 279-284.

[Янковская, 2000-3] Отражение образного мышления специалиста в интеллектуальной распознающей системе патогенеза заболевания / Янковская А.Е., Тетенов Ф.Ф., Черногорюк Г.Э. // Компьютерная хроника. – 2000. – №6. – С. 77-92.

[Янковская, 2011-1] Логические тесты и средства когнитивной графики / А.Е. Янковская // (Издательский Дом: LAP LAMBERT Academic Publishing, 2011)

[Янковская, 2011-2] Янковская А.Е. Смешанные диагностические тесты – новая парадигма создания интеллектуальных обучающих и контролирующих систем // Материалы Всероссийской научно-практической конференции "Новое качество образования в новых условиях". – Томск: ТОИПКРО, 2011. – Т.1. – с. 195-203. (ISBN 978-5-903029-30-3)

[Albu, 1990] COGR – Cognitive Graphics System, Design, Development, Application / V.A.Albu, V.F.Khoroshevskiy // Russian Academy of Science Bulletin. Technical Cybernetics - 1990. - № 5 (in Russian)

[Axelrod, 1976] The Structure of Decision: Cognitive Maps of Political Elites / R. Axelrod // Princeton University Press, 1976

[Basaker, 1965] Finite Graphs and Networks: An Introduction with Applications / R.G.Basaker, T.L.Saati // Research Analysis Corp., Mc Graw Hill Company, NY-London-Toronto, 1965

[Kobriniski, 1995] Analysis and use of associative knowledge in medical expert system / B.A.Kobriniski, A.E.Feldman // Artificial Intelligence News - N3, 1995

[Kobriniski, 1996] Why should we take in account imaginary thinking and intuition in medical expert systems / B.A.Kobriniski // Artificial Intelligence – 96. Proceedings of the 5th National Conference with International Participation. Volume II. – Kazan, 1996 (in Russian)

[Pospelov, 1991] How to combine left and right? / D.A.Pospelov, L.V.Litvintseva // News of Artificial Intelligence, N2, 1996. (in Russian)

[Pospelov, 1992] Cognitive Graphics – a window into the new world / D.A. Pospelov // Software products and systems, 1992, 4-6 (in Russian)

[Yankovskaya, 1996] Yankovskaya A.E. Design of Optimal Mixed Diagnostic Test With Reference to the Problems of Evolutionary Computation// Proceedings of the First International Conference on Evolutionary Computation and Its Applications (EVCA'96). – Moscow, 1996. – pp. 292-297.

[Yankovskaya, 2003] IMSLOG-2002 Software Tool for Supporting Information Technologies of Test Pattern Recognition / Yankovskaya A. E., Gedike A. I., Ametov R. V., Bleikher A. M. // Pattern Recognition and Image Analysis, 2003. Vol. 13. No. 4. – pp. 650-657.

[Yankovskaya, 2004] Optimization of a set of tests selection satisfying the criteria prescribed / A.E. Yankovskaya, V.I. Mozheiko // 7th International Conference on Pattern Recognition and Image Analysis: New Information Technologies (PRIA-7-2004). Conference Proceedings, Vol. I, St. Petersburg: SPbETU 2004, 145-148

[Yankovskaya, 2009] Cognitive Computer Based on n-m Multiterminal Networks for Pattern Recognition in Applied Intelligent Systems / A. Yankovskaya, D. Galkin // Proceedings of Conference GraphiCon'2009. – М.: МАКС Пресс, 2009. – pp. 299-300.

[Yankovskaya, 2010] Computer Visualization and Cognitive Graphics Tools for Applied Intelligent Systems / A. E. Yankovskaya, D. V. Galkin, G. E. Chernogoryuk // Proceedings of the IASTED International Conferences on Automation, Control and Information Technology, v.1. – 2010. – pp. 249-253.

[Yankovskaya, 2013-1] Cognitive Tool for the Representation of Test Results of the Blended Education / Yankovskaya A., Semenov M., Semenov D. // Interactive System: Problem of Human-Computer Interaction. - Collection of scientific papers. - Ulyanovsk: USTU, 2013. pp. 305-312.

[Yankovskaya, 2013-2] Cognitive Graphics Tool Based on 3-Simplex for Decision-Making and Substantiation of Decisions in Intelligent System / Yankovskaya A., Krivdyuk N. // Proceedings of the IA STED International Conference Technology for Education and Learning (TEL 2013) November 11 - 13, 2013 Marina del Rey, USA - pp. 463-469.

[Zenkin, 1991] Cognitive Computer Graphics / A.A.Zenkin // М.: Nauka, 1991 (in Russian)

APPLICATION OF COGNITIVE GRAPHICS TOOLS BASED ON THE 3-SIMPLEX IN INTELLIGENT TRAINING-TESTING SYSTEMS

Yankovskaya A.E. *, Yamshanov A.V. **, Krivdyuk N.M. **

** Tomsk State University of Architecture and Building
National Research Tomsk State University
Tomsk State University of Control Systems and Radioelectronics
Tomsk, Russia
ayyankov@gmail.com*

*** Tomsk State University of Control Systems and Radioelectronics
Tomsk, Russia
yav@keva.tusur.ru
knm@kcup.tusur.ru*

The new approach to the quality estimation of respondents education, the mathematical foundations of the object under investigation displaying into 3-simplex, the architecture of training-testing system are proposed. Displaying of testing results and the dynamics of learning with using 3-simplex are considered. Illustrative examples are given. The partitioning of the respondents into groups according to learning speed is proposed. The software realization of the visualization module of 3-simplex is described. The advantages of using 3-simplex in creating intelligent training-testing systems are described. The prospects for further development of the visualization module are given.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514;004.4'275;81'32

ТТР-СИСТЕМА: ИНТЕГРАЦИЯ ЕСТЕСТВЕННОГО ЯЗЫКА И ИЗОБРАЖЕНИЙ

Курбатов С.С. *, Литвинович А.В. **, Лобзин А.П. *, Хахалин Г.К. *

** Научно Исследовательский Центр Электронной Вычислительной Техники, Москва, Россия*

curbatow.serg@yandex.ru

lobzin@rambler.ru

gkhakhalin@yandex.ru

*** ООО "Телеком-Защита", Москва, Россия*

artlav@land.ru

В работе рассматривается задача преобразования естественно языкового текста в изображение. Дается типология современных ТТР-систем. Описаны этапы синтеза графических изображений по текстам на естественном языке в разрабатываемой системе. Формулируются особенности языка алгоритмического описания графики GRASP, обладающего повышенным уровнем представления графических объектов, возможностью взаимодействия с прикладной онтологией и расширяемостью. Приводятся результаты экспериментов по конструированию объектов и по созданию графических композиций

Ключевые слова: лингвистический анализ, онтология, ТТР-система, концептуальный синтезатор изображений.

Введение

Актуальность задачи синтеза графических изображений по текстам на естественном языке (ЕЯ) обусловлена существованием многочисленных предметных областей, в которых главную роль играет наглядность представления текстовой информации. Это относится к изучению родных и иностранных языков, при общении с людьми, не имеющими письменность или неграмотными людьми, с людьми с различными нарушениями речи (с глухонемыми и аутистами) и при реабилитации людей с черепно-мозговыми травмами.

Интересны приложения по реконструкции дорожно-транспортных происшествий (для страховых компаний) и восстановление сцен преступлений по их словесному описанию.

В связи с прикладными областями обучения и общения возникла даже новая парадигма для трансляции [Mihalcea et al., 2009]: трансляция текстов на ЕЯ с помощью картинок (pictorial translation). Эта парадигма основана на присущей людям способности понимать содержание картинки независимо от языка, на котором они говорят. Например, слово 'яблоко' различно в разных языках, но картинка яблока может быть понята

всеми людьми одинаково, если она расположена с множеством написаний слова *яблоко*.

Автоматическая трансляция входного текста в картинки – это нетривиальная задача. И есть ограничения присущие использованию визуальных представлений для целей коммуникации. Первое, существует сложная информация, которая не может быть представлена картинками, например, *“An inhaled form of insulin won federal approval yesterday”* (Ингаляционная форма инсулина вчера получила федеральное одобрение), и требует более адекватного представления, которое может быть получено только кодировкой в ЕЯ. Второе, существует множество понятий, которые имеют уровень абстракции, препятствующий визуальному представлению, например, *политика* или *возрождение*, *материализм* или *схоластика*. И последнее, культурные различия могут приводить к различным уровням понимания определенных понятий. Например, базовое изображение для понятия *дом* может отличаться в разных странах.

1. Типология ТТР-систем

Любая ТТР-система реализует три этапа работы:

- Анализ входного текста.
- Выделение графических описателей, соответствующих представлению входного текста.

- Синтез изображения.

Существующие (и разрабатываемые) ТТР-системы отличаются друг от друга в зависимости от имеющихся средств реализации каждой составляющей (и области приложения).

Если анализатор текста реализует только выделение ключевых слов и фраз, важных для «рисования», то в этом случае графическими описателями являются понятия, соответствующие лексемам или стандартным формам словосочетаний, с которыми связаны заранее выделенные изображения из коллекции изображений. Тогда синтез изображений по тексту заключается в выборе из коллекции соответствующих картинок (фотографий) и их компоновки на поле визуализации в зависимости от строя ЕЯ-предложений.

По этой стратегии строятся дуальные словари, которые комбинируют для основных понятий лингвистическое представление и представление в картинках (например, система PicNet [Mihalcea et al., 2009]). Или системы «усиления» коммуникации, где основной акцент сделан на выделение блоков изображений и их разбивки в зависимости от структуры текста и общих правил расположения для того, чтобы сделать картинку интуитивно «читательной». [Zhu et al., 2007].

Для конвертирования текста на ЕЯ в графическую сцену необходим уже синтаксический и семантический анализ текста. Примером такой системы является система WordsEye [Coyne et al., 2001].

Преобразование текста в динамическую 3D сцену (например, для графического реконструкции дорожно-транспортного происшествия по его естественно-языковому описанию) требует модуль извлечения из текста информации, существенной для визуализации данных, и модуль визуализации, который интерпретирует эти данные и показывает в динамике трехмерную сцену ДТП (система CarSim [Svensson et al., 2002]).

Из современных отечественных разработок интересны подходы, развитые в работах [Усталов и др., 2012; Ермилов, 2007]. В этих подходах декларируется обработка естественного языка, использование онтологий и синтез графики. Информационные ресурсы системы ТТР системы Utkus [Усталов и др., 2012] разделяются на онтологию, тезаурус, правила изображения (depiction rule — правило преобразования семантического элемента в графический описатель), и галерею графических примитивов. Основными элементами онтологии являются акторы (actor — узел-оператор), которые имеют свойства (цвет, размер и т.п.) и имеют методы — функции, определённым образом изменяющие свойства акторов: *падать*, *лежать* и др. Слова и семантические отношения между ними представлены в тезаурусе. К каждому слову в

тезаурусе может быть привязано несколько графических примитивов из галереи. Онтология содержит в себе класс Actor, экземпляры которого привязываются к синсетам тезауруса [синсет, synset — множество синонимов]. Таким образом, для каждого набора синсетов может быть задан экземпляр класса Actor со специфичными для него свойствами. Для экземпляров класса Actor описаны свойства — объекты, привязанные к глагольным синсетам в тезаурусе, выражающие всевозможные отношения между акторами, например, *fall(actor)* и *fallTo(actor1 actor2)*. Для экземпляров класса Actor описаны свойства-значения, выражающие параметры этих акторов, например, *position* и *rotation*.

Интеграция лингвистической трансляции с онтологиями и средствами визуализации является сложной проблемой, требующей комплексного подхода [Курбатов и др., 2011]. Если программные средства визуализации получили достаточное развитие, количество онтологий весьма велико, обработка естественного языка достигла определённого уровня, то интеграция в рамках целостной системы пока далека от концептуальной и технологической зрелости. Отдельные аспекты этой проблемы исследованы в работах [Курбатов и др., 2012; Литвинович, 2013; Khakhalin et al., 2012].

В большинстве ТТР систем изображение составляется путём комбинации готовых изображений из библиотек, подобранных и взаимно расположенных по смыслу исходного естественно языкового описания. Недостатком такого подхода является его фиксированность — в подобных системах нельзя сгенерировать произвольное изображение вне библиотеки готовых картинок (галереи графических изображений) [Goldberg et al., 2009; Rouhizadeh et al., 2012; Yamada et al., 1992; [Coyne et al., 2001]; Li et al., 2008].

2. Система LAT&CSI

Система LAT&CSI (Language Analyzes of Text & Conceptual Syntheses of Images) представляет собой ТТР-систему синтеза изображений по ЕЯ-выражениям через общую, интегрированную онтологию предметной области.

Блок-схема системы LAT&CSI представлена на рис. 1.

Такой выбор схемы определен проектом, интегрирующим системы анализа/синтеза ЕЯ-текста и изображений на основе общей предметной онтологии [Khakhalin et al., 2012].



Рисунок 1 – Схема-схема системы LAT&CSI

В LAT&CSI процесс синтеза изображения по тексту проходит в три этапа:

- лингвистический анализ исходного текста — разбиение текста на составляющие, его морфологическая и синтаксическая разметка, а также получение семантического представления текста;
- генерация набора графических описателей, соответствующих семантическому представлению исходного текста;
- синтез изображения — построение растрового изображения на основе множества графических примитивов, расположенных в соответствии с описателями исходного текста.

Хотя это и стандартные этапы ТТР-системы, однако, развиваемый в данной работе подход отличается большей общностью, наличием онтологии и возможностью действительного синтеза изображений с вкраплениями готовых картинок.

2.1. Лингвистический анализ предметно-ориентированного креолизованного текста

Креолизованный текст характеризуется сочетанием вербализованных и невербализованных (иконических) компонентов, что наблюдается в научных и особенно научно-технических текстах. К иконическим элементам в этих текстах относятся таблицы, символические изображения, формулы и т.п. С учетом такой специфики лингвистический анализ предложений креолизованного текста реализуется полным морфологическим анализом и редуцированным и контекстно-ориентированным семантическим анализатором, который по мере необходимости вызывает синтаксический анализатор.

Для лингвистического анализа креолизованных текстов нет необходимости проводить полный и последовательный синтаксический разбор предложения. Поэтому выбрана семантически-ориентированная архитектура лингвистического анализа, которая подключает соответствующий синтаксический компонент в зависимости от неопределенности и сложности самого текста.

Примером текста на предметно-ориентированном креолизованном естественном языке может быть следующее описание:

Арматура типа В с числом блоков 5. В точке 0 арматуры присоединена цистерна Tank_V через точку 0. К цистерне прикреплен солнечная панель типа С, точки сцепления 1 и 0. В точке 1 арматуры присоединена вторая цистерна Tank_V через точку 0. Через точку 16 с арматурой сцеплен стыковочный элемент типа В, точка сцепки 0. Переходник типа В соединен с арматурой, точки сцепки 8 и 0, соответственно. Через точки 1 и 0 к переходнику подцеплена арматура типа В с числом блоков 2. На эту арматуру подцеплены три стыковочных элемента типа Б, точки сцепки 2:0, 3:0 и 6:0 соответственно.

В рамках данной работы реализован работоспособный вариант лингвистической трансляции, описание которого дано ниже. Однако помимо этого варианта разработаны и реализованы элементы лингвистического обеспечения, непосредственно не включенные в текущую версию трансляции. В то же время это обеспечение является весьма важным для снятия ряда ограничений на используемый язык при предполагаемом дальнейшем развитии системы.

Реализованная версия лингвистической трансляции основана на базе концепции, аналогичной известному методу перифразирования, развитому в лаборатории Апресяна [Апресян и др., 1988] и адаптированной для данной системы.

Версия использует следующие основные ограничения на входной язык:

- каждый объект со свойствами задается именной группой, где имя объекта стоит в начале, а далее следуют свойства со значениями;
- каждое сцепление объектов описывается в отдельном предложении;
- новым считается объект либо впервые упомянутый, либо которому предшествует слово типа “новый” или “другой”. В противном случае объект считается последним ранее упомянутым;
- точки сцепления задаются явно;
- при описании множества объектов, сцепленных с другим, указывается количество объектов, а их типы и точки сцепления задаются непрерывным списком.

Вводится понятие канонического (стандартного) описания объектов и их сцеплений. Исходный текст, описывающий конструкцию, с помощью последовательного (многократного) применения правил перифразирования приводится к каноническому виду. Достоинством такого метода является наглядность процедуры перифразирования, позволяющей на каждом шаге контролировать процесс обработки (выдача диагностики на естественном языке), что существенно облегчает отладку.

2.2. Прикладная онтология

Разрабатываемая онтология характеризуется комплексом отличительных свойств:

- Естественнo-языковой интерфейс для редактирования и доступа.
- Средства автоматизированного пополнения онтологии путем сканирования произвольных текстов ЕЯ.
- Наличие априорной модели высокоуровневых знаний и ее программная поддержка.
- Использование гиперграфовой организации знаний.

Онтологическое представление визуализируемых объектов базируется на языке представления знаний (ЯПЗ), позволяющем описывать сущности с помощью структурированной

семантической сети и гиперграфов [Хахалин, 2009]. На этом языке описываются как предметные сущности (визуализируемые объекты), так и результаты лингвистической трансляции.

Предполагаемая для использования онтология обладает достаточным для целей данного исследования набором функциональных возможностей – создание вершин семантической сети, свойств/значений вершин, отношений между вершинами, операциями редактирования и т.д. Механизмы работы с онтологией позволяют описать в предметно-ориентированной онтологии визуализируемые объекты с учетом наследования свойств, корректности устанавливаемых между объектами отношений и т.д. Инструментальная СУБД (Progress) обеспечивает развитые средства индексации и быстрого создания приложений для отладки декларированной технологической цепочки: естественный язык → онтология → синтез и визуализация объекта.

2.3. Концептуальный синтез изображений

Основная задача интерфейса между онтологией и визуализацией:

- вычленив из онтологического описания информацию, необходимую для визуализации;
- сформировать по этой информации программу на GRASP (см. ниже);
- вызвать интерпретатор GRASP для выполнения собственно визуализации.

Язык алгоритмического описания графики GRASP (GRAphics Situation Planner) [Литвинович, 2012] обладает следующими характеристиками, совокупность которых отличает его от наиболее известных программных средств для визуализации:

- Повышенный уровень описаний, ориентация на элементы вида "объект", "отношение", "свойство".
- Тесное взаимодействие с онтологией.
- Адаптивность и расширяемость.

Подчеркнем, что описание изображения на естественном языке не влечет автоматически наличие высокого уровня представления. "Понятность" описания естественный язык обеспечивает, а вот "высокий уровень" не обязательно. Например, описание изображения на естественном языке в виде множества фраз типа "Точка N125 объекта имеет координаты $X = 2543$, $Y = 5439$, $Z = 8799$, и синий цвет" понятно большинству носителей языка, но оно отнюдь не является высокоуровневым.

Разумеется, это связано с универсальностью естественного языка, допускающего описание изображения на любом уровне сложности. Выбор же блоков, обеспечивающих разумный компромисс между универсальностью и высоким уровнем описания, является весьма нетривиальной задачей. Предлагаемый в языке GRASP подход разбивает эту задачу на две подзадачи, два подуровня,

описываемых разными способами [Литвинович, 2013].

Первый подуровень образует Д-язык, описывающий конструкцию из достаточно крупных объектов, связанных отношением "сцепление". Физически это отношение интерпретируется как прочная скрепленность объектов, образующих единое целое. Это отношение имеет более общий характер, чем отношения типа "находится справа" ("находится над" и т.д.). Действительно, если два сцепленных объекта вращаются как целый объект, то перечисленные выше отношения изменяются во времени, а отношение "сцепление" сохраняется.

При этом эти крупные объекты рассматриваются как экземпляры классов, обладающие набором различных свойств (не выходящих за пределы, определенные классом). Эти свойства могут задавать размеры объектов, особенности формы, цвет, элементы текстуры и т.д. Специальное свойство – "точка сцепки" определяет локализацию и особые характеристики (тип, ориентация) отношения "сцепление". При формировании этих экземпляров выполняется обращение к П-языку, с указанием имени класса и конкретного набора характеристик.

П-язык образует второй подуровень описания и для каждого класса содержит стандартные выполняемые операторы (например, оператор цикла – $\text{for } i = \dots$). В результате выполнения операторов формируется конкретный экземпляр объекта визуализации. Базовые элементы нижнего уровня включают *цилиндр*, *дуга* и т.д., служащих минимальными блоками для формирования экземпляра класса. Детали и нюансы, связанные с группировкой, категориями и особенностями анимации графического объекта, приводятся в описании П-языка.

Наличие этих под-языков позволяет, с одной стороны, предложить весьма высокий уровень описания, а с другой стороны, обеспечить большое разнообразие изображений. Это разнообразие тем больше, чем удачнее выбраны базовые объекты и чем мощнее комбинаторика значений синтезируемых объектов.

GRASP – полноценный язык программирования с синтаксисом, результат интерпретации которого визуализируется средствами OpenGL. Процедурная часть (П-язык) задаёт алгоритм синтеза объекта, по элементам которого он синтезируется; декларативная часть (Д-язык) задаёт связи (сцепки) между элементами и параметры элементов для процедурной части.

3. Эксперименты

Эксперименты велись в двух направлениях, одно связано с синтезом графического объекта по описанию на предметно-ориентированном ЕЯ, другое – с визуализацией разных композиций.

3.1. Конструирование объекта по ЕЯ-описанию

Для экспериментов была выбрана предметная область, содержащая классы объектов, ориентированных на визуализацию. Каждый объект имеет свойства и может находиться в отношении с другими объектами. Объекты и их свойства имеют естественно-языковые описания: *арматура, цистерна, адаптер, переходной элемент, число блоков, радиус* и т.п. Основным отношением между объектами является *сцепление*. Каждый объект имеет несколько точек, через которые осуществляется сцепление. К каждой точке может быть подцеплен только один объект. В онтологии классы объектов представлены концептами, допускающими произвольные свойства и отношения, а не только те, которые доступны для визуализации.

Приведенный в разделе 2.1 текст является одним из реальных тестовых примеров, демонстрирующих возможности лингвистического транслятора и языка GRASP. Синтезированное по этому описанию изображение и текст на GRASP приведены на рис. 2. Ряд других тестовых примеров и возможности лингвистической адаптации приведены в [Сайт результатов экспериментов, 2013]. В частности, средства лингвистической адаптации позволяют пользователю вводить конкретную перифразировку, по которой система формирует новое обобщенное правило перифразирования.

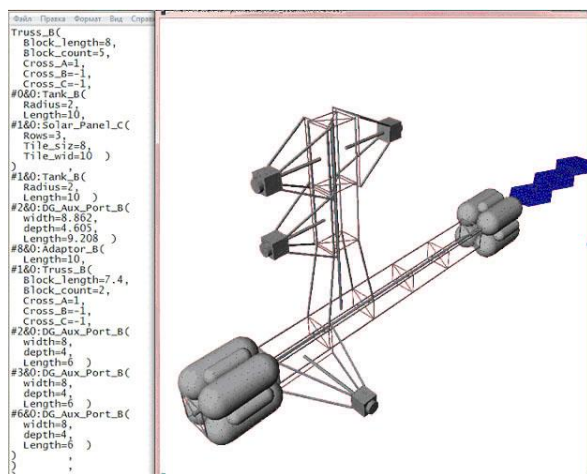


Рисунок 2 – Текст на языке GRASP и результат статической визуализации

3.2. Создание графических композиций

Как отмечалось выше, в большинстве ТТР-систем изображение составляется путём комбинации готовых изображений из библиотек, подобранных и взаимно расположенных по смыслу исходного естественно языкового описания.

Система концептуального синтеза, представленная в данном исследовании, на наш взгляд позволяет снять некоторые ограничения присущие системам трансляции «ЕЯ → картинки».

На рисунке 3 приведен результат экспериментов

по визуализации композиции, которая может соответствовать предложению на естественном языке: *В центре домик, слева чуть впереди одноколесная тачка, справа от домика большая ромашка.*

Эксперименты показывают, что изображения удовлетворяют, по крайней мере, трем требованиям (точность, разнообразие, компактность), предъявляемым для систем преобразующих слова в изображения.

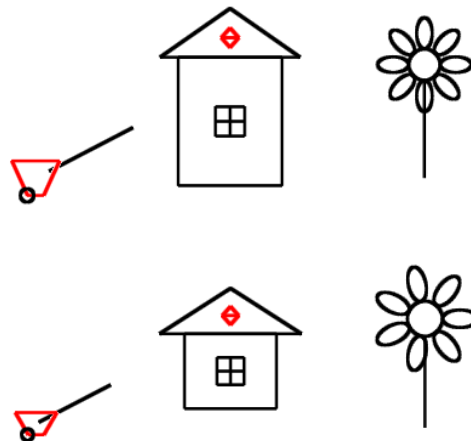


Рисунок 3 – Вариант графической композиции

Каждая графическая композиция представляет собой единое целое (следовательно, к нему можно применять команды сдвига, масштабирования, поворота), что разные композиции могут учитывать различные нюансы ЕЯ-текста (*домик* и *тачка* разных размеров, *цветок* с разным количеством лепестков и т.п.) и что на изображениях нет ничего лишнего. Само же описание на GRASP'e достаточно компактно.

Заключение

Одним из перспективных направлений дальнейших исследований является развитие возможностей данной системы для интеграции с реальными САПР. В этом случае будет интересным как визуализация конструкции по ЕЯ-описаниям из фрагментов технического задания на конструкцию, так и использование онтологии для поиска аналогичных конструкций.

Работа выполнена при поддержке гранта РФФИ, проект № 12-07-00531а.

Библиографический список

- [Coyne et al., 2001] Bob Coyne and Richard Sproat. Wordseye: an automatic text-to-scene conversion system. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pages 487-496. ACM Press, 2001
- [Goldberg et al., 2009] Goldberg A., Rosin J., Zhu X. Dyer C. Toward Text-to-Picture Synthesis //In NIPS 2009 Mini-Symposia on Assistive Machine Learning for People with Disabilities. — 2009.
- [Khakhalin et al., 2012] Khakhalin G., Kurbatov S., Naidenova K., Lobzin A. Integration of the Image and NL-text Analysis/Synthesis Systems. In book: "Intelligent Data Analysis for

Real-Life Applications: Theory and Practice” (Eds. Rafael Magdalena et al.) – USA: IGI Global, 2012.

[Li et al., 2008] Haojie Li, Jinhui Tang, Guangda Li, Tat-Seng Chua. Word2Image: Towards Visual Interpreting of Words. School of Computing, National University of Singapore // MM’08, October 26–31, 2008.

[Mihalcea et al., 2009] Rada Mihalcea, Chee Wee Leong. Toward communicating simple sentences using pictorial representations. Springer Science+Business Media B.V. 2009.

[Rouhizadeh et al., 2012] Masoud Rouhizadeh, Richard Sproat, Bob Coyne. Collecting Spatial Information for Locations in a Text-to-Scene Conversion System. Department of Computer Science, Columbia University, 2012.

[Svensson et al., 2002] Hans Svensson & Ola Akerberg. Development and Integration of Linguistic Components for an Automatic Text-to-Scene Conversion System. Department of Computer Science Lund Institute of Technology, Sweden, August 2002.

[Yamada et al., 1992] Atsushi Yamada et al. Reconstructing Spatial Image from Natural Language Texts // Proc. OF COLING-92, NANTES, AUG. 23-28, Kyoto University, Japan, 1992, 1279-1283.

[Zhu et al., 2007] Zhu X., Goldberg A., Eldawy M., Dyer C. A text-to-picture synthesis system for augmenting communication // Proceedings of The National Conference of the Artificial Intelligence. — 2007. — Vol.22 — P.1590–1595.

[Апресян и др., 1988] Апресян Ю.Д. и др. Лингвистическое обеспечение системы ЭТАП-2 - М.: Наука, 1988.

[Дональд и др., 2005] Дональд Херн, М. Паулин Бейкер. Компьютерная графика и стандарт OpenGL = Computer Graphics with OpenGL. – 3-е изд. – М.: Вильямс, 2005. – 1168 с.

[Ермилов, 2007] Ермилов В. В. Вариационное параметрическое геометрическое моделирование в САПР на основе онтологий, Дис. . канд. техн. наук – Н.Новгород, 2007. - 170 с.

[Курбатов и др., 2011] Курбатов С.С., Литвинович А.В., Хахалин Г.К. Синтез визуальных объектов по естественно-языковому описанию // Труды второй Международной научно-технической конференции «Компьютерные науки и технологии» (КНиТ-2011). – Белгород: БГУ, 2011. С. 595-600.

[Курбатов и др., 2012] Курбатов С.С., Литвинович А.В., Лобзин А.П., Хахалин Г.К. Концептуальный синтез графических образов по структурам прикладной онтологии // Тринадцатая национальная конференция по искусственному интеллекту с международным участием (КИИ-2012), Т. 2.. – Белгород: БГУ, 2012. – С. 120-127.

[Литвинович и др., 2011] Литвинович А.В., Курбатов С.С., Хахалин Г.К. Синтез выражений русского жестового языка по естественно языковому тексту // XIV международная конференция «Речь и компьютер». SPECOM-2011. – Казань, 2011, С. 420-425.

[Литвинович, 2012] Литвинович А.В. Язык описания графических объектов GRASP // еирокомпытеры: разработка, применение, № 10, 2012, С. 26-30.

[Литвинович, 2013] Литвинович А.В. Система синтеза изображений по тексту на естественном языке // Динамика сложных систем — XXI век, № 1, 2013, С. 65-68.

[Сайт результатов экспериментов, 2013] – Электронный ресурс: <http://www.eia--dostup.ru/RFFI-12-14.htm>.

[Усталов и др., 2012] Усталов Д., Кудрявцев А. Применение онтологии при синтезе изображения по тексту, ФГАОУ ВПО «УрФУ имени первого Президента России Б.Н. Ельцина», 2012.

[Хахалин, 2009] Хахалин Г.К. Прикладная онтология на языке гиперграфов // Труды второй Всероссийской Конференции с международным участием "Знания-Онтологии-Теории" (ЗОНТ-09), Новосибирск, 2009, с. 223-231.

TEXT-TO-PICTURE SYSTEM: INTEGRATION OF NATURAL LANGUAGE AND IMAGES

Kurbatov S.S. *, Litvinovich A.V. **,

Lobzin A.P. *, Khakhalin G.K. *

* *Research Centre of Electronic Computing Engineering, Moscow, Russia*

curbatow.serg@yandex.ru

lobzin@rambler.ru

gkhakhalin@yandex.ru

** *“Telecom-Guard”, Moscow, Russia*

artlav@land.ru

Subject of this work is the problem of synthesizing images by description in natural language. We describe the stages in conversion of natural language expression into the corresponding image. Among them is the description and specifics of GRASP algorithmic image description language – a high level image description language, designed for extensibility and ease of integration with applied ontologies. We also describe the results of the object modeling and image composition construction experiments, performed with the said system

Introduction

Importance of image synthesis from natural language description lies in multitude of tasks where the obviousness and appearance of the data, represented by textual information, is important. We provide the typology of TTP systems and outline the stages of the system for image synthesis from natural language texts.

Main Part

Synthesis of image from text consists of three stages. Linguistic analysis of the input text; generating a set of graphical descriptors, corresponding to the semantic presentation of the input text; constructing an image based on a set of graphical primitives, positioned based on the descriptors from the input text.

Implemented version of the linguistic analysis is based on the known concept of rephrasing, adapted for the given task.

Ontological presentation of the synthesized objects is based on a knowledge representation language that allows object description in structured semantic networks and hypergraphs.

An algorithmic graphics description language, GRASP, was developed for the image synthesis subsystem. It converts the part of the ontological description that is graphically interpretable into an intermediate vector representation, compatible with OpenGL graphics system. Procedural part of GRASP describes a parametric algorithm for synthesizing an element; the declarative part defines the relations between elements, and their parameters.

In the experimental section we provide the results for modeling an object and for image composition.

Conclusion

We consider intellectual-assist virtual prototyping to be one of the promising applications for this class of systems and research.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514; 004.932

РАЗБОР ГРАФИЧЕСКИХ ИЗОБРАЖЕНИЙ ОНТОЛОГИЧЕСКИМ КОМПОНЕНТОМ ГИБРИДНОЙ СИСТЕМЫ РАСПОЗНАВАНИЯ

Хахалин Г.К., Курбатов С.С., Лобзин А.П.

Научно-Исследовательский Центр Электронной Вычислительной Техники, Москва, Россия

gkhakhalin@yandex.ru

curbatow.serg@yandex.ru

lobzin@rambler.ru

В работе рассматривается онтологический компонент гибридной системы анализа изображений. В качестве базы знаний для этого компонента служит прикладная онтология, которая представлена на языке семантического гиперграфа.

Ключевые слова: гибридная система; концептуальный разбор изображения; предметная онтология; семантический гиперграф.

Введение

Проблематика машинного зрения столь привлекательна для современных исследователей по той причине, что аппаратные возможности, предоставляемые в данной области последними достижениями электроники и вычислительной техники, достигли такого уровня, что они уже во многом приближаются к «техническим характеристикам» глаза человека [Визильтер и др., 2007]. Но главная задача в машинном зрении – задача «понимания» изображений – сегодня еще далека от своего окончательного решения.

При решении задачи понимания изображений необходимо сместить цели распознавания от только наименования объектов к их структурно-логическим описаниям. Это позволит не просто именовать знакомые объекты, но также сообщать неизвестные аспекты известных объектов; говорить что-либо о неизвестных объектах; и обучать распознавать новые объекты на небольшом количестве зрительных примеров или вообще при их отсутствии [Farhadi et al., 2009].

Для процессов такой обработки изображений даже введен термин «разбор изображений» (image parsing). В настоящее время большинство систем разбора изображений [Endres et al., 2013; Shrivastava et al., 2011] реализуется с помощью одной системы распознавания статистического типа, в которой используются кроме дискриминантных признаков семантические свойства объектов. Последние

характеризуют не только различия одного класса объектов от другого, но и их сходство.

В области машинного зрения принято выделять следующие основные этапы обработки данных: предобработка изображений; сегментация; выделение геометрической структуры; определение относительной структуры и семантики. Связанные с этими этапами уровни обработки обычно называются соответственно: обработка нижнего уровня, среднего уровня, высокого уровня. В то время как алгоритмы обработки нижнего уровня (кодирование, сглаживание, фильтрация) могут рассматриваться как хорошо проработанные и детально изученные, алгоритмы среднего уровня (сегментация) продолжают сегодня оставаться центральным полем приложения инженерных и исследовательских усилий. Методы обработки высокого уровня, относящиеся собственно к «пониманию (разбору) изображений», находятся еще в начальной фазе развития и по-прежнему представляют собой «вызов» для сообщества исследователей в области компьютерного зрения и искусственного интеллекта [Визильтер и др., 2007].

Исходя из различной проработанности этих этапов, оптимальной нагруженности и невозможности на пиксельном уровне в общем случае решить задачу априорной сегментации, в данной работе предлагается гибридная система понимания (разбора) изображения, состоящая из двух компонентов: PR-система (Pattern Recognition) для распознавания «непроизводных» элементов при использовании эффективных дискриминантных (числовых) методов, и KB-система (Knowledge

Based) для распознавания композиционных объектов в целом, обладающих существенной структурной информацией, где ведущая роль принадлежит структурным (символьным) методам. Типология методов распознавания и обоснованность гибридного подхода представлены в [Курбатов и др., 2012b; Хахалин и др., 2013].

Под задачей понимания (разбора) изображения подразумевается описание распознаваемых объектов и их расположение на изображении в терминах и в структурах языка прикладной онтологии. Основным результатом разбора является описание зрительной ситуации, состоящей из экземпляров распознанных классов объектов (с означенными характеристиками) и отношений между ними.

1. О гибридной системе анализа изображений

На входном изображении в общем случае задается не один образ, а целая ситуация (композиция) и ее надо описать в терминах взаимосвязей означенных структур классов объектов.

В качестве модельной предметной среды рассматривается мир плоских двумерных изображений объектов: «детские» рисунки и планиметрические фигуры (*кораблики, домики, ромашки, паровозики, зайцы* и т.п. и *прямоугольники, трапеции, окружности* и т.п.).

PR-система непосредственно работает с реальным изображением. KB-система работает с моделью изображения (по результатам работы PR-системы), но с мощным языком символьного (концептуального) описания среды. Они работают последовательно с возможностью управления процессом распознавания на нижнем уровне с концептуального уровня, реализуя т.н. «эксклюзивный» анализ (см. ниже). Функциональная схема двухуровневой системы анализа изображений представлена на рис. 1.

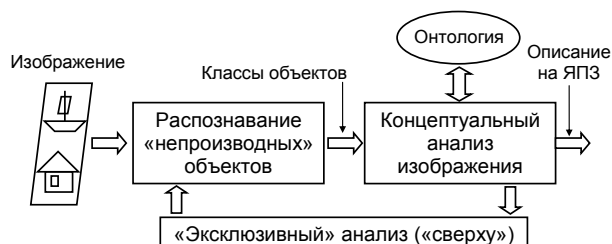


Рисунок 1 – Схема двухуровневой системы разбора изображений

Данная гибридная система разбора изображений не является изолированной, а представляет собой элемент интегральной системы, в которую входят дополнительно системы концептуального синтеза изображений и анализа и синтеза текстов естественного языка [Курбатов и др., 2012b]. Взаимодействие этих систем осуществляется на основе общей прикладной онтологии [Khakhalin et al., 2012].

Для иллюстраций в выбранной предметной области мы используем реальную PR-систему MyScript Notes [MyScript Notes, 2007]. Эта система решает стандартные задачи предварительной обработки изображений, включая кодирование, сглаживание, фильтрацию, сегментацию и т.д., избавляя от этих процедур KB-систему. На вход программы MyScript Notes подаются нарисованные от руки геометрические рисунки. На выходе получаем для каждого класса распознаваемых объектов: имя класса, вероятность распознавания и значения геометрических параметров, привязанных к полю изображения (координатной сетке). Кроме того дается альтернативная информация распознавания, когда приводятся возможные другие кандидаты на распознавание данного объекта с соответствующей вероятностью.

Результатом распознавания будет список (его фрагмент дан на рис. 2), где представлены имена предполагаемых классов объектов с указанием вероятности распознавания и конкретные характеристики экземпляров объектов, привязанные к системе координат изображения (подчеркнутые имена классов имеют максимальную вероятность).

```
rectangle 0.753047 625.545 395.106 764.597 390.574 777.322 780.953 638.269 785.485
parallelogram 0.752861 624.92 390.086 764.34 396.322 777.918 785.459 638.498 779.224
rectangle 0.894784 877.211 785.921 876.636 587.349 1391.45 585.859 1392.02 784.432
parallelogram 0.89469 880.424 785.418 872.933 586.889 1388.14 586.386 1395.63 784.915
rectangle 0.956212 1440.37 1090.52 584.024 1099.19 580.978 798.668 1437.32 789.989
parallelogram 0.95617 1445.39 1091.08 589.294 1098.52 575.977 798.106 1432.07 790.666
circle 0.67984 1281.93 1187.42 75.2383
ellipse 0.679622 1281.93 1187.42 78.6176 71.8591 -1.4689
arc 0.679404 1281.93 1187.42 78.6176 71.8591 -1.4689 0 6.28319
circle 0.609414 746.45 1192.97 82.0498
ellipse 0.609176 746.45 1192.97 83.6068 80.4929 -0.325574
arc 0.608938 746.45 1192.97 83.6068 80.4929 -0.325574 0 6.28319
circle 0.609228 1024.86 1177.54 75.0216
ellipse 0.60899 1024.86 1177.54 80.0398 70.0035 -0.590248
arc 0.608752 1024.86 1177.54 80.0398 70.0035 -0.590248 0 6.28319
```

Рисунок 2 – Фрагмент результатов распознавания PR-системой

Результаты распознавания композиции с привязкой к координатной сетке приведены на рисунке 3. На нем представлены распознанные объекты (некоторые *прямоугольники, треугольники* и др.), неправильно распознанные объекты (другие *прямоугольники, эллипсы* и др.), и не распознаваемые на этом уровне классы объектов (*домик, кораблик, паровозик, зайчик* и др.).

Для нераспознанных и нерасознаваемых PR-системой объектов как раз и служит верхний уровень гибридной системы – KB-система.

2. KB-система

На вход KB-системы подается результат распознавания «непроизводных» объектов PR-системой (имена объектов с конкретными характеристиками на координатной сетке).

Основным результатом работы KB-системы является описание на языке представления знаний, состоящее из экземпляров распознанных классов объектов и отношений между этими экземплярами.

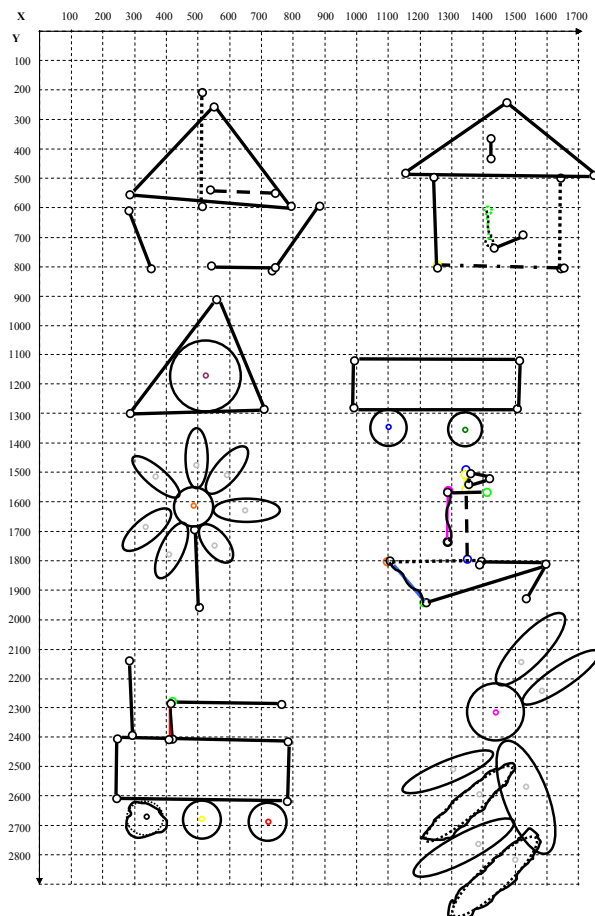


Рисунок 2 – Результаты распознавания композиции фигур в координатной сетке

Поскольку система работает в комплексе, результат может быть также представлен в виде:

- перечня имен всех «понятых» (распознанных) объектов композиции;
- текста на естественном языке (какие объекты и в каких отношениях друг с другом присутствуют на изображении);
- изображения, полученного с помощью системы концептуального синтеза изображений с теми же самыми характеристиками объектов и/или «аналогичного» с характеристиками тех же классов, но со значениями, не выходящими за пределы этих классов.

2.1. Структура прикладной онтологии

Прикладная онтология представлена в виде семантического гиперграфа со структурами, каждая из которых соответствует обобщенному описанию некоторого класса объектов, и взаимосвязями с другими классами объектов. Фрагмент прикладной онтологии в виде семантического гиперграфа представлен на рис. 4., где большие кружки – концепты онтологии (объекты среды), тонкие стрелки – родовидовые связи концептов, маленькие кружки и тонкие стрелки – свойства объектов, замкнутые области – описание структур объектов, толстые стрелки – связь типа «часть-целое» («входит_в_структуру»). Каждая структура объекта представляет собой подгиперграф, отражающий

взаимоотношения его частей. Вершины и дуги (ребра) онтологии на языке семантического гиперграфа помечаются цепочками символов некоторого алфавита (в данном случае, словами русского языка) для различения вершин (концептов) и связей между ними (отношений). На рисунке дана только часть разметок.

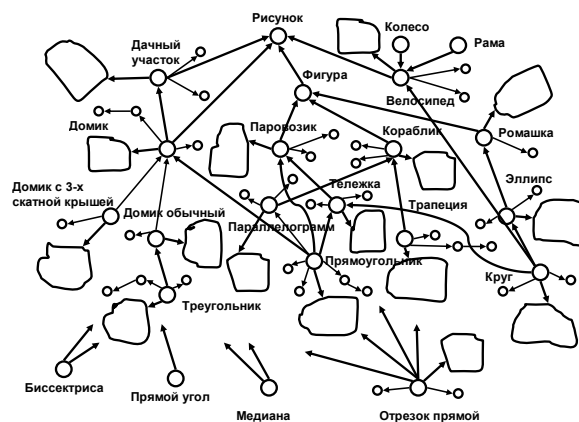


Рисунок 4 – Фрагмент прикладной онтологии

В прикладной онтологии представлены понятия нижнего уровня (*точка, треугольник* и т.д.), дублирующие классы PR-системы, для исправления ошибок распознавания и для анализа тех классов, которые PR-системой не распознаются в принципе (например, *трапеция*), и структурные образы рисунков, составленных из классов PR-системы (*кораблики, домики, ромашки* и т.д.).

2.2. Функции концептуального компонента

КВ-система выполняет следующие функции по концептуальному разбору изображения:

- представление результатов работы PR-системы в виде концептов «непроизводных» объектов с означенными характеристиками (*точки, отрезки прямых, прямоугольники, окружности, дуги* и т.д.) – получение «сырого» графа изображения с привязкой к координатной сетке;
- «вычисление» путей в онтологии для формирования списка гипотез на сопоставление (по связям «вх_в_структуру», «род-вид»), т.е. формирование ранжированного списка гипотез;
- сопоставление «сырого» графа изображения с графами структур комплексных объектов прикладной онтологии (поиск полного либо частичного изоморфизма);
- вычисление параметров «ареала» (области) по структурному описанию объекта из онтологии для проведения поиска на изображении гипотетического непроизводного объекта с помощью PR-системы;
- подтверждение результатов сопоставления, осуществляемое на реальном изображении при посредстве «эксклюзивного» анализа.

2.3. Онтологический разбор композиции

КВ-система первоначально преобразует

результаты, полученные PR-системой, в граф, вершины которого помечены именами классов производных объектов. То есть задается модель изображения в виде графа производных понятий со своими атрибутами, которые может распознать PR-система (например, *отрезок прямой* с координатами концевых точек и длиной отрезка; *окружность* с координатами центра окружности и радиусом).

По этому графу составляется ранжированный список производных понятий $L_{v1} (v_1, v_2, \dots, v_k)$, где на первых местах стоят более «крупные» понятия (*треугольники, эллипсисы, квадраты* и т.д.), а на последних – *точки, отрезки, дуги и произвольные кривые (drawing)*. Ранг определяется иерархией прикладной онтологии.

Для выбранного понятия v_i с наибольшим рангом по онтологии формируется список сложных, комплексных понятий $L_{s1} (h_1, h_2, \dots, h_n)$, в структуру которых входит выбранное понятие (используются онтологическое отношение «вх_в_структуру»). Выбираем из этого списка гипотезу h_1 . В онтологии с этим именем выделяем структуру онтологического понятия и частично означиваем ее по атрибутам производного понятия v_1 . По значениям этих атрибутов с учетом числовых отношений в структуре вычисляем ареал (область) на координатной сетке, в котором могут присутствовать другие производные объекты данной онтологической структуры.

По списку L_{v1} составляем список производных понятий, которые попадают в этот ареал. Тем самым из модельного графа изображения выделяем некоторый подграф. И этот подграф используется для сопоставления со структурой гипотезы h_1 (граф структуры онтологического понятия).

Если граф структуры онтологического понятия полностью сопоставлен хотя бы с частью подграфа модели изображения, то комплексный объект считается идентифицированным. И тогда: 1) формируется означенный экземпляр онтологического понятия; 2) из L_{v1} исключаются все понятия, которые вошли в структуру h_1 ; 3) осуществляется переход к следующему производному понятию в модифицированном списке L_{v1} ; и 4) формируется список $L_{v2} (h_1)$ – список подтвержденных гипотез для следующей итерации (для более сложных объектов и ситуаций).

Если граф структуры онтологического понятия не полностью сопоставлен с подграфом модели изображения, то переходим на эксклюзивный анализ реального изображения, при этом: 1) уточняется ареал по распознанным составляющим (т.е. определяется «где надо искать»); 2) по структуре гипотезы h_1 задается список производных объектов, которые необходимо искать в этом ареале (т.е. определяется «что искать»).

Далее работает PR-система в пределах только вычисленной области на реальном изображении.

После работы PR-системы получаем модифицированные результаты распознавания выделенного ареала изображения. По этим результатам модифицируется подграф модельного графа и осуществляется сопоставление структуры гипотезы h_1 с этим подграфом.

Если после эксклюзивного анализа сопоставление найдено, то считаем, что комплексный объект распознан. В противном случае, переходим к следующей гипотезе h_2 .

Далее в цикле обрабатываются оставшиеся понятия из множества $\{v_i\}$.

Если в процессе всего концептуального анализа ничего не смогли распознать, то получаем пустой список $L_{v2}()$.

Если частично идентифицированы комплексные рисунки (это общий случай), то получаем список $L_{v2} (h_1, h_2, \dots, h_m, v_1, v_2, \dots, v_p)$, где подсписок (h_1, h_2, \dots, h_m) – комплексные рисунки, а подсписок (v_1, v_2, \dots, v_p) – «остатки» в виде производных объектов. Для более сложных композиций (если есть описания в прикладной онтологии) этот список берется в качестве начального для следующей итерации.

Если на изображении только комплексные рисунки и KB-система их идентифицировала, получаем список $L_{v2} (h_1, h_2, \dots, h_m)$. Это говорит о том, что все сложные объекты распознаны и кроме них на изображении ничего нет (это крайне редкий случай).

Вообще целью концептуального разбора изображения является получение списка $L_{v2} (h_1, h_2, \dots, h_m, v_1, v_2, \dots, v_p)$ с минимизацией мощности множества $\{v_1, v_2, \dots, v_p\}$. По данному списку формируется основной результат разбора изображения, а уже на его основе – другие формы представления результата: текст на естественном языке и/или синтезированное изображение композиции.

Проиллюстрируем процесс эксклюзивного анализа на следующем примере. Предположим, что на композиционном изображении кроме всего прочего присутствует объект *Домик* (рис 5а).

PR-система распознала *треугольную крышу* – *треугольник*, а все остальные элементы (*избу, окошко и слуховое окно*) не распознала (рис 5б). Тогда KB-система, выйдя на структуру понятия *домик* и найдя частичное сопоставление (часть вершин этой структуры будет означено), выдвинула гипотезу о том, что в определенном месте изображения (вычисляется ареал по местоположению и размерам *треугольника*) могут присутствовать недостающие для *домика* элементы (формируется список требуемых элементов) с определенными ограничениями по размерам (вычисляются на основе размеров *треугольника*). Подав эту информацию на вход PR-системы, происходит повторное распознавание в только

определенной части реального изображения, что повышает робастность всей системы.

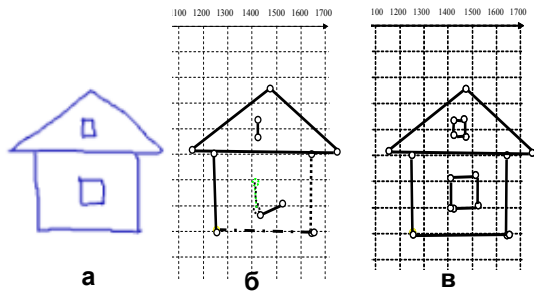


Рисунок 5 – Иллюстрация эксклюзивного анализа

Процесс сегментации при таком методе обработки изображения носит более адекватный характер, т.е. сегментация осуществляется на основе частичных результатов распознавания, а не априори – только с учетом пиксельных характеристик изображения.

Реальные модельные эксперименты показывают более надежные результаты даже на «нижнем» уровне распознавания (на уровне PR-системы), когда сегменты изображения «под конструкции» выделяются более осмысленно. Результат, полученный PR-системой по схеме эксклюзивного распознавания, показан на рис. 5в.

3. Использование концептуального синтеза

Концептуальный синтезатор используется в двух ипостасях: как средство отображения результатов работы гибридной системы распознавания и как средство отладки при модификации и пополнении онтологии.

Первый случай подробно рассмотрен в [Хахалин и др., 2013]. Здесь рассмотрим кратко процесс модификации и пополнения онтологии.

Естественно, что КВ-система «видит только то, что знает», т.е. распознает комплексные объекты, информация о которых присутствует в онтологии. Поэтому важными вопросами являются задачи пополнения и коррекции прикладной онтологии.

В интегральной системе модификация и расширение онтологии предполагает использование двух каналов: для инженера по знаниям (для разработчика) – непосредственный ввод структур новых концептов и/или модификацию старых на языке семантического гиперграфа; для пользователя – то же самое на естественном языке. В отличие от стандартных методов обучения для статистических систем такое обучение носит «дедуктивный» характер. Чтобы это дедуктивное обучение не приводило к «хаосу», необходимо применить методику наполнения онтологии с верификацией вводимых описаний с помощью системы концептуального синтеза изображений [Курбатов и др., 2012а; Литвинович, 2013].

По новой введенной структуре система синтезирует множество изображений. Если с точки зрения учителя (разработчика или пользователя) множество корректно, то структура фиксируется в онтологии. Если среди синтезированных объектов присутствуют изображения, не относящиеся к данному классу, то структура модифицируется. И так до тех пор, пока множество синтезированных изображений не будет корректным.

Проиллюстрируем это на примере концепта *Тачка*. До ввода структуры объекта *Тачка* в онтологию гибридная система распознавания представит его описание в виде взаимосвязанного набора *равнобокой трапеции*, *окружности* и *отрезка прямой*. Введя в онтологию описание структуры понятия *Тачка*, как это представлено на рис. 7, и получив множество синтезированных изображений по этой структуре, представленных на рис. 8, можно предположить, что теперь гибридная система будет распознавать изображение как целостный объект планиметрической среды.

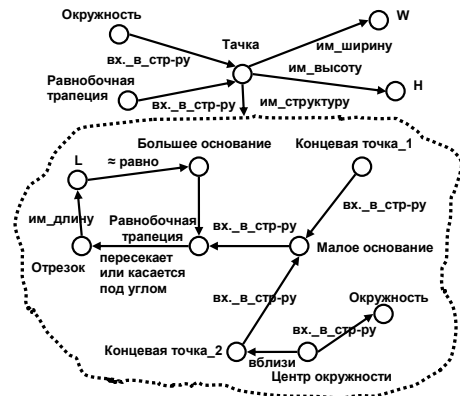


Рисунок 7 – Онтологическая структура концепта *Тачка*

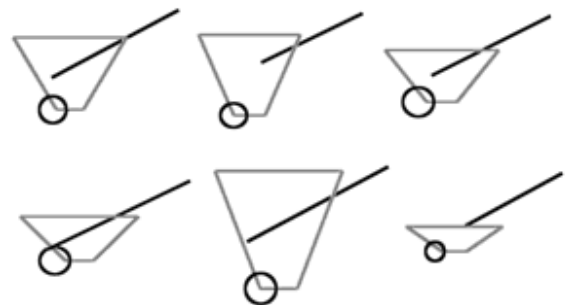


Рисунок 8 – Множество синтезированных изображений концепта *Тачка*

Заключение

Модельные эксперименты показывают, что наиболее оптимальными для решения задачи понимания изображений будут системы, использующие преимущества статистических и структурных методов анализа.

Если учитывать погружение гибридной системы в интегральную систему, то обучение онтологического компонента может происходить без визуальных примеров по описаниям классов на

естественном языке. Для верификации такого процесса обучения привлекается концептуальный синтезатор изображений.

Будущие исследования гибридной системы предполагают решение задач анализа перекрывающихся (маскирующихся) рисунков и поиска объекта на изображении по запросу пользователя.

По мере развития гибридной системы предполагается переход от контурных рисунков к изображениям реальных фрагментов среды (например, для решения задач навигации роботов в реальной среде; для распознавания лиц и т.п.).

Интересным вопросом является «передача знаний» с онтологического уровня к статистическому, позволяющая подключать параллельно несколько PR-систем, обученных на разные ограниченные множества классов.

Работа была выполнена при поддержке гранта РФФИ, проект № 12-07-00531а.

Библиографический список

- [Endres et al., 2013] Ian Endres, Kevin Shih, Johnston Jiaa, and Derek Hoiem. Learning Collections of Part Models for Object Recognition. CVPR, 2013.
- [Farhadi et al., 2009] A. Farhadi, I. Endres, D. Hoiem, and D.A. Forsyth. Describing Objects by their Attributes. CVPR 2009.
- [Khakhalin et al., 2012] Khakhalin G., Kurbatov S., Naidenova K., Lobzin A. Integration of the Image and NL-text Analysis/Synthesis Systems. In book: "Intelligent Data Analysis for Real-Life Applications: Theory and Practice" (Ed. Rafael Magdalena et al.). – N.Y., USA: IGI Global, 2012, p. 160-185.
- [MyScript Notes, 2007] Режим доступа: <http://www.visionobjects.com/en/webstore/myscript-studio/description/>. – [Электронный ресурс].
- [Shrivastava et al., 2011] Abhinav Shrivastava, Tomasz Malisiewicz, Abhinav Gupta, Alexei A. Efros. Data-driven visual similarity for cross-domain image matching. ACM Trans. Graph. 30(6): 154 (2011).
- [Визильтер и др., 2007] Визильтер Ю. В., Желтов С. Ю., Князь В. А., Ходарев А. Н., Моржин А. В. Обработка и анализ цифровых изображений с примерами на LabVIEW IMAQ Vision. – М.: ДМК Пресс, 2007. – 464 с.
- [Курбатов и др., 2012а] Курбатов С.С., Литвинович А.В., Лобзин А.П., Хахалин Г.К. Концептуальный синтез графических образов по структурам прикладной онтологии // Труды XIII национальной конференции по Искусственному Интеллекту, Т. 2.. – Белгород: БГТУ, 2012. – С. 120-127.
- [Курбатов и др., 2012б] Курбатов С.С., Лобзин А.П., Найденова К.А., Хахалин Г.К. Гибридная схема анализа изображений // Труды Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» OSTIS-2012. 16-18 февраля 2012, г. Минск, Беларусь. – 2012. – С. 327-334.
- [Литвинович, 2013] Литвинович А.В. Система синтеза изображений по тексту на естественном языке // Динамика сложных систем — XXI век, № 2, 2013, С. 65-68.
- [Хахалин и др., 2013] Хахалин Г.К., Курбатов С.С., Лобзин А.П. Концептуальный компонент в гибридной системе анализа изображений // Труды 23-й Международной конференции по компьютерной графике и зрению, Владивосток, Россия. – Владивосток: Институт автоматизации и процессов управления ДВО РАН, 2013. – С. 222-225.

PARSING OF GRAPHIC IMAGES ON THE BASIS OF APPLIED ONTOLOGY IN HYBRID SYSTEM OF RECOGNITION

Khakhalin G.K., Kurbatov S.S., Lobzil A.P.

*Research Centre of Electronic Computing
Engineering, Moscow, Russia*

gkhakhalin@yandex.ru

curbatow.serg@yandex.ru

lobzin@rambler.ru

It is considered conceptual component of the hybrid system of an image analysis. The knowledge base for this component is the applied ontology which is presented in the semantic hypergraph language

Introduction

Now systems of image understanding it is realized by means of one system of recognition only statistical type in which semantic properties of objects are used except for discriminant attributes.

The description of distinguished objects and their arrangement is meant a problem of understanding (parsing) of the images in terms and in structures of applied ontology language. The basic result of analysis is the description of the visual situation consisting of occurrences of distinguished classes of objects and relations between them.

Main Part

For the decision of a problem of images understanding the hybrid system of the analysis of the images, consisting of a statistical and ontology components is proposed.

The statistical component directly works with the real image. The ontological component works with model of the image (by results of work of statistical system).

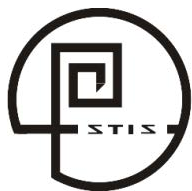
The algorithm of work of the analysis of a composition on the image is described and the examples illustrating process of understanding are given.

Functions of an ontological component are listed. The algorithm of work of the analysis of a composition on the image is described and the examples illustrating process of understanding are given.

The scheme of automation of process of ontology augmentation with attraction of naturally language interface and conceptual system of image synthesis is considered.

Conclusion

The future researches of hybrid system assume the decision of problems of the analysis of overlapped (masking) figures and search of object on the image by inquiry of the user.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

СЕМАНТИЧЕСКОЕ КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ ЕСТЕСТВЕННО-ЯЗЫКОВОГО ИНТЕРФЕЙСА ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ

Елисеева О.Е.*

** Белорусский государственный университет,
г. Минск, Республика Беларусь*

olae@open.by

В работе осуществляется проектирование концепции естественно-языкового интерфейса интеллектуальных систем на основе открытой семантической технологии проектирования интеллектуальных систем OSTIS. Изложение основных результатов предлагается непосредственно на графовом семантическом языке SC, в результате чего формируется фрагмент БЗ подсистемы естественно-языкового интерфейса. Визуализация знаний о процессе общения и о естественном языке в виде семантической сети является инструментом сформирования более глубокого понимания соответствующих предметных областей.

Ключевые слова: естественно-языковой интерфейс; интеллектуальная система; лингвистическая база знаний; семантическая сеть; семантический язык.

Введение

Актуальность создания естественно-языкового интерфейса (ЕЯИ) для любой интеллектуальной системы является вполне очевидной. Тем не менее, до настоящего времени открытым остаётся вопрос массового использования ЕЯИ. Главными причинами такого положения дел являются:

- Широкое использование ставшего стандартом графического интерфейса с разнообразными панелями инструментов в виде кнопок, переключателей, форм и пр., к которым все пользователи уже привыкли, в связи с чем возникает проблема грамотного сочетания стандартных интерфейсных средств с естественно-языковыми;

- Значительное количество проблем, связанных со сложностью автоматического анализа и синтеза естественного языка. Предметная ориентированность прикладных систем, а также отсутствие универсальной технологии создания ЕЯИ фактически приводят к тому, что для каждой отдельной системы приходится создавать ЕЯИ практически с нуля, что слишком затратно и ресурсоёмко.

В данной работе осуществляется очередная попытка (наряду с уже представленными в работах [Гецевич и др., 2012], [Елисеева, 2009], [Корончик, 2013] и др.) проектирования концепции ЕЯИ интеллектуальных систем, которая ориентирована

на массовое универсализированное применение. В качестве основы для проектирования взята открытая семантическая технология проектирования интеллектуальных систем (Open Semantic Technology for Intelligent Systems - OSTIS) [Ostis, 2012] и соответствующие ей графовые языки представления знаний на основе семантических сетей [Голенков и др., 2013].

Изложение основных результатов в статье предлагается непосредственно на графовом семантическом языке SC (Semantic Code), в его графической нотации SCg-коде. Таким образом, в данной работе ставится своеобразный эксперимент, в результате которого должна быть доказана либо опровергнута гипотеза о том, что представление знаний в виде семантической сети может способствовать более эффективному пониманию соответствующей информации.

1. Основные понятия

Для начала уточним понимание термина «семантическое проектирование». В данной работе осуществляется анализ смысла (семантики) каждого понятия (термина), используемого в процессе проектирования, с целью определения семантических связей между компонентами ЕЯИ, выделения основных (ключевых) технологических моментов, а также формирования базовой онтологии понятий и отношений предметной области, соответствующей понятию ЕЯИ. Процесс такого переосмысления и представления его

результатов в виде sc-текстов будем называть **семантическим проектированием**.

1.1. Моделирование процесса общения на естественном языке

Фактически ЕЯИ является компьютерной (программной) **моделью процесса общения**. Для краткости в данной работе мы не рассматриваем все возможные определения понятия «общение» (такая попытка осуществлялась автором в [Елисеева, 2009]), а будем опираться на формальную модель общения, предложенную Э.В.Поповым в книге [Попов, 1982]. Следуя рассуждениям в [Попов, 1982], представим модель общения в виде фрагмента семантической сети на языке SC (sc-текста, рис. 1). С целью облегчения восприятия в данном sc-тексте не уточнена семантика связей между указанными понятиями.



Рисунок 1 – Модель общения

Согласно базовой денотационной семантике языка SC, узлы с идентификаторами являются знаками соответствующих понятий, а дуги обозначают наличие связей между ними. На рис.1 задействован также специальный тип sc-дуги,

которая изображена в виде двойной линии со стрелкой и является знаком ориентированной пары. Для уточнения семантики связи между понятиями в указанные дуги должны быть проведены дуги из знаков соответствующих отношений. Достаточно часто для этого используются теоретико-множественные отношения.

Рассмотрим более подробно каждый из перечисленных в указанной модели компонентов. База знаний (БЗ) предметной области в каждой конкретной ИС, как правило, является некоторой частью модели окружающего мира. Рассмотрение технологии создания указанной БЗ не является целью данной работы, т.к. этому вопросу посвящено достаточное количество других работ. В следующей главе данной работы БЗ будет проанализирована на предмет наличия в ней «естественно-языковых объектов».

1.2. Субъекты общения и обмен сообщениями

Основой данной модели является наличие двух субъектов общения: пользователя и интеллектуальной системы. Начнём с рассмотрения примера описания знаний о субъектах общения в рамках базы знаний ЕЯИ. Предположим, мы имеем некоторую прикладную интеллектуальную систему и некоторого её пользователя *X*. Опишем данное утверждение в линейной нотации языка SC:

пользователь ИС -> Пользователь X;
ИС -> Прикладная ИС-X;

Из данного фрагмента sc-текста видно, что мы ввели два знака множеств:

- всех пользователей интеллектуальных систем (ИС), которое задали sc-узлом с идентификатором *пользователь ИС*. При этом узел с идентификатором *Пользователь X* обозначает некоторого конкретного пользователя, и в общем случае из узла с идентификатором *пользователь ИС* будет выходить ровно столько дуг, сколько пользователей у системы. Все узлы, в которые указанные дуги входят, относятся к константному типу и являются некоторыми «виртуальными представителями» (моделями) конкретных пользователей, каждый из которых относится к объектам окружающего мира (см. рис.1). Для описания этого в языке SC имеются соответствующие средства;

- всех интеллектуальных систем, которое задали sc-узлом с идентификатором *ИС*. Аналогично сказанному выше, узел *Прикладная ИС-X* является знаком конкретной прикладной интеллектуальной системы. Если мы будем рассматривать ЕЯИ конкретной ИС, то, по всей видимости, данное описание может оказаться избыточным. Но если следовать логике рассуждений Э.В.Попова ([Попов, 1982]), то в ИС должна быть представлена некоторая «модель себя». Следовательно, такое описание может оказаться вполне востребованным, например, в

рамках компонента ИС, отвечающего за компонент «Помощь (руководство пользователя)».

В процессе общения субъекты обмениваются сообщениями (рис. 2). Из sc-текста на рис. 2 мы видим, что для описания сообщений в рамках ЕЯИ необходимо ввести новое тернарное отношение *сообщение-автор-адресат**, описывающее семантические связи между автором (обозначено атрибутом *автор_* рассматриваемого отношения), адресата (*адресат_*) и собственно сообщения (*сообщение_*).

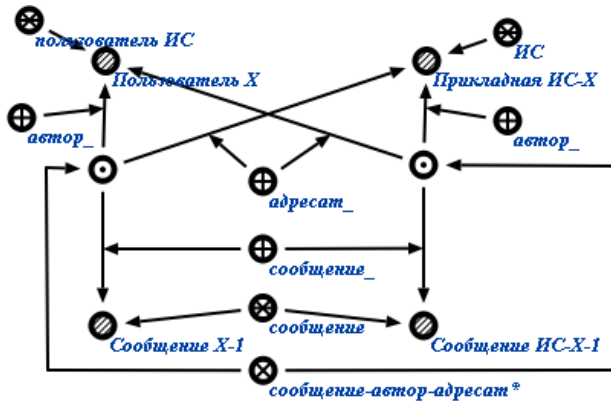


Рисунок 2 – Сообщения субъектов общения

Уточним понимание использованных на рис. 2 символов и обозначений. Для описания тернарного (как и любого другого) отношения используются следующие изображения sc-узлов ([Ostis, 2012]), задающие семантику (роли) каждого компонента отношения:

- ⊗ - собственно знак отношения;
- ⊕ - знаки атрибутов отношения;
- ⊙ - знак связки отношения.

Очевидно, что в прикладной ИС сообщения пользователя и ИС взаимосвязаны по принципу вопрос – ответ (импульс – реакция). Данное бинарное асимметричное отношение опишем с использованием ключевого узла *вопрос-ответ*.

Помимо базовой взаимосвязи типа «вопрос – ответ» очевидно, что при проектировании и создании интерфейса ИС, как и любой другой прикладной системы, все сообщения соответствующим образом классифицируются, представляются в различной форме и реализуются различными способами. В рамках ЕЯИ ставится задача автоматического анализа поступающих от пользователя сообщений, в результате которого выполняется (рис. 3):

- распознавание типа сообщения;
- определение соответствующего запроса к базе знаний (БЗ), его параметров и операндов либо обработка введенных пользователем данных/знаний в ответ на запрос ИС.

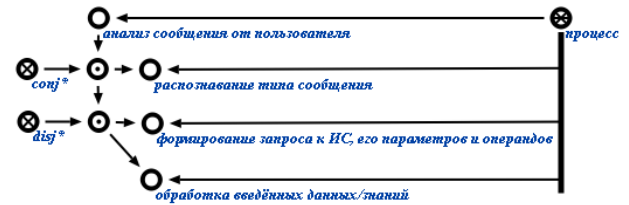


Рисунок 3 – Процесс анализа сообщений от пользователя

Из сказанного выше следует, что в процессе проектирования ЕЯИ необходимо определить типологию сообщений от пользователя. В первом приближении все сообщения можно разбить на 2 группы: запросы и ответы. При этом в качестве запросов можно выделить: поиск информации в БЗ; преобразование информации; добавление информации в БЗ; уточняющий вопрос и др.

Исходя из приведённой типологии, на данном этапе проектирования можно сформулировать первые правила анализа входных сообщений в форме следующих импликативных высказываний:

1. «Если входное сообщение от пользователя относится к типу «запрос», то в результате его анализа необходимо сформировать соответствующий запрос к БЗ и определить перечень его параметров и операндов» (рис. 3);
2. «Если входное сообщение от пользователя относится к типу «ответ», то в результате его анализа необходимо обработать введенные пользователем данные или знания и сформировать запрос на добавление этих данных/знаний в соответствующую БЗ и/или базу данных (БД)», если таковая используется в ИС (рис. 4).

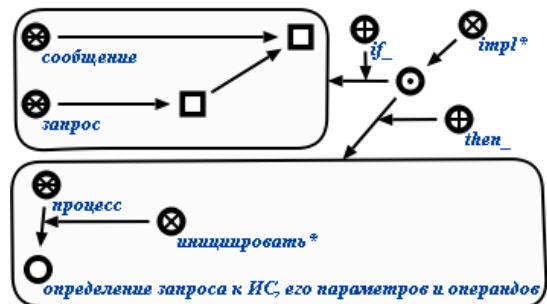


Рисунок 3 – Правило № 1 анализа сообщения от пользователя

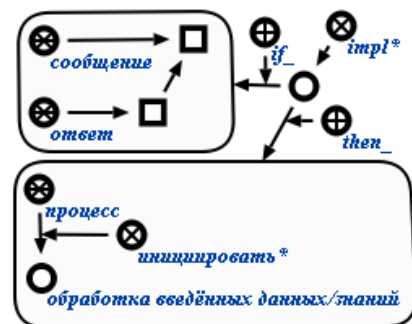


Рисунок 4 – Правило № 2 анализа сообщения от пользователя

Следует отметить, что из указанных формулировок также следует, что первым этапом анализа сообщения от пользователя является

именно определение его типа. Таким образом, мы можем начать описывать упорядоченный сценарий (последовательность операций) обработки сообщений (рис.5).



Рисунок 5 – Описание процесса анализа сообщений от пользователя

После обработки сообщения от пользователя ИС в результате обработки БЗ должна сгенерировать ответное сообщение. Ответные сообщения также классифицируются и разбиваются на две группы: ответы и запросы. Среди ответов, в частности, можно выделить:

- прямой ответ, т.е. ответ, полностью соответствующий (релевантный) запросу пользователя;
- косвенный ответ – ответ, в котором перехватывается инициатива и уточняется понимание [Попов, 1982];
- инструкция – ответ, в котором ИС сообщает (объясняет) пользователю, что и как он должен делать и др.

Очевидно, что рассмотренные классификации сообщений пользователя и ИС весьма приблизительные и требуют значительных уточнений, но на данном этапе концептуального проектирования мы такую задачу не ставим. Распознавание типа сообщения при этом представляет собой достаточно нетривиальную технологическую-лингвистическую задачу, в постановке которой следует учесть два момента:

- какие запросы и ответы могут быть реализованы ИС и как;
- каким образом и на основании каких лексических и грамматических составляющих тот или иной тип сообщения можно распознать в результате автоматической обработки соответствующего естественно-языкового текста (ЕЯ-текста).

Указанную задачу в данной работе мы также лишь обозначаем, выделив на концептуальном уровне набор соответствующих операций. Отметим, что при реализации «традиционного» графического интерфейса типология сообщений (которые в

сущности представляют собой команды или операции) осуществляется в процессе проектирования и представляет собой заранее заданную совокупность, которая поддерживается соответствующими элементами интерфейса: кнопками, переключателями и пр.

1.3. Моделирование диалога

В модели общения (рис.1) показано, что процесс взаимного обмена сообщениями пользователя и ИС реализуется в форме диалога. Таким образом, следующим этапом моделирования процесса общения и его семантического описания является уточнение всех понятий, связанных с **диалогом**. Сообщения при этом являются основными единицами, из которых строится диалог. При создании диалоговых систем говорят о проектировании общей и конкретной структуры диалога. Рассмотрим, каким образом это можно описать на языке SC.

На рис.6 представлен пример конкретной структуры диалога, которая, очевидно, является линейной последовательностью сообщений.

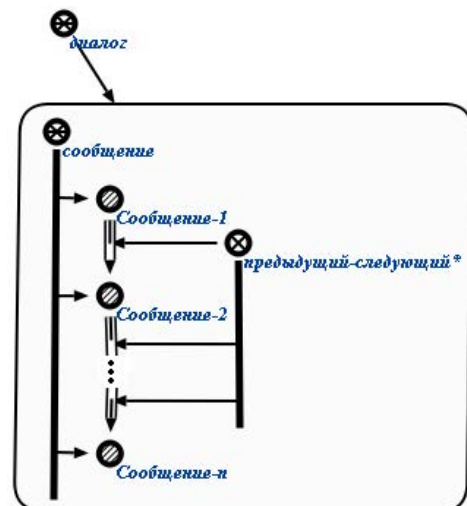


Рисунок 6 – Линейная последовательность сообщений в конкретной структуре диалога

Не менее очевидно также, что общая структура диалога представляет собой направленную, разветвленную структуру. Для некоторых конкретных ИС на этапе их разработки можно попытаться описать первоначальные конструкции, отражающие возможные варианты развития диалога пользователя с данной системой. В реальности же общение на естественном языке является недетерминированным процессом, описать который заранее практически невозможно. Поэтому в данной работе будем исходить из следующих положений:

- при проектировании ЕЯИ описывается некоторая базовая совокупность общих структур диалога, соответствующих различным типам общения (например, диалог-приветствие (рис.7), диалог-справка о системе, диалог-запрос на решение задачи по предметной области, диалог-

информационно-поисковый запрос по предметной области);

- в процессе создания конкретной ИС по конкретной предметной области инженер знаний может описать дополнительные общие структуры аналогичным образом;

- в рамках концептуального проектирования ЕЯИ ставится задача разработки методов и алгоритмов автоматического формирования общих структур диалогов различного типа на основе анализа и обобщения конкретных взаимодействий с пользователями (блок самообучения ЕЯИ).

В sc-тексте на рис.7 мы ввели ряд семантических связей, отражающих сценарий работы с пользователем на этапе входа в ИС. Помимо сообщений, которыми обмениваются пользователь и ИС, целесообразно говорить о происходящих в процессе рассматриваемого взаимодействия **событиях**, для первичного описания необходимости обработки которых вводится бинарное ориентированное отношение *обработка события** (рис.8).

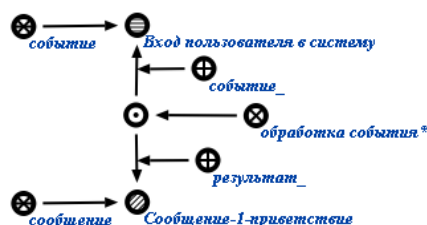


Рисунок 7 – Пример использования отношения *обработка события*

Использованные на рис.6 и 7 идентификаторы sc-констант весьма условны и введены лишь для лучшего понимания читателем данного изложения. В процессе реального функционирования ЕЯИ эти идентификаторы не нужны, т.к. семантика соответствующих узлов описывается с помощью соответствующих введенных в БЗ ЕЯИ знаков множеств и отношений.

1.4. Естественный язык

Центральным элементом рассматриваемой модели общения является **естественный язык** как основное средство взаимодействия пользователя и ИС в рамках ЕЯИ. Следуя логике размышлений данной работы, для начала необходимо уточнить понимание понятия «естественный язык». Уточним высказывание, представленное на рис.1, указав семантику представленных на нём узлов и дуг (рис.8).

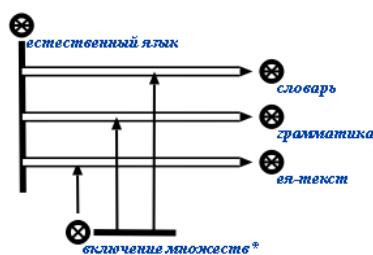


Рисунок 8 – Состав естественного языка

Заметим, что в данной работе мы не претендуем на полную представленность описания, а лишь обозначаем основные моменты. Кроме того, некоторые примеры и подходы к формализации знаний о естественном языке были представлены в [Гецевич и др., 2012], [Елисеева, 2009], [Елисеева и др., 2013].

Очевидно, что в процессе создания ЕЯИ необходимо обеспечить эффективное решение двух основных задач:

- автоматический анализ ея-текстов;
- автоматическая генерация ея-текстов.

Таким образом, в рамках представляемого концептуального моделирования ЕЯИ можно говорить о соответствующих указанным задачам двух наборах (множествах) процедур и/или операций, а также исходных данных для них. На данном этапе размышлений не достаточно ясно, каким образом соотносятся между собой указанные два множества. Поэтому здесь мы пока используем теоретико-множественное отношение **включение множеств***. Множества, заданные sc-узлами **автоматический анализ ея-текстов** и **автоматическая генерация ея-текстов** могут включать как декларативные, так и процедурные знания, относящиеся к любому из компонентов языка, перечисленных на рис.8.

В процессе проведённого анализа основных понятий был выполнен первый шаг формализации предметной области «Общение» и создан фрагмент соответствующей базы знаний, а точнее – определён набор ключевых sc-узлов. Все знания в процессе создания конкретной ИС хранятся в общей БЗ системы и представляются единым образом. Благодаря этому, мы получаем возможность использовать одни и те же инструменты как для обработки БЗ ЕЯИС, так и для обработки БЗ предметной области.

2. Лингвистическая база знаний и база знаний предметной области: формы представления языка

Отметим следующее:

- очевидно, что при создании ЕЯИ необходимо говорить о создании лингвистической БЗ. Однако на уровне БЗ предметной области также можно обнаружить достаточно много «лингвистических конструкций»;
- в процессе формализации знаний об окружающем мире и представления их в БЗ ИС на языке SC мы используем слова естественного языка;
- связи между объектами, явлениями и пр. окружающего мира, а также их базовые семантические характеристики, мы задаём либо с использованием специально вводимых обозначений и символов языка представления знаний SC, либо с использованием соответствующих ключевых

sc-узлов с идентификаторами-словами естественного языка;

- ввиду сказанного становится очевидно, что sc-тексты содержат элементы естественного языка, которые могут быть использованы в реализации естественно-языкового общения с пользователем.

Итак, в БЗ ИС информацию о ЕЯ мы имеем:

- в составе идентификаторов ключевых sc-узлов – это основные понятия соответствующей предметной области;

- в формальных высказываниях, описывающих определения понятий, аксиомы, теоремы и их доказательства, утверждения, задачи и т.п. (см., например, [Давыденко и др., 2011]);

- в виде протоколов решения задач из предметной области;

- в виде формальных описаний конкретных объектов предметной области. Например, для ИС решения задач по геометрии [Давыденко и др., 2011] это описания геометрических фигур и их соотношений (рис.9);

- в текстовых содержимых sc-узлов;

- в нетекстовых содержимых sc-узлов, которые можно описать на ЕЯ. Например, для ИС по геометрии это чертежи геометрических фигур. Для таких содержимых в предметной области создаются мета-описания и/или подписи на ЕЯ (рис.10).

Исходя из сказанного выше, можем сформировать первоначальную совокупность операций и исходных данных для них, необходимых для автоматической обработки естественного языка:

автоматический анализ ея-текстов

⇒ формирование sc-текста на основе ея-текста

{⇒ формирование sc-идентификатора на основе ея-слов

{⇒ автоматическое извлечение из ея-текста ключевых понятий

⇒ лемматизация

⇒ модель управления слова }

⇒ формирование sc-высказывания на основе фрагмента ея-текста

⇒ формирование связей между sc-высказываниями }

⇒ обработка текстового содержимого sc-узла

автоматическая генерация ея-текстов

{⇒ перевод sc-текста на ея

{⇒ преобразование идентификатора в ея-словоформу

{⇒ правила словоизменения

⇒ словарь словоформ }

⇒ формирование ея-словосочетания

{⇒ правила согласования словоформ

⇒ типы сочетаемости слов

⇒ шаблоны словосочетаний

⇒ лексическая функция }

⇒ формирование ея-предложения

{⇒ типология предложений

⇒ правила построения предложений

⇒ модель управления слова

⇒ шаблоны предложений }

⇒ правила трансляции sc-высказываний на ея

{⇒ правила трансляции теретико-множественных высказываний

⇒ правила трансляции логических высказываний

{⇒ правила трансляции имплицитивных высказываний

⇒ правила трансляции высказываний о существовании }

⇒ формирование ея-текста

{⇒ стиль текста

⇒ структура текста

⇒ определение последовательности предложений в тексте }

⇒ обработка текстового содержимого sc-узла }

На рис. 11 представлен пример описания в БЗ синонимичных высказываний о составе естественного языка, которые являются трансляцией (переводом) sc-текста, представленного на рис.8, на русский язык.

Из представленного на рис.11 примера видно, что одно и то же sc-выражение может быть транслировано на естественный язык различными способами. В дополнение к характерному для естественного языка явлению синонимии мы имеем синонимию, связанную с особенностями представления базы знаний на основе теоретико-множественных рассуждений. Очевидно, что конечным (неподготовленным) пользователем ИС наиболее понятно и естественно будет воспринято первое предложение: «Естественный язык состоит из словаря, грамматики и текстов.». Отметим, что из исходного sc-текста указанное предложение «напрямую» не следует. Для его получения необходимо выполнить следующие операции:

- транслировать идентификаторы в соответствующие словоформы и словосочетания (результат см., например, на рис.12);

- выбрать наиболее подходящий (естественный) эквивалент из списка синонимичных слов и словосочетаний;

- определить синтаксические роли и связи отобранных словоформ и словосочетаний;

- построить синтаксическую структуру результирующего предложения;

- синтезировать результирующее предложение.

На рис.13 приведён пример описания словообразовательной парадигмы, которая либо должна быть занесена в лингвистическую БЗ экспертом-лингвистом, либо сгенерирована

автоматически на основе правил словоизменения. На основе указанной парадигмы выбирается необходимая для синтеза ея-текста словоформа.

На рис.14 показана связь синонимии, благодаря которой в результирующем ея-тексте вместо слова «ея-тексты» можно получить слово «тексты».

На рис.15-17 представлены фрагменты лингвистической БЗ, на основе которого генерируется рассматриваемый нами пример ея-предложения. В частности, на рис.15 указано следующее:

- слово «включение», являющееся частью словосочетания «включение множеств», синонимичного идентификатору включение множеств*, является производным отглагольным существительным от слова «включать»;
- слово «включать» имеет синонимы «содержать», «состоять из»;
- для предиката «состоять из» описана модель управления слова (МУС) [Мельчук, 1974], в которой задана соответствующая синтаксическая, морфологическая и семантическая информация.

На рис.16 описано преобразование перевода (трансляции) словосочетания «включение множеств» в эквивалентное его представление в виде sc-текста, в котором уточнён тип связи слов в словосочетании.

Наконец, на рис. 17 приведён пример эквивалентного соответствия (перевода) ея-предложения и его представления в виде синтаксической структуры на языке SC.

В завершение данного рассмотрения следует отметить, что здесь в некотором приближении был рассмотрен процесс генерации ея-текста на основе sc-текста, что фактически представляет собой перевод с одного языка (языка представления знаний) на другой (естественный язык). Предлагаемый подход продиктован тем, что фактически обмен сообщениями пользователя и ИС в процессе естественно-языкового диалога представляет собой ни что иное, как перевод ея-текстов на язык SC и наоборот. Важным преимуществом языка SC при этом является его изначальная ориентация на семантическое описание различных предметных областей, что в значительной степени облегчает интерпретацию многих высказываний.

Следовательно, из сказанного выше, можно предположить, что в процессе реализации предлагаемой концепции возможно получение вполне надёжных универсальных правил, методов, алгоритмов и процедур автоматического перевода текстов.

3. От базы знаний – к пониманию. Когнитивно-семиотический аспект

Как сказано выше, важнейшей особенностью языка SC и его подязыков является изначальная

ориентация на семантическое представление знаний различного вида. Особенно важно, что на этом же языке реализуются и программы обработки рассмотренных sc-текстов. Таким образом, мы имеем интегрированную среду, в которой знания и процедуры их обработки представляются единым образом.

Решение задач обработки естественного языка, опираясь на sc-представления, таким образом, даёт нам возможность описывать единым образом все уровни языка: лексику, морфологию, синтаксис, семантику, - а также процедуры обработки знаний каждого уровня. При этом естественным образом мы поддерживаем взаимосвязь всех уровней языка и фактически не сталкиваемся с проблемой специального выделения каждого уровня и их разграничения. Заметим, что аналогичная ситуация характерна и для самого естественного языка, в котором, например, в процессе синтаксического анализа/синтеза так или иначе приходится апеллировать к семантике. Таким образом, мы сохраняем эту особенность в БЗ ЕЯИ.

В данной работе для иллюстрирования основных моментов проектировании сознательно не использовались никакие иные средства, кроме изобразительных возможностей графовой нотации языка SC – SCg-кода. Автор надеется, что уважаемый читатель не испытывал особых трудностей в понимании приведённых в тексте sc-конструкций. Если это действительно так, то это, с одной стороны, является доказательством семантической мощи языка SC. С другой стороны, следует ещё раз отметить, что фактически в процессе рассуждений мы создавали фрагменты реальной БЗ. Очевидно, что многие моменты в статье, учитывая объективные ограничения, опущены, но если всё изложенное представить на SC, мы получим полноценную БЗ естественно-языковой подсистемы ИС.

Использование технологии формирования и уточнения понятийной системы предметной области путём введения ключевых sc-узлов в качестве знаков понятий и явного описания теоретико-множественных и логических взаимосвязей между ними в виде семантической сети даёт возможность более чётко осознать семантику соответствующей предметной области. При этом эта семантика вполне естественным образом визуализируется в виде sc-текстов. Таким образом, можно утверждать, что язык SC является мощным инструментом понимания. Эксперт, создающий БЗ на языке SC, получает возможность глубже понять соответствующую предметную область. Чтение же и понимание смысла sc-текстов – не намного более сложная, а порой даже более простая задача, чем чтение и понимание естественно-языкового текста, т.к., несмотря на первоначальное ощущение некоторой громоздкости, sc-тексты более структурированы по сравнению с ея-текстами. Так же, как в памяти человека информация представляется нелинейным образом, sc-тексты

Учитывая сказанное, мы приближаемся к моделированию процессов понимания в БЗ ИС.

В данной работе представлено семантическое концептуальное моделирование ЕЯИ ИС, в результате которого были созданы фрагменты БЗ соответствующей ея-подсистемы в виде семантической сети. К сожалению, формат статьи

Кроме того, одной из задач данной работы явилось иллюстрирование экспертам-лингвистам, а также специалистам любых других предметных областей, поэтапного процесса формализации знаний, представленных на естественном языке. Всё повествование в работе построено по принципу перевода сформулированных на естественном языке размышлений на семантический язык представления знаний.

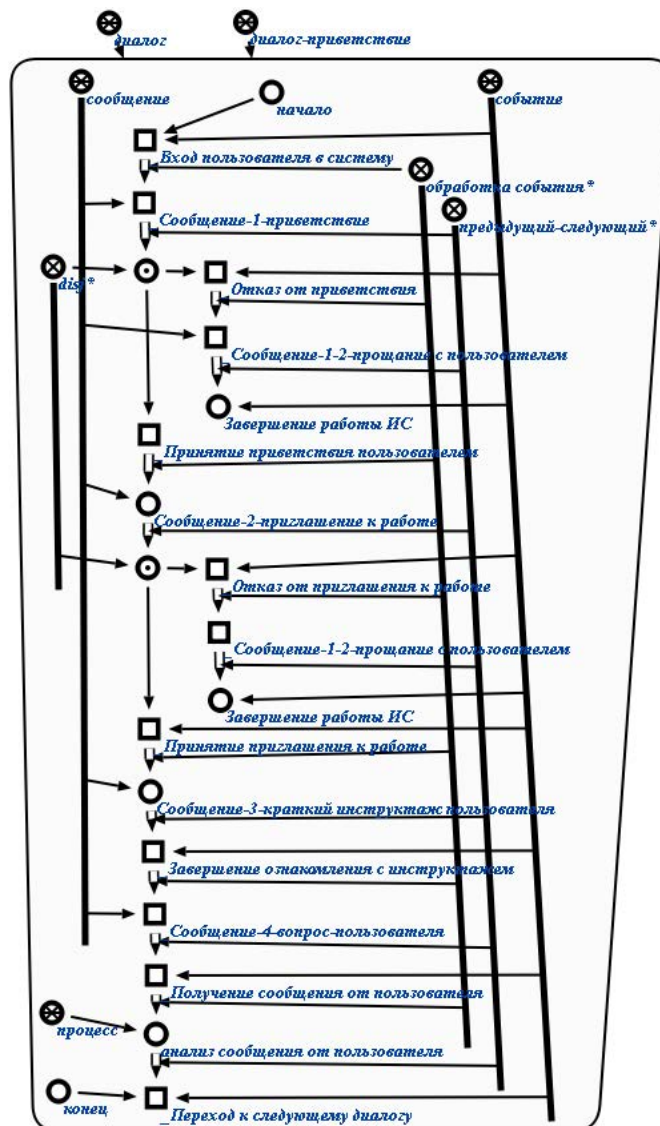


Рисунок 9 – Общая структура диалога-приветствия

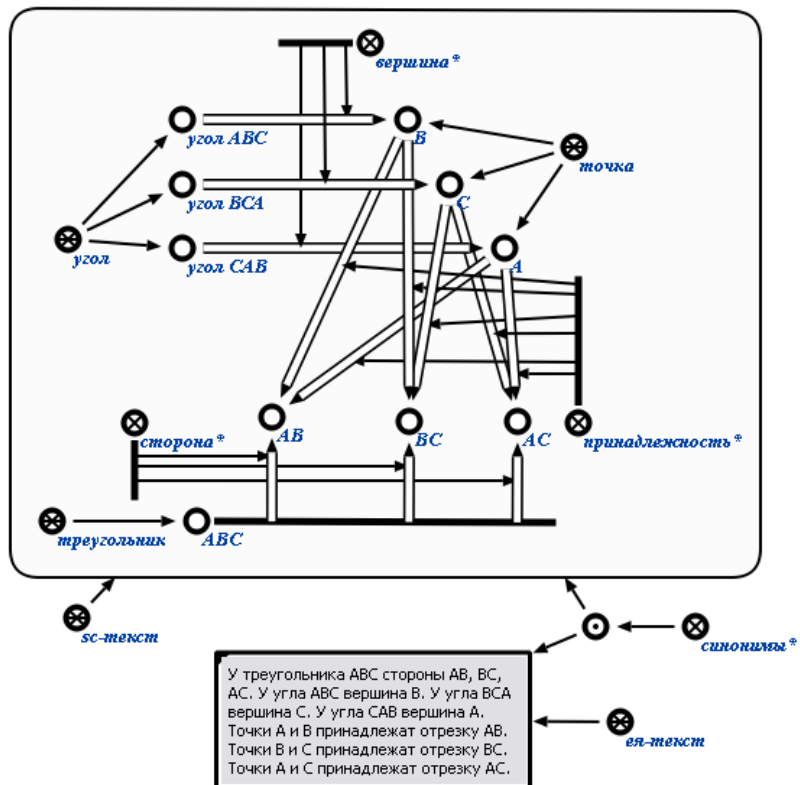


Рисунок 10 – Пример описания геометрических фигур в виде sc-текста и синонимичного ея-текста

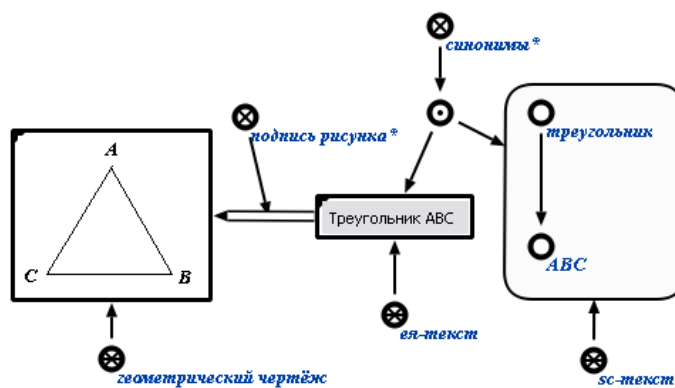


Рисунок 11 – Пример подписи геометрического чертежа

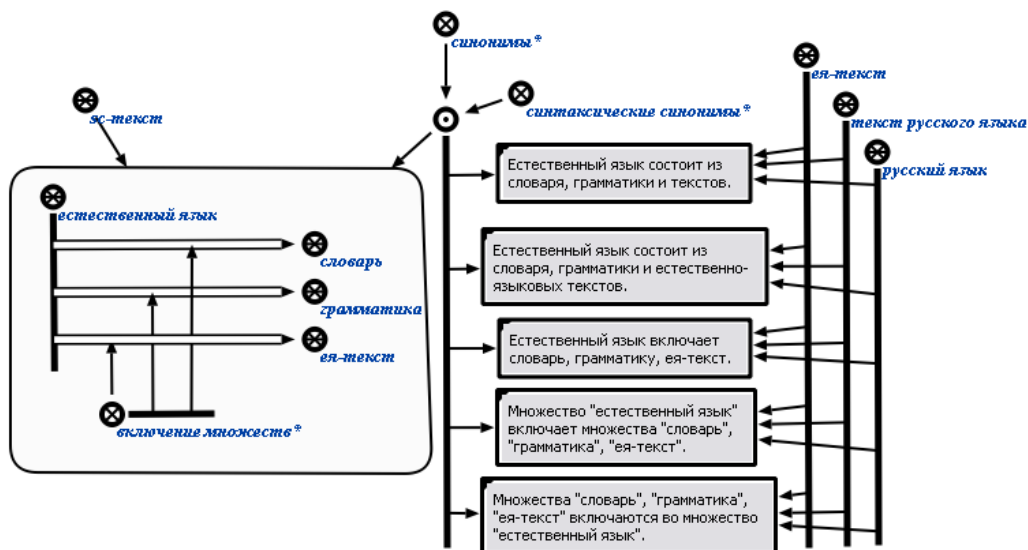


Рисунок 12 – Пример синонимии sc-текстов и ея-текстов

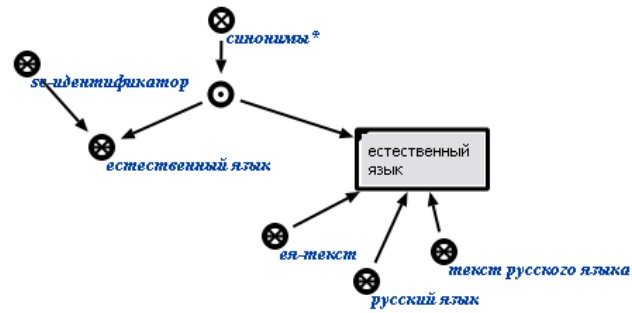


Рисунок 13 – Описание синонимии идентификатора и его записи в виде ея-текста

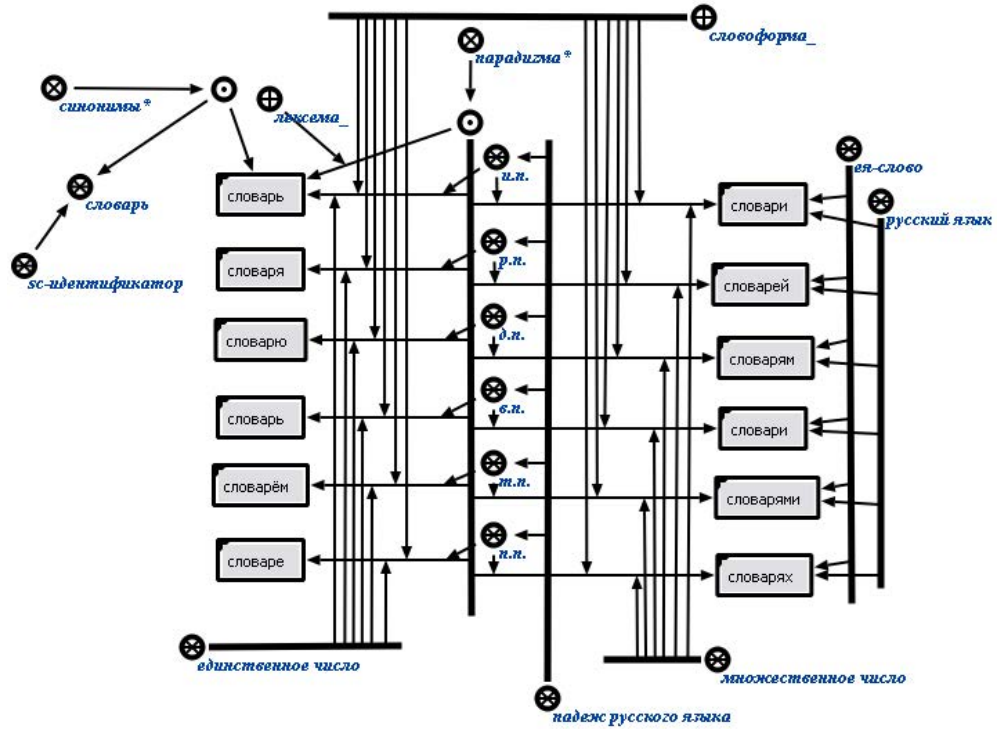


Рисунок 14 – Пример словообразовательной парадигмы

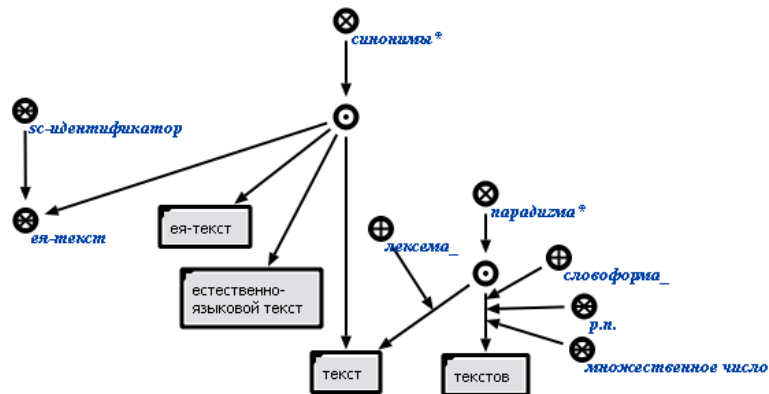


Рисунок 15 – Несколько синонимов для ея-представления идентификатора

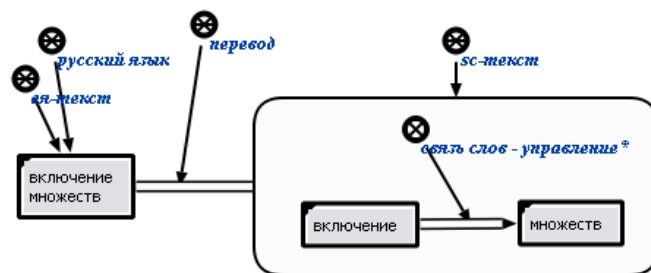


Рисунок 16 – Пример перевода ея-словосочетания в формат сс-текста

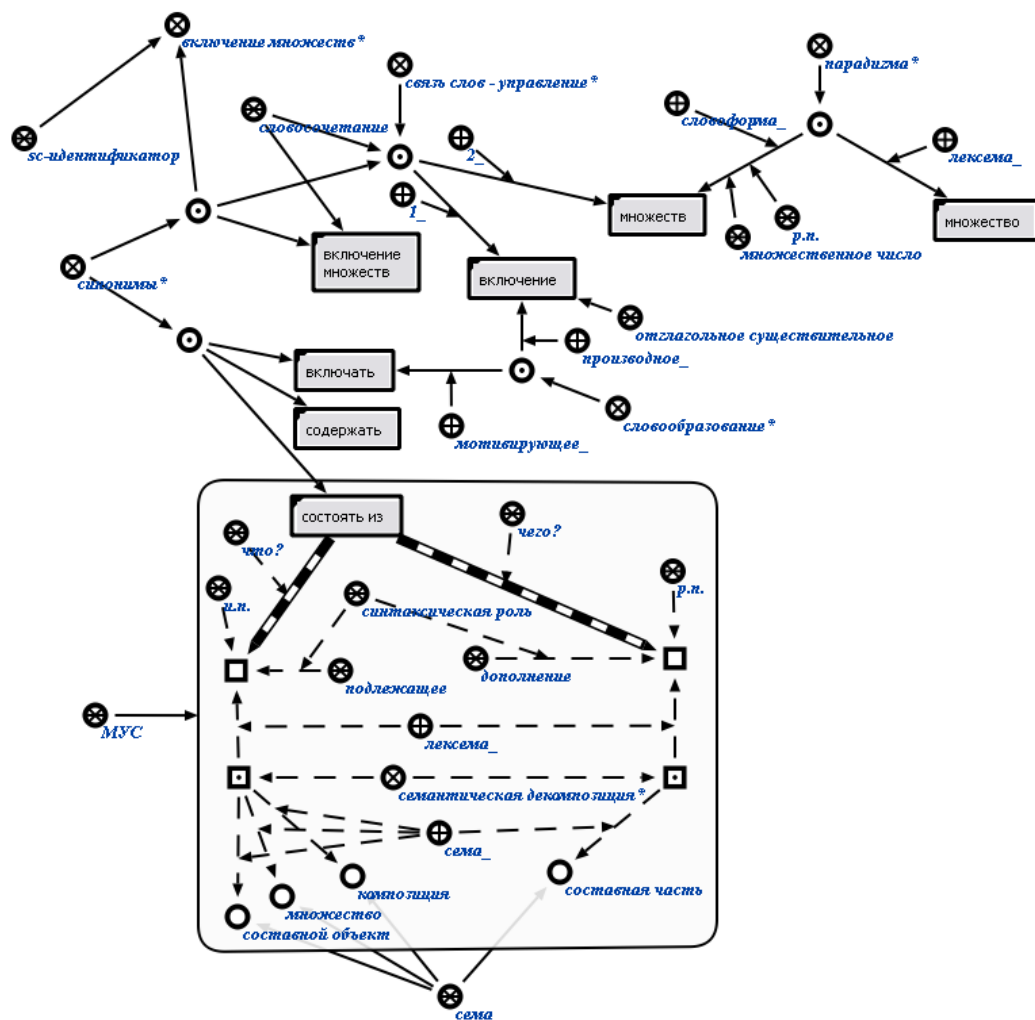


Рисунок 17 – Фрагмент лингвистической БЗ

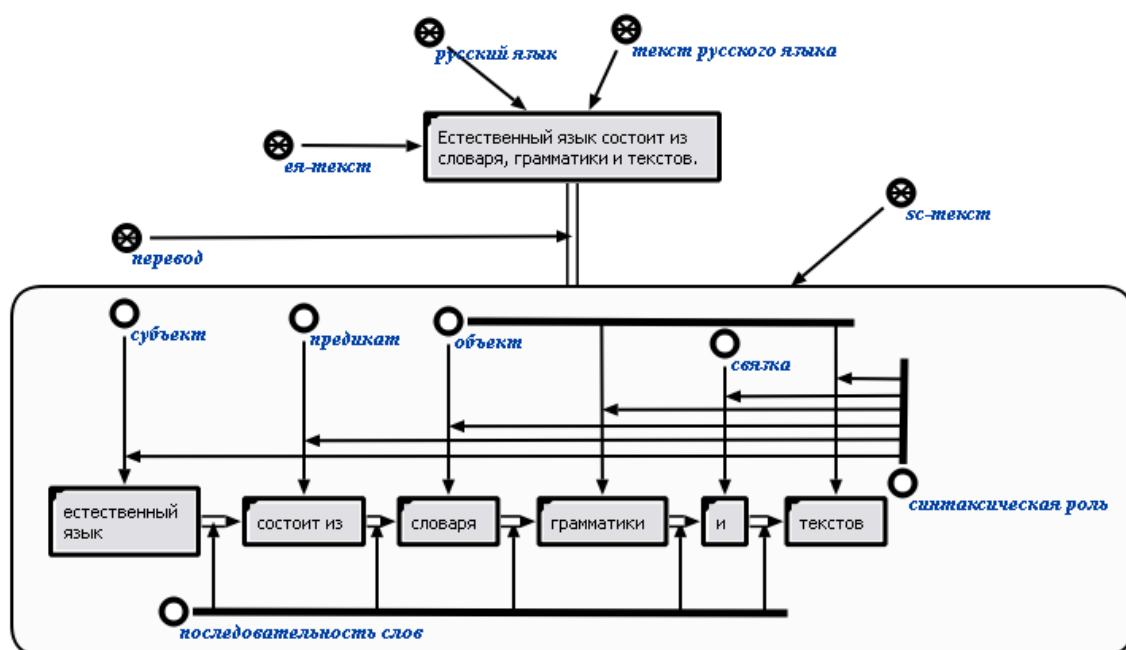


Рисунок 18 – Соответствие ея-предложения и его синтаксической структуры в формате sc-текста

Библиографический список

- [Апресян, 1995] Апресян, Ю. Д. Избранные труды, том II. Интегральное описание языка и системная лексикография / Ю. Д. Апресян – Москва: Школа «Языки русской культуры», 1995.
- [Гецевич и др., 2012] Семантическая технология проектирования белорусско- и русскоязычных ея-интерфейсов вопросно-ответных систем / Ю.С. Гецевич [и др.] // сборник трудов Международной научно-технической конференции OSTIS-2012, Минск, 2012 / БГУИР – Минск, 2012.
- [Голенков и др., 2013] Голенков, В.В., Гулякина, Н.А. Открытый проект, направленный на создание технологии компонентного проектирования интеллектуальных систем / В.В. Голенков, Н.А. Гулякина // сборник трудов Международной научно-технической конференции OSTIS-2013, Минск, 2013 / БГУИР – Минск, 2013.
- [Давыденко и др., 2011] Интеллектуальная справочная система по геометрии / Давыденко И.Т. [и др.] // сборник трудов Международной научно-технической конференции OSTIS-2011, Минск, 2011 / БГУИР – Минск, 2011.
- [Елисеева, 2009] Елисеева О.Е. Естественно-языковой интерфейс интеллектуальных систем: учеб. пособие / О. Е. Елисеева; под науч. ред. проф. В. В. Голенкова. – Минск: БГУИР, 2009.
- [Елисеева и др., 2013] Елисеева О.Е., Русецкий К.В. Подходы к формализации естественного языка на семантическом графовом языке представления знаний // Седьмые Карповские научные чтения. Сборник научных статей, 2013.
- [Корончик, 2013] Корончик, Д.Н. Унифицированные семантические модели пользовательских интерфейсов интеллектуальных систем и технология их компонентного проектирования / Д.Н. Корончик // сборник трудов Международной научно-технической конференции OSTIS-2013, Минск, 2013 / БГУИР – Минск, 2013.
- [Мельчук, 1974] Мельчук, И.А. Опыт теории лингвистических моделей «Смысл-Текст». Семантика, Синтаксис / И.А. Мельчук. – Москва, 1974.
- [Попов, 1982] Попов, Э.В. Общение с ЭВМ на естественном языке / Э.В. Попов. – Москва: Наука, 1982.
- [Ostis, 2012] Open Semantic Technology for Intelligent Systems. [Электронный ресурс]. – 2012. – Режим доступа: <http://www.ostis.net/>. – Дата доступа: 20.12.2013.

SEMANTIC CONCEPTUAL DESIGN OF NATURAL LANGUAGE INTERFACES FOR INTELLIGENT SYSTEM

Yeliseyeva O.E. *

**Belarusian State University, Minsk, Republic
of Belarus*

olae@open.by

In the work conceptual design of natural language interfaces of intelligent system based on Open Semantic Technology for Intelligent Systems OSTIS is performed. Exposition of the main results is offered directly on graph semantic language SC, thereby fragment of knowledge base for natural language interface subsystem is formed. Visualization of knowledge about the process of communication and natural language as a semantic network is a tool for a better understanding of the relevant domains.

Introduction

Urgency of creation of natural language interface to any intelligent system is quite obvious. However, until now an open question is mass usage of natural language interface. The main reasons for this situation are:

- The widespread use of the standard graphical interface with multiple toolbars in the form of buttons, switches, forms, etc., to which all users have become accustomed. Therefore there is a problem an intelligent combination of standard interface tools with natural-language;

- A significant number of problems are associated with the complexity of the automatic analysis and synthesis of natural language. Object-oriented application systems, as well as the absence of a universal technology of natural language interface cause the that each system necessary to create natural-language interface virtually from scratch that is too costly and resource intensive.

Main Part

Natural language interface is a computer (software) model of the communication process. The basis of this model is the two subjects of communication: the user and intelligent system. In the process of communicating subjects are exchanging messages. In the design process of natural language interface we must define a typology of messages. Next we have to formulate the rules of analysis of input messages.

A process of messaging user and intelligent system is implemented in the form of a dialogue. When it's creating a dialogue systems, general and specific structure of dialogue is designed.

The central element of the model of communication is the natural language. Relationships between objects, events, etc. of the world, as well as many of their characteristics, we set either using specially introduced signs and symbols of the language knowledge representation SC, or by using appropriate key sc-node identifiers words of natural language. Sc-texts contain elements of natural language, which may well be used in the natural language of communication with the user.

Conclusion

In this paper we presented a semantic conceptual modeling of natural language interface for intelligent system. As a result the fragments of knowledge base of natural language subsystem were specified as a semantic network. Unfortunately, the article format does not allow to provide full ontology of concepts of natural language interface. Some raised in the work questions require a more thorough thinking by involving of experts from the field of linguistics.

In addition, one of the objectives of this work was to illustrate the linguistic experts, as well as professionals of any other domains, a gradual process of formalization of knowledge represented on natural language. All narrative in the work built on the principle of translation of thoughts formulated in natural language to the semantic knowledge representation language.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК [004.522+004.934+004.91]:004.89

МЕТАД ПАБУДОВЫ КАМПАМЕНТАЎ СІНТЭЗУ МАЎЛЕННЯ ПА ТЭКСЦЕ ДЛЯ НАТУРАЛЬНА-МАЎЛЕНЧАГА ІНТЭРФЕЙСА ПРЫ ДАПАМОЗЕ NOOJ

Гецэвіч Ю.С. *, Скопінава А.М. *, Окрут Т.І. *

** Аб'яднаны інстытут праблем інфарматыкі Нацыянальнай акадэміі навук Беларусі,
г. Мінск, Беларусь*

{yury.hetseвич, skelena777, tatberrie}@gmail.com

У дадзеным артыкуле акрэсліваецца падыход да пабудовы адасобленых кампанентаў для натуральна-маўленчага інтэрфейса інтэлектуальных сістэм праз вырашэнне камп'ютэрна-лінгвістычных задач сінтэзу маўлення, у прыватнасці праз праграмны сродак NooJ. Аўтарамі пазтапна апісаныя і алгарытмічна прадстаўленыя рашэнні задачы пошуку і класіфікавання колькасных выразаў з адзінкамі вымярэння, а таксама задачы аўтаматызаванага афармлення дыялогаў.

Ключавыя словы: кампанент; канчатковы аўтамат; NooJ; колькасны выраз з адзінкай вымярэння; ідэнтыфікацыя выразаў; агучванне дыялогаў.

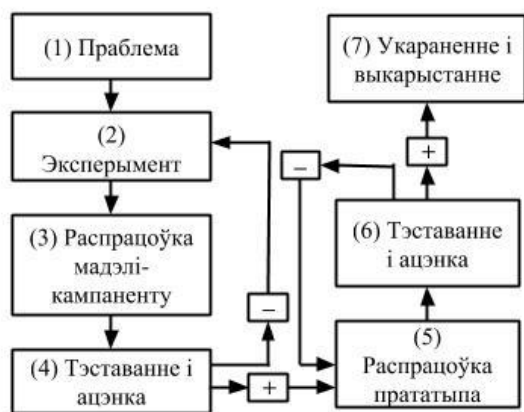
Уводзіны

Натуральна-маўленчы інтэрфейс інтэлектуальных сістэм патрабуе інтэграцыі складаных кампанентаў і модуляў для забеспячэння якаснага ўзаемадзеяння між карыстальнікам і сістэмай. Карыстальнік хоча чуць ад сістэмы агучаны тэкст самых шырокіх тэматычных даменаў у форме звычайнага маўлення. Для вырашэння гэтай агульнай задачы патрэбна звярнуцца да сінтэзатара маўлення па тэксце, які мае добра распрацаваны лінгвістычны працэсар тэксту. Аўтары артыкула прапануюць выкарыстаць праграмны сродак NooJ [NooJ, 2002] і апісаць метады для распрацоўкі якасных кампанентаў для паляпшэння лінгвістычнага працэсара сінтэзатара маўлення па тэксце праз вырашэнне розных камп'ютэрна-лінгвістычных задач.

Пад камп'ютэрна-лінгвістычнай задачай па электронным тэксце будзем разумець такую задачу (праблему), якая, па-першае, ставіцца адносна электроннага тэксту; па-другое, закранае пытанні канкрэтнага пошуку, класіфікацыі ці перапрацоўкі паслядоўнасцяў знакаў і сімвалаў жаданага электроннага тэксту; па-трэцяе, яе канчатковым рашэннем павінна быць камп'ютэрная праграма для папярэдняй апрацоўкі тэксту, праца якой можа быць правярана карыстальнікам на неабмежаванай колькасці іншых электронных тэкстаў [Гецэвіч і інш., 2013а]. Такое вызначэнне з'яўляецца абагульненым падыходам да шэрагу задач, якія фармуляваліся і вырашаліся ў працах [Гецэвіч,

2011], [Гецэвіч і інш., 2012], [Гецэвіч і інш., 2013b], [Skopinava et al., 2013].

Разгледзім пазтапна працэс ад пастаноўкі пэўнай задачы да яе рашэння з улікам камп'ютэрных сродкаў і ўмоў. На малюнку 1 відаць, што дадзены працэс прадугледжвае ажыццяўленне сямі этапаў ад вызначэння задачы да стварэння прадукта (яе рашэння) для карыстальніка. Такім чынам, спачатку даследчыкам фармулюецца праблема (1). Пасля гэтага эксперыментальнымі шляхамі (пастаноўка і аспрэчванне гіпотэз) эксперт знаходзіць рашэнні дадзенай праблемы адносна невялікага фрагмента тэксту (2); для гэтых рашэнняў мадэлююцца кампаненты, якія прадугледжваюць наяўнасць пэўных рэсурсаў і алгарытмаў у выглядзе канчатковых аўтаматаў NooJ (3) [NooJ, 2002]. Затым атрыманыя канчатковыя аўтаматы тэстуюцца на адносна вялікай колькасці тэкстаў (4). Калі вынікі тэставання паводле адзнак дакладнасці і паўнаты не здавальняючыя, то мадэлі-кампаненты вяртаюцца для дапрацоўкі на стадыю (2); у процілеглым выпадку яны перадаюцца для стварэння на іх базе эксперыментальна-праграмных комплексаў (5). Пры гэтым тэставыя дадзеныя, распрацаваныя на этапе (4), выкарыстоўваюцца і на этапах (5) і (6) для дакладнай распрацоўкі, тэставання і ацэнкі праграма прадукта. Пры становачай ацэнцы (у межах дапушчальнай памылкі) праграма трапляе ў рукі карыстальніка (7) і набывае статус канчатковага прадукта, а пры адмоўнай – адбываецца вяртанне на стадыю (5).



Малюнак 1 – Абагульненая схема працэсу вырашэння камп’ютэрна-лінгвістычнай задачы

Пад карыстальнікам будзем разумець або экспертаў-лінгвістаў, або прамых карыстальнікаў, якія могуць ужываць атрыманыя праграмныя сродкі для папярэдняй апрацоўкі тэксту. У пацверджанне выкарыстанасці метаду, які прыводзіцца, разгледзім рашэнні дзвюх розных камп’ютэрна-лінгвістычных задач у прыкладанні да сінтэзу маўлення па тэксце.

1. Апрацоўка колькасных выказаў з адзінкамі вымярэння

Важна, каб натуральна-маўленчы інтэрфейс інтэлектуальнай сістэмы мог агучыць наступныя літарна-знакавыя выразы: $8,024129(3) \cdot 10^{23} \text{ моль}^{-1}$, 44 мА , $-95 \text{ }^\circ\text{C}$, $3 \text{ тыс. гадоў назад}$ і г.д. Колькасныя апісанні ўласцівых і агульнай навуковай карціне свету, і побытавай сферы жыцця. Апроч складанаструктураванасці, усюдыіснасці, яны характарызуюцца і варыятыўнасцю формаў напісання і пазначэння. Так і ўзнікае неабходнасць спецыяльна распрацаваць паўнаважныя кампаненты алгарытмаў з рэсурсамі для апрацоўкі КВАВ у прыкладанні да сінтэзу маўлення па тэксце.

Перавагай сістэмы NooJ датычна задачы пошуку і апрацоўкі КВАВ з’яўляецца тое, што распрацаваныя з дапамогай убудаванага візуальнага рэдактара і ў выглядзе канчатковых аўтаматаў кампаненты валодаюць навочнасцю, дзякуючы чаму іх можна адносна лёгка карэктаваць і папаўняць, што выключна важна ў сілу вышэй пералічаных уласцівасцяў КВАВ.

Вынікаючы з вышэй апісанай у частцы 1 схемы вырашэння камп’ютэрна-лінгвістычнай праблемы, для пачатку выразна сфармулюем задачу (1): знайсці, класіфікаваць і апрацаваць у электронных тэкстах колькасныя выразы з адзінкамі вымярэння. На этапе (2) экспертам праводзіліся назіранні і аналіз КВАВ адносна іх будовы і выкарыстання на матэрыяле электронных тэкстаў навукова-тэхнічнага і прававога тэматычнага дамена на беларускай і рускай мовах. На дадзены момант на этапе (3) аўтарамі змадэляваныя тры ўзаемадапаўняльныя складаныя кампаненты з алгарытмамі і рэсурсамі для пошуку колькасных

выказаў з адзінкамі вымярэння ў вялікіх карпусах тэкстаў, якія дазваляюць: знаходзіць КВАВ і класіфікаваць іх па трох тыпах паводле міжнароднай сістэмы адзінак СІ (асноўныя, вытворныя, пазасістэмныя); знаходзіць КВАВ з метралагічнымі прыстаўкамі (кратнымі ці дольнымі, скарачанымі ці ў поўнай форме) і класіфікаваць іх паводле словаўтваральных асаблівасцяў; пераўтвараць КВАВ у арфаграфічныя словы (малюнак 2).

Этап тэставання (4) паказаў, што, напрыклад, першы кампанент дае пошукавыя вынікі з дакладнасцю ў 72 %. Гэты адносна высокі паказчык даў падставу для распрацоўкі эксперыментальнага праграмнага комплексу (5), які б знаходзіў КВАВ і класіфікаваў іх па трох тыпах адносна міжнароднай сістэмы адзінак СІ (асноўныя, вытворныя, пазасістэмныя). Праграма атрымала назву “QEMU Identifier”, г.зн. “Quantitative Expressions with Measurement Units Identifier” ці “Ідэнтыфікатар колькасных выказаў з адзінкамі вымярэння”.

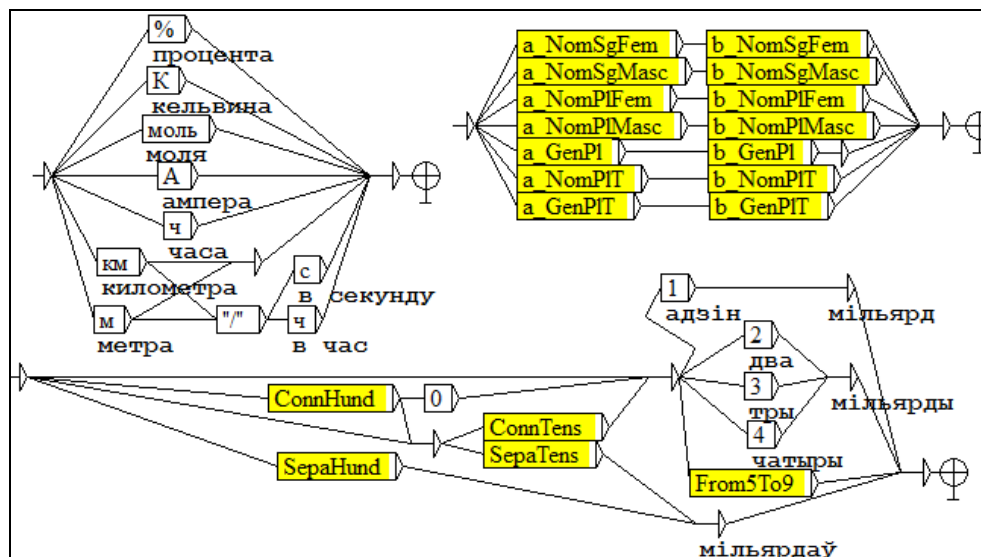
Разгледзім працу QEMU Identifier на прыкладзе аналізу фрагментаў беларуска- і рускамоўных тэкстаў з адпаведных навукова-тэхнічных тэкставых карпусоў. Тэксты для аналізу можна ўводзіць адвольна ў верхняе поле або загружаць тэкставыя файлы (у фармаце txt) праз адпаведную каманду ва ўкладцы меню.

Каманда “*Пераўтварыць*” (Transform) ажыццяўляе пошук КВАВ і прысвойвае ім пэўныя маркеры. Да прыкладу, на малюнку 3а было ідэнтыфікавана 40°C як $\langle \text{MEAS} + \text{Temperature in Celsius scale} + D \rangle$. Гэта азначае, што дадзены літарна-знакавы набор з’яўляецца колькасным выразам з адзінкай вымярэння тэмпературы па шкале Цэльсія, прычым па стандартах сістэмы СІ дадзеная адзінка з’яўляецца вытворнай.

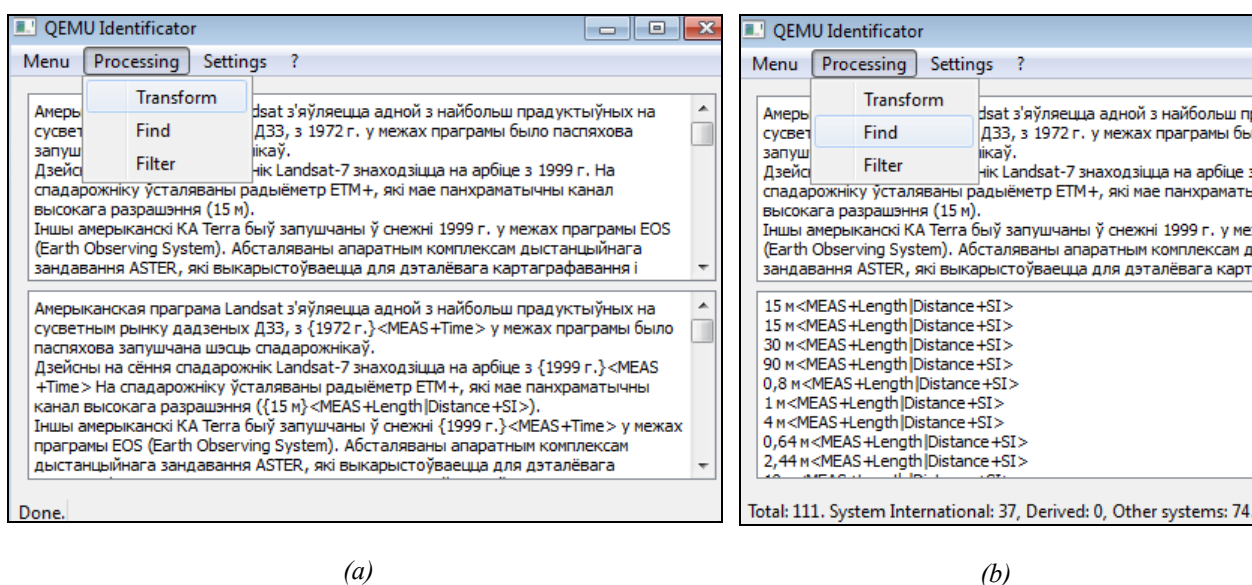
Каманда “*Знайсці*” (Find) на выйсці выдае спіс усіх знойдзеных КВАВ і падае колькасныя звесткі (малюнак 3б). У дадзеным тэставым тэксце знайшлося 111 КВАВ, з якіх 37 выказаў уключаюць асноўныя мерныя адзінкі СІ, 0 выказаў з вытворнымі ад СІ адзінкамі, а 74 колькасныя выразы змяшчаюць пазасістэмныя адзінкі.

Выкананне каманды “*Фільтраванне*” (Filter) дае магчымасць здзяйсняць разнастайныя пошукавыя запыты праз фармальную мову рэгулярных выказаў. Напрыклад, пасля ўводу *Voltage|Frequency* спіс знойдзеных КВАВ абмяжоўваецца толькі тымі, якія ўтрымоўваюць адзінкі вымярэння або электрычнай напругі, або чашчыні току (малюнак 3с): 1 Гц , 50 В , 150 В .

Зараз праграма QEMU Identifier тэстуецца распрацоўшчыкамі і карыстальнікамі. У далейшым плануецца палепшыць гэты праграмны прадукт, а таксама праграма рэалізаваць астатнія кампаненты-мадэлі для пошуку колькасных выказаў з адзінкамі вымярэння.



Малюнок 2 – Кампанент для пераўтварэння колькасных выказаў з адзінкамі вымярэння ў арфаграфічныя паслядоўнасці слоў



Малюнак 3 – Выкананне каманд “Пераўтварыць” (a), “Знайсці” (b), “Фільтраванне” (c) для ідэнтыфікацыі КВАВ у электронных тэкстах на беларускай (a, b) і рускай (c) мовах праз убудаваны ў праграму OEMU Identifier марфалагічныя і сінтаксічныя кампаненты NooJ

2. Ідэнтыфікацыя і шматгалосая агучка дыялогаў у электронных тэкстах

Дзякуючы ўбудаванай у NooJ магчымасці анатаваць з дапамогай сінтаксічных кампанентаў электронны тэкст стала магчымым рашэнне яшчэ адной камп'ютэрна-лінгвістычнай задачы – ідэнтыфікаваць і агучваць рознымі сінтэзаванымі галасамі дыялогі розных персанажаў. Гэта дазволіць зрабіць натуральна-маўленчы інтэрфейс больш адпаведным семантычнаму сэнсу, які ён перадае.

Першым крокам для вырашэння дадзенай праблемы была распрацоўка кампанентаў для аўтаматычнай ідэнтыфікацыі прастай мовы ў тэксце. Эксперты прааналізавалі тэкставы матэрыял на прадмет выяўлення структур афармлення прастай мовы, якія пасля закладваліся ў сінтаксічны кампанент NooJ [Гецэвіч і інш., 2013с]. Атрыманыя вынікі ляглі ў аснову кампанента *DS All* (малюнак 4), які спрацоўвае тады, калі прастая мова пачынаецца з працяжніка, затым ідуць словы персанажа; пераход ад слоў аўтара і наадварот пазначаецца праз камбінацыю працяжніка з коскай, кропкай, клічнікам, пыталнікам ці іх спалучэннямі:

1. Словы персанажа (М) без слоў аўтара (А):
 $M(?!|!!!|?|?!|...|.)$.
2. Словы персанажа са словамі аўтара ў канцы:
 $M(?!|!!!|?|?!|...|.)-A(...|.)$.
3. Словы персанажа з адной ці некалькімі аўтарскімі ўстаўкамі:
 $M(?!|!!!|?|?!|...|.)-A(,|...|.|:|.)-M(?!|!!!|?|?!|...|.)(-A(,|...|.|:|.)-M(?!|!!!|?|?!|...|.))$.

Калі пасля слоў персанажа ідуць словы аўтара, алгарытм працягвае пошук завяршэння фразы дыялогу. У падграфіях *Speaker* і *Author* асобна разглядаюцца ўнутраныя знакі прыпынку, прычым аўтарскі тэкст мае іх меншую варыянтнасць: коска, кропка, шматкроп'е і дужкі.

Наступным крокам стала ідэнтыфікацыя роду персанажаў пасродкам аналізу слоў аўтара, якія характарызуюцца наяўнасцю такіх індикатараў, як дзеясловы мінулага часу адзіночнага ліку (*казаў, казала*); імёны ўласныя (*Алесь, Майка*); назоўнікі, якія абазначаюць размоўцу (*бацька, дзяўчынка*). З трэніравальнага тэксту былі вынятыя рэплікі з аўтарскімі ўстаўкамі і размечаныя па родзе. Затым з улікам знойдзеных індикатараў быў створаны канчатковы аўтамат для ідэнтыфікацыі роду. Для гэтага ў падграф *Author* былі дададзеныя родазалежныя падграфы *VERBSfeminine* (малюнак 5) і *VERBSmasculine* для вызначэння жаночага і мужчынскага роду. Пашырэнне спісу дзеясловаў-індикатараў прывяло да стварэння цэлага рэсурсу ў выглядзе лінгвістычнага слоўніку (малюнак 6). У ім парамі прадстаўленыя дзеясловы мінулага часу ў формах жаночага і мужчынскага роду.

Падчас распрацоўкі алгарытмаў даследавалася і праблема граматычных амонімаў (амаформаў). Так, дзеяслоў *кажа* можа належаць і мужчынскаму, і

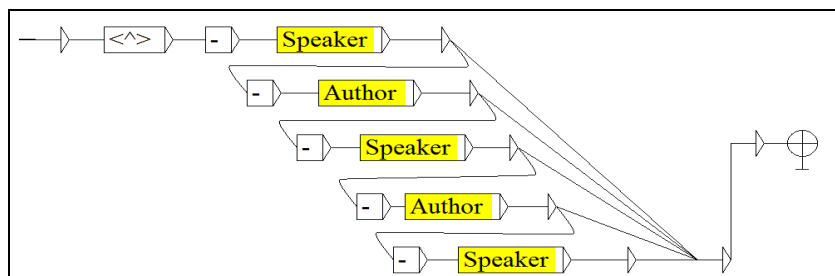
жаночаму роду. У выніку праз такія формы паасобку нельга высветліць катэгорыю роду персанажа. Таму быў створаны дадатковы звязак графаў “дзеяслоў-назоўнік”, дзе першы граф уключае граматычныя амаформы маўленчых дзеясловаў, а другі граф падае спіс выражаных назоўнікамі паказчыкаў роду. Атрыманыя графы прымяняюцца паслядоўна да мэтавага тэксту праз NooJ, прычым пазнакі аднаго захоўваюцца падчас спрацоўвання другога. Такім чынам, на выйсці тэкст атрымлівае разметку па рэпліках для мужчынскага і жаночага галасоў.

Пасля таго як дакладнасць кампанентаў на трэніравальным тэкставым матэрыяле (звыш 100 тыс. словаўжыванняў з твору У.С. Караткевіча “Каласы пад сярпом тваім”) дасягнула больш 95%, быў створаны тэставы корпус з урыўкаў твораў мастацкай літаратуры аб'ёмам каля 24 тыс. словаўжыванняў. Па падліках эксперта тэставы корпус утрымоўвае 481 рэпліку (рэплікі з працяжнікамі), 165 рэплік персанажаў мужчынскага роду і 68 рэплік персанажаў жаночага роду. Тэставанне на тэкставым корпусе паказала, што кампаненты ідэнтыфікуюць простую мову ў электронных тэкстах і вызначаюць род персанажаў з дакладнасцю вышэй за 90%, што дазваляе пачаць іх укараненне ў сістэму сінтэзу маўлення па тэксце (СМТ). Па-першае, каб выкарыстанне ў сістэме СМТ пад стандарт SAPI 5.1 стала магчымым, неабходна прывесці тэксты да выгляду SAPI TTS XML [XML TTS Tutorial, 2013]. Па-другое, для выбару сістэмай адпаведнага голасу сінтаксічная анатацыя, якую генеруе кампанент, павінна быць адаптаваная пад наступны код:

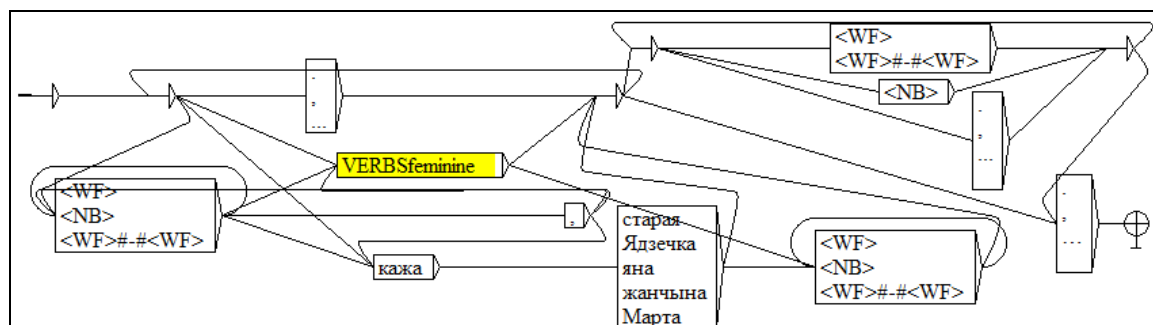
<VOICE Required="name=[Назва голасу ў сістэме]">...Тэкст для агучвання...</VOICE>.

Для гэтага ў падграфы *DS_M* і *DS_F* былі ўстаўленыя маркеры пазначэння шляхоў, па якіх спрацоўваюць кампаненты. Маркеры наладжаныя так, што неразмечаны тэкст (малюнак 7а) і словы аўтара агучваюцца голасам *AlesiaBel*, мужчынскія рэплікі – голасам *BorisBel*, а жаночыя – *ElenaBel*. Апрацаваныя кампанентамі сказы набываюць выгляд, які дэманструецца малюнкам 7б. Нарэшце ўжо размечаны тэкст можна падаваць на ўваход сістэмы СМТ. На малюнку 7с адлюстравана, як праграма SAPI5 TTSAPP аўтаматычна пераключае пастаўленыя ў сістэме галасы *AlesiaBel*, *BorisBel*, *ElenaBel* з дапамогай опцыі *Process XML*.

Такім чынам, распрацаваныя мадэлі-алгарытмы паказалі станоўчыя вынікі ў спалучэнні з сістэмай СМТ і ў далейшым могуць выкарыстоўвацца для стварэння дадатковага блока аўтаматычнага выбару мужчынскага альбо жаночага голасу сістэмы СМТ. Шматгалосы сінтэзатар маўлення па тэксце дасць магчымасць натуральна-маўленчаму інтэрфейсу захоўваць індывідуальныя асаблівасці персанажаў, якія гавораць між сабой у паведамленні.



Малюнак 4 – Сінтаксічны кампанент для аўтаматычнай ідэнтыфікацыі прастай мовы ў тэксе



Малюнак 5 – Сінтаксічны падкампанент для вызначэння рэплік персанажаў жаночага роду

трымаў, VERB+SpeechAct+Masculine
 трымала, VERB+SpeechAct+Feminine
 ударыў, VERB+SpeechAct+Masculine+FLX=ŷVERB1
 ударыла, VERB+SpeechAct+Feminine+FLX=ŷVERB1

Малюнак 6 – Фрагмент слоўніка індэксараў роду, выражаных дзеясловамі мінулага часу

(a) - Можа, і ёсца тут праўда... - амаль з вяковым прыдыханнем сказала яна. Зусім чыста па-мужыцку
 - Ты што ж... І гаварыць можаш? - спытаў ён. - Чаго ж прыкідвалася?

↓ ↓ ↓

(b) 2 <VOICE Required="name=ElenaBel">- Можа, і ёсца тут праўда...</VOICE> <VOICE Required="name=AlesiaBel">- амаль
 з вяковым прыдыханнем сказала яна. Зусім чыста па-мужыцку...</VOICE>
 3 <VOICE Required="name=BorisBel">- Ты што ж... І гаварыць можаш?</VOICE> <VOICE Required="name=AlesiaBel">-
 спытаў ён.</VOICE> <VOICE Required="name=BorisBel">- Чаго ж прыкідвалася?</VOICE>

↓ ↓ ↓

(c)
 Mouth Position: Яна вагалася. Нават уздыхнула ў цёпры. І тут ён пачуў нешта такое...
 <VOICE Required="name=ElenaBel">- Можа, і ёсца тут праўда...</VOICE> <VOICE Required="name=AlesiaBel">- амаль з вяковым прыдыханнем сказала яна. Зусім чыста па-мужыцку...</VOICE>
 <VOICE Required="name=BorisBel">- Ты што ж... І гаварыць можаш?</VOICE> <VOICE Required="name=AlesiaBel">- спытаў ён.</VOICE>
 <VOICE Required="name=BorisBel">- Чаго ж прыкідвалася?</VOICE>

Малюнак 7 – Працэс аднаўлення электроннага тэксту (a) праз VoiceXML (b) для аўтаматычнага пераключэння галасоў да шматгаласога агучвання (c) аўтаматычна размечанага тэксту

Заклучэнне

Такім чынам, быў прапанаваны метада пабудовы адасобленых кампанентаў для натуральна-маўленчага інтэрфейса інтэлектуальных сістэм праз вырашэнне дзвюх камп'ютэрна-лінгвістычных задач сінтэзу маўлення. Для гэтага выкарыстоўваўся наладжвальны лінгвістычны працэсар NooJ.

Распрацаваныя мадэлі рашэнняў з'яўляюцца самастойнымі незалежнымі кампанентамі, кожны з якіх можа або выкарыстоўвацца паасобку, або ўсе кампаненты разам могуць быць убудаваныя ў іншыя сістэмы. У будучыні аўтарамі плануецца павышэнне дакладнасці працы кампанента ўжо генеравання арфаграфічнага тэксту па колькасных выразках з адзінкамі вымярэння, а таксама распрацоўка кампанента ідэнтыфікацыі роду персанажаў у рэпліках без уставак слоў аўтару.

Бібліяграфічны спіс

[Гецэвіч і інш., 2013а] Гецэвіч, Ю.С. Мадэляванне і распрацоўка сістэм пошуку колькасных выказаў з адзінкамі вымярэння ў электронных тэкстах на беларускай і рускай мовах / Ю.С. Гецэвіч, А.М. Скопінава, А.Ф. Есіс // Развитие информатизации и государственной системы научно-технической информации (РИНТИ-2013) : доклады XII Международной конференции (Минск, 20 ноября 2013 г.). – Минск : ОИПИ НАН Беларуси, 2013. – С. 282–287.

[Гецэвіч, 2011] Гецэвіч, Ю.С. Аўтаматызаваная апрацоўка сімвалічных выказаў у тэкстах для сістэмы сінтэзу беларускага маўлення / Ю.С. Гецэвіч // Информатика. – 2011. – № 4. – С. 82–93.

[Гецэвіч і інш., 2012] Гецэвіч, Ю.С. Ідэнтыфікацыя выказаў з адзінкамі вымярэння ў навукова-тэхнічных і прававых тэкстах на беларускай і рускай мовах / Ю.С. Гецэвіч, А.М. Скопінава // Развитие информатизации и государственной системы научно-технической информации (РИНТИ-2012) : доклады XI Междунар. конф., Минск, 15 нояб. 2012 г. – Минск : ОИПИ НАН Беларуси, 2012. – С. 260–265.

[Гецэвіч і інш., 2013б] Гецэвіч, Ю.С. Кампаненты ідэнтыфікацыі колькасных выказаў з адзінкамі вымярэння ў тэкстах на беларускай і рускай мовах / Ю.С. Гецэвіч, А.М. Скопінава // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2013) : материалы III Междунар. науч.-техн. конф., Минск, 21–23 февр. 2013 г. – Минск : БГУИР, 2013 г. – С. 319–328.

[Skopinava et al., 2013] Skopinava, A.M. Processing of quantitative expressions with units of measurement in scientific texts as applied to Belarusian and Russian text-to-speech synthesis / Yu. S. Hetsevlch, A.M. Skopinava, B.M. Lobanov // Компьютерная лингвистика и интеллектуальные технологии : материалы Междунар. конф. «Диалог», Московская обл., г. Бекасово, 29 мая – 2 июня 2013 г. – Вып. 12 (19). – В 2 т. – Т.1. – М. : Изд-во РГГУ, 2013. – С. 634–651.

[NooJ, 2002] Лінгвістычны працэсар NooJ [Электронны рэсурс]. – 2002. – Рэжым доступу : <http://www.nooj4nlp.net/pages/nooj.html>. – Дата доступу : 01.07.2012.

[Гецэвіч і інш., 2013с] Гецэвіч, Ю.С. Аўтаматызацыя шматгаласавога стварэння аўдыёкніг на беларускай мове з дапамогай сінтэзатараў маўлення па тэксце / Ю.С. Гецэвіч, Т.І. Окрут, Б.М. Лабанаў // Развитие информатизации и государственной системы научно-технической информации (РИНТИ-2013) : доклады XII Международной конференции (Минск, 20 ноября 2013 г.). – Минск : ОИПИ НАН Беларуси, 2013. – С. 269–276.

[XML TTS Tutorial, 2013] XML TTS Tutorial (SAPI 5.3) // Microsoft Developer Network [Electronic resource]. – 2013. – Mode of access : <http://msdn.microsoft.com/en-us/library/ms717077%28v=vs.85%29.aspx>. – Date of access : 29.07.2013.

METHOD OF CONSTRUCTING TEXT-TO-SPEECH COMPONENTS FOR NATURAL LANGUAGE INTERFACES WITH THE HELP OF NOOJ

Hetsevlch Yu.S.^{*}, Skopinava A.M.^{*}, Okrut T.I.^{*}

^{*}*United Institute of Informatics Problems of the National Academy of Sciences, Minsk, Republic of Belarus*

{yury.hetsevlch, skelena777, tatberrie}@gmail.com

This article outlines an approach to construction of detached components for natural language interfaces through solution of computer-linguistic problems by means of the linguistic processor NooJ. The authors focus on two different problems and describe stage-by-stage solutions to them.

INTRODUCTION

In order to make text interfaces more “natural”, systems of human computer interaction should be able to voice electronic texts. High-quality text-to-speech synthesis cannot be achieved without solving various computer-linguistic problems. Under the term “a computer-linguistic problem”, we mean a task, which refers to electronic texts; and concerns identifying, classifying, and processing of words and symbol sequences; to solve it means to develop a program for preliminary text processing.

MAIN PART

The first problem consists in processing quantitative expressions with measurement units (QEMU). By the moment three complementary blocks of components and resources have been built for Belarusian and Russian. They allow: identifying and classifying QEMU according to the International Bureau of Weights and Measures (expressions with SI-basic, SI-derived, and off-system measurement units); classifying QEMU according to word formation peculiarities (full or shortened, multiple or submultiple prefixes); expanding QEMU into orthographic words.

The second problem lies in many-voiced scoring of dialogues in Belarusian and Russian. The authors have created a component with algorithms and resources, which choose and switch voices automatically for systems of text-to-speech synthesis. Multi-voiced scoring of dialogues allows voicing electronic books, and preserving sound distinction among personages.

CONCLUSION

We have described a general approach to construction of components for natural language interfaces with the help of NooJ. In future it is planned to expand the resources in order to cover wider groups of measurement units, and to extract more gender indicators.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

UDK 681.3.

ЗНАНИЕ-ОРИЕНТИРОВАННЫЕ СРЕДСТВА ПОДДЕРЖКИ СЕМАНТИЧЕСКОГО ПОИСКА В WEB

Рогошина Ю.В.

Институт программных систем НАН Украины, Киев, Украина

ladamandraka2010@gmail.com

В статье рассматриваются основанные на семантике направления развития поиска в Web. Проанализированы средства представления знаний (онтологии и тезаурусы), методы их пополнения и сопоставления. Предложены способы применения персональной и коллаборативной информации при семантическом поиске.

Ключевые слова: семантический поиск, онтологическая модель, тезаурус, индуктивный вывод, рекомендующие системы, коллаборативный поиск.

Введение

Значительная часть современных приложений ориентированы на функционирование в открытой информационной среде, в частности, на извлечение из ресурсов Web знаний об интересующей пользователя предметной области (ПрО). Таким образом, проблема поиска в Web оказывается составной частью самых разных информационных систем, а общая тенденция экспоненциального увеличения объема доступной через Web информации обуславливает потребность в переходе при информационном поиске от обработки больших объемов данных к обработке знаний – более компактных, но имеющих значительно более сложную структуру.

В связи с этим возрастает актуальность как методов извлечения знаний, так и их повторного использования. Основой для интероперабельного представления знаний может послужить онтологический анализ, базирующийся на стандартах и программных средствах, разработанных в рамках проекта Semantic Web.

То, что онтологическое моделирование является адекватным средством для описания различных ПрО, является на сегодня общепризнанным фактом, а широкий выбор онтологий, доступных через Web, подтверждает популярность этого подхода среди различных групп разработчиков и пользователей Web-приложений.

При этом возникает ряд вопросов, связанных с формализацией семантики, в частности, построения формализованной семантической модели знаний, которая обеспечивает однозначную и

автоматизируемую их интерпретацию. На сегодня уже разработан и используется ряд достаточно эффективных средств семантического моделирования, позволяющих адекватно отображать сведения о реальном мире, например, RDF Schema, однако их выразительные возможности требуют расширения. Такие возможности предоставляют онтологии. Онтология – это система, состоящая из множества понятий и множества утверждений об этих понятиях, на основе которых можно строить классы, объекты, отношения и теории [Бездушный, 2003]. Онтологическое позволяет интероперабельно описать предметную область.

Важным вопросом является выбор языка для представления онтологий, который основывается на том, какие именно логические формализмы лежат в их основе. При этом онтология рассматривается как логическая теория.

Необходимо, чтобы логический язык предоставлял достаточно выразительных возможностей для формализации знаний в описываемой ПрО. Кроме того, следует учитывать разрешимость и вычислительную сложность тех онтологий, которые могут быть созданы при помощи такого языка, так как от этого зависит сама возможность создания программных средств для обработки таких онтологий и необходимые для этого вычислительные ресурсы. В настоящее время для описания онтологий используются такие логические формализмы, как логика предикатов первого порядка (в языке Ontolingua), фреймовая логика (в языке F-Logic) и дескриптивная логика (в языках DAML+OIL, OWL DL). Разрешимость логического языка гарантирует получение ответа от

системы логического вывод, но время его получения зависит от вычислительной сложности языка.

Чем выше выразительность логического языка, тем точнее он позволяет описывать ПрО, но тем больше времени затрачивается на логический вывод. Различные дескриптивные логики обладают различной выразительной мощностью. Поэтому использование логических языков представления онтологий на основе дескриптивных логик позволяет обеспечить приемлемый компромисс между выразительными возможностями языка и необходимыми для обработки таких онтологий вычислительными ресурсами: можно в зависимости от задачи выбрать язык с достаточной выразительностью и минимальной вычислительной ресурсоемкостью. OWL DL базируется на дескриптивной логике SHIQ и обеспечивает достаточно точное прогнозирование сходимости и времени работы различных методов обработки знаний, представленных в виде онтологий.

Онтологический анализ в семантическом поиске

При традиционном информационном поиске производится сопоставление запроса пользователя, характеризующего его информационную потребность, со сведениями о контенте проиндексированных информационных ресурсов (ИР), и в ответ на запрос пользователя формируется группа информационных объектов, известных ИПС и по тем или иным параметрам сопоставленных с запросом, т.е. $I = \{i_j, j = \overline{1, n}\} = f(z, DB_{unc})$.

При этом субъектами обработки являются пары (запрос, ИР), а сопоставление производится унифицированно для всех пользователей. Такой поиск обычно основывается на обнаружении лексического соответствия ключевых слов и терминов, содержащихся в документе, на основе минимальных предварительных знаний, рассматривая текст как набор слов

Как правило, такой поиск применяют для обработки неструктурированной текстовой информации. Кроме того, часто учитывается расстояние между обнаруженными терминами. Для структурированных документов имеет значение месторасположение ключевых слов (например, в заголовке или в метаописании).

Анализ исследований в области информационного поиска показал, что дальнейшее усовершенствование средств описания и алгоритмов сопоставления запросов и ИР практически не улучшает ситуацию, т.к. для более эффективного поиска нужно использовать дополнительные знания – о ПрО, интересующей пользователя, о сообществах пользователей с подобными интересами и о качестве ИР.

При персонифицированном поиске субъектами обработки являются уже тройки (запрос,

пользователь, ИР) $I_{pers} = f(z, u, DB_{unc})$. При этом сопоставление производится уже по-разному для различных пользователей в зависимости от их персональных предпочтений. Сведения о пользователе хранятся в БД поисковой системы и могут содержать его формальные характеристики (например, перечень естественных языков, которые знакомы пользователю), историю ранее выполненных запросов и знания об интересующих его предметных областях.

К сожалению, не всегда опыта конкретного пользователя достаточно для того, чтобы определить полезность тех или иных ресурсов (например, при обращении к новой ПрО или к новой группе ресурсов). В таком случае полезным может оказаться опыт других пользователей, производивших ранее поиск в той же области и имеющих сходные информационные потребности. При коллаборативном поиске целесообразно использовать методы, применяемые в рекомендующих системах. Если мы говорим о семантическом поиске, то предполагается, что информация в процессе сопоставления должна обрабатываться на семантическом уровне, с использованием знаний (о пользователе, ресурсах, предметной области и т.д.).

При простом знание-ориентированном поиске при обработке учитываются не только формальные сведения о запросе, пользователе и ИР, но и более сложно структурированные знания о них. Тогда при их сопоставлении необходимо будет оценивать степень подобия этих знаний. В частности, при онтологическом подходе к представлению знаний для каждого ИР может указываться онтология ПрО, характеризующая его контент, а для пользователя – онтология интересующей его ПрО, а при сопоставлении ИР и запроса необходимо будет выполнить сопоставление этих двух онтологий.

Семантический поиск определяется как метод информационного поиска, в котором релевантность документа запросу определяется семантически (по близости смысла), а не синтаксически (по встречаемости ключевых слов в документе).

То, какие именно знания используются, как они представлены и как они обрабатываются, зависит как от специфики разрабатываемой ИПС, так и от концепции, выбранной ее разработчиками, но в общем случае $I_s = \{i_j, j = \overline{1, n}\} = f(z, DB_{unc}, KB_{unc})$.

При этом часто предоставляются возможности для нечеткого поиска (например, обрабатываются неправильно написанные ключевые слова), поиска с учетом контекста

Если же речь идет о семантическом поиске в Web, то следует учитывать, что при этом в Web могут быть размещены не только информационные объекты, среди которых осуществляется поиск, но и внешние базы знаний, используемые при поиске. Поэтому при создании таких систем следует

учитывать, что такие внешние БЗ могут менять контент, структуру и доступность независимо от разработчиков ИПС.

$$I_{web_s} = \{j, j = \overline{1, n}\} = f(z, DB_{ипс}, KB_{ипс}, \{KB_{web_k}, k = \overline{1, m}\}).$$

Следует учитывать, что сегодня многие ИПС (например, Google) стремятся накапливать и использовать опыт взаимодействия с конкретным пользователем. Но часто информационные потребности пользователя оказываются ограниченными во времени (например, накопив информацию для выбора нового телефона, пользователь покупает его и больше не нуждается в сведениях о телефонах, а ИПС продолжает предлагать их ему) либо вообще не связанными с ним (например, запрос выполняется по чьей-либо просьбе). Кроме того, часть своих информационных потребностей пользователь не хочет делать открытой информацией – к примеру, запросы, связанные с отдыхом или здоровьем не хочет смешивать с запросами по работе. Поэтому более целесообразно при выполнении запросов дать возможность пользователю включить его в один из своих профилей либо вообще не сохранять для дальнейшей обработки

Таким образом, можно выделить основные направления развития поиска: 1. от формального – к семантическому; 2. от унифицированному – к персонифицированному; 3. от индивидуальному – к коллаборативному; 4. от закрытого – к управляемому; 5. от монотонного – к тематическому (с учетом динамики и конечности информационных потребностей). Наиболее полно удовлетворить информационные потребности пользователя позволяет интегрированное использование всех этих возможностей, т.е. персонифицированный, Web-ориентированный коллаборативный поиск, основанный на знаниях. Как правило, такой поиск является надстройкой над уже существующими поисковыми системами (например, Google), которая позволяет переупорядочить результаты поиска.

Чтобы повысить эффективность поиска, целесообразно как можно более точно определить, к какой ПрО относится информационная потребность пользователя и какой именно тип информации необходим для ее удовлетворения. В первом случае пользователю нужно выбрать одну из существующих онтологий ПрО и при необходимости модифицировать ее в соответствии со спецификой его проблемы. Так как не только создание и модификация онтологий, но и анализ содержащихся в уже созданных онтологиях знаний представляет собой достаточно сложную задачу, то такой подход приемлем только в том случае, если в дальнейшем такая онтология будет использоваться для выполнения единичного запроса, а для достаточно большого числа связанных с этой ПрО запросов. Такая ситуация характерна, например, для научных исследований, когда пользователь старается обнаруживать новые публикации и разработки по интересующей его проблематике. Для

таких ситуаций характерно как достаточно глубокое и структурированное понимание пользователем специфики ПрО (что и позволяет ему создавать онтологическую модель ПрО), так и отсутствие полностью удовлетворяющей его потребностям и общепринятой онтологической модели ПрО, так как научные исследования проводятся именно в тех областях, где не для всех вопросов уже найдены удовлетворительные решения и именно с целью нахождения таких решений.

Во втором случае пользователю необходимо воспользоваться какой-либо онтологией или таксономией информационных объектов, информация о которых содержится в IP Web. Например, воспользовавшись организационной онтологией, пользователь может указать, что ему необходимы сведения об организациях с теми или иными свойствами, о людях, имеющих определенную квалификацию, либо о проектах, решающих какие-то проблемы.

Но и этих сведений недостаточно в том случае, если условиям запроса удовлетворяют много различных IP – для пользователя достаточно трудоемко самому анализировать все предложения информационно-поисковой системы (ИПС). Возникает проблема ранжирования списка найденных релевантных IP. Кроме уже широко используемых унифицированных подходов, когда такое ранжирование будет одинаковым для всех пользователей, целесообразно учитывать как персональный опыт конкретного пользователя, так и опыт того сообщества пользователей, которое пользователь признает авторитетным для себя именно в этом запросе. Следует отметить, что в большинстве случаев ИПС, использующие опыт тех или иных сообществ, либо обобщают информацию, поступающую от всей совокупности пользователей, либо самостоятельно вычлениают некоторую подгруппу пользователей (кластер), которую считают подобно тому пользователю, для которого выполняется запрос. При этом возникает три проблемы: 1) как правило, пользователю непонятно, по каким критериям формируется такая группа; 2) группа формируется одинаково для любых запросов пользователя; 3) пользователю не предоставляются средства для явного формирования такой группы.

Постановка задачи

Чтобы обеспечить для различных интеллектуальных приложений эффективный доступ к ресурса открытой информационной среде Web, необходимо разработать интегрированную формальную модель, обеспечивающую интероперабельное представление знаний о пользователях, ресурсах и специфике предметных областей (в частности, учитывающих Semantic Web и социальный Web). При разработке методов обработки представленных в этой модели знаний необходимо проанализировать уже существующие средства онтологического анализа, возможности

использования тезаурусов, методы индуктивного извлечения знаний и алгоритмы выработки рекомендаций на основе накопленного сообществом пользователей опыта об интересующей пользователя предметной области.

Семантический поиск в Google

В последнее время многие разработчики ИПС в той или иной степени декларируют применение онтологий и поддержку семантического поиска.

Рассмотрим, что понимают под семантическим поиском в Google его разработчики. Это достаточно важно, так как на сегодня именно Google является наиболее широко используемым средством поиска в Web. Семантический поиск Google включает три основных компонента [Amerland D., 2013]: адрес URI; RDF; значение (онтологию). При этом онтологии позволяют связывать друг с другом различные данные, описанные при помощи RDF и адресованные через URI. Кроме того, для поддержки семантического поиска в Google применяется новый инструмент, обеспечивающий семантический поиск, – Knowledge Graph (Сеть знаний), который обеспечивает связь между различными элементами проиндексированного контента ИП и позволяет объединять информацию из различных источников.

Knowledge Graph позволяет непосредственно на странице с результатами поиска получать информацию об объекте поиска и связанные с ним факты: справа от результатов поиска на экран выводится информационная панель, отображающая сведения о географических объектах, людях, фильмах и т.п. Это позволяет пользователю получить информацию, не переходя на сайт, послуживший источником информации.

Сейчас в базе находится (по различным оценкам) от 500 до 700 миллионов объектов и около 3.5 миллиардов связей между ними. Объем знаний, доступный различным национальным версиям инструмента, различается. Информация поступает в базу данных сервиса в основном из открытых источников: для России в основном из русскоязычной Википедии, а в США используется также онлайн-справочник CIA World Factbook и база структурированных данных Freebase, собранная из множества отдельных вики-проектов: Google в 2010 году приобрел компанию Metaweb Technologies – разработчика базы данных Freebase, которая на тот момент содержала около 12 млн. сущностей.

Предполагается, что Knowledge Graph позволяет поисковой системе понимать смысл запроса. При помощи Knowledge Graph Google не просто идентифицирует ключевые слова, но и понимает смысл поискового запроса и ищет информацию о нем в базе данных Google, которая содержит сведения о различных объектах — например, людях, местах, вещах, фильмах, произведениях искусства и

прочем. Сеть знаний является первой ступенью на пути Google к интеллектуальному поиску.

В Web доступна информация, поступающая из Web-сайтов, социальных сетей, локальных сетей, профилей, баз данных и собственно от Google Search. Однако этой информации недостаточно. Google использует правила вывода, чтобы определить, как информация объединяется в группы и что она значит. Эти правила формируются на основе того, как пользователи работают с данными в социальных сетях (распространяемый контент, комментарии и взаимодействие), форумах и при поиске в Google. Кроме того, отслеживается поведение пользователей на различных Web-сайтах и анализируются последовательности их действий. Важным источником информации служат также сервисы, определяющие местонахождение пользователей по сигналу GPS от их мобильных устройств, определение географического местоположения по IP-адресу и сведения, получаемые при взаимодействии устройств.

Основная причина, по которой Google отображает на своей странице с результатами поиска прямые ответы на запросы – дольше удерживать на этой странице пользователя, т.к. это время является важным параметром для рекламодателей. Конкуренты Google, в первую очередь Facebook, заметно упрочили свои позиции в борьбе за внимание пользователей — а следовательно, и за деньги рекламодателей: посещение Facebook занимает около 13% того времени, которое средний американец проводит в Сети, а на долю сервисов Google, включая YouTube, остается лишь 11%. Помимо этого Facebook в значительной степени перенял у Google ее бизнес-модель: персонализированные предложения в этой социальной сети — это не что иное, как контекстная реклама Google, формирующаяся под влиянием запросов и интересов пользователей.

Google подчеркивает, что персонализация поиска в сочетании с семантическим поиском дадут новый вариант релевантности выдачи. Тем не менее, следует отметить, что сейчас такая информация для запросов – не только на русском и украинском языках, но и на английском – предоставляется в ответ только на самые короткие и простые запросы и по охвату значительно меньше, чем набор статей в Википедии. Поэтому остается открытым вопрос, насколько будет полезен пользователям новый сервис Google и насколько на самом деле в нем задействован семантический поиск. Главная проблема для пользователей – отсутствие явной модели поиска и сведений об онтологии, используемой при поиске, что позволяло бы хоть в какой-то мере прогнозировать его результаты.

Кроме того, непонятно, какую политику применяет Google в тех случаях, когда ответы не однозначны и зависят не только от местоположения пользователя, языка, на котором он говорит, времени года и т.п., но и от более сложных и субъективных факторов (например, запрос «самые

красивые собаки» или «выдающиеся политики», на которые не может быть однозначного ответа).

Онтологическая модель взаимодействия пользователей и ресурсов в Web

Семантический поиск представляет собой надстройку над традиционным информационным поиском, в котором с целью повышения pertinентности поиска используется обработка знаний, которые касаются как самого пользователя и его информационных потребностей (персонализация поиска), так и об информационных ресурсах, среди которых осуществляется поисковая процедура. Таким образом, семантический поиск состоит из формирования информационных моделей пользователя, интересующей его предметной области, задачи, которую решает пользователь, и информационных моделей доступных информационных ресурсов (ИР), которые характеризуют их семантику, и их дальнейшего сопоставления.

Сейчас для интероперабельного представления различных знаний в Web широко применяются онтологии, обеспечивающие явное формализованное представление семантики представленной информации и обеспечивающие возможность логического вывода на них. Поэтому представляется целесообразным для поддержки семантического поиска разрабатывать именно онтологические модели взаимодействия пользователей и ИР в информационном пространстве Web, а также методы их сопоставления и пополнения.

Онтологическая модель пользователя представляет собой класс онтологии, экземплярами которого являются сведения о зарегистрированных в ИПС пользователях (рис.1).

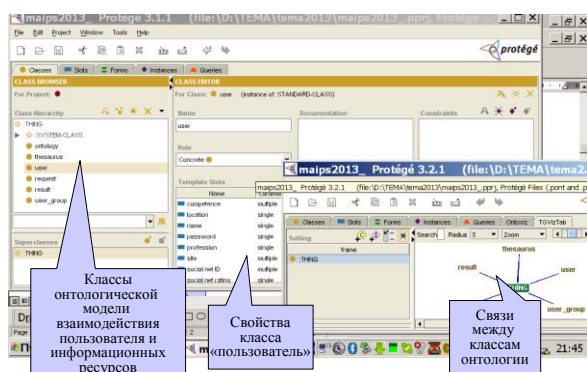


Рисунок 1 - Онтологическая модель взаимодействия пользователей и ИР

В онтологической модели взаимодействия пользователей и ИР описаны следующие классы:

- **онтология** ПрО, которая описывает область, к которой относятся информационные

потребности

пользователя

$$O_{\text{ПрО}_i} = \langle T_{\text{ПрО}_i}, R_{\text{ПрО}_i}, F_{\text{ПрО}_i} \rangle, i = \overline{1, n};$$

- **лексическая онтология** ПрО, которая содержит сведения о лексемах естественных языков, соответствующих терминам онтологии ПрО

$$O_{\text{lex}_i} = \langle T_{\text{ПрО}_{\text{lex}_i}}, R_{\text{lex}_i}, f \rangle, i = \overline{1, n},$$

т.е. $\forall t \in T_{\text{ПрО}_i} \exists t_{\text{lex}} \in T_{\text{ПрО}_{\text{lex}_i}}$, свойствами которого является множество лексем естественных языков, соответствующих данному термину онтологии, а отношения описывают связи между ними;

- **тезаурус** задачи – множество пар, первым элементом которых являются термины онтологии, совокупность которых характеризует ту конкретную задачу из ПрО, которую в данный момент решает пользователь, а вторым – вес (положительный или отрицательный) этого термина для данной задачи $Th_{i_j} = \langle th_{k_{ij}} \in T_{\text{ПрО}_i}, v_{k_{ij}} \rangle, k = \overline{1, s_{i_j}}, j = \overline{1, m_i};$

- **запрос** – множество ключевых слов, характеризующих одну из информационных потребностей пользователя, связанный с конкретной задачей при помощи тезауруса $z = \langle \{k_q\}, Th_{i_j} \rangle, q = \overline{1, u};$

- **тема** – множество запросов, связанных с одной информационной потребностью $thema = \langle id_{thema}, \{z_q\} \rangle, q = \overline{1, u}$, которое может объединять запросы разных пользователей, базирующиеся на различных онтологиях и тезаурусах, и позволяющее объединять семантически связанные запросы;

- **результат запроса** – множество пар, первым элементом которых являются ссылки на ИР, а вторым – оценки этих ИР пользователем $rez = f(z, u) = \langle \{id_{ir}, rating_{ir}\} \rangle;$

- **пользователь** u – класс, имеющий более сложную структуру и имеющий следующие атрибуты, которые можно разделить на несколько групп:

1. регистрационная информация:

- идентификатор пользователя;
- пароль для доступа к ИПС;

2. опыт взаимодействия ИПС с пользователем:

- список онтологий, которые пользователь применял для описания своих информационных интересов;

- список тезаурусов, которые пользователь применял в поисковых запросах;

- список ранее выполненных запросов;

- список результатов выполненных запросов с оценками пользователя для найденных результатов;

3. сведения, импортируемые из внешних

источников (необязательные сведения, могут отсутствовать):

- идентификаторы пользователя в социальных сетях, позволяющие динамически обновлять сведения о нем;
- рейтинги пользователя в социальных сетях;
- адрес пользователя в Википедии и других вики-ресурсах;
- адрес сайта пользователя;
- сфера компетенций пользователя (ключевые слова, импортируемые из социальных сетей);
- ссылки на публикации пользователя;

4. собственные характеристики пользователя:

- сфера компетенций пользователя (список ключевых слов, вводимых пользователем непосредственно);

5. формальные данные о пользователе (необязательные сведения, предоставляющие ИПС дополнительные сведения для формирования групп пользователей со схожими информационными потребностями):

- место жительства;
- возраст;
- профессия, образование и т.д.
- **группа пользователей** – класс, свойствами которого являются идентификатор группы и список пользователей, по тем или иным причинам объединенных в одну группу (группы могут формироваться явно путем выбора пользователя или автоматически на основе соответствия каким-либо условиям, например, группы пользователей со сходными формальными данными или выполняющих похожие запросы)
$$gr = \langle id_{gr}, \{u_i\}, i = \overline{1, n} \rangle,$$
- **информационный ресурс** – сведения о найденных ранее ресурсах и их оценках
$$\langle U_{url}, \{ \langle z_i, m_i, q_i \rangle, i = \overline{1, n} \} \rangle,$$
 включающие идентификатор ресурса, запросы, по которым он был обнаружен, оценку пользователя, которому он был предоставлен, и его уровень читабельности для этого пользователя.

Рекомендующие системы и семантический поиск

Рекомендующие системы (РС) [Ricci, 2011] отличаются от ИПС тем, что пользователю не надо явным образом формулировать поисковый запрос – система сама, на основании имеющихся сведений о пользователе, предлагает ему рекомендуемые элементы (РЭ). Чем выше и точнее информированность РС о потребностях

пользователя, тем более эффективны результаты ее работы. Персонализированные рекомендации – это упорядоченные списки РЭ, т.е. работа РС сводится именно к ранжированию доступных РЭ.

При этом упорядочении РС, основываясь на предпочтениях и ограничениях пользователей, пытается прогнозировать, какие товары или сервисы наиболее подходят пользователю. Для этого РС накапливает информацию о предпочтениях пользователя и о его действиях. Эти подходы целесообразно использовать и при семантическом поиске, учитывая оценки, данные различным ИР различными сообществами пользователей, сгруппированными на основе подобных информационных потребностей [Рогущина, 2013]

Формально создание рекомендаций в РС может быть представлена следующим образом: пусть C – множество пользователей РС, S – множество предлагаемых РЭ (товаров, книг, фильмов, сервисов и т. д.). U – функция полезности, описывающая интерес пользователя $c \in C$ к РЭ $s \in S$, т. е. $U: C \times S \rightarrow R$, где R – количественная оценка. Цель РС – для каждого потребителя $c \in C$ выбрать такой РЭ $s' \in S$, что $U(c, s') = \max_{s \in S} U(c, s)$. Каким именно образом определяется функция полезности, зависит от типа РС и от специфики РЭ.

При классификации РС обычно выделяют следующие подходы к отбору РЭ:

- *персональный* подход – анализ профиля конкретного пользователя, его ранее проявленных предпочтений и явным образом выраженных условий;
- *социальный* (коллаборативный) подход – анализ предпочтений других пользователей, которые по тем или иным причинам могут распространяться и на того пользователя, для которого делается выбор;
- *контент-ориентированный* подход, при котором анализируются сами РЭ, предлагаемые пользователю;
- *доверительный* подход – анализируется качество предлагаемых пользователю РЭ и анализируется степень доверия к ним.

Следует отметить, что в большинстве реальных РС все эти подходы реализуются интегрированно, но им придается различное внимание.

В большинстве РС подобие между двумя пользователями основывается на том, какие оценки они дали одним и тем же товарам. Наибольшее распространение получили корреляционный метод и метод линейного сходства.

Для эффективной работы РС надо предвидеть оценки, исходя из небольшого количества примеров.

Для преодоления проблемы разреженности оценок следует при поиске похожих пользователей использовать также сведения из их профилей и обнаруживать пользователей со схожими профилями, например, относящихся к одному демографическому сегменту.

Анализ основных направлений развития современных РС [Middleton, 2009] связывает их с использованием онтологий для представления знаний как о пользователях, так и о РЭ. При персональном подходе РС необходимо накопить достаточно сведений о пользователе, чтобы в дальнейшем их обобщать и анализировать. Фоновый мониторинг работы пользователя обеспечивает положительные примеры того, что этот пользователь ищет, не мешая его нормальной работе. Для нахождения отрицательных примеров из наблюдаемого поведения также могут применяться эвристики, (хотя в целом с меньшей точностью). Эта идея лежит в основе тех РС, которые наблюдают за поведением пользователей и рекомендуют им те новые РЭ, которые коррелируют с профилями пользователей. Например, если пользователь регулярно просматривает сайты определенной тематики, то РС, проанализировав контент этих сайтов, может предложить ему другие сайты той же направленности.

Другой способ рекомендовать РЭ базируется на рейтингах, предоставляемых теми людьми, которые ранее оценили РЭ. Коллаборативные РС для этого запрашивают у пользователей явные оценки РЭ, а затем рекомендуют те РЭ, которые высоко оценили похожие пользователи.

Рекомендация относительно новых РЭ для пользователей может формироваться на основе его сравнения с подобными РЭ (фильтрация на основе контента), отзывов об РЭ в сообществе пользователей (коллаборативной фильтрации), семантических отношений между РЭ (эвристические рекомендации) или сочетания этих подходов. Во многих случаях выбор подхода зависит от того, насколько доступны метаданные об РЭ и есть ли обратная связь с пользователями (явно и неявно). Методы на основе контента хорошо работают, если есть достаточная обучающая выборка, а коллаборативные методы – когда система имеет большое сообщество пользователей. Однако на сегодня не выработаны общепринятые правила для выбора стратегии recommendations.

Коллаборативная фильтрация использует рейтинги, предоставляемых сообществом пользователей, чтобы рекомендовать РЭ конкретному пользователю. Существуют два взаимодополняющих подхода к коллаборативной фильтрации: на основе пользователя или на основе РЭ. При коллаборативной фильтрации на основе пользователя находят группы подобных пользователей, а затем конкретному пользователю рекомендуют те РЭ, которые понравились другим пользователям из той же группы. При коллаборативной фильтрации на основе РЭ

группируются те РЭ, которые одинаково оцениваются людьми. Для того, чтобы выполнить коллаборативную фильтрацию, должен быть создан профиль пользователя на основании имеющихся документов о том, какие РЭ были этим пользователем рассмотрены и оценены.

Так же, как и основанном на пользователе методе, сходство РЭ определяется при помощи того, сколько пользователей оценили эти РЭ как подобные. При этом определяются наборы похожих предметов. Этот метод хорошо масштабируется, поскольку новые РЭ добавляются к окрестностям на основе того, как пользователи оценивают их, без необходимости явного использования онтологии.

На основе вышеприведенного анализа можно предложить следующие подходы к работе РС с использованием онтологий: формирование модели пользователя; формирование модели РЭ; создание онтологии РЭ; накопление сведений об экземплярах РЭ и экземплярах пользователей; накопление оценок РЭ пользователями; анализ экземпляров РЭ; классификация (или кластеризация) пользователей на группы с подобными интересами; формирование набора стратегий, которые пользователь может явно выбирать для получения рекомендации; построение метода, позволяющего уточнить класс необходимого пользователю РЭ.

Широко распространенные классификации подходов к выработке рекомендаций, подразделяющие все существующие методы на базирующиеся на пользователе и базирующиеся на РЭ, а также на персональные и коллаборативные, являются слишком общими и, как правило, бинарными. Кроме того, в РС большое внимание уделяют алгоритмам вычисления подобия пользователей и РЭ, и значительно меньшее – методам классификации пользователей и РЭ.

На практике целесообразно не только использовать больше критериев, значимых для выработки рекомендаций, но и предоставить пользователю РС возможность самостоятельно формировать стратегию recommendations, явным образом указывая значимость для данной задачи тех или иных критериев.

При выработке рекомендаций в РС многое зависит от специфики искомого РЭ.

Все РЭ можно подразделить на две категории с точки зрения возможности их *повторного использования*: используемые однократно и многократно. К первой категории относятся различные предметы материального мира и связанные с их использованием услуги. К ним относятся, например, технические устройства, продукты питания, авиабилеты, турпоездки. Если у пользователя уже есть такой РЭ, то ему может понадобиться такой же или похожий на него (после поломки первого, использования и т.д.). К второй категории относятся, как правило, информационные объекты, т.е. такие объекты, что наличие одного

экземпляра позволяет создавать произвольное количество его копий. К ним относятся, например, электронные книги и фильмы. Если у пользователя уже есть такой РЭ, то маловероятно, что ему понадобится еще один (хотя возможна утрата или поломка).

Кроме того, РЭ можно классифицировать (на два или более классов – в зависимости от требуемой точности рекомендаций) на *редко или часто* используемые. К примеру, прогноз погоды нужен пользователю почти ежедневно, а выбор модели холодильника актуален для большинства раз в 10 лет. Для наиболее часто используемых РЭ большинство пользователей склонны ориентироваться на собственный опыт, а для редко используемых – на совокупный опыт сообщества пользователей РС. При этом следует учитывать, что один и тот же РЭ может одним пользователям быть интересен редко, а другим – часто. Например, большинство людей редко интересуется особенностями газовых плит или мебельной фурнитурой, но специалист по комплексным ремонтам может выполнять такие запросы регулярно. Поэтому надо дать пользователю возможность самому явно определять, насколько часто он интересуется такими РЭ (т.е. насколько его собственное мнение о них компетентно).

Еще один важный параметр РЭ – *субъективность* оценивания. Если, к примеру, при оценке бытовой техники или автомобилей достаточно легко сформулировать те отличия, по которым пользователь более высоко оценивает один РЭ, чем другой (например, надежность работы, стоимость, простота обслуживания, функциональные возможности), и вследствие этого каждому пользователю может быть полезен опыт всего сообщества, то при оценке предметов искусства, музыки, фильмов такие отличия практически невозможно формализовать, и потому при выработке рекомендаций для конкретного пользователя важен только опыт некоторого подмножества сообщества с аналогичными вкусами, причем это подмножество может быть сформировано при помощи методов машинного обучения и на основе традуктивного вывода.

На способ выработки рекомендаций влияет и то, насколько оценивание РЭ требует *специальных знаний* в конкретной предметной области. Особенно важно это для тех РЭ, которыми большинство пользователей интересуются редко и потому сами, как правило, не имеют о них глубоких знаний. Вследствие этого они оценивают, как правило, лишь конкретный предмет (а не все предметы данного класса) и практически не имеют возможности сравнивать его с другими подобными РЭ. Например, оценивая телевизор, пользователь может оценить лишь ту модель телевизора, которую он купил, и сравнивать ее лишь с несколькими теми моделями, которыми он пользовался. Поэтому в некоторых областях важнее ориентироваться на мнение экспертов, а не на мнение большинства.

Открытым остается вопрос о том, каким образом формировать множество экспертов для той или иной ПрО. В частности, в большинстве социальных сетей существует как возможность зафиксировать как связь пользователя с набором тем, так и средства определения общего рейтинга пользователя (влиятельность его мнения для других пользователей, оценка его действий другими пользователями и т.д.), но, как правило, отсутствует возможность дифференцировать компетентность пользователя в той или иной тематике. Например, один и тот же пользователь, проявляющий интерес к компьютерам и кулинарии, может быть экспертом в компьютерной технике (и его оценки различных экземпляров компьютеров будут высоко точными), а в приготовлении пищи разбираться очень плохо (и именно потому и интересоваться этой областью) и потому оценивать различные РЭ крайне неправильно.

Следует отметить, что в процессе развития Web и систем электронной коммерции появилось много источников, обеспечивающих доступ к одним и тем же РЭ. При этом речь идет как о материальных, так и об информационных продуктах и услугах. Но при этом разные источники предлагают разные условия доступа и качества обслуживания. Поэтому значительного внимания заслуживает степень доверия к источникам, которую можно определить по совокупности оценок сообщества пользователей. Этот критерий рекомендаций выделяется отдельно, так как речь идет не об оценивании РЭ, а об оценивании *источников РЭ*. Например, многие Web-сайты предлагают бесплатно скачивать статьи, книги и фильмы, но некоторые из них требуют регистрации, отправки SMS-сообщений (не бесплатных), оплаты пароля на распаковку архива и т.д. очевидно, что такие сайты оцениваются значительно ниже, чем те, которые предлагают свободный доступ к электронной библиотеке. Электронные магазины, предлагающие покупку материальных предметов, также работают не всегда честно – они могут задерживать доставку, предоставлять бракованную продукцию, требовать дополнительной оплаты доставки, значительно завышать цену по сравнению с обозначенной (например, на сайте цена представлена в долларах, а оплату следует осуществлять в гривнах по крайне невыгодному курсу, отличающемуся от официального). Поэтому следует предоставить пользователям оценить такие магазины ниже, чем те, которые работают корректно.

В большинстве используемых на практике РС вырабатываются рекомендации относительно какого-то довольно узкого класса РЭ, и нет возможности связывать профили одного и того же пользователя, сформированные различными РС (например, нельзя связать предпочтения пользователя относительно выбора художественной литературы, покупок в электронном магазине и при просмотре новостей). Использование онтологических моделей пользователей позволяет в какой-то мере решить эту задачу и интегрировать

различные РС. При этом возникает дополнительная задача – классификация РЭ, относительно которого пользователь нуждается в рекомендации (и, соответственно, переадресация запроса к соответствующей специализированной РС). Следует учитывать, что нередко пользователь нуждается в рекомендациях по набору взаимосвязанных вопросов (например, выбор места для отдыха связан и с выбором турпутевки, и с прогнозом погоды, и с рекомендациями относительно транспорта). Для этого необходимо разработать общую онтологию РЭ, которая должна быть достаточно компактной и несложной, но при этом охватывать основные классы РЭ, относительно которых пользователи часто нуждаются в рекомендациях. Специфические знания предметных областей такая онтология не должна включать, т. к. они должны содержаться в онтологиях специализированных РС.

Рассмотрим также, оценки каких именно групп пользователей целесообразно применять для коллаборативной фильтрации. Самый простой случай группы – это группа, состоящая всего из одного пользователя, для которого и осуществляется поиск рекомендаций.

Можно сказать, что при этом коллаборативная фильтрация сводится к персональной. Противоположный случай – когда для пользователя значимы оценки РЭ *всем сообществом* в целом. Это может иметь место для тех ПрО, к которым пользователь обращается впервые и еще не имеет собственного мнения не только о самой области, но и о критериях нахождения в ней экспертов. К промежуточным случаям относятся анализ оценок экспертов в ПрО (целесообразно предоставить пользователю возможность явно задавать приемлемый уровень их квалификации).

В целом следует оценивать выбранную стратегию рекомендации по трем направлениям – учет мнения самого пользователя, учет мнения сообщества, анализ самого РЭ. В таком трехмерном пространстве можно разместить большинство типичных объектов рекомендации.

Выбор пользователем значения по каждому из трех параметров для стратегии рекомендации для интересующих его РЭ и наличие онтологии (или хотя бы таксономии РЭ) позволяет достаточно точно профилировать интересы самого пользователя, оценить его собственную компетентность для оценивания РЭ (и, соответственно, значимость его мнения для других пользователей) и выявлять группы пользователей со сходными интересами. Следует отметить, что более объективным является признание своей некомпетентности в оценивании РЭ, чем декларирование своей высокой квалификации.

Тезаурусы как средство представления знаний

Следует отметить, что важным требованием

пользователя к работе любой информационной системы является понятность и предсказуемость ее действий. В общем случае онтология ПрО – достаточно сложная структура, а конкретная задача, для решения которой пользователь ищет информацию, использует только часть содержащихся в такой онтологии знаний. Поэтому представляется целесообразным использовать для моделирования знаний пользователя об интересующей его ПрО частного случая онтологии – тезауруса, который можно рассматривать как проекцию онтологии на задачу.

Тезаурус можно представить как комплекс лингвистических знаний [Браславский, 1997]. Сегодня именно с тезаурусами связаны многие направления усовершенствования семантического поиска [Лукашевич, 2011].

Тезаурус в ИТ – это полный систематизированный набор данных о какой-либо области знаний, позволяющий человеку или вычислительной машине в ней ориентироваться. Тезаурус – это $Ts = \langle T, R \rangle$, где T – множество

терминов, а R – множество отношений между этими терминами. Множества T и R конечны. Множество терминов тезауруса T соответствует множеству концептов X онтологии O . Тезаурусы позволяют моделировать знания как о пользователях, так и о тех ресурсах, которые они ищут [Гладун, 2006].

Чтобы формализовать область своих интересов – ПрО поиска – пользователь должен создать тезаурус, моделирующий интересующую его ПрО, в котором содержатся основные термины ПрО и связи между ними. Тезаурус можно создать вручную или автоматизированно. Основой для автоматического создания тезауруса может послужить обработка набора релевантными этой ПрО или ранее созданная онтология ПрО, из которой пользователь отбирает только необходимые ему термины. Все эти подходы могут комбинироваться друг с другом.

Для создания тезаурусов ИР и РЭ предлагается использовать упрощенный алгоритм построения тезауруса: по полному перечню слов, используемых в ИР, строится словарь терминов, из которого отбрасываются стоп-слова, содержащиеся в специально разработанном пользователем списке. Этот алгоритм применяется только для тех ИР, которые не сопровождаются метаописаниями. В противном случае из метаописаний (в формате RDF или OWL) извлекаются термины тезауруса и связи между ними, которые дополняют построенный по контенту ИР словарь. Аналогично строятся тезаурусы РЭ – обрабатываются их метаописание, контент, отзывы о них других пользователей.

Пользователь вводит запрос, приблизительно идентифицируя свою информационную потребность с помощью ключевых слов или выбирая класс интересующего его РЭ (возможно, с набором условий и ограничений), например, РЭ

класса «художественная литература/фантастика/фэнтези», изданная после 2005 года. В ответ РС формирует набор РЭ, доступных системе и соответствующих этому приблизительному запросу – n ссылок на РЭ и их кратких описаний $I = \{Ref_j, D_j\}$, $j = \overline{1, n}$. Здесь Ref_j – ссылка на соответствующий РЭ (или его описание), а d_j – информация об этом РЭ, доступная РС.

Если множество I не пусто, причем РС найден в ответ на запрос более чем один РЭ ($n \geq 1$), то нужно установить порядок, в каком предлагать пользователю сведения о найденных РЭ. Тогда для всех РЭ из этого множества $I = \{Ref_j, D_j\}$, $j = \overline{1, n}$ формируются их упрощенные тезаурусы $Ts(ИР_j) = \langle T_{j_j}, \emptyset \rangle$, $j = \overline{1, n}$ и соответствующие им словари терминов $T_j = \{t_{j_{lw}}\}$, $j = \overline{1, n}$, $w = \overline{1, q_j}$. $t_{j_{lw}}$ – это слова, которые используются в информации о j -м РЭ, найденном РС, т. е. в D_j , $j = \overline{1, n}$. q_j , $j = \overline{1, n}$ – это количество различных слов, используемых в описании D_j , $j = \overline{1, n}$. Если слова в описании повторяются, то в словаре терминов они фиксируются только один раз.

Затем пользователь формирует тезаурус интересующей его ПрО (или указывает на ранее сформированный тезаурус) $Ts_{ПрО}$ и соответствующий ему словарь терминов этой ПрО $T_{ПрО} = \{t_m\}$, $m = \overline{1, q}$. $T_{ПрО}$ – это множество, состоящее из m терминов, относящихся к интересующей пользователя ПрО. Это множество строится аналогично словарю терминов РЭ и обычно формируется как объединение словарей терминов, содержащихся в документах, которые пользователь нашел ранее и посчитал релевантными интересующей его ПрО (как в их контенте, так и в метаописаниях).

Производится сравнение $T_{ПрО}$ и T_j , $j = \overline{1, n}$, вычисляется коэффициент их близости

$$K_j = \sum_{m=1}^q \sum_{w=1}^{w_j} f(t_{j_w}, t_m), \quad m = \overline{1, q}, \quad w = \overline{1, w_j},$$

где

$$f(t_1, t_2) = \begin{cases} 0, & \text{если } t_1 \neq t_2, \\ 1, & \text{если } t_1 = t_2. \end{cases} \quad (1)$$

Коэффициент (1) представляет собой количество терминов, которые встретились как в тезаурусе РЭ, так и в тезаурусе ПрО. Найденные ИР упорядочиваются в зависимости от значений K_j , пользователю предъявляются в первую очередь те

ИР, которые имеют наиболее высокий коэффициент близости к ПрО.

При использовании коэффициента (1) возникает следующая проблема: слова, соответствующие одному термину, но являющиеся, например, различными словоформами, синонимами или переводами на различные языки, обрабатываются как разные термины. Поэтому представляется целесообразным использовать онтологию ПрО и выделять группы слов, соответствующих одному термину. Для этого пользователь должен связать РЭ словаря терминов тезауруса ПрО с одним из терминов онтологии ПрО, т. е. $\forall t_m \in T_{ПрО}, m = \overline{1, q}$ задать функцию $g(t_m) \in X$. Затем для вычисления коэффициента близости K^O эта функция используется следующим образом:

$$K^O_j = \sum_{m=1}^q f(t_{j_w}, t_m), \quad m = \overline{1, q}, \quad w = \overline{1, w_j},$$

$$\text{где } f(t_1, t_2) = \begin{cases} 0, & \text{если } g(t_1) \neq g(t_2), \\ 1, & \text{если } g(t_1) = g(t_2). \end{cases} \quad (2)$$

Коэффициент (2) представляет собой количество терминов, которые встретились как в тезаурусе РЭ, так и в тезаурусе ПрО и при этом ссылаются на один и тот же термин онтологии ПрО. По сравнению с коэффициентом (1) коэффициент (2) позволяет использовать меньший объем документов для построения тезауруса ПрО, но требует большее время для вычислений.

При создании тезауруса ПрО, которая интересует пользователя РС, необходимо явно указать основные понятия ПрО и связи между ними. К сожалению, большинству пользователей достаточно сложно это сделать (даже имея соответствующие знания и применяя их в своей деятельности). На первом этапе формирования тезауруса пользователь может выбрать одно из следующих решений: 1) самостоятельно построить с помощью одного из редакторов онтологий онтологическое описание интересующей его ПрО; 2) найти (например, в Web) онтологию на языке OWL, которую описывает ПрО, близкую к области его информационных интересов; 3) сформировать множество понятий ПрО, содержащее наиболее характерные слова и словосочетания, встречающиеся в интересующих его ИР.

Использование индуктивного вывода для пополнения онтологий

При построении онтологий и тезаурусов ПрО, а также при коллаборативном подходе к поиску важно определить, какие связи между терминами ПрО являются существенными для описания информационной потребности пользователя. Пользователю достаточно сложно самостоятельно обнаружить все важные закономерности и отбросить несущественные.

Для их выявления можно воспользоваться методами индуктивного и традиционного извлечения знаний из данных. Существуют независимые подходы к реализации подобных методов: ID3, ACLS, CART и т. д. Наиболее интересным, в связи со спецификой проводимой работы, оказался алгоритм ID3 [Quinlan, 1979], который специально разработан для извлечения ценной информации из больших объемов слабо структурированных данных. При работе этого алгоритма время вычислений зависит линейно от числа введенных примеров, числа атрибутов, используемых для описания примеров, и числа узлов в строящемся дереве решений, что отличает его от таких известных алгоритмов построения деревьев решений, как INDUCE, SPROUTER, ROTH-P, в которых усилия, требующиеся для решения задачи, резко возрастают вместе со сложностью задачи.

Если методы, подобные МГУА (метод группового учета РЭ), предназначены для нахождения закономерностей по набору количественных измерений параметров и полученному по ним результату, то методы, подобные ID3 и его вариациям (C4.5, ID4 и т.д.), предназначены для обобщения опыта экспериментов, параметры и результаты которых описаны через качественные оценки (лингвистические переменные). В большинстве случаев между их значениями невозможно установить даже относительное упорядочение (например, различные симптомы и диагнозы пациентов). К таким задачам относится и проблема, которую решают рекомендующие системы. ID3 принадлежит к невозрастающим алгоритмам, то есть при добавлении к набору классифицированных примеров определенного количества новых нужно обрабатывать снова как старые, так и новые примеры. Но ID3 предназначен для построения только бинарного дерева решений, а этого недостаточно удобно для представления закономерностей многих ПрО.

Поэтому для пополнения онтологий предлагается использовать ID3m [Rogushina, 2012] – модификацию ID3 для произвольного (конечного) количества решений. Он также принадлежит к невозрастающим алгоритмам. В данном случае, примерами обучающей выборки являются РЭ, доступные РС, а параметрами, по которым они описываются, являются их свойства, описанные в метаданных и в онтологии РЭ, а также термины тезауруса пользователя. Значения, соответствующие терминам тезауруса, – "Термин отсутствует в описании РЭ", "Термин встречается в описании РЭ редко", "Термин встречается в описании РЭ часто". В качестве результата используется оценка, данная пользователем найденному РЭ (качественная оценка, имеющая два и более значений).

На вход алгоритма поступает обучающая выборка H – набор из n классифицированных

(получивших одну из возможных оценок) примеров одинаковой размерности $H = \{h_i\}$, $i = \overline{1, n}$.

Каждый пример из выборки представляет собой упорядоченную последовательность значений s атрибутов и результирующего атрибута $h_i = \langle a_1, \dots, a_s, r \rangle$, $i = \overline{1, n}$. Значения атрибутов принадлежат конечным множествам: $a_{j_u} \in A_j$, $j = \overline{1, n}$, $u = \overline{1, n_j}$, $r_y \in R$, $y = \overline{1, n_r}$.

Если обучающая выборка содержит примеры, в которых все значения атрибутов одинаковы, а решения различны, то введенная информация недостаточна для построения классификационного правила. Если множество примеров пустое, то можно произвольно связать его с любым решением. Если все примеры относятся к одному классу, строится один лист дерева решений, связанный с этим классом. В противном случае необходимо выбрать один из атрибутов и разделить множество атрибутов на подмножества в зависимости от значения этого атрибута и применить алгоритм к каждому из полученных подмножеств.

На каждом шаге работы алгоритма вычисляется, какой атрибут m несет наибольшее количество информации о результате.

$$C_{\max} = \max \{C_z, z = \overline{1, s}\} = \max_z \left\{ \sum_i \sum_j \frac{C(a_{z_i} \in A_z, r_j \in R_j)}{d_z} \right\}, \quad (3)$$

где $C(x, y)$ – количество информации, d_m – стоимость получения значения m -го атрибута. В результате работы алгоритма ID3m формируется дерево решений, в котором каждый лист связан с одним из решений, каждый узел – с именем одного из атрибутов, а выходящие из такого узла ветви – со значениями этого атрибута.

Такое дерево решений позволяет РС по параметрам вновь найденного РЭ прогнозировать, как именно оценит его пользователь, и предлагать пользователю в первую очередь те РЭ, которые соответствуют его индивидуальным предпочтениям. Так как точные значения вероятностей событий из обучающей выборки неизвестны, то они аппроксимируются на основе рассматриваемого множества примеров.

Предложенный выше подход к формированию рекомендаций основывается на использовании знаний пользователя о ПрО, характеризующей его информационные потребности. Пользователь может явно указывать интересующие его термины и получать те РЭ, которые соответствуют его потребности. Такой подход ориентирован на пользователя с относительно стабильными информационными потребностями, не являющегося специалистом в области информационных

технологий, и позволяет пользователю избежать рутинной работы по фильтрации результатов поиска в Web.

МАИПС и выработка рекомендаций

МАИПС – мультиагентная информационно-поисковая система с развитыми средствами интеллектуализации ее поведения, предназначенная для поиска информации в описанных пользователем относительно узких предметных областях, связанных с профессиональными или научными интересами пользователей, и рекомендует пользователю те результаты поиска, которые относятся к интересующей его ПрО и соответствуют его информационным потребностям. Ее можно рассматривать как рекомендующую систему, ориентированную на формирование рекомендаций относительно естественно-языковых и мультимедийных информационных ресурсов, доступных через Web.

Система МАИПС ориентирована на пользователей, имеющих в сети постоянные информационные интересы и требующих постоянного поступления соответствующей информации. Для этого МАИПС позволяет сохранять и повторно выполнять запросы, учитывая реакцию пользователя на ранее предложенные ему ИР (персональная фильтрация), отслеживать появление аналогичных запросов у других пользователей (коллаборативная фильтрация), сохранять формальное описание области интересов пользователя в виде онтологии (семантическая фильтрация) и т.д. Кроме того, в МАИПС при профилировании пользователей используется специфичный для естественно-языковых ИР критерий оценивания – сложность текста для понимания. Особенностью системы является использование оригинального знание-ориентированного алгоритма, позволяющего определить сложность понимания текста для конкретного пользователя (для этого используются тезаурусы предметных областей, интересующих пользователей) [Рогущина, 2007].

Важной особенностью МАИПС является то, что в этой системе интегрированы различные средства и методы создания, пополнения, обработки и использования онтологий, которыми пользователь может управлять явным образом. Для представления знаний об интересующей пользователя ПрО используются онтологии и тезаурусы ПрО: тезаурус строится пользователем по соответствующей онтологии самостоятельно, а онтология выбирается из набора предложенных на сайте либо импортируются из Web.

Пользователь МАИПС может обращаться к онтологиям, созданных другими пользователями – пересматривать их, задавать по ним контекст поиска, копировать из них нужные фрагменты, но не имеет права изменять их. ИПС может обеспечить поиск онтологий, которые содержат введенные

пользователем термины, а также поиск онтологий, похожих на выбранную пользователем онтологию. Это позволяет создавать группы пользователей с общими информационными интересами и предотвратить дублирование в выполнении одинаковых многоразовых запросов разных пользователей.

Основой МАИПС являются технологии Semantic Web, в частности, язык представления онтологий OWL DL и средства его обработки (рис.2).

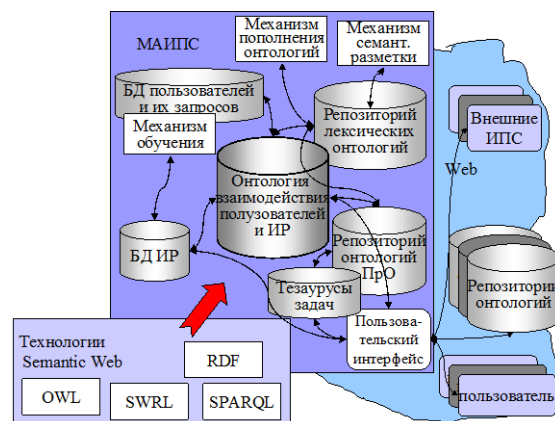


Рисунок 2 - Использование онтологической модели взаимодействия пользователей и ИР для семантического поиска в МАИПС

По мере развития МАИПС возникла потребность в подключении репозитория онтологий, чтобы пользователи могли повторно использовать знания о ПрО, доступные в Web. При этом поиск может осуществляться не только по ключевым словам, а и по другим важным свойствам онтологий. Поэтому в дальнейшем представляется целесообразным реализовать в МАИПС средства взаимодействия с репозиториями онтологий, поддерживающие поиск нужной пользователю онтологии, обнаружение похожих на выбранную пользователем онтологий, а также сопоставление построенного пользователем тезауруса с другими онтологиями и тезаурусами.

Онтологическая модель, описывающая семантику взаимодействия пользователей и ресурсов МАИПС в информационном пространстве Web, обеспечивает знания для выполнения следующих действий, связанных с поиском информации и основанные на рассмотренных выше методах:

1. *предварительный этап*, когда в систему вводятся сведения о окружающем мире, то есть

- создается онтологическая модель, описывающая структуру информации относительно основных элементов, с которыми работает система (пользователей, ресурсов, результатов поиска и т.д.) (рис.1);

- вводятся онтологии ПрО, которые могут быть полезны для поиска, и ссылки на внешние репозитории и средства поиска онтологий, которые

пользователь может применить для работы в более специфических ПрО;

- по имеющимся онтологиям создается несколько примеров тезаурусов, которые могут быть использованы при поиске;

2. этап *регистрации пользователя*, на котором пользователь вводит сведения, необходимые для создания нового экземпляра класса «пользователь»;

3. этап *создания нового запроса* пользователя, на котором пользователю последовательно нужно выполнить следующие действия :

- Выбрать базовую онтологию, знания которой обеспечат семантическую обработку запроса;

- По выбранной онтологией создать тезаурус запроса одним из следующих способов:

- * выбрать несколько классов или экземпляров классов из базовой онтологии;

- * выбрать несколько классов из базовой онтологии и классы, находящиеся от них на семантической дистанции не более указанной пользователем величину;

- * выбрать несколько классов из базовой онтологии и их надклассы и подклассы указанной пользователем глубины;

- * выбрать несколько классов из базовой онтологии и классы, связанные с выбранными классами выбранным пользователем отношением, специфичным для ПрО;

- * вручную ввести термины тезауруса, характеризующие интересующую пользователя задачу;

- * над построенными ранее тезаурусами применить теоретико-множественные операции объединения, пересечения и дополнения;

- * указать вес каждого из терминов тезауруса, который отражает его важность для конкретной задачи.

- создать список ключевых слов, характеризующих конкретный информационный запрос, и соединить его с одним из ранее созданных тезаурусов;

- если нужно, присоединить запрос к одной из ранее построенных групп запросов или создать для него новую группу.

4. этап *выполнения запроса*, в процессе которого поисковый запрос по ключевым словам перенаправляется внешней ИПС (Google), затем МАИПС получает найденные результаты и переупорядочивает их в соответствии с количеством найденных в них терминов тезауруса и их весом.

Кроме того, для упорядочения могут учитываться другие свойства ИР, например, если пользователь указывает желаемый уровень

читаемости, то этот параметр тоже влияет на рейтингование ИР.

Если некоторые из найденных ИР ранее были предложены другим пользователям МАИПС, то по желанию того пользователя, который предоставляет запрос, их оценки могут учитываться либо непосредственно, либо с учетом таких факторов, как степень подобия между этими пользователями и их рейтинг в данной ПрО, который вычисляется как по их собственным оценкам и сведениям о них из социальных сетей, так и по статистике, накопленной МАИПС.

5. этап *обработки запросов* для повышения эффективности поиска. Кроме непосредственного выполнения запросов МАИПС выполняет следующие функции:

- определения уровня компетентности пользователей для различных ПрО, соответствующие имеющимся онтологиям ПрО, которое базируется на таких параметрах, как :

- * Количество запросов пользователя, основанных на данной онтологии;

- * Собственная оценка пользователем своей осведомленности в интересующих его ПрО;

- * Выбранный пользователем уровень читаемости текстовых ИР для данной ПрО - как заданный явно, так и средний для избранных им ИР по запросам, базирующихся на данной онтологии (этот параметр является наиболее объективным)

- * Релевантность собственных публикаций пользователя этой ПрО, которая определяется путем сопоставления тезауруса, построенного по онтологией ПрО, с тезаурусами публикаций;

- * Рейтинг пользователя в социальных сетях;

- * Количество других пользователей МАИПС, выбирающих пользователя за эксперта для поиска рекомендаций в данной ПрО.

6. *создание рекомендаций* по результатам обработки. В отличие от большинства существующих рекомендуемых систем, МАИПС позволяет пользователю явно, непосредственно и динамично управлять средствами создания рекомендаций.

Пользователь может учитывать оценки:

- * Всего сообщества пользователей МАИПС;

- * Подмножеств пользователей, запросы которых базируются на тех же онтологиях;

- * Тех пользователей, которые используют наиболее подобные тезаурусы и ключевые слова для запросов;

- * Запросов с выбранной пользователем темы, в которую могут входить как только собственные запросы пользователя, так и запросы различных пользователей с различными онтологиями и тезаурусами;

* Явно указанной подмножества пользователей МАИПС;

* Подмножества пользователей МАИПС, построенной по введенным пользователем формальными условиями (например, по месту жительства или по возрасту);

* Подмножества запросов самого пользователя, отвечающих определенным условиям (например, построенные в указанный интервал времени и по использованию ключевого слова);

Такие условия пользователь может предоставлять как для каждого запроса по отдельности, так и для определенной группы своих запросов. Результаты построения рекомендаций 1) влияют на упорядочение результатов запроса 2) позволяют рекомендовать пользователю те ИР, которые не встречаются в результатах его собственных запросов, но считаются МАИПС интересными для него

7. автоматизированное *пополнение профиля*. МАИПС функционирует в открытом информационном среде Web и поэтому нуждается в динамическом обновлении своих знаний об этом среде. Для этого используется проактивный поиск новых сведений о своих пользователях в Web (например, поиск их новых публикаций или тех публикаций, содержащих ссылки на них, экспорт новых сведений из социальных сетей) и автоматизированное обновление онтологий ПрО (например, путем обработки семантических вики-ресурсов, экспорта внешних онтологий).

8. *семантическая разметка* предоставляемых пользователю ИР, которая выполняется автоматически на основе лексической онтологии, соответствующей выбранной пользователем онтологии ПрО. В разметке, в зависимости от желания пользователя, могут применяться как все термины, присутствующие в этой онтологии, так и некоторое их подмножество – например, только те, которые присутствуют и в онтологии, и в построенном на ее основе тезаурусе задачи, или же еще и те, которые связаны с терминами из тезауруса выбранными пользователем отношениями.

Заключение

Совместное использование формальной онтологической модели взаимодействия пользователей с ресурсами и методов выработки рекомендаций, доступа к внешним источникам информации, индуктивного извлечения знаний и технологий Semantic Web при семантическом поиске в Web позволяет более эффективно обеспечить пользователя необходимыми сведениями, а явным образом выбранные методы рекомендаций и онтологические описания ПрО обеспечивают пользователю понимание поведения такой системы.

Библиография

- [Amerland, 2013] Amerland D. Google Semantic Search: Search Engine Optimization (SEO) Techniques That Gets Your Company More Traffic, Increases Brand Impact and Amplifies Your Online Presence. – Que Publishing, 2013. – 230 p.
- [Middleton, 2009] Middleton S., De Roure D., Shadbolt N. Ontology-Based Recommender Systems // in Handbook on Ontologies, Edt. by S.Staab, R.Studer, Springer, 2009. – P. 779-796.
- [Quinlan, 1979] Quinlan J.R. Discovery rules from large collections of examples: a case study // Expert Systems in the Microelectronic Age. – Edinburg, 1979. – P. 87-102.
- [Ricci, 2011] Ricci F., Rokach L., Shapira B., Kantor P. Recommender Systems Handbook. – Springer, 2011. – 842 p.
- [Rogushina, 2012] Rogushina J., Gladun A. Ontology-based competency analyses in new research domains // Journal of Computing and Information Technology. V.20, N. 4, 2012. – P.277-293.
- [Бездушный, 2003] Бездушный А.Н., Гаврилова Э.А., Серебряков В.А., Шкотин А.В. Место онтологий в единой интегрированной системе РАН // Современные технологии в информационном обеспечении науки, М.: Научный Мир, 2003. С.97-115.
- [Браславский, 1997] Браславский П.И., Гольдштейн С.Л., Ткаченко Т.Я. Тезаурус как средство описания систем знаний // Информационные процессы и системы. – 1997. – № 11, Серия 2. – С. 16-22.
- [Гладун, 2006] Гладун А.Я., Рогушина Ю.В. Онтологии и мультилингвистические тезаурусы как основа семантического поиска информационных ресурсов Интернет // The Proc. of XII-th Intern. Conf. KDS'2006, Varna, Bulgaria. – P. 115-121.
- [Лукашевич, 2011] Лукашевич Н.В. Тезаурусы в задачах информационного поиска. – М.: Издательство Московского университета, 2011. – 512 с.
- [Рогушина, 2007] Рогушина Ю.В. Использование критериев оценки удобочитаемости текста для поиска информации, соответствующей реальным потребностям пользователя // Проблемы программирования. – 2007. – № 3. – С. 76-87.
- [Рогушина, 2013] Рогушина Ю.В. Использование онтологических знаний в рекомендующих системах // Проблемы програмування, 2013, №2. – С.71-86.

KNOWLEDGE-ORIENTED MEANS OF SEMANTIC SEARCH INTO THE WEB

Rogushina J.

Institute of software systems of National Academy of Sciences Ukraine, Kiev, Ukraine

ladamandraka2010@gmail.com

Semantic-based approach for Web search is proposed. Means of knowledge representation (ontologies and thesauri), methods of their refinement and matching are analysed. Ways of personal and collaborative information use are proposed.

Keywords: semantic search, ontological model, thesaurus, inductive inference, recommending systems, collaborative search.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

UDK 681.3.

USE OF THE ONTOLOGICAL APPROACH TO SEMANTIC SEARCH IN THE ENVIRONMENT OF THE INTERNET OF THINGS

Rogushina J. *, Gladun A. **, Andrushevich A. ***, Kurbatski A. ***

** Institute of software systems of National Academy of Sciences Ukraine, Kiev, Ukraine*

ladamandraka2010@gmail.com

*** International Research and Training Center Information Technologies and Systems of National Academy of Sciences Ukraine, Kiev, Ukraine*

glanat@yahoo.com

**** Software Engineering Department, Faculty of Applied Mathematics and Computer Science, Belarusian State University, Minsk, Belarus*

{lastname}@bsu.by

In the article the multilayer ontological reference model of the Internet of Things and methods of it's design and use for semantic search are proposed.

Keywords: Internet of Things, ontological model.

Introduction

The term «Internet of Things» (IoT) goes back into presentation of K. Ashton in 1999. Now IoT is undoubtedly becoming a reality, it now goes far beyond the original concept, encompassing not only RFIDs, but also sensors, actuators, mobile devices etc. In the new vision, all of these are considered to be things that can act upon, measure, or provide services based on real-world entities. The number of different interconnected through ICT physical objects/devices as well as their virtual representation has been growing rapidly in recent years. According to different estimations there will be around 25 billion devices connected to the Internet by 2015 and 50 billion by 2020.

Such numerous amounts of widely distributed and highly heterogeneous devices and technologies will need to communicate in different scenarios under various conditions. This implies that interoperability becomes one of the most fundamental requirements to support object discovery and retrieval as well as information representation, storage and exchange. IoT has started getting considerable interest from academia and industry that work on the related technologies. New protocols and industrial standards for low-level device communications in resource-constrained environments are also being actively developed.

However, IoT still has a lot of unresolved issues like high heterogeneity (networks of objects in IoT will be deployed by distinct entities, and the deployed hardware will display different operating characteristics, such as sampling rates and error distributions); scalability; unknown topology; incomplete metadata etc.

This complex subject requires a lot of joint integrated efforts from communities working in different fields, such as telecommunications, IT, manufacturing and material engineering etc. Moreover, it should be laid on foundations of modern knowledge-oriented technologies that assure full future interoperability of networks and devices and shared access to various IoT resources.

Data collected by different sensors and devices is usually polysemantic (temperature, light, sound, video, etc.). The diversity, volatility and ubiquity of the real world IoT data will cause a lot of problems with their processing, integrating and interpreting. Therefore the dynamic and resource-constrained nature of IoT requires special design of knowledge representation and processing that must take into account the semantic web technologies such as ontological analysis, semantic metadata, Semantic Web, Linked Data and Social Web.

Semantic descriptions can support the IoT integration by enabling interoperability between

different sources. But analysis and mapping between different semantic description models is still required to facilitate the IoT data integration with other existing knowledge domains and establish the general reference model of the IoT semantics (Fig.1).

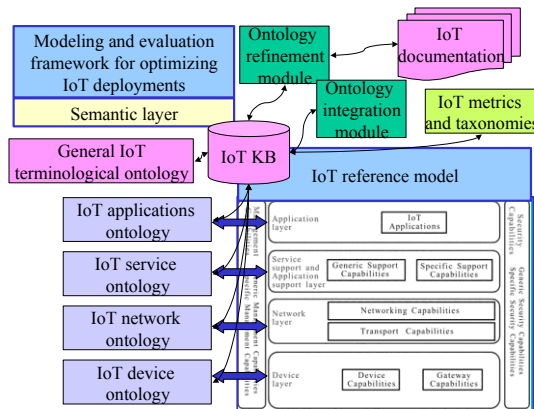


Figure 1 - IoT reference model

Current state of research in the field

The Internet of Things [Kurakova, 2013] may be considered as a heterogeneous network formed by autonomic and human controlled devices that communicate with each other in order to accomplish some specific, often delay or mission-critical, task(s).

The unprecedented design challenges related to IoT has caused it to receive great attention from both academia and industry as of late. Having a world of interconnected and automated devices facilitates the development of unprecedented applications and breaks with the current markets.

Taxonomy of IoT applications

Even by its term, the “Internet of Things“ has an inherent fuzziness and dichotomy due to trying to merge the networking plane (Internet) and the objects‘ plane (Things). It is therefore very important to precede any foray into this field by achieving a clear understanding of the overall scientific as well as practical engineering contexts and objectives. Although IoT research field is comparatively young, there already exists sufficient body of literature that could be used as a basis for building such understanding.

Each of these fields may in turn contain an endless number of different visions, which will all together morph into creating a rich ecosystem of IoT technologies and applications. Even simply enumerating the technologies is a challenge, let alone harnessing their potential in a most efficient manner that would be both technically optimal and at the same time economically viable.

IoT Semantic Formal Modeling

IoT components (devices – sensors, actuators, mobile devices etc. – and services) are heterogeneous

and dynamic, with unknown nature of the network topology. As one of the fundamental constituents of future Internet, IoT has attracted tremendous interests from various research communities and industry [Hachem, 2011]. Advancement in wireless sensor networks has led to a potential interest in integrating data and capabilities provided by physical world objects into the Internet.

Already, we have seen many applications using semantic Web technologies in IoT research, in particular the SSN ontology for annotating sensors and sensor networks; Linked Data [Bizer, 2009] for sensor data publishing [Pschorr, 2010], and semantic sensor observation services (SemSoS) [Henson, 2009].

Now IoT is undoubtedly becoming a reality but it needs some integration means with modern semantic-oriented Web technologies (for example, with standards of the Semantic Web approach) including semantic modeling that can be used for Scalable Search in the Internet of Things [Andrushevich, 2013]. An ontological approach can be used as a base for interoperable reused description of their semantics. Now a lot of researchers try to develop different ontological models of different IoT domains, like for example Semantic Building [Andrushevich, 2010].

Ontology for the Internet of Things provides all the necessary semantics for the specification of IoT devices as well as the specifications of the IoT solution (input, output, control logic) that is deployed using these devices. Technologies developed in the Semantic Web such as ontologies, semantic annotation, Linked Data and semantic Web services can be used as principal solutions for the purpose of realizing the IoT but there is no one universal ontology or a set of aligned ontologies that can be considered as a standard for all IoT community. A common solution to all challenges above is the use of semantic technologies to increment knowledge with metadata. The representation of the IoT-based real world divides into some different layers: a physical layer (things); an information layer (data and metadata about knowledge provided by things); and a functional layer comprising services provided by things.

To match this vision of the real world and its representation by IoT, we have to develop the integrated set of ontologies that actually models all three layers. The physical layer is represented by Device Ontology that is specific for IoT. The information layer is represented by the set of Domain Ontologies that can be exported and reused from different intelligent software. The service layer is represented by Functional Ontology that can use the research results in semantic Web-services area.

The other important problem is considered with the development of the algorithms and techniques that are oriented on efficient processing of IoT ontologies: for example, methods for semantic search of IoT devices interested for particular user; methods of collaborative search that can acquire the dependences in IoT world; methods of integration with domain ontologies etc.

Now we have a long-term experience in investigation of knowledge management problems in the Semantic Web environment in such associated areas as the Semantic Web technologies and their use in the design of intelligent applications in different domains that we try to apply in the IoT domain: design of intelligent and personalized ontology-based retrieval and recommending systems in the context of the Semantic Web technologies [Gladun, 2009]; design of the methods of ontological analysis – thesauri forming, mereological analysis, semantic markup of natural language texts with lexical ontologies [Gladun, 2012]; cognitive networks and ontological analysis in increasing of service quality in heterogeneous wireless environment [Rogushina, 2010].

Scientific significance of the planned research

IoT applications have to be intelligent and knowledge-based. Different relevant ontologies that define such domains as geographical objects, people, units of measurement can be reused in combination with specific IoT ontologies.

We think that IoT formal model on base of ontological approach becomes an important tool for integration and retrieval of different IoT objects and services. The use of standardized terminology provides the facility for formalized semantic definition of IoT elements. An important dimension of research is a development of means and instrumental tools for automated population and refinement of this model by knowledge acquired from natural language texts and other informational resources relevant to the new IoT elements. This model is an efficient mean of cooperation and collaboration in community of IoT researchers and their mutually beneficial communications with results of investigation in domains of the Semantic Web, semantic Web services and Semantic Grid, semantic informational retrieval etc.

IoT formal ontological model

Design of formal ontological model of the IoT objects includes some main parts:

- Analysis of the IoT domain knowledge structure, main objects and relations: use of mereological and ontological methods to acquire the IoT domain terminology and design of the OWL ontology structure;
- Reuse of existing taxonomies and ontologies of the IoT domain: retrieval and analysis of knowledge representation of the IoT means and standards, overview of relevant ontologies and other knowledge structures, integration of existing domain taxonomies;
- Design of methods for automated refinement of the IoT formal ontological model: design of methods for automated acquisition of knowledge (terms and relations) from texts relevant to the IoT domain,

methods of automated linguistic processing of the natural language texts and refined OWL ontology;

- Semantic retrieval in IoT domain on base of domain ontology: semantic search of the IoT devices, analysis of the web search result lists in the IoT domain;
- Semantic search of the IoT services: methods of semantic search of the IoT services, recommendations about use of IoT devices and services and methods of the collaborative search in IoT community.

Semantic search in IoT

IoT connects a lot of embedded devices equipped with sensors to perceive their surroundings. Thereby, the state of the real world will be available online and in real-time and can be combined with other data and services in the Internet to realize novel applications such as Smart Homes, Smart Cities, Smart Grids, or Smart Healthcare. This requires an open representation of sensor data and scalable search over data from diverse sources including sensors [Andrushevich, 2013]. The Semantic Web technologies RDF (an open semantic data format) and SPARQL (a query language for RDF-encoded data) can be used to address those challenges: prediction models can be employed for scalable sensor search, encoded as RDF and queried by means of SPARQL.

We propose to use the structure of IoT domain as a base of for semantic search of the IoT items pertinent to user needs (Fig.3).

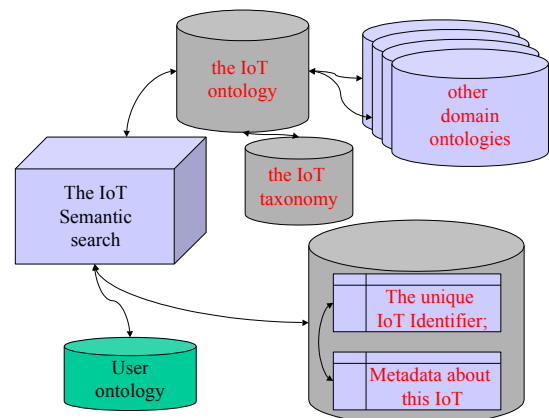


Figure 2 - Semantic search for the IoT

By this approach the formal model consists of: 1. the unique IoT Identifier – URI from the IoT repository; 2. metadata about this IoT corresponding into this IoT Repository to identified IoT; 3. the IoT ontology where different the IoT rules and correspondences are stored (this ontology can be supplemented by other domain ontologies).

Like regular search, semantic search can be deconstructed into its components. To create a semantic search that can use not only keywords but their meanings, for example, in [Amerland, 2013] semantic search is based on the three basic elements: Identifier (URI), RDF and an ontology library.

Universal Resource Identifier – a URL or a URI of the Web resource, or it can be a Universal Resource Name (URN). The URI is needed for the initial set of Web information resources (but that initial set of data is not sufficient for efficient search). But URI has to be refined before it can be proposed to user. This refinement is achieved with the help of metadata information – RDF that can be used as a set of transforming rules for data from URI database to another database without loss of meaning.

Ontologies are used for knowledge structuring by classes and rules that support an inference on them. Metadata from RDF and domain knowledge from ontologies can be used to infer the meaning of IR content in the search process. The IoT Device Ontology identifies what things should be looked up to satisfy an application's requirements. Different applications that use this ontology can identify devices relevant to provided services.

In [Smyth, 2011] the problems of mainstream web search engines that suggest the one-size-fits-all results of Web search are described: two different users with the same query will, more or less, receive the very same result-list, despite their different preferences. Web search needs to become more personalized so that the implicit needs and preferences of searchers can be accommodated. Now a lot of different approaches to personalizing web search by harnessing different types of user preference and context information to influence the search experience. For personalization of search for the IoT we have to use the knowledge about every user of the IoT and about the community of users in general.

We propose to integrate the knowledge about the IoT items with the knowledge about personal needs of users and domain knowledge represented in ontological form. For use of all of this information in semantic search process the ontological elements are matched.

Conclusion

Development of the IoT formal ontological model that integrates all main concepts (items, services, protocols, benchmarks etc.) and their relations of this subject domain becomes an important instrument of integration, retrieval and interoperability of heterogeneous objects and services of the IoT on the semantic level.

Bibliography

- [Kurakova, 2013] Kurakova T. Overview of Internet of Things // Proc. of the INTHITEN (INternet of THings and ITs ENablers) conference, Saint Petersburg, Russia, 2013. – P. 1-13.
- [Hachem, 2011] Hachem S., Teixeira T., Issarny V. Ontologies for the internet of things // Proc. MDS'2011, N.3, ACM, New York, NY, USA 2011. – <http://hal.archives-ouvertes.fr/docs/00/64/21/93/PDF/IotMiddleware.pdf>.
- [Bizer, 2009] Bizer C., Heath T., Berners-Lee T. Linked Data - The Story So Far // International Journal on Semantic Web and Information Systems (IJSWIS), 2009.
- [Barnaghi, 2010] Barnaghi P., Presser M., Moessner K. Publishing Linked Sensor Data // Proc. 3rd International Workshop

on Semantic Sensor Networks (SSN), in conjunction with the 9th International Semantic Web Conference (ISWC 2010), 2010.

[Pschorr, 2010] Pschorr J., Henson C., Patni H., Sheth A. Sensor discovery on linked data // Proc. of the 7th Extended Semantic Web Conference, ESWC2010, Heraklion, Greece, 2010.

[Henson, 2009] Henson C., Pschorr J. K., Sheth A. P., Thirunaryan K. SemSOS: Semantic Sensor Observation Service // Proc. of the 2009 International Symposium on Collaborative Technologies and Systems (CTS 2009), Baltimore, MD, 2009.

[Andrushevich, 2013] Andrushevich, A., Kurbatski, A., Basko, V., Klapproth, A. Towards information retrieval characteristics in the Internet of Things // Proc. of the International Congress on Computer Minsk, 2013.

[Andrushevich, 2010] Andrushevich, A., Staub, M., Kistler, R., Klapproth, A. Towards Semantic Buildings: Goal-driven approach for building automation service allocation and control // Proc. of the 15th IEEE International Conf. on Emerging Technologies and Factory Automation, Bilbao, Spain, 2010.

[Gladun, 2009] Gladun A., Rogushina J. Use of the Semantic Web technologies in design of informational retrieval systems// in Book "Building and Environment", 2009 Nova Scientific Publishing, New-York, USA. – P.89-103.

[Gladun, 2012] Gladun A., Rogushina J. Use of Semantic Web Technologies and Multilingual Thesauri for Knowledge-Based Access to Biomedical Resources // International Journal of Intelligent Systems and Applications, 2012, №1, P.11-20. – <http://www.mecspress.org/ijisa/ijisa-v4-n1/IJISA-V4-N1-2.pdf>.

[Rogushina, 2010] Rogushina J., Lesko O. Use of the specialized lexical otology for automated formation of the domain ontologies on base of natural language texts // Information Models of Knowledge. Edited by K.Markov, V.Velychko, O.Voloshin. – ITHEA, Kiev-Sofia, 2010. – P.93-100.

[Smyth, 2011] Smyth B., Coyle M., Briggs P. Communities, Collaboration, and Recommender Systems in Personalized Web Search // in Ricci F., Rokach L., Shapira B., Kantor P. Recommender Systems Handbook. – Springer, 2011. – P.579-615.

[Amerland, 2013] Amerland D. Google Semantic Search: Search Engine Optimization (SEO) Techniques That Gets Your Company More Traffic, Increases Brand Impact and Amplifies Your Online Presence. – Que Publishing, 2013. – 230 p.

ИСПОЛЬЗОВАНИЕ ОНТОЛОГИЧЕСКОГО ПОДХОДА К СЕМАНТИЧЕСКОМУ ПОИСКУ В СРЕДЕ ИНТЕРНЕТА ВЕЩЕЙ

Рогушина Ю. *, Гладун А. **, Андрушеви А. ***, Курбатский А. ***

* *Институт программных систем НАНУ, Киев, Украина*

ladamandraka2010@gmail.com

** *МНУЦІТС НАНУ и МОНУ, Киев*

glanat@yahoo.com

*** *Белорусский государственный университет, Минск, Беларусь*

{lastname}@bsu.by

В статье предложена многоуровневая онтологическая эталонная модель для the Internet of Things и методы ее разработки, пополнения и использования при семантическом поиске.

Ключевые слова: Интернет вещей, онтологическая модель



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

УНИВЕРСАЛЬНЫЙ СЕМАНТИЧЕСКИЙ КОД КАК ТЕХНОЛОГИЯ АБСТРАГИРОВАНИЯ ДЛЯ РЕШЕНИЯ НЕЧЕТКО ПОСТАВЛЕННЫХ ТЕХНИЧЕСКИХ ЗАДАЧ

Канделинский С. Л.^{*}, Бойко И. М.^{**}

**Объединенный Институт проблем информатики НАН Беларуси,
kandel@yandex.ru*

*** Cisco Inc., San Jose, CA, USA
igboyko@cisco.com*

В работе приводится технология абстрагирования в составе моделей решения сложных нечетко поставленных задач на начальном этапе проектирования. В качестве основного инструмента используется Универсальный Семантический Код (УСК).

Ключевые слова: абстрагирование, глагол, УСК, аксиоматический вывод

Введение

Нечетко поставленными задачами, будем называть такие задачи, из условий которых нельзя однозначно определить, что является исходными данными, что - результатами, какова связь между ними, а так же обширным полем перебора вариантов возможных решений [Горбатов, 2000]. Причем из данных часть необходимых может отсутствовать, а часть быть избыточной, что допускает множество интерпретаций проблемы.

Проблема может быть определена в виде:

$$\Pi = \langle \text{ИС}, \text{СЭ}, \text{НЭ-ПЭ}, \text{Т}, \text{О} \rangle, \quad (1)$$

где: ИС - Исходная Ситуация

СЭ - Сравнение с Эталоном

НЭ - Нежелательный Эффект – негативное последствие позитивного эффекта (ПЭ)

Т – Цель: устранение НЭ

О – Ограничения на степень допустимых изменений проблемной системы.

Ограничивающими условиями решения задачи могут быть, например:

- Запрет на изменение конструкции объекта;
- Запрет на изменение принципа действия объекта;
- Запрет на изменение элементной базы;
- Запрет на изменение функции объекта.

Ограничения могут быть неявными и при этом как истинными, так и ложными.

Схема процесса решения задачи может быть описана согласно [Khomenko et al, 2007] в виде сходящейся к конкретному решению Р последовательности вида:

$$P = (\{K3\} \rightarrow \{A3\} \rightarrow \{AP\} \rightarrow \{KP\} \rightarrow UP \rightarrow PP \rightarrow BP),$$

где:

КЗ – множество конкретных задач

АЗ – множество абстрактных задач

АР – множество абстрактных решений

КР – множество частичных решений

УР – концептуальное решение,

РП – решение-прототип,

ВР – внедряемое решение.

Этап абстрагирования, как отвлечение от окружения и выделение главных компонентов проблемы, оказывается практически ключевым для обеспечения быстрого сужения поля поиска к наиболее желаемому решению задачи.

Существуют эффективные технологии решения задач [Альшуллер, 1985] и основанные на них системы CAI (computer aided innovation systems) [Щуриков, 1991], [Pesetsky et al., 2004], где абстрагирование происходит путем применения эвристических правил.

Известны подходы, применяющие лингвистические и семантические средства для обработки формулировки задачи. Например,

использования синонимичных понятий для переподстановки при уточнении и решении задачи [Broadbent, 1966].

Однако, все известные подходы предназначены для человека, в очень малой степени работают с семантикой естественного языка и не предназначены для компьютерного исчисления.

В отличие от других методов, Универсальный Семантический Код (УСК) предоставляет средства автоматизации обработки и решения задач.

1. Универсальный семантический код

1.1. Основы УСК

УСК как формальная теория естественного языка был разработан в 70-х годах проф. В.В.Мартыновым и прошел эволюцию из шести версий [Martynov, 2001; Мартынов, 2009].

Основной постулат УСК: вывод знаний не может быть эффективным без семантического представления знаний. УСК предлагает вычислительную модель, основанную на формальном представлении значения глагола.

Текст УСК представляется последовательностью цепочек УСК, которые связаны между собой заданными отношениями. Минимальная цепочка УСК состоит из трёх компонентов располагающихся вокруг глагола: субъект, инструмент, объект, причем субъект одновременно рассматривается как объект при воздействии самого на себя. Максимальная цепочка содержит четыре компонента, включая второй объект.

Глаголы разбиты на 108 функциональных классов делящихся на две группы физических и информационных глаголов. Каждый класс задает количество элементов вокруг глагола. Название класса представлено наиболее абстрактным глаголом. Каждый класс содержит список конеретизирующих глаголов. Например, класс 'создавать' представлен глаголами строить, генерировать, производить и т.д. Некоторые глаголы состоят из совокупности классов. Например, глагол 'переносить' состоит из двух классов 'двигать' и 'удерживать', причем оба действия (класса) осуществляются одновременно.

Формальная часть алгебры УСК основана на логике Лукасевича [Lukasiewicz, 1958] и определена как $A = \{M, \rightarrow, '\}$, где: M – набор элементов, \rightarrow – бинарная некоммукативная и неассоциативная операция импликации на заданном множестве, $'$ – одноместная операция дополнения на заданном множестве.

Рассмотрим фразу на естественном языке и ее представление посредством УСК. Например, фраза 'Локатор облучает цель' состоит из глагола 'облучать' и двух окружающих компонентов 'локатор' и 'цель'.

В соответствии с УСК классификатором глагол 'облучать' должен иметь три компонента, а это значит, что в начальной фразе один компонент должен быть восстановлен. Очевидно, что есть субъект 'локатор' и объект 'цель', но не хватает инструмента. Добавим инструмент и получим фразу 'Локатор посредством зондирующего сигнала облучает цель'.

В нотации УСК компонентам присваиваются значения переменных для последующего формального представления и вывода знаний. Так: X = субъект = локатор, Y = инструмент = зондирующий сигнал, Z = объект = цель.

1.2. Аксиоматика УСК

Все цепочки УСК состоят из двух частей, где левая часть это стимул, а правая это реакция. Например, цепочка для глагола 'создавать' $((XY)Z)((ZW)W)$ относится к группе физических глаголов и состоит из левой части $((XY)Z)$ и правой части $((ZW)W)$. Все аксиоматические преобразования производятся над правой частью цепочки. Приведем примеры нескольких аксиом.

1.2.1. Аксиома транспозиции

Аксиома определяет смену позиций скобок в правой части формулы: $((XY)Z)((ZW)W) \rightarrow ((XY)Z)(Z(W)W)$ т.е. если 'создавать', то 'материализовываться'.

1.2.2. Аксиома диффузии

Правая часть формулы может быть преобразована одним из трех способов:

- замена переменной во второй позиции переменной из первой позиции
 $((XY)Z)((ZW)W) \rightarrow ((XY)Z)((ZZ)W)$
- замена переменной в третьей позиции переменной из второй позиции
 $((XY)Z)((ZZ)W) \rightarrow ((XY)Z)((ZZ)Z)$
- замена переменной в третьей позиции переменной из первой позиции, если и только если все три переменные различны
 $(XY)Z((ZY)W) \rightarrow ((XY)Z)((ZY)Z)$.

1.2.3. Аксиома дополнения

Аксиома определяет преобразование одной формулы в другую изменением правой части в соответствии с пространственными отношениями:

$((XY)Z)((ZZ)W) \rightarrow ((XY)Z)((ZZ)W')$ \rightarrow
 $((XY)Z)((ZZ)W'')$ = ввести, вставить (внутри) \rightarrow
приблизиться (на, вблизи) \rightarrow прицелиться
(выбрать цель и направиться к ней).

1.2.4. Аксиома корреляции

Аксиома определяет преобразование информационной цепочки в физическую: $((XY)X)((XY)Y) \rightarrow ((XY)Z)((ZY)Y)$ = аннулировать \rightarrow уничтожать.

Аксиомы показывают, что последовательность глаголов в предложении может быть явным образом установлена и канонизирована для естественного

языка. Так фраза 'ребенок ест руками' аксиоматически реконструируется в полную: 'ребенок ест ртом, удерживая пищу руками'. Такая реконструкция часто не важна для читателя, но очень важна для автоматического вывода.

На основе аксиоматики и классификатора глаголов формируются алгоритмы перехода от исходной ситуации к целевой на абстрактном уровне. Фактически происходит преобразование фраз естественного языка [Бойко и др., 1991].

Аксиомы УСК могут одновременно применяться к исходному глаголу, тем самым задвая возможные параллельные одновременные выводы других глаголов из него, как возможные альтернативные решения.

2. Приложение УСК

2.1. Алгоритм УСК перехода

Мы предлагаем следующий алгоритм перевода текста с естественного языка на УСК:

1. Выделить главные отношения представленные глаголами в предложении.
2. Выделить компоненты этого отношения.
3. Заменить найденные компоненты переменными.
4. Выделить второстепенные отношения предложения.
5. Выделить компоненты второстепенных отношений.
6. Заменить найденные компоненты второстепенных отношений переменными
7. Построить последовательность действий из предложений как последовательность глаголов.
8. Соотнести каждый глагол с классами глаголов и УСК цепочкой.
9. Построить последовательность формул как абстрактное описание процесса.
10. В соответствии с задачей оставить как есть или преобразовать последовательность УСК цепочек используя аксиомы.
11. Перейти к описанию конкретного процесса.

С помощью таких последовательностей можно описать различные технологические, физические, химические и информационные процессы, а также решать задачи из разных предметных областей.

2.2. УСК представление процессов

Рассмотрим применение алгоритма на примере описания работы системы контроля доступа на световых барьерах. Принцип действия светового барьера основан на обнаружении прерывания узконаправленного потока инфракрасного (ИК) излучения с кодо-импульсной модуляцией [Брашко, 2009].

Блок управления генерирует кодо-импульсную последовательность электрических сигналов и по проводной линии связи передаёт её на излучатель. В соответствии с принятыми сигналами излучатель

преобразует эту последовательность в импульсы ИК-излучения. Приёмник принимает импульсы ИК-излучения, преобразует их в соответствующую последовательность импульсов электрических сигналов и по проводной линии связи передаёт её на блок управления.

Блок управления сравнивает последовательность импульсов, переданную на излучатель, с последовательностью импульсов, принятой от приёмника, и на основании выбранного алгоритма оценивает соответствии этих последовательностей между собой.

А) В случае соответствия импульсов, переданных на излучатель и полученных от приёмника, блок управления вырабатывает управляющий сигнал 'включение' и передаёт его на реле. Реле, приняв сигнал 'включение', замыкает контакты, через которое включается оборудование.

Б) В случае несоответствия импульсов излучателя и приёмника, блок управления вырабатывает управляющий сигнал 'выключение' и передаёт его на реле, которое, приняв сигнал 'выключение', размыкает контакты и таким образом блокирует работу оборудования.

Переписываем текст в виде отдельных предложений содержащих один значащий глагол и специфицируем компоненты каждого глагола.

1. Блок управления *генерирует* кодо-импульсную последовательность электрических сигналов.
2. Блок управления посредством проводной линии связи *передает* кодо-импульсную последовательность электрических сигналов излучателю.
3. Излучатель *принимает* кодо-импульсную последовательность электрических сигналов.
4. Излучатель *преобразует* кодо-импульсную последовательность электрических сигналов в импульсы ИК-излучения.
5. Приёмник *принимает* импульсы ИК-излучения.
6. Приёмник *преобразует* импульсы ИК-излучения в электрические сигналы.
7. Приёмник *передает* электрические сигналы на блок управления по проводной линии связи.
8. Блок управления *сравнивает* последовательность импульсов, переданную на излучатель, с последовательностью импульсов, принятой от приёмника.
9. Блок управления *оценивает* соответствии последовательностей между собой.
10. А) Блок управления *вырабатывает* управляющий сигнал 'включение'.
11. А) Блок управления *передает* управляющий сигнал 'включение' на реле.
12. А) Реле *принимает* управляющий сигнал 'включение'.
13. А) Реле *замыкает* контакты.
14. А) Реле *включает* оборудование.

10. Б) Блок управления *вырабатывает* управляющий сигнал 'выключение'.

11. Б) Блок управления *передает* управляющий сигнал 'выключение' на реле.

12. Б) Реле *принимает* управляющий сигнал 'выключение'.

13. Б) Реле *размыкает* контакты.

14. Б) Реле *блокирует* оборудование.

Теперь мы извлекаем последовательность глаголов, представляющую весь процесс [Boyko, 2001, 2009] и которая выглядит как: генерировать → передать → принимать → преобразовать → принимать → преобразовать → передавать → сравнивать → оценивать → вырабатывать → передавать → принимать

замыкать → проводить или
размыкать → блокировать

Соотнесим последовательность глаголов с УСК классификатором глаголов для определения УСК цепочек задающих количество переменных каждого глагола и их естественно-языковую интерпретацию. Упрощая, оставляем только правые части цепочек:

генерировать – класс создавать – $(ZW)W$

передавать – класс продвигать – $(ZZ)W'$

принимать – класс вводить (в себя) – $(ZZ)W$

преобразовывать – класс преобразовать – $Z(ZZ)$

сравнивать – класс сравнивать – $(XX)X$

оценивать – класс оценивать – $(XX)X'$

вырабатывать – класс создавать – $(ZW)W$

замыкать – класс соединять – $(ZY)W$

прерывать – класс разъединять – $(ZW)Y$

размыкать – класс разъединять – $(ZW)Y$

проводить – класс двигать – $(ZZ)Z'$

блокировать – объединение классов удерживать и изолировать – $((ZY)Z'')(Z(WY''))$

Формируем последовательность правых частей УСК цепочек:

$(ZW)W$ (создавать) → $(ZZ)W'$ (продвигать) → $(ZZ)W$ (вводить) → $Z(ZZ)$ (преобразовывать) → $(ZZ)W$ (вводить) → $Z(ZZ)$ (преобразовывать) → $(ZZ)W'$ (продвигать) → $(XX)X$ (сравнивать) → $(XX)X'$ (оценивать) → $(ZW)W$ (создавать) → $(ZZ)W'$ (продвигать) → $(ZZ)W$ (вводить)

↓
 $(ZY)W$ (соединять) или $(ZW)Y$ (разъединять)
↓
 $(ZZ)Z'$ (двигать) $((ZY)Z'')(Z(WY''))$ (удерживая изолировать).

Теперь мы можем переписать все шаги процесса абстрагируясь от конкретных глаголов и уточняя недостающие компоненты в окружении глаголов и не упомянутые в тексте.

1. $(ZW)W$ – Блок управления посредством генератора создает электрические сигналы в проводной линии связи. Здесь появляется новый компонент, а точнее инструмент 'генератор', не

упомянутый в тексте и подразумеваемый по умолчанию.

2. $(ZZ)W'$ – Блок управления посредством проводной линии связи продвигает электрические сигналы к излучателю.

3. $(ZZ)W$ – Излучатель посредством приемного модуля вводит (в себя) электрические сигналы.

4. $Z(ZZ)$ – Излучатель посредством преобразователя преобразует электрические сигналы в импульсы ИК-излучения.

5. $(ZZ)W$ – Приемник посредством приемного модуля вводит (в себя) импульсы ИК-излучения.

6. $Z(ZZ)$ – Приемник посредством преобразователя сигналов преобразует импульсы ИК-излучения в электрические сигналы.

7. $(ZZ)W'$ – Преобразователь сигналов посредством электрической цепи продвигает электрические сигналы к блоку управления.

8. $(XX)X$ – Блок управления посредством компаратора сравнивает соответствие импульсов излучателя и приёмника.

9. $(XX)X'$ – Блок управления посредством компаратора оценивает последовательность импульсов излучателя и приёмника.

10. $(ZW)W$ – Блок управления посредством генератора сигналов создает управляющий сигнал.

11. $(ZZ)W'$ – Блок управления посредством передатчика продвигает управляющий сигнал к реле.

12. $(ZZ)W$ – Реле посредством приемного модуля вводит (в себя) управляющий сигнал.

13. А) $(ZY)W$ – Реле посредством электрического импульса соединяет контакт 1 с контактом 2.

14. А) $(ZZ)Z'$ – Электрические сигналы посредством соединительной линии двигают энергию.

13. Б) $(ZW)Y$ – Реле посредством электрического импульса разъединяет контакт 1 от контакта 2.

14. Б) $((ZY)Z'')(Z(WY''))$ – Реле посредством контактов удерживает и изолирует электрические сигналы от оборудования.

Практически, вся последовательность операций была переведена на абстрактный функциональный уровень. Кроме того, при переводе текста на язык УСК произошла итеративная интерпретация используемых переменных, например:

X - блок управления,

Y - генератор,

W - проводная линия связи,

Z - электрические сигналы.

Или

X - блок управления,

Y - компаратор,

Z - электрические импульсы.

2.3. УСК поиск решения

В данном процессе иногда случается сбой аппаратуры: при генерации сигнала 'выключение' оборудование не отключается. Т.е. необходимо

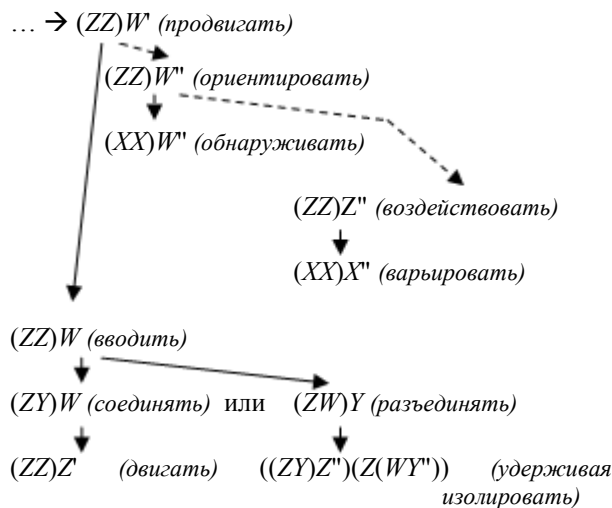
решить задачу устранения такого НЭ (1) в виде последовательности цепочек УСК, сходной с последовательностью описания процесса, но включающую дополнительные элементы, связанные с обнаружением, устранением и/или предотвращением указанной ситуации.

Задачу обнаружения, в частности, можно поставить целевым глаголом 'обнаружить', который относится к классу 'воспринимать' и представлен цепочкой $(XX)W''$.

Проследивая с помощью аксиом УСК, из какой цепочки выводится целевой глагол, находим, что это 'ориентировать' $(ZZ)W''$, который выводится из глагола 'продвигать' $(ZZ)W'$.

В свою очередь из глагола 'ориентировать' выводится глагол 'воздействовать' $(ZZ)Z''$.

Дополняем последовательность цепочек УСК, начиная с последнего глагола 'продвигать'. Получаем граф параллельных последовательностей цепочек УСК:



Применяя УСК классификатор глаголов, мы находим, что глагол 'ориентировать' является компонентом глагола 'облучать', который состоит из трех классов 'выделять и ориентировать и фокусировать'.

... → $(ZZ)W'$ (продвигать) → $(ZZ)W''$ (ориентировать) → $((ZW)Z)((ZZ)W'')((XY)X'')$ (выделяя ориентируя фокусировать) → $(XX)W''$ (обнаруживать)

Среди причин указанной ситуации может быть, например, «залипание» контактов одного из исполнительных реле.

Найденная последовательность глаголов дает подсказку, что для обнаружения 'залипания' можно использовать «параллельное облучение».

В терминах конкретной области, при запрете на изменение элементной базы, параллельное 'облучение' можно интерпретировать как добавление к основному исполнительному каналу параллельного канала с дополнительной парой нормально-разомкнутых и сопряженной ей парой нормально-замкнутых контактов. И основные и

дополнительные нормально-разомкнутые контакты подключены параллельно, но если рабочие контакты при подаче сигнала «включить» замыкаются, то параллельные им дополнительные нормально-разомкнутые контакты должны быть разомкнуты. При подаче тестирующей последовательности ('параллельного облучения') эти контакты должны одновременно срабатывать так, что, когда одна пара нормально-разомкнутых контактов замкнута, вторая – разомкнута, и наоборот. И, таким образом, оборудование при их нормальной работе остается нормально включенным. Однако, если нормально-замкнутые контакты любого из каналов окажутся постоянно разомкнутыми – то это будет признаком «залипания» нормально-разомкнутых контактов и, как следствие, - формирования сигнала тревоги. [Ероховец и др., 2008].

Кроме того можно оказывать некое воздействие, варьирующее определенный параметр в системе извещающий о неисправности или устраняющий ее. Характер воздействия может быть уточнен конкретизирующими глаголами УСК и верифицирован экспертом данной предметной области.

Заключение

В статье показано как УСК абстрагирование может применяться на ранних стадиях проектирования включающих этапы описания, постановки и решения задач.

Последовательности цепочек УСК представляют предписание и маршруты решения задач. УСК классификатор глаголов позволяет перейти с конкретного уровня представления ситуации на абстрактный и обратно на конкретный.

Структура цепочек УСК схожа со структурой элементарных компонентов обобщенных моделей вещественно-полевого анализа системы Изобретательских Стандартов [Альтшуллер, 1985] и ее модификаций [Royzen, 2008]. Поэтому система стандартов может быть переведена на УСК.

УСК разрабатывался как интеллектуальная система поддержки принятия интеллектуальных решений, которая может работать в интерактивном режиме с пользователем, если нужна дополнительная информация, или полностью автоматически, если исходных данных достаточно.

Кроме того, УСК это лингвистический инструмент, который может быть посредником при переводе текстов с одного языка на другой. При этом необходимо использовать многоязычный УСК классификатор глаголов, где каждая цепочка УСК выступает посредником, представляющим независимое от языка значение.

Благодарности

Авторы статьи благодарны, что имели

возможность работать вместе и учиться у Виктора Владимировича Мартынова. Виктор Владимирович всегда предлагал оригинальные решения, при разработке различных проектов, и внимательно относился ко всем предложениям своих учеников и коллег, что показвало не только его высокий профессионализм, а также и замечательные человеческие качества.

Библиографический список

- [Альтшуллер, 1985] Альтшуллер Г.С., Стандарты на решение изобретательских задач. / Рукопись, 1985. <http://www.altshuller.ru/triz/standards1.asp>.
- [Бойко и др., 1991] Бойко И.М., Гуминский А.П., Мартынов В.В., Семантическое кодирование и решение интеллектуальных задач // Журнал ТРИЗ, 2.1.91 (№ 3), сс 43-46.
- [Брашко и др., 2009] Брашко Н. Н., Шуляк В. В., Юдаева Л. А., Мелех О. В. Разработка световых барьеров/завес для защиты работников от травматизма в полиграфической промышленности // Труды БГТУ. Сер. IX, Издат. дело и полиграфия. 2009. Вып. XVII. С. 45–49.
- [Горбатов, 2000] В.А. Горбатов. Фундаментальные основы дискретной математики. / Информационная математика. 2000.
- [Ероховец и др., 2008] Ероховец В.К. Световой барьер ограничения доступа. / Патент РБ 4622, 2008.08.30, МПК (2006) G 08B 13/18, G 01B 11/00.
- [Мартынов, 2009] Мартынов В.В. В центре сознания человека. / БГУ. Минск., 2009.
- [Цуриков, 1991] Цуриков В.М. Проект "Изобретающая машина" - интеллектуальная среда поддержки инженерной деятельности // ТРИЗ. — 1991. — №2.1.—С. 17 -35
- [Boyko, 2001] Boyko, I. Computer Semantic Search of Inventive Solutions. / TRIZ Journal. USA. March. <http://unsemcode.com>.
- [Boyko, 2009 edition] Boyko, I. Formal Semantics Of Verbs For Knowledge Inference. // Fifth Workshop On Inference in Computational Semantics (ICoS-5), 20 - 21 April 2006, Buxton, England. (2009 Edition). <http://unsemcode.com>.
- [Broadbent, 1966] Broadbent G.H., Creativity: The design method. / London. 1966.
- [Khomenko, 2007] Khomenko N. and others, A Framework for OTSM-TRIZ Based Computer Support to be used in Complex Problem Management. // International Journal of Computer Application in Technology (IJCAT). Volume 30 issue 1/2 - 2007.
- [Lukasiewicz, 1958] Lukasiewicz J., Elementy Logiki Matematycznej. / Warszawa, 1958.
- [Martynov, 2001] Martynov V. 2001. Foundations of semantic coding. Summary. / European Humanity University. Minsk. <http://www.unsemcode.com>.
- [Pesetsky et al., 2004] Pesetsky S., Duan H., Zhang M. Innovations Through Enhanced RCA, Ontological Search and TRIZ Based Reasoning. / <http://www.iwint.com/en/tech/paper-20090104-121051.html>.
- [Royzen, 2008] Royzen, Z. Solving Problems Using TOP-TRIZ / TRIZ Consulting, Inc. 2008.

UNIVERSAL SEMANTIC CODE AS AN ABSTRACTING TECHNOLOGY FOR FUZZY ENGINEERING PROBLEM SOLVING

Kandelinsky S.L. *, Boyko I.M. **

** United Institute of Informatics Problems of National Academy of Sciences of Belarus, Minsk, Republic of Belarus*

kandel@yandex.ru

***Cisco Systems Inc., San Jose, CA, USA*

igboyko@cisco.com

This paper is focused on a Prof. V.V. Martynov semantic model Universal Semantic Code (USC) which is applied to engineering problem solving.

Introduction

There is a discussion about alternative not formal approaches to represent a fuzzy formulated engineering problem and USC providing a classification of verbs paired with their formal meaning representation and axiomatic inference of meanings from each other.

Main Part

USC is a system of sense calculation and has been developed as a knowledge representation model for natural language processing. UCS postulates: knowledge can be kept using some semantic code therefore inference of knowledge from the kept knowledge can be done based on semantic theorems and axioms.

Using power of the USC axioms and classifier of verbs, we are able to describe engineering, physical, chemical, biological, informational and other processes on an abstract level.

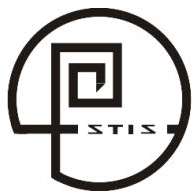
We are demonstrating USC power on the example of processing the engineering problem and providing step by step conversion of knowledge on an abstract level.

Conclusion

Converting natural language texts into strings of the semantic code is a powerful solution to build semantic Artificial Intelligence systems being able to solve different types of problems and communicate both with a human and with each other.

Formal representation of verbs and axioms of their inference are the essential properties of USC to reach this goal.

Besides, USC may be used for language translation if similar verb classifiers will be developed for other natural languages and mapped to each other by means of USC strings.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 005.8:615.478

ПРИМЕНЕНИЕ ОНТОЛОГИЙ В ПРОЕКТНОМ УПРАВЛЕНИИ

Веремьев В.Л.^{*}, Горовая Д.О.^{**}

^{*} *Санкт-Петербургский государственный политехнический университет,
г. Санкт-Петербург, Россия
veremyev.victor@gmail.com*

^{**} *ЗАО "Диджитал Дизайн", г. Санкт-Петербург, Россия
dana.gorovaya@gmail.com*

Статья посвящена мало изученной теме – использованию онтологий в проектном управлении. Актуальность этого подхода обусловлена, с одной стороны, расширяющейся в нашей стране практикой проектного управления и, как следствие, необходимостью совершенствования методологии и инструментов управления проектами, а с другой стороны, ростом интереса, разработок и внедрений в области информационных систем, построенных с использованием онтологий.

Ключевые слова: управление проектом; база знаний; онтология; RDF.

Введение

В условиях глобальной конкуренции для быстрого проведения необходимых изменений в организационной, технической и социальной сферах все шире используются проектные методы управления. В настоящее время признано как за рубежом так и в России, что методология управления проектами является эффективным инструментом развития любого предприятия.

Согласно данным по международным компаниям, включая AT&T, Boeing, Hewlett Packard, IBM, GM, непосредственными результатами от внедрения процессов управления проектами являются уменьшение времени вывода продукта на 35–75%, снижение содержательных и инжиниринговых изменений проекта на 45–68%, рост среднего показателя прибыли на 6%, повышение рентабельности инвестиций в отдельных случаях на 20%. Таким образом можно сделать вывод о том, что проектный менеджмент является стратегическим преимуществом организации в конкурентной борьбе [Ферзалиева, 2011].

Стабильной тенденцией, благоприятствующей дальнейшему развитию сферы УП, стал нарастающий темп перемен в хозяйственной жизни. Важным обстоятельством является то, что влияние факторов, обуславливающие рост роли науки управления проектами, со временем только увеличивается.

В ряде стран мира правительство и органы государственного управления увидели в методологии проектного менеджмента важный инструмент обеспечения эффективного государственного развития и заявили о формировании национального потенциала в области управления проектами [Аншин и др., 2013].

С ростом востребованности методов УП неизбежно совершенствуется методология и инструменты самого проектного управления. При этом требуется, чтобы в условиях, когда в проекте участвуют и со стороны заказчика и со стороны исполнителя специалисты с различной степенью подготовки, квалификации и опыта доступность информации и необходимых проектных знаний, оперативность и форма их получения не являлась тормозящим фактором.

Совершенствование этих свойств в быстро меняющихся условиях современного глобализирующегося бизнеса очень важно. В сложноорганизованных областях, где обрабатывается разнородная информация в различных форматах, в последнее время завоевывает все большее признание семантические методы организации и поиска информации, основанные на онтологическом подходе [Гаврилова и др., 2000].

В информационных технологиях и компьютерных науках под онтологией подразумевается явная, спецификация концептуализации, где в качестве концептуализации

выступает описание множества объектов и связей между ними. Онтология состоит из понятий терминов, организованных иерархическую систему, их описаний и правил вывода [Википедия].

Большое значение имеет накопление знаний и трансформация успешного опыта управления проектами в управленческую технику многократного использования. Это уменьшает необходимость полагаться на знания и способности отдельных руководителей проектов и других членов команды, повышает индивидуальную квалификацию каждого сотрудника, а также эффективность управления проектами компании в целом. При этом все больше специалистов уделяют особое внимание управлению знаниями, но лишь превращение знаний в технологию может повысить эффективность выполнения проекта [Динг, 2009].

Онтологический подход уже находит в мире многообразные практические применения [Гурьянова и др., 2011; Кудрявцев, 2010]. В сфере проектного управления за рубежом он также уже используется в реальных системах [Загорулько, 2008; Гурьянова и др., 2011;]. По некоторым аспектам его применения в проектном управлении появились и отечественные исследования [Клейменова и др., 2013; Зуева и др., 2009].

Таким образом, освещение вопросов построения на основе онтологий базы знаний по методологии и инструментам проектного управления представляется актуальным.

1. Обзор публикаций об использовании онтологий в управлении проектами

Поскольку в доступных информационных источниках отсутствует общее освещение работ, в которых обсуждаются основные вопросы использования онтологического подхода в проектном управлении, считаем целесообразным привести их краткий обзор в данной статье.

За рубежом, как указано выше, онтологии уже применяются в реальных задачах УП. Сюда относят следующие [Гурьянова и др., 2011; Загорулько, 2008]:

- онтологии в проекте TOVE, предназначенные для моделирования бизнес-процессов и бизнес-деятельности;
- APM для проектирования предприятия Enterprise Design Workbench;
- приложения для области планирования авиационных компаний, разработанные на основе онтологии SENSUS.

Онтологии TOVE, созданные в лаборатории интеграции предприятий университета Торонто, включают Онтологию проектирования предприятия, Онтологию проекта, Онтологию календарного планирования и Онтологию сервиса.

APM Enterprise Design Workbench - это среда проектирования, которая позволяет пользователю исследовать множество проектов предприятия. Процесс исследования – это либо проектирование, либо анализ или перепроектирование. Этот APM обеспечивает сравнительный анализ альтернативных проектов предприятий и руководство проектировщиком.

Онтология SENSUS имеет более 70 000 понятий, организованных в иерархию в соответствии с их уровнями абстракции. Ее текущее содержание было получено путем извлечения и слияния (вручную) информации из различных электронных источников знаний. Онтология для планирования военных авиационных кампаний, созданная на основе SENSUS, включают набор базовых элементов, характеризующих планы авиационных кампаний: кампания, сценарий, участники, военачальники и др. Она также включает онтологии оружия, систем, горючего и т.п.

Среди отечественных публикаций описанию принципов построения специализированной онтологии управления НИР и ОКР предприятия посвящена работа [Клейменова и др., 2013]. В ней предлагается модель онтологии, применяемая для процессов планирования распределения ресурсов при управлении проектами НИР и ОКР.

Сложность задачи планирования заключается в том, что процесс управления проектом или проектами состоит из переменного ряда взаимосвязанных этапов, на каждом из которых необходимо учитывать множество факторов и критериев для успешного достижения поставленных целей.

Предлагаемая онтология предприятия, используемая в ходе проектного управления, позволяет, с одной стороны, описывать объекты и процессы, особенности задач управления проектами, учитывая структуру предприятия, а также детализировать и накапливать информацию о конкретных проектах предприятия.

С другой стороны, разработанная онтология описывает особенности конкретных сотрудников, их личностные качества, навыки и опыт. Эта информация может быть использована для формирования индивидуальных траекторий обучения и повышения их квалификации.

Рассматриваемая онтологическая модель может использоваться для представления плана работ как сети взаимосвязанных задач исполнителей: этап построения плана работ, этап исполнения плана, контроль и анализ результатов работы. Описанная модель используется для формализации знаний по управлению проектами НИР и ОКР ОАО РКК «Энергия» [Клейменова и др., 2013]. Однако в этой работе не приводятся сведений об использовании онтологического подхода для построения базы знаний методик, методов, шаблонов, регламентов и инструментов самого проектного управления.

Онтологии предметных областей все больше привлекают внимание исследователей систем поддержки принятия решений. В частности, рассматриваются вопросы применения онтологий для управления интеллектуальными ресурсами предприятия [Муромцев, 2011]. База знаний по управлению проектами можно отнести к этой категории ресурсов.

Предлагается схема построения системы управления интеллектуальными ресурсами на базе онтологии. Основная задача такой системы - это хранение и повторное использование накопленных интеллектуальных активов. Ключевой возможностью системы является поиск требуемых ресурсов.

Методология, направления развития и опыт управления проектами обсуждаются в издающемся с 2005 г. профессиональном журнале «Управление проектами и программами». Среди его публикаций к рассматриваемой теме применения онтологического подхода в УП можно отнести статью [Зуева и др., 2009], посвященную применению таксономии в управлении проектами и программами.

Таксономия - учение о принципах и практике классификации и систематизации. Математически таксономией является древообразная структура классификаций определенного набора объектов. Формально онтология состоит из понятий терминов, организованных в таксономию, их описаний и правил вывода.

Системно и целенаправленно таксономия в бизнес-практике современных российских предприятий не применяется. Основное внимание в указанной работе направлено на рассмотрение применения таксономии в процессах разработки и использования классификаторов проектов и ближайших смежных с этим аспектов.

Рациональная деятельность человека всегда направлена на систематизацию имеющихся ресурсов, получаемой информации, знаний и опыта. Эти особенности таксономии в масштабе отдельного проекта, портфеля проектов могут в значительной мере помочь в преодолении управленческого, методического, технологического «хаоса», увеличить эффективность и прибыльность предприятия. Но при этом подчеркивается, что метод таксономии только тогда становится наиболее эффективным, когда его используют не в отдельных бизнес-процессах, а по отношению ко всей структуре управления.

Приведенные доступные источники демонстрируют интерес исследователей к использованию онтологического подхода к сфере управления проектами. Однако сама область знаний по проектному управлению, которую целесообразно рассматривать самостоятельно в качестве современного эффективного управленческого инструмента, пока не описана онтологической

моделью. Рассмотрим далее эту предметную область управления проектами и ее объекты.

2. Пространство знаний в проектном управлении

Управление проектами в современном виде возникло в 1950-е годы за рубежом применительно к важнейшим программам военного назначения. В 1950–1960-е годы был сформирован базовый инструментарий, основные методические и практические подходы. Далее в 1970–1990-е годы создавались профессиональные организации и развивалась практика и методы стандартизации в управлении отдельными проектами. В 1990-е годы возникла концепция управления при помощи проектов и управление проектами пришло непосредственно в компании. Возникли методологии управления портфелями проектов и программами, на многих предприятиях и в организациях стали создаваться корпоративные системы управления проектами [Аньшин и др., 2013].

В настоящее время управление проектами быстро развивается и стало в университетах одним из курсов программ MBA. За период с 1990 по настоящее время количество членов Института управления проектами США (PMI) увеличилось с 7,7 до 300 тысяч человек из 170 стран [PMI].

Ставя задачу исследовать вопросы построения онтологии предметной области методологии и инструментов управления проектами необходимо рассмотреть набор документов, которые содержат основные понятия этой области.

Структура методологического обеспечения корпоративной системы управления проектами может быть представлена в виде иерархии со следующими уровнями [Аньшин и др., 2013]:

- 1) стандарты организационного управления проектами, определяющие подходы к оценке и наращиванию зрелости компании в области управления проектами, программами и портфелями проектов (OPM3) – верх иерархии;
- 2) стандарты управления портфелями проектов (SPfM);
- 3) стандарты управления программами (SPgM, P2M);
- 4) стандарты, определяющие компетенции в управлении проектами (PMCDF, PM ICB, НТК, GAPPS);
- 5) стандарты управления проектами (PMBOK, ISO 10006, P2M, PRINCE2).

Сегодня стандарт PMBOK (A Guide to the Project Management Body of Knowledge - Свод знаний по управлению проектами) получил наибольшее распространение во всем мире, в том числе и в России, и является ведущим международным стандартом де-факто. Являясь сводом знаний по управлению проектами, он претендует быть основой для разработки онтологии.

К перечисленным стандартам необходимо добавить недавно принятый международный стандарт ISO 21500, который, как считают специалисты, послужит основой для дальнейшего развития и совершенствования ранее разработанных национальных и международных стандартов [Полковников, 2013].

Разработаны российские стандарты в области УП:

- ГОСТ Р 54869—2011 Проектный менеджмент. Требования к управлению проектом.
- ГОСТ Р 54870—2011 Проектный менеджмент. Требования к управлению портфелем проектов.
- ГОСТ Р 54871—2011 Проектный менеджмент. Требования к управлению программой.
- Управление проектами: Основы профессиональных знаний, Национальные требования к компетентности специалистов [НТК, 2010].

Первые 3 из перечисленных стандартов лаконичны – в них приведены перечни типовых процессов, используемых в УП. Отечественный стандарт по компетенциям [НТК, 2010] основан на наиболее авторитетном международном стандарте PM ICB и содержит требования к структуре компетенций, знаниям, личным качествам.

Российскими специалистами в сфере УП сам процесс управления проектом рассматривается как система. Предложены системная модель управления проектами [Воропаев, 2006] и подход по сравнению и гармонизации международных и национальных стандартов компетентности [Ципес, 2013] (здесь их модели не приводятся из-за ограниченного места). А в работе [Аншин и др., 2013] «сделана попытка системного подхода к рассмотрению управления проектами, раскрытию процессов, протекающих при формировании портфелей, программ и отдельных проектов, встраивании проектной методологии в стратегический бизнес-процесс компании».

Основными элементами структуры методологии управления проектами являются [Аншин и др., 2013]:

1. Методологические подходы к управлению проектами, сформулированные ведущими исследователями в сфере управления проектами:

- логико-структурный;
- системный;
- интегрированный.

2. Методы управления проектами:

- структуризации;
- сетевого планирования;
- метод освоенного объема и многие другие методы, применяемые в различных областях знаний управления проектами.

3. Модели управления проектами:

- модели зрелости организационного управления проектами;

- сетевые и другие модели.

4. Стандарты управления проектами, программами и портфелями проектов различного уровня (глобального, международного, национального, отраслевого).

5. Частные (корпоративные и отраслевые) методологии управления проектами.

Причинами разработки системной методологии управления проектами и программами (УПП) стали:

- отсутствие полного системного понимания всего спектра вопросов, касающихся управления проектами и программами;
- отсутствие системной, единой концепции УПП, надлежащим образом структурирующей знания, функции, процессы, процедуры и т.д.;
- необходимость определения технологической взаимосвязи и последовательности решения задач УПП;
- необходимость обеспечения эффективной интеграции всех элементов дисциплины управления проектами;
- необходимость развития методов и инструментов УПП, обусловленных потребностями новых и традиционных областей приложений УПП;
- сложности взаимодействия и взаимопонимания между экспертами и практиками в области управления проектами в силу многообразия технологий и терминологий в различных профессиональных сферах и литературе по УПП.

На предприятиях, где заботятся о накоплении, обмене и развитии «банка знаний», где сформирована система внутреннего обучения, повышается уровень управляемости ресурсами и организацией в целом, снижается зависимость от ключевых экспертных ресурсов при передаче знаний и навыков новым сотрудникам и при расширении масштабов бизнеса, тиражировании собственных решений.

В проектном управлении уже накоплен большой объем знаний и актуален вопрос выбора эффективной формы его организации. Управление знаниями в проектном управлении надо рассматривать в контексте существующих проблем в этой сфере. Анализируя доступные источники, следует выделить следующие существующие проблемы в управлении проектами [Полковников, 2013; Аншин, 2013; Динг, 2009; Зуева, 2009]:

- частая ограниченность высококвалифицированных ресурсов при выполнении проектов;
- недостаточная интеграция всех видов внешних и внутренних ресурсов для быстрого формирования необходимых условий в интересах эффективного выполнения проекта;
- не высокая эффективность управления огромными проектами со сложными взаимозависимостями;

- низкий уровень определения и многократного использования компонентов проекта;
- зачастую отсутствие отработанных механизмов и инструментов накопления опыта по выполнению проектов, в том числе и в виде базы знаний повторно используемых компонент.

Онтологические подходы при построении и использовании систем управления проектом, на наш взгляд, способны предоставить эффективные механизмы для решения перечисленных проблем.

Следует отметить особенности процессов в проектном управлении. Так стандарт РМВОК [РМВОК, 2008] выделяет 4 фазы жизненного цикла: начало проекта, планирование, исполнение, мониторинг и контроль, завершение. Указанные группы процессов сами сложны. Так, например, только в группу процессов планирования входит 21 процесс. Далее в управление проектами, как правило, входят определение требований, удовлетворение потребностей и ожиданий сторон, уравнивание конкурирующих ограничений. Последнее включает 6 факторов.

Все перечисленные процессы взаимодействуют между собой сложным образом, это нельзя полностью задать и изобразить в документе или с помощью рисунков [Аньшин и др., 2013].

Рассматривая сферу проектного управления ее разделяют на две основные области:

- целевая предметная область, на изменение которой направлен проект;
- область методик, методов, шаблонов, регламентов и инструментов самого проектного управления.

Онтологию УП, как инструмент многократного применения, целесообразно, прежде всего, создавать для второй из двух областей. При этом из-за сложности взаимодействия множества процессов разрабатывать онтологию для процедур планирования, мониторинга и руководства проектом затруднительно. Поэтому целью создания онтологии для проектной деятельности выберем построение базы знаний методологии, инструментов, шаблонов и опыта УП.

3. Принципы построения онтологических моделей УП

Перечисленные проблемы в построении существующей системы знаний по УП и их использовании в ходе проектной работы возможно решать с помощью их организации в онтологическую базу знаний. Этот перспективный подход обеспечит повышение эффективности поиска и анализа информации для пользователей с различной степенью подготовки.

В настоящее время исследуются вопросы использования онтологического подхода в построении как баз знаний [Гаврилова, 2000; Кудрявцев, 2010], так и систем управления

реального времени [Скобелев, 2012; Клейменова, 2013]. Для наших целей целесообразно остановиться на открытой децентрализованной базе знаний, которая способна развиваться в плохо структурированной среде. Такая задача продиктована потребностью создания отечественной базы знаний УП, создаваемой поэтапно, модульно, в том числе путем изменения ее структуры и параметров, т.е. способной к саморазвитию.

Требованием сегодняшнего дня к исследуемой базе знаний является возможность использования Интернета и соответствующих веб-сервисов. Следовательно, к ней предъявляется требование обеспечения высокого уровня интероперабельности. Интероперабельность — это способность продукта или системы, интерфейсы которых полностью открыты, взаимодействовать и функционировать с другими продуктами или системами без каких-либо ограничений доступа и реализации [Википедия]. Синтаксическая интероперабельность реализуется за счет открытости интерфейса доступа к своим сервисам путем использования единой формы для обмена данными, а именно XML и объектной модели представления документов DOM. Семантическая интероперабельность обеспечивается за счет представления знаний в виде онтологии.

Термин "онтология", как известно, в философии используется давно, а в области компьютерных наук впервые введен Томасом Грубером в 1993 году. При разработке базы знаний предметной области под онтологией понимают формальное явное описание понятий (классов) в рассматриваемой предметной области, свойств каждого понятия, описывающих различные свойства и атрибуты понятия (слотов), отношений между понятиями, ограничений, наложенных на слоты (фацетов) и утверждений, построенных из этих понятий, их свойств и связей между ними. Онтология вместе с набором индивидуальных экземпляров классов образует базу знаний [Гаврилова, 2000].

Подчеркивают, что онтология не самоцель. Основными причинами ее разработки являются:

- необходимость анализа предметной области;
- необходимость общего использования людьми и программными агентами;
- необходимость повторного использования знаний в предметной области.

Каждая интеллектуальная система может предоставлять несколько онтологических описаний, соответствующих различным областям хранящихся в ней декларативных знаний и, таким образом, выступает как хранилище библиотеки онтологий [Лапшин, 2010].

В настоящее время разработаны языки описания онтологий, которые являются стандартными. Это язык RDF (Resource Description Framework), RDFS (RDF Schema) и OWL (Web Ontology Language).

RDF представляет собой язык для описания ресурсов и эти записи могут читать и понимать компьютеры. RDFS предоставляет специальный словарь для RDF, который может быть использован для определения таксономий классов, свойств и др. Для записи семантики предметных областей в онтологиях служит язык OWL.

Разработка онтологии предусматривает несколько этапов [Лапшин, 2010]:

- определение отрасли и масштаба онтологии;
- рассмотрение вариантов повторного использования существующей онтологии;
- перечисление важных терминов в онтологии;
- определение классов и иерархии классов;
- определения свойств классов - слотов;
- определение фактов свойств;
- создание экземпляров.

Для редактирования документов с онтологиями, написанными на языке OWL, широко используют программу открытого доступа Protégé. Редактор Protégé предоставляет возможность задавать к онтологии запросы на языке SPARQL, а также производить над ней различные логические манипуляции.

Заключение

В данной работе сделана попытка поставить задачу исследования такой мало изученной области как построение на основе онтологий базы знаний по методологии и инструментам управления проектами.

Использование онтологий в базах знаний по проектному управлению может быть эффективным для работы в проектных командах и подготовке специалистов.

Библиографический список

- [Аншин, 2013] Управление проектами: фундаментальный курс / под ред. В. М. Аншина, О. Н. Ильиной. — М.: Изд. дом Высшей школы экономики, 2013. — 620 с.
- [Верзук, 2008] Верзук Э. Управление проектами: ускоренный курс по программе MBA. — М.: ООО И.Д. Вильямс, 2008. — 480 с.
- [Воропаев, 2006] Воропаев В.И., Секлетова Г.И., Воропаева-Кейтс М.В. Системная модель управления проектами как основа структуризации профессиональных знаний и компетентности специалистов // Управление проектами и программами. — 2006. — №4(8). — с. 304-317.
- [Гаврилова и др., 2000] Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем: Учебник. М.: Питер, 2000. — 382 с.
- [Гаврилова, 2010] Гаврилова Т.А. Онтологический инжиниринг. БИГ-Петербург. http://www.big.spb.ru/publications/bigspb/km/ontolog_engeneering.shtml
- [Гурьянова и др., 2011] М.А. Гурьянова, И.В. Ефименко, В.Ф. Хорошевский Онтологическое моделирование экономики предприятий и отраслей современной России. Часть 2. Мировые исследования и разработки: аналитический обзор: препринт WP7/2011/08 (ч. 2). — М.: Изд. Высшей школы экономики, 2011. — 88 с.
- [Динг, 2009] Динг Р. Повышение эффективности работы исполнителей в различных проектах за счет использования

стандартизированных компонентов// Управление проектами и программами. — 2009. - №2(18), с. 106-118.

[Загорюлько Ю.А., 2008] Загорюлько Ю.А. Методы и методологии разработки, сопровождения и реинжиниринга онтологий// Ин-т систем информатики им. А.П. Ершова СО РАН, 2008 г. — 73 слайда.

[Зуева, 2009] Зуева А.Г., Неизвестный С.И. О применении таксономии в проектном управлении// Управление проектами и программами. - 2009. - №2(18), с. 94-104.

[Клейменова и др., 2013] Е.М. Клейменова, П.О. Скобелев, В.Б. Ларюхин, Д.С. Косов, Е.В. Симонова Разработка и использование онтологии интеллектуальной системы управления проектами НИР и ОКР// Вестник Самарского государственного технического университета. Сер. Технические науки. — 2013. - №2. — с. 18-25.

[Кудрявцев, 2010] Кудрявцев Д.В. Системы управления знаниями и применение онтологий. — СПб.: Изд-во Политехнического университета., 2010. — 343 с.

[Лапшин, 2010] Лапшин В.А. Онтологии в компьютерных системах. - М.: Научный мир, 2010. — 224 с.

[Муромцев и др., 2011] Д.И. Муромцев, Г.В. Варгин, И.А. Семерханов Применение онтологии в системе управления интеллектуальными ресурсами// Научно-технический вестник Санкт-Петербургского государственного университета информационных технологий, механики и оптики, 2011, № 2 (72), с. 170.

[НТК, 2010] Управление проектами: Основы профессиональных знаний, Национальные требования к компетентности специалистов. Версия 3.0. - М.: ЗАО Проектная Практика, 2010.

[Полковников, 2013] Полковников А.В. Стандартизация в области управления проектами: текущее состояние и направления развития// Управление проектами и программами. - 2013. - №2, с. 124-132.

[Скобелев, 2012] Скобелев П.О. Онтологии деятельности для ситуационного управления предприятиями в реальном времени // Онтология проектирования. — 2012. - №1(13). — с.6-38.

[Ферзалиева, 2011] Ферзалиева А.О. Корпоративная система управления проектами как эффективный инструмент управления инновационной деятельностью организации // Креативная экономика. — 2011. — № 10 (58). — с. 34-41.

[Ципес, 2013] Ципес Г.Л., Воропаев В.И., Товб А.С., Клименко О.А. Стандарты компетенции – гармонизация через структуризацию// Управление проектами и программами. — 2013. — №1(33). — с.52-60.

[РМВОК, 2008] Руководство к Своду знаний по управлению проектами (Руководство РМВОК). — 4-е изд. Project Management Institute. — 2008. — 408 с.

[PMI] Институт управления проектами США (Project Management Institute), www.pmi.org.

ONTOLOGIES FOR PROJECT MANAGEMENT

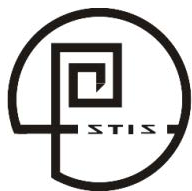
Veremyev V.L. *, Gorovaya D.O. *

**Saint-Petersburg State Polytechnic University,
Saint-Petersburg, Russia*

veremyev.victor@gmail.com

***Digital Design, Saint-Petersburg, Russia
dana.gorovaya@gmail.com*

The article deals with one of the under investigated topics of using ontologies in project management. The topicality of such approach is based on the emerging role of project management practices in Russia. Also the new methodology should be developed aiming the refinement and improvement of practical project management tools and methods. Ontological approach is a novel promising technology that is now widely used in information systems solutions.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8 + 004.942

ОНТОЛОГИЧЕСКИЙ ПОДХОД К ПОСТРОЕНИЮ СИСТЕМЫ ИМИТАЦИИ, НАСТРАИВАЕМОЙ НА КОНКРЕТНУЮ ПРЕДМЕТНУЮ ОБЛАСТЬ

Замятина Е.Б., Михеев Р.А.

*Пермский государственный национальный исследовательский университет
г. Пермь, Россия*

e_zamyatina@mail.ru

mikheev@prognoz.ru

В работе рассматривается модельно-ориентированный подход к построению системы имитации, настроенной на конкретную предметную область. Модельно-ориентированный подход базируется на онтологиях, определенных в системе имитации TriadNS. Система имитации TriadNS предназначена для проектирования и моделирования компьютерных сетей.

Ключевые слова: имитационное моделирование, онтология, адаптируемость, компьютерные сети.

Введение

Имитационное моделирование является широко применяемым, а иногда и единственным, методом анализа сложных динамических систем, каковыми являются, например, компьютерные сети. Однако для того, чтобы применение методов имитационного моделирования было эффективным, необходимо привлекать к исследованиям специалистов из разных областей знаний. Так, например, при разработке алгоритмов маршрутизации целесообразно применять теорию графов (нахождение кратчайшего расстояния, определение характеристик графа, вершинами которого являются вычислительные узлы компьютерной сети). Исследование трафиков обычно выполняется с применением теории массового обслуживания, для исследования параллельных и распределенных программных систем используют теорию сетей Петри. Поэтому важно, чтобы при разработке имитационной модели исследователь смог оперировать привычными для него терминами, понятиями. В этом случае целесообразно применять онтологический подход, поскольку онтологии представляют собой концепты (понятия предметной области) и связи между ними. В статье рассматривается и другой подход, который основан на использовании языкового инструментария. Кроме того, для тщательного исследования проблемы, ситуации, сложной динамической системы целесообразно разрабатывать различные их модели и преобразовывать (трансформировать) одну модель в

другую. Таким образом, возникает возможность строить многомодельные (полимодельные) комплексы [Соколов, 2005]. Полимодельные комплексы способствуют улучшению качества имитационного исследования (квалиметрия имитационной модели).

1. Применение онтологий в имитационном моделировании

Известно, что онтология – это описание типов сущностей предметной области, их свойств и отношений. Каждая предметная область (некая часть реального мира) может быть описана с помощью онтологий. Онтологии создаются и используются во множестве областей знаний, в том числе, известны примеры их успешного применения в имитационном моделировании. Однако создание онтологий для моделирования является достаточно сложной задачей, поскольку этот метод используют для исследования самых разнообразных систем, относящихся к различным предметным областям (химическим, физическим, транспортным и т.д.). Кроме того, методы имитационного моделирования основаны на математических, вероятностных и статистических расчетах, и, таким образом, онтологии для этих областей должны служить основой для всех остальных. Онтологии используют на различных этапах имитационного моделирования, начиная с этапа сбора информации о моделируемой системе и заканчивая этапом валидации модели [Sargent, 2004]. Примерами использования онтологий моделирования могут служить управляемые онтологиями среды

моделирования, а также подходы к объединению различных федератов, разрабатываемые для High Level Architecture (HLA) [Rathman, 2005]. Подход, разрабатываемый для HLA, использует онтологии для описания требований, которым должны удовлетворять интерфейсы федератов для успешного взаимодействия в федерации, а так же для разработки этих требований с учетом знаний о моделируемой предметной области. В работе [Liang, 2006] представлена онтология портов, рассматриваемая как средство автоматизации построения моделей из компонентов. Порты описывают интерфейс, определяющий границы компонентов или подсистем в конфигурации системы. Система представлена как конфигурация подсистем или компонентов, соединенных друг с другом через четко определенные интерфейсы. Онтологии успешно применяются и в других работах по имитационному моделированию [Benjamin, 2005], [Benjamin, 2006]. Итак, перед авторами стоит задача выполнить настройку системы имитации на конкретную предметную область с целью проведения качественных исследований. Рассмотрим различные подходы к решению этой задачи.

2. Настройка на конкретную предметную область

Для настройки системы имитации можно воспользоваться либо программными средствами, которые предоставляют языковые инструментарии (или DSM-платформы), предназначенные для создания предметно-ориентированных языков (DSL, Domain Specific Language), либо программными средствами на основе онтологий. Рассмотрим первый способ.

2.1. Использование языкового инструментария для построения DSL

Существуют языковые инструментарии, которые позволяют строить DSL (метамодели), с использованием которых создаются модели соответствующих предметных областей (MetaEdit+, DSL Tools и др.). В Пермском университете был разработан языковой инструментарий MetaLanguage [Сухов, 2013], который лишён ряда ограничений, присущих другим DSM-платформам. В этом случае порядок построения системы имитации, настраиваемой на предметную область, сводится к следующим действиям:

- Создание новой метамодели с указанием её имени и описания (если это необходимо). Метамодель в данном случае – это предметно-ориентированный язык моделирования, который используется для создания моделей, ориентированных на решение конкретных задач.

Построение метамодели выполняется с помощью графического редактора моделей. При создании метамодели в первую очередь определяются базовые конструкции языка. Базовыми элементами, которые используются для

создания метамodelей в MetaLanguage, являются сущность, отношение, ограничение. Таким образом, в процессе создания DSL определяются сущности метамодели, отношения между ними, задаются ограничения, налагаемые на метамодели.

- Используя полученный DSL, исследователь создает модели, содержащие объекты, описывающие конкретные сущности предметной области и связи между ними.

- По окончании построения модели необходимо проверить, удовлетворяет ли она ограничениям, которые были на неё наложены, – выполнить валидацию созданной модели. Если какие-либо ограничения не выполняются, пользователь будет проинформирован об этом и сможет внести изменения в модель.

Перечисленные выше программные средства предназначены не только для разработки имитационных моделей в конкретных системах имитации (например, GPSS-модели). Они имеют гораздо более широкое применение и используются, в частности, для построения моделей бизнес-процессов [Замятин и др., 2013].

2.2. Онтологический подход к построению предметно-ориентированной модели

Вторым способом настройки на предметную область является использование модельно-ориентированного подхода на основе онтологий. Этот подход используется для построения графического интерфейса различных программных продуктов [Грибова, 2005], [Грибова, 2006], [Грибова, 2010]. В качестве примера можно рассмотреть подход к построению пользовательских интерфейсов сетевой информационно-вычислительной системы, в рамках которого сначала строится высокоуровневое описание интерфейса, его модель, а затем по ней генерируется программный код с использованием технологии сборки [Веселов, 2009]. Авторы отмечают преимущества такого подхода, которые заключаются в следующем: 1) уменьшение временных затрат на разработку графического интерфейса и упрощение модификации, поскольку разработчику предоставляется только декларативное описание; 2) использование для сборки интерфейса как собственные разработки, так и заёмные компоненты. Реализация предлагаемого подхода позволяет использовать ресурсы географически удаленных кластеров членов сетевого общества и выполнять отдельные части проекта, не выходя за пределы своей корпоративной среды и понятий своей предметной области.

Для настройки системы имитации на предметную область в TriadNS используют онтологический подход, что обусловлено следующими причинами:

- В TriadNS разработаны базовая и предметно-ориентированная онтологии, которые используются для автоматического доопределения

модели по семантическим признакам.

- Компьютерные сети являются примером быстро меняющейся предметной области (новые типы сетей, новые протоколы, алгоритмы передачи сообщений, алгоритмы маршрутизации). Поэтому необходимо располагать языковыми и программными механизмами, позволяющими выполнять оперативную автоматическую или автоматизированную настройку на динамически изменяющуюся предметную область.

- Компьютерные сети могут включать большое количество вычислительных узлов, поэтому на имитационный эксперимент требуется большое количество времени. Выходом в данной ситуации является распараллеливание имитационного прогона, распределение объектов имитационной модели по узлам высокопроизводительной вычислительной среды. При этом необходимо сохранять каузальность событий. Для распараллеливания имитационного прогона существуют классические алгоритмы синхронизации (оптимистический и консервативный). Усилия многих авторов направлены на усовершенствование этих алгоритмов. В TriadNS также разрабатываются усовершенствованные алгоритмы синхронизации (на базе оптимистического алгоритма). Особенность этих алгоритмов заключается в том, что они используют знания о модели, а знания эти частично извлекаются из онтологий. Более того, для равномерного распределения вычислительной нагрузки на узлы также используют семантические знания о модели. Разработан мультиагентный подход, агенты действуют по правилам, извлекаемым из онтологий, как базовой, так и предметно-ориентированной [Миков, 2013].

- Верификация и валидация моделей также выполняются с применением онтологий [Замятина, 2013], из онтологий извлекаются правила, которым соответствует модель.

- Использование онтологий позволяет автоматически интегрировать компоненты имитационной модели, включать в уже разработанные модели компоненты, подготовленные в другой имитационной системе.

Так как онтологии применяются во всех перечисленных выше случаях, целесообразно использовать их и для настройки на конкретную предметную область. Исследования и проведенные эксперименты показали, что онтологии позволяют поддерживать адаптируемость системы моделирования и ее открытость [Замятина, 2012].

Прежде, чем более подробно рассмотреть особенности разработки онтологий в TriadNS, рассмотрим принципы построения имитационной модели, лингвистических и программных средств сбора и обработки информации о модели и т.д.

3. Представление имитационной модели в симуляторе компьютерных сетей TriadNS

Симулятор компьютерных систем разработан на базе системы автоматизированного проектирования и моделирования вычислительных систем Triad [Миков, 1995].

3.1. Описание имитационной модели

Описание имитационной модели в Triad состоит из трех слоёв: слоя структур (*STR*), слоя рутин (*ROUT*) и слоя сообщений (*MES*). Таким образом, модель в системе Triad можно определить как $M = \{STR, ROUT, MES\}$.

Слой структур представляет собой совокупность объектов, взаимодействующих друг с другом посредством посылки сообщений. Каждый объект имеет полюсы (входные P_{in} и выходные P_{out}), которые служат соответственно для приёма и передачи сообщений. Слой структур можно представить графом. В качестве вершин графа следует рассматривать отдельные объекты. Дуги графа определяют связи между объектами. Объекты действуют по определённому алгоритму поведения, который описывают с помощью программы, которую в Triad принято называть рутинной (*route*). Рутинная представляет собой последовательность событий e_i , планирующих друг друга ($e_i \in E, i=1 \div n$, – множество событий, множество событий рутинной является частично упорядоченным в модельном времени). Выполнение события сопровождается изменением состояния объекта. Состояние объекта определяется значениями переменных рутинной. Таким образом, система имитации является событийно-ориентированной. Рутинная так же, как и объект, имеет входные (Pr_{in} и выходные Pr_{out}) полюса. Входные полюса служат соответственно для приёма сообщений, выходные полюса – для их передачи. В множестве событий рутинной выделено входное событие e_{in} . Все входные полюса рутинной обрабатываются входным событием. Обработка выходных полюсов осуществляется остальными событиями рутинной. Для передачи сообщения служит специальный оператор **out** (**out** <сообщение> **through** <имя полюса>). Совокупность рутин определяет слой рутин ROUT.

Слой сообщений (*MES*) предназначен для описания сообщений сложной структуры. Алгоритмом имитации называют совокупность объектов, функционирующих по определённым сценариям, и синхронизирующий их алгоритм. Последовательный алгоритм является событийно-ориентированным. Каждая рутинная имеет свой локальный календарь событий (*tloc*). При выполнении очередного события осуществляется поиск события с минимальным временем (поиск в календарях событий всех рутин) и управление передается рутинной, которая включает это событие.

3.2. Сбор и обработка информации о функционировании имитационной модели

Для сбора, обработки и анализа имитационных моделей в системе TriadNS существуют специальные объекты – информационные процедуры и условия моделирования, реализующие алгоритм исследования. *Информационные процедуры* ведут наблюдение только за теми элементами модели (событиями, переменными, входными и выходными полюсами), которые указаны пользователем. Если в какой-нибудь момент времени имитационного эксперимента пользователь решит, что следует установить наблюдение за другими элементами или выполнять иную обработку собираемой информации, он может сделать соответствующие указания, подключив к модели другой набор информационных процедур. *Условия моделирования* анализируют результат работы информационных процедур и определяют, выполнены ли условия завершения моделирования.

3.3. Запуск модели на выполнение

Имитационная модель, описанная на языке Triad, может быть запущена на моделирование с помощью специального оператора *simulate*: *simulate* <список элементов моделей, за которыми установлено наблюдение> *on conditions of simulation* <наименование> (список фактических настроечных параметров)]<список входных и выходных фактических параметров интерфейса>]. Следует отметить, что исследователь может одновременно запустить на моделирование сразу несколько моделей, указав элементы, за которыми будут следить информационные процедуры, перечисленные в условиях моделирования. Кроме того, можно указать сразу несколько условий моделирования. Например, при моделировании компьютерных сетей целесообразно рассматривать эти компьютерные сети с разных точек зрения: исследователей может интересовать *стоимость* проектируемых компьютерных сетей, их *производительность*, их *надежность*, их *доступность*. Во всех этих случаях набор собираемых данных может быть различным, разными способами обрабатываются наблюдаемые данные, условия завершения моделирования также могут отличаться друг от друга. В этом случае целесообразно провести над моделью несколько имитационных экспериментов, используя различные критерии (условия моделирования).

4. Многомодельное имитационное исследование с использованием онтологий

Итак, как уже было сказано ранее, многомодельное (полимодельное) исследование позволяет привлечь к анализу результатов имитационного исследования специалистов различных предметных областей, дать им возможность работать в привычных для них

терминах, оперировать определенным кругом понятий. Для настройки на предметные области в TriadNS используют базовую и предметно-ориентированные онтологии. В настоящее время разработаны онтологии, учитывающие особенности реализации компьютерных сетей и онтологии, для представления модели в терминах системы массового обслуживания. Кратко рассмотрим особенности реализации.

4.1. Базовая онтология

Базовая онтология основывается на принципах построения имитационной модели в TriadNS. Базовую онтологию используют для представления семантических знаний, необходимых для доопределения моделей. Для представления онтологий был выбран язык OWL, поскольку существует большое количество инструментальных средств работы с онтологиями OWL, поддерживающих возможность публиковать созданные онтологии в сети Internet и объединять информацию из различных источников, как локальных, так и находящихся в глобальной сети. Для работы с онтологиями использовался инструментарий Jena OWL API. Автоматическое доопределение модели предполагает автоматическое извлечение рутины из базы рутин и ее наложение на терминальную вершину, поведение которой заранее не определено (вершина без указания алгоритма ее поведения и называется терминальной). Извлеченная рутина должна соответствовать трем условиям автоматического доопределения:

- условию спецификации (проверка семантического имени, если семантическое имя – «маршрутизатор», то должна быть извлечена рутина, реализующая алгоритм работы маршрутизатора),
- условию конфигурации (проверка количества входов и выходов вершины и рутины, количество их должно совпадать),
- условию декомпозиции (проверка графо-окружения, т.е. топология графа, образованного из вершин, смежных терминальной должно совпадать с «шаблоном» из базы знаний).

Семантические знания, которые необходимы для проверки условий автоматического доопределения модели, требуется извлечь из онтологического описания слоя структур. В качестве такого описания используется базовая онтология, импортируемая всеми создаваемыми онтологиями. В этой онтологии определены следующие классы: Model (класс, описывающий множество моделей языка Triad), SubMod (класс, описывающий множество всех экземпляров рутин и структур), Graph (класс, описывающий множество структур моделей, является подклассом SubMod), Routine (множество экземпляров рутин, является подклассом SubMod), Object (множество всех вершин структуры модели, является суперклассом для всех семантических типов) и т.д. Фрагмент базовой онтологии представлен на рисунке 1. Для работы с самими

онтологиями реализован класс *OntoManager*, поддерживающий загрузку нескольких онтологий, создание и сохранение онтологий. Кроме того, реализован класс *TypeManager* для работы с семантическими типами и для поиска соответствующего экземпляра рутин.

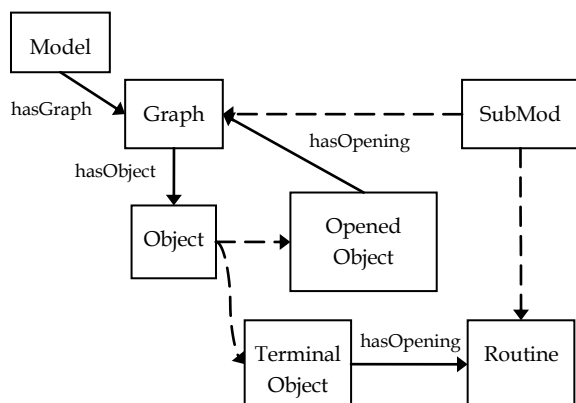


Рисунок 1 – Фрагмент базовой онтологии

4.2. Онтология для представления компьютерных сетей

Предметно-ориентированная онтология системы *TriadNS* дополняет базовую онтологию. Введены специализированные для области компьютерных сетей подклассы основных классов базовой онтологии: *ComputerNetworkModel* – модель компьютерной сети; *ComputerNetworkStructure* – структура модели компьютерной сети; *ComputerNetworkNode* – элемент компьютерной сети, изначально содержит подклассы *WorkStation*, *Server*, *Router*; *ComputerNetworkRoutine* – рутина элемента компьютерной сети и т.д.

В онтологии компьютерных сетей определены два специальных свойства для полюса вершины, представляющей элемент компьютерной сети:

- *isRequired(ComputerNetworkRoutinePole, Boolean)* определяет, обязательно ли полюс должен быть соединен с другим полюсом;
- *canConnectedWith(ComputerNetworkRoutinePole, ComputerNetworkRoutine)* определяет семантический тип элемента, с которым полюс можно соединить.

Для построения модели может быть использован язык *Triad* или графический редактор. Графический редактор по сути является визуальным языком и используется для построения как имитационной модели, так и онтологий соответствующей предметной области.

4.3. Онтология для представления модели компьютерной сети в виде системы массового обслуживания

Онтология семантических типов элементов систем массового обслуживания (СМО) дополняет базовую онтологию в следующем. Во-первых, вводятся специализированные для области систем массового обслуживания подклассы основных

классов базовой онтологии: *MpsModel* – модель системы массового обслуживания, *MpsStructure* – структура модели системы массового обслуживания и т.д. Во-вторых, вводятся четыре подкласса класса вершины, соответствующие основным элементам систем массового обслуживания: *Generator*, *Queue*, *Channel*, *Terminator*. В-третьих, вводятся классы для представления специфической для теории очередей информации: типа распределения времени, типа процедуры выборки из очереди и т.д. Иерархия семантических типов в онтологии представляется в виде иерархии классов рутин, верхним элементом которой является подкласс класса *Routine* базовой онтологии – класс *MpsRoutine*, то есть рутина системы массового обслуживания. При формировании иерархии семантических типов большую роль играет множественное наследование. На рисунке 2 представлен фрагмент системы массового обслуживания.

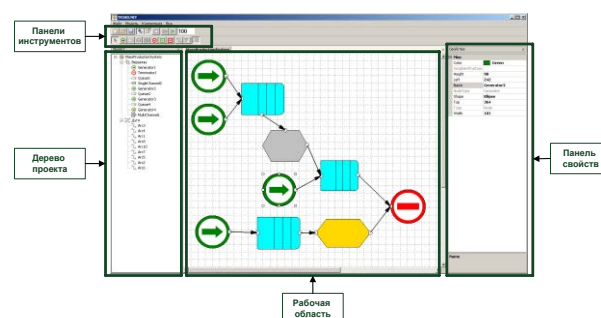


Рисунок 2 – Фрагмент графического представления модели СМО

Заключение

В работе представлен онтологический подход к настройке системы имитации на конкретную предметную область в симуляторе компьютерных сетей *TriadNS*. Настройка на конкретную предметную область позволяют улучшить качество имитационных исследований благодаря тому, что к ним могут быть привлечены специалисты из разных областей знаний, но при этом они могут работать в привычной для них предметной области и использовать определенный круг понятий, концептов, терминов. В настоящее время построены базовая и предметно-ориентированная системы имитации для компьютерных сетей и систем массового обслуживания. Коллектив авторов предполагает разработать программные и языковые средства, позволяющие выполнить преобразования (трансформации) одной модели в другую (многомодельное имитационное исследование).

Работа выполнена при поддержке РФФИ (проекты № 12-07-00302-а «Теоретические и технологические основы интеллектуального управления жизненным циклом сложных изделий» и № 13-07-96506_р-юг-а «Разработка деонтических основ мультиагентных технологий формирования баз знаний для виртуальных лабораторий»), и гранта Министерства образования и науки (проект № 8.5782.2011) «Автоматизация и оптимизация этапов имитационного моделирования».

Библиографический список

- [Веселов, 2009] Веселов А.В., Остапкевич М.Б., Пискунов С.В. Автоматизированная генерация пользовательских интерфейсов для сетевой информационно-вычислительной системы // Тр. 7-й Междунар. конф. памяти акад. А.П. Ершова "Перспективы систем информатики". Новосибирск: Сиб. науч. изд-во, 2009. С. 84–90.
- [Грибова, 2005] Грибова В.В., Клещев А.С. Использование методов искусственного интеллекта для проектирования пользовательского интерфейса // Информационные технологии. - 2005. - №8. - С.58-62.
- [Грибова, 2006] Грибова В.В., Клещев А.С. Управление проектированием и реализацией пользовательского интерфейса на основе онтологий // Проблемы управления. 2006. №2. С.58-62.
- [Грибова, 2008] Грибова В.В. Модель генерации кода пользовательского интерфейса для различных типов диалога // Научно-технические ведомости СПбГПУ. – Санкт-Петербург: Изд-во Политехнического университета. –2008. –№3. –С.145-151.
- [Грибова, 2010] Грибова В.В., Черкезшвили Н.Н. Развитие онтологического подхода для автоматизации разработки пользовательских интерфейсов с динамическими данными // Информационные технологии. -2010. -№10. -С. 54-58.
- [Замятина и др., 2013] Замятина Е.Б., Лядова Л.Н., Сухов А.О. О подходе к интеграции систем моделирования и информационных систем на основе DSM-платформы MetaLanguage // В кн.: Технологии разработки информационных систем ТРИС-2013. Материалы IV Международной научно-технической конференции / Т.1. Таганрог: Издательство Технологического института ЮФУ, 2013. С. 61-70.
- [Замятина и др., 2011] Замятина Е.Б., Миков А.И., Михеев Р.А. Применение онтологий в системе проектировании и моделирования компьютерных сетей TRIADNS. Труды Конгресса по интеллектуальным системам и информационным технологиям. «IS&IT'11», Научное издание в 4-х томах, М. Физматлит, 2011, Т.1, стр. 253-260
- [Замятина и др., 2012] Замятина Е.Б., Миков А.И. Программные средства системы имитации Triad.Net для обеспечения ее адаптируемости и открытости. Информатизация и связь. 2012. № 5. С. 130–133.
- [Миков и др. а, 2010] Миков А.И., Замятина Е.Б. Проблемы повышения эффективности и гибкости систем имитационного моделирования. Проблемы информатики, №4(8), Новосибирск, Институт вычислительной математики и математической геофизики СО РАН, 2010. С.49-64.
- [Миков и др. б, 2010] Миков А.И., Замятина Е.Б., Козлов А.А. Мультиагентный подход к решению проблемы равномерного распределения вычислительной нагрузки. Natural and Artificial Intelligence, ITHEA, Sofia, Bulgaria, 2010. P.173-180.
- [Соколов, 2005] Соколов Б.В., Юсупов Р.М. Концептуальные основы квалиметрии моделей и полимодельных комплексов // Имитационное моделирование. Теория и практика: сб. докл. 2-й всерос. науч.-практ. конф. ИММОД–2005. Т. 1. СПб: ЦНИИТС. 2005. С. 65–70.
- [Сухов, 2013] Сухов А.О. Инструментальные средства создания визуальных предметно-ориентированных языков моделирования / Фундаментальные исследования. – 2013. – № 4 (ч. 4). – С. 848-852.
- [Benjamin, 2005] Benjamin P., Akella K.V., Malek K., Fernandes R. An Ontology-Driven Framework for Process-Oriented Applications // Proceedings of the 2005 Winter Simulation Conference / M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.,– pp. 2355-2363.
- [Benjamin, 2006] Benjamin P., Patki M., Mayer R. J. Using Ontologies For Simulation Modeling // Proceedings of the 2006 Winter Simulation Conference/ L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds. –pp.1161-1167
- [Liang, 2003] Liang V.-C., Pardis C.J.J. A Port Ontology for Automated Model Composition // Proceedings of the 2003 Winter Simulation Conference / S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds., - pp. 613-622.
- [Mikov, 1995] Mikov A.I. Formal Method for Design of Dynamic Objects and Its Implementation in CAD Systems // Gero J.S. and F.Sudweeks F.(eds), Advances in Formal Design Methods for CAD, Preprints of the IFIP WG 5.2 Workshop on Formal Design Methods for Computer-Aided Design, Mexico, Mexico, 1995. pp.105-127.
- [Mikov, 2007] Mikov A., Zamyatina E., Kubrak E. Implementation of simulation process under incomplete knowledge

using domain ontology //Proceedings of the 6-th EUROSIM Congress on Modeling and Simulation. University of Ljubljana, Slovenia, 2007, Vol.2. Book of full papers. p.1-7.

[Mikov, 2013] Mikov A., Zamyatina E., Kozlov A., Ermakov S. Some Problems of the Simulation Model Efficiency and Flexibility. Proceedings of «2013 8th EUROSIM Congress on Modelling and Simulation EUROSIM 2013», Cardiff, Wales, United Kingdom. P. 532-538.

[Rathnam, 2004] Rathnam T., Paredis C.J.J. Developung Federation Object Models Using Ontologies // Proceedings of the 2004 Winter Simulation Conference / R. G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, eds.,– pp. 1054-1062.

[Sargent, 2004] Sargent R.G. Some Recent Advances In The Process World View. // Proceedings of the 2005 Winter Simulation Conference / M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.,– pp. 293-298.

ONTONTOLOGICAL APPROACH TO A SIMULATION SYSTEM CONSTRUCTING FOR THE PARTICULAR SUBJECT AREA

Zamyatina E.B., Mikheev R.A.

*Perm State National Researching University,
Perm, Russian Federation*

e_zamyatina@mail.ru

mikheev@prognoz.ru

This paper discusses the problem of simulation system constructing for the specific domain. Authors suggest the ontological approach using. The simulation system TriadNS is considered. This simulation system is devoted for computer networks design and analysis. Authors represent base ontology and some another ontologies describing the concepts of specific domain.

Introduction

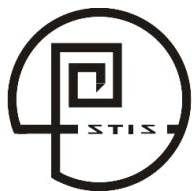
Introduction considers the actuality of problem. Indeed the adjusting of simulation model for the specific domain permits differ specialists take part in common investigations. So computer networks may be considered from the point of view of graph theory, or as queue networks, or as Petri nets and so on. The investigations become to be more quality.

Main Part

Main part includes the overview of ontology using in simulation. The overview shows that ontological approach may be useful during design, verification and validation of simulation model. Furthermore the simulation system TriadNS is described, and base ontology and ontologies for specific domains are presented.

Conclusion

The results of investigations and new directions in investigations are presented. The authors intend to design rules of transformation of one simulation model to another.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8 + 004.4

ИНТЕГРАЦИЯ СИСТЕМ МОДЕЛИРОВАНИЯ НА ОСНОВЕ DSM-ПЛАТФОРМЫ С ИСПОЛЬЗОВАНИЕМ ОНТОЛОГИЙ

Сухов А.О.^{*}, Лядова Л.Н.^{*}, Замятина Е.Б.^{**}

^{} Национальный исследовательский университет «Высшая школа экономики» (Пермский филиал),
г. Пермь, Россия*

LNLyadova@gmail.com

Sukhov.PSU@gmail.com

*^{**} Пермский государственный национальный исследовательский университет,
г. Пермь, Россия*

E_Zamyatina@mail.ru

Рассматривается подход к интеграции различных систем моделирования, основанный на использовании возможностей DSM-платформы (языкового инструментария) MetaLanguage. Инструментальные средства MetaLanguage позволяют разрабатывать предметно-ориентированные языки (DSL) и описывать их трансформации. Предметно-ориентированное моделирование обеспечивает возможность работы аналитиков в терминах предметной области при создании моделей. Компонент трансформации является основой для разработки средств экспорта и импорта моделей, их интеграции. Онтологии формализуют знания о предметных областях и моделях и используются средствами автоматизации разработки DSL, а также средствами поиска и доопределения моделей.

Ключевые слова: моделирование, предметно-ориентированные языки, онтология, адаптируемость, трансформации.

Введение

Информационные и аналитические системы, используемые для решения различных задач управления, создаются с применением технологий, основой которых являются модели. Эти модели разрабатываются с использованием различных инструментальных средств, выбор которых зачастую определяет и выбор языков моделирования. Таким образом, используемый разработчиками инструментарий фактически «навязывает» им и пользователям систем соответствующий формальный аппарат, отсутствует возможность адаптации используемых средств к решаемым задачам, потребностям пользователей. Кроме того, часто отсутствует возможность использования ранее созданных моделей из-за невозможности или трудоёмкости их преобразования при переходе к новым средствам моделирования [Замятина и др., 2013].

Максимальная гибкость средств моделирования достигается при реализации многоуровневых моделей, описывающих моделируемые системы с разных точек зрения и с разной степенью

детализации с использованием языков моделирования, соответствующих особенностям предметных областей и решаемых пользователями задач, – предметно-ориентированных языков моделирования (DSL – Domain Specific Language). Мультиязыковые системы моделирования могут быть созданы на основе DSM-платформ (DSM – Domain Specific Modeling), основное назначение которых – разработка высокоуровневых предметно-ориентированных языков, предназначенных для создания моделей, ориентированных на решение задач в различных предметных областях, учитывающих их специфику, а также потребности аналитиков, экспертов, занимающихся разработкой моделей, их анализом [Hessellund, 2009]. При создании DSL могут быть учтены не только особенности предметной области, но и квалификация пользователей. Создание таких средств обеспечивает основу для интеграции различных информационных и аналитических систем, созданных на основе моделей и использующих их для решения задач пользователей [Balasubramanian and others, 2007; Bräuer and others, 2007].

Предметно-ориентированные языки можно рассматривать как метамодели, с помощью которых разрабатываются модели предметных областей [Сухов, 2013]. Для снижения трудоёмкости разработки DSL и моделей предлагается использовать ранее разработанные метамодели, доопределяя и уточняя их [Elokhov and others, 2013]. Онтологический подход широко используется для решения задач поиска. В сочетании с другими методами его можно успешно использовать для поиска (мета)моделей, удовлетворяющих заданным требованиям.

При решении задачи интеграции систем моделирования, информационных и аналитических систем, для разработки которых использован модельно-ориентированный подход, актуальной является задача трансформации построенных моделей. Визуальные языки моделирования обладают преимуществами – модели, построенные с их помощью наглядны, легче воспринимаются. Средства трансформации моделей, построенных с использованием графических нотаций, реализуются чаще всего на основе графовых грамматик, методов перезаписи графов. Однако эти средства не обеспечивают в достаточной степени учёт семантики предметных областей в процессе преобразования моделей, выполнение трансформаций приводит к потере «точности». Усовершенствовать средства трансформации можно, реализовав доопределение построенных моделей. При решении этой задачи используются различные методы, в частности, задача доопределения моделей сводится к задаче устранения неполноты информации (применяются методы интерполяции [Епифанов, 2011]). Одним из наиболее перспективных подходов здесь является подход, основанный на использовании онтологий [Миков и др., 2008; Миков и др., 2009].

1. Разработка моделей в MetaLanguage

При разработке DSM-платформы, которая может стать основой для интеграции систем моделирования, информационных и аналитических систем, должны быть обеспечены следующие возможности:

- построения языков моделирования для широкого спектра предметных областей и решаемых задач;
- многоуровневого и мультязыкового моделирования;
- динамического изменения языков моделирования с поддержанием в согласованном состоянии метамodelей и моделей;
- экспорта и импорта построенных моделей;
- поиска и доопределения моделей.

Одно из ключевых требований – доступность

инструментария для различных категорий пользователей: профессиональных разработчиков (программистов, системных аналитиков и др.) и экспертов в предметных областях, конечных пользователей (бизнес-аналитиков и пр.).

Перечисленные требования в полном объёме в настоящее время не реализуются ни в одной из существующих DSM-платформ [Сухов, 2013]. Языковой инструментарий MetaLanguage ориентирован на решение поставленных задач.

На первом этапе разработки моделей в языковом инструментарии MetaLanguage необходимо *создать метамодель* (в данном случае метамодель – это предметно-ориентированный язык моделирования, DSL, который используется для создания моделей, ориентированных на решение конкретных задач). Построение метамодели выполняется с помощью редактора моделей. При создании DSL в первую очередь определяются базовые конструкции языка. Базовыми элементами, которые используются в MetaLanguage для создания метамodelей, являются сущность, отношение, ограничение. В процессе создания DSL определяются *сущности* метамодели, *отношения* между ними, задаются *ограничения*, налагаемые на сущности и отношения. После построения метамодели разработчик получает в распоряжение расширяемый, динамически настраиваемый визуальный язык моделирования (рисунок 1).

Используя полученный DSL, пользователи (аналитики, эксперты) могут *создавать модели*, содержащие объекты, описывающие конкретные сущности предметной области и связи между ними. При разработке моделей выполняется их *валидация* (проверка, удовлетворяет ли модель ограничениям, которые были заданы для сущностей и отношений при создании метамодели).

Разработанный DSL, в свою очередь, может использоваться в качестве метаязыка – на его основе, таким образом, может быть создана *иерархия языков*, позволяющих работать с моделями различных уровней абстракции, ориентированных на решение различных задач разными категориями пользователей [Замятин и др., 2013]. При внесении изменений в метамодель система автоматически внесёт все необходимые изменения в модели, созданные с помощью этой метамодели.

Средства *трансформации* позволяют в соответствии с заданными правилами трансформаций (вертикальных и горизонтальных), созданными в той же среде, преобразовать модели. Таким образом, разработанная модель может быть переведена на нужный язык и передана во внешние системы для решения соответствующих задач.

Пользователь может сохранить построенные (мета)модели в репозитории, сгенерировать на их основе документацию к системе.

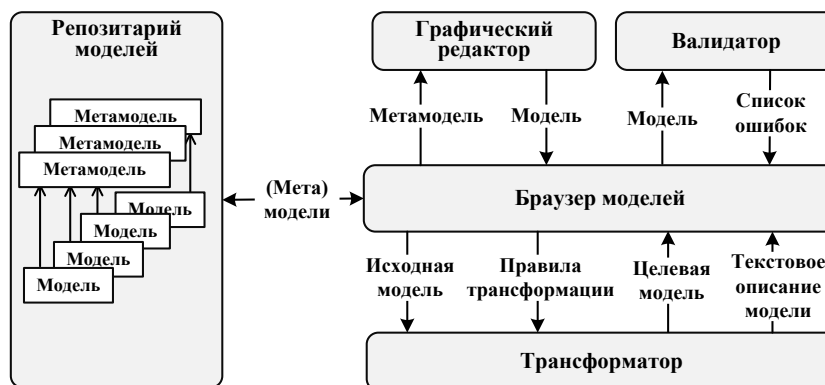


Рисунок 1 – Компоненты DSM-платформы MetaLanguage

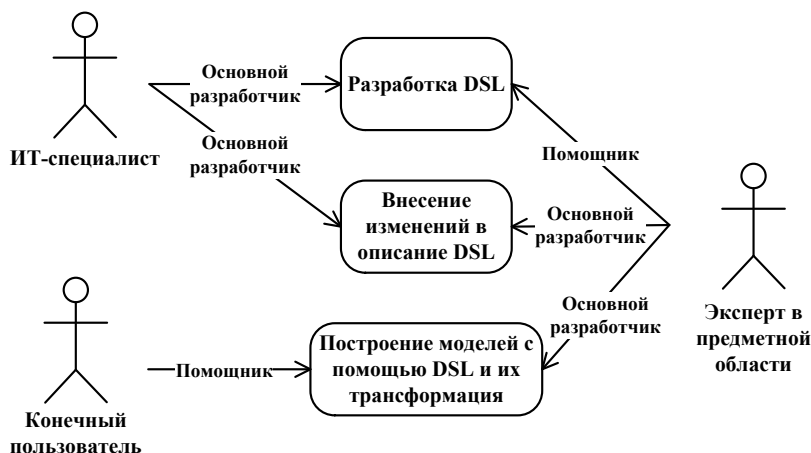


Рисунок 2 – Участие пользователей в разработке DSL и моделей

Специальные средства позволяют при необходимости выполнить поиск наиболее подходящих моделей в репозитории на основе созданных онтологий [Elokhov and others, 2013]. Поиск моделей осуществляется с использованием онтологий, описывающих предметные области.

Пользователи могут принимать активное участие как в разработке языков, так и в создании моделей на их основе (рисунок 2). Средства настройки языков, возможность динамического переопределения DSL с автоматическим поддержанием согласованности, без необходимости изменять код редакторов моделей, разрабатывать программы, делают DSM-платформу доступной для использования различными категориями специалистов.

2. DSM-платформа как основа средств интеграции

В разных системах, применяемых на предприятиях, используются различные модели, методы представления и обработки данных, ориентированные на решение соответствующих задач. При обмене данными, при передаче их из одной системы в другую выполняется преобразование данных в соответствии с заданными правилами средствами, обеспечивающими возможность интеграции систем. Методы и средства

решения этих задач проработаны и реализуются во всех промышленных информационных и аналитических системах. Однако более полная интеграция требует не только обмена данными, но и передачи созданных моделей из одной системы в другую. Модель, созданная с использованием какого-либо языка моделирования в одной системе, ориентированной на решение некоторых задач, может быть передана в другую систему для решения других задач. При разработке информационных систем различного назначения применяются разные типы моделей и языки моделирования. Для интеграции этих систем необходимо учитывать особенности используемых в них языков и моделей. При этом может потребоваться перевод модели с одного языка моделирования на другой, её трансформация. Оперативность решения задач пользователей зависит от того, смогут ли они самостоятельно настраивать используемые средства, создавать и модифицировать модели, задавать правила их преобразования.

Таким образом, DSM-платформа, обеспечивающая возможность создания новых языков моделирования и правил трансформаций созданных языков и моделей, обеспечивающая их согласованность, становится центральным звеном (рисунок 3), обеспечивающим интеграцию различных систем [Замятина и др., 2013].

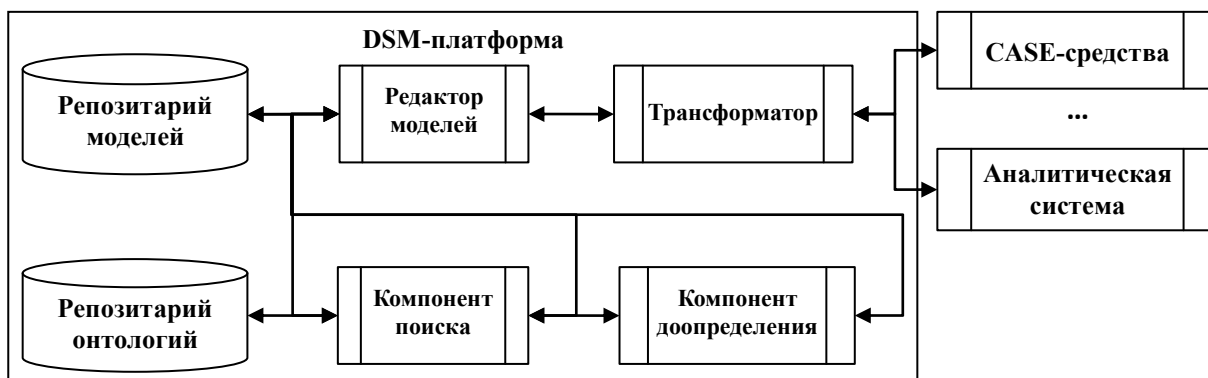


Рисунок 3 – DSM-платформа как основа средств интеграции

3. Использование онтологий для реализации DSM-платформы

В соответствии с наиболее общим определением онтология представляет описание сущностей предметной области, их свойств и отношений. Они создаются и используются во множестве областей знаний (поиск информации, автоматизация построения моделей и пр.). Онтологии применяются для решения различных задач моделирования (здесь моделирование рассматривается и как процесс построения моделей, и как процесс их исследования): от поиска информации о моделируемой системе до доопределения и валидации созданных моделей. В частности, онтологии часто рассматриваются как средство автоматизации построения моделей из заданных компонентов, как средство конфигурирования [Benjamin and others, 2006; Liang and others, 2003; Fishwick and others, 2004]. Онтологии служат для решения задачи рефакторинга систем, выполнения настройки моделей в соответствии с меняющимися требованиями.

Создание онтологий для решения задач моделирования является достаточно сложной задачей, поскольку модели используются для исследования самых разнообразных систем, относящихся к различным предметным областям. Кроме того, методы построения и анализа моделей основаны на применении математического аппарата, теории графов, вероятностных и статистических методов. Таким образом, онтологии этих областей знания должны служить основой для разработки онтологий для других областей [Миков и др., 2008; Миков и др., 2009], требуется разрабатывать многоуровневые онтологии предметных областей. При этом необходимо обеспечить преемственность, «переиспользовать» ранее созданные модели.

Предлагается снизить трудоёмкость разработки моделей: при решении многих задач можно найти ранее созданные предметно-ориентированные языки и модели для той же или близкой предметной области; доработка найденных моделей, их настройка на изменившиеся условия может оказаться менее трудоёмкой, чем создание DSL и

моделей «с нуля». Поиск моделей (метамodelей, DSL) предлагается выполнять с использованием онтологий соответствующих предметных областей. Однако и сам процесс построения онтологий является трудоёмким, требующим знаний не только в предметной области, но и навыков использования инструментальных средств, языков описания и редакторов онтологий. Помочь в создании онтологий и поиске моделей могут средства, базирующиеся на анализе документов, описывающих предметные области и решаемые задачи [Elokhov and others, 2013].

Найденные модели требуют доопределения. Решение этой задачи также основывается на создании онтологий. Реализуются средства полуавтоматического доопределения найденных DSL (метамodelей) или моделей (рисунок 4).

На основе документов, описывающих предметную область и решаемые задачи, получаемых из различных источников, формируются требования к целевой модели (или к DSL – метамodelи).

Эти требования формализуются в виде онтологии, которая создаётся в результате анализа собранных документов (корпуса) на основе ранее разработанных базовых онтологий: из документов, включённых в корпус, извлекаются ключевые понятия и связи между ними, устраняется синонимия и т.д.

Пользователь может внести изменения в построенную онтологию, доработать её, используя редактор системы: дополнить множество понятий, связей, ограничений и т.д.

Далее осуществляется поиск в репозитории модели (или метамodelи, DSL), наиболее подходящей для описанной предметной области и решаемой задачи. Найденная модель (или метамodelь) рассматривается как основа для построения целевой модели или DSL.

Выявляются все несоответствия найденной модели (метамodelи) формализованному с помощью онтологии описанию предметной области – требованиям к целевой модели (новому DSL).

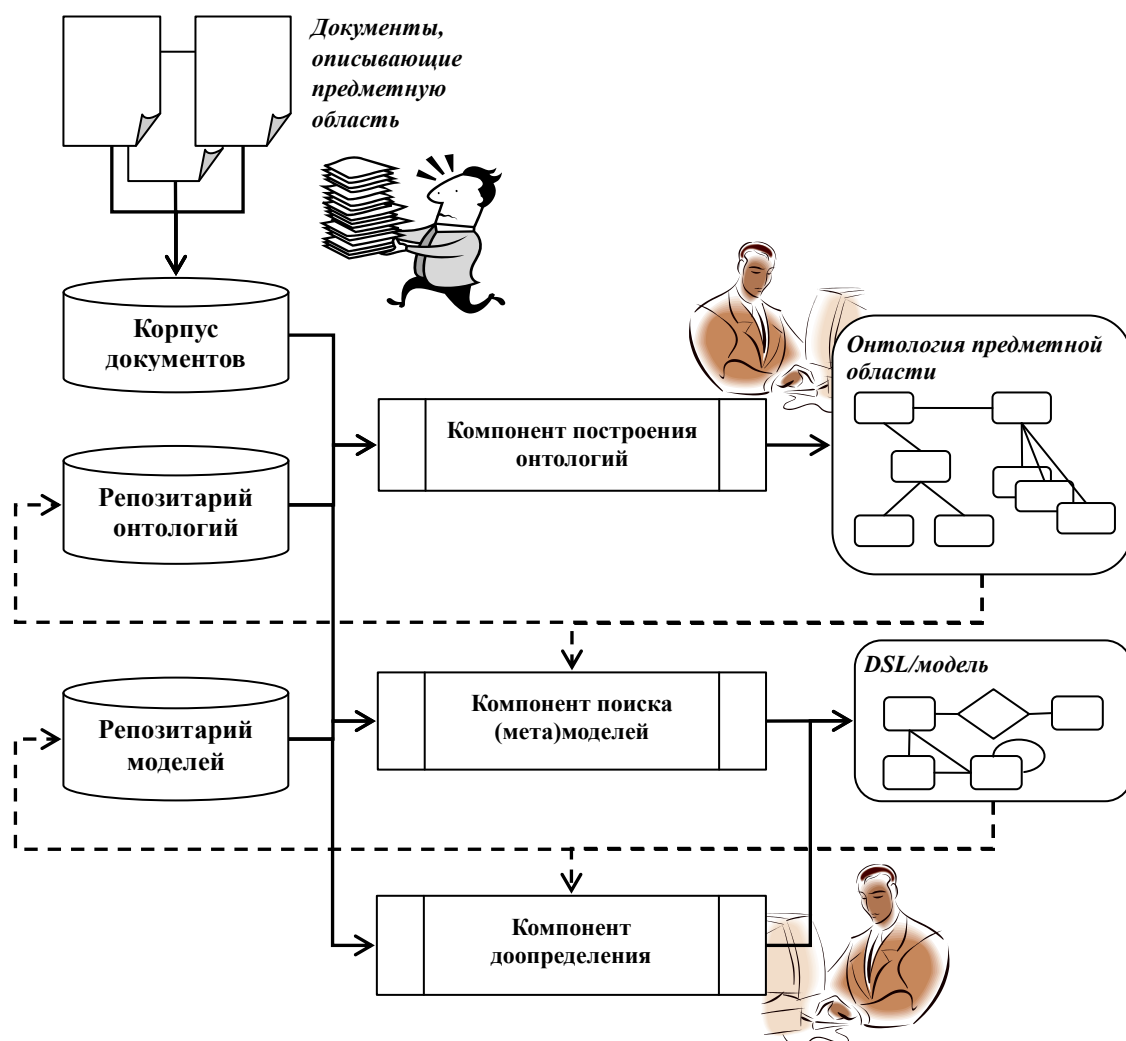


Рисунок 4 – Схема поиска и доопределения моделей с использованием онтологий

Если найденная модель (или метамодель) не удовлетворяет требованиям, то пользователь может доработать найденную модель (DSL) «вручную» или же использовать возможности компонента автоматического доопределения моделей с использованием репозитория моделей и онтологий.

Для доопределения модели (DSL) пользователь может повторить поиск в репозитории, исключив из исходного множества ранее выбранную и уже использованную модель (метамодель).

Результаты поиска могут соответствовать «фрагментам» построенной онтологии предметной области, т.е. сформированным для поиска требованиям, для которых на предыдущем шаге не было найдено соответствий. Таким образом, найденная новая модель (DSL) может быть использована для доопределения ранее выбранной из репозитория модели (метамодели).

Поиск и автоматическое доопределение моделей или DSL могут продолжаться, пока в построенной в результате поиска и доопределения модели (или в DSL) будут выявляться несоответствия, формализованные в виде онтологии требованиям, и

в репозитории будут находиться модели (метамодели), «покрывающие» их.

Пользователь может внести изменения в найденные модели (в DSL) вручную, доопределить построенную модель или предметно-ориентированный язык, используя редактор системы. Таким образом, если в репозитории были сохранены языки и модели, предназначенные для решения каких-либо задач в определённой предметной области, то с их помощью можно создать новую (мета)модель, для решения более сложной задачи, относящейся к той же предметной области, разбив новую задачу на «подзадачи», которые были решены ранее. Можно также расширить возможности DSL для новой предметной области, дополнив найденные языки вручную.

Заключение

В настоящее время разработаны исследовательские прототипы описанных средств. Языковой инструмент MetaLanguage прошёл апробацию, с его помощью были разработаны несколько предметно-ориентированных языков и

моделей, описаны трансформации, которые позволили экспортировать созданные модели во внешние системы, в частности сгенерировать модели на языке моделирования GPSS и провести имитационные эксперименты; построить базу данных для описанной предметной области, автоматически определив её схему на основе разработанной с использованием DSL модели и правил трансформации.

Работы выполнены при поддержке Научного фонда НИУ ВШЭ (проект № 13-09-0143) и РФФИ (проект № 14-07-31330-мол-а).

Библиографический список

- [Елифанов, 2011] Елифанов А.С. Доопределение частично заданных законов функционирования дискретных детерминированных систем / Проблемы управления. – 2011. – № 2. – С. 23-30.
- [Замятина и др., 2013] Замятина Е.Б., Лядова Л.Н., Сухов А.О. О подходе к интеграции систем моделирования и информационных систем на основе DSM-платформы MetaLanguage // В кн.: Технологии разработки информационных систем ТРИС-2013. Материалы IV Международной научно-технической конференции / Т. 1. Таганрог: Издательство Технологического института ЮФУ, 2013. С. 61-70.
- [Миков и др., 2008] Миков А.И., Замятина Е.Б., Кубрак Е.Н. Онтологический метод доопределения имитационной модели // Advanced Studies in Software and Knowledge Engineering: International Book Series, Supplement to International Journal "Information Technologies and Knowledge". N4, Vol.2. Sofia, 2008. Pp. 133-140.
- [Миков и др., 2009] Миков А.И., Замятина Е.Б., Кубрак Е.Н. Использование онтологий в распределённой системе имитационного моделирования Triad.Net // "Знания-Онтологии-Теории" (ЗОНТ-09): Материалы Всероссийской конференции с международным участием 22-24 октября 2009 г. / Новосибирск, 2009. С. 197-206.
- [Сухов, 2013] Сухов А.О. Инструментальные средства создания визуальных предметно-ориентированных языков моделирования / Фундаментальные исследования. – 2013. – № 4 (ч. 4). – С. 848-852.
- [Balasubramanian and others, 2007] Balasubramanian K. Component-Based System Integration via (Meta)Model Composition / K. Balasubramanian, D.C. Schmidt, Z. Molnar, A. Ledeczi // Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'07). Tucson, Arizona. March 26 29, 2007. P. 93-102.
- [Benjamin and others, 2006] Benjamin P., Patki M., Mayer R. Using Ontologies for Simulation Modeling // Proceedings of Winter Simulation Conference, p.1161-1167, 2006.
- [Bräuer and others, 2007] Bräuer M. Towards Semantic Integration of Multiple Domain-Specific Languages Using Ontological Foundations / M. Bräuer, H. Lochmann // Proceedings of the 4th International Workshop on (Software) Language Engineering (ATEM'07), Nashville, 2007. 15 pp.
- [Elokhov and others, 2013] Elokhov E. An Approach to the Selection of DSL Based on Corpus of Domain-Specific Documents / E. Elokhov, E. Uzunova, M. Valeev, A. Yugov, V. Lanin // Proc. of the 7th Spring/Summer Young Researchers' Colloquium on Software Engineering. M.: Изд-во Инст. сист. прогр. РАН, 2013. – P. 139-143.
- [Fishwick and others, 2004] Fishwick P.A., Miller J.A. Ontologies for Modeling and Simulation: Issues and Approaches. In: Proceedings of, Winter Simulation Conference, p. 259-264, 2004.
- [Hessellund, 2009] Hessellund A. Domain-specific Multimodeling: PhD thesis. IT-University of Copenhagen, Software Development Group, 2009. 190 p.
- [Liang and others, 2003] Liang V.-C., Paredis C.J. A Port Ontology for Automated Model Composition // Proceedings of Winter Simulation Conference, p. 613-622, 2003.

INTEGRATION OF MODELING SYSTEMS ON THE BASIS OF DSM-PLATFORM WITH ONTOLOGIES

Sukhov A.O. *, Lyadova L.N. *, Zamyatina E.B. **

* National Research University «Higher School of Economics» (Perm Branch),
Perm, Russia

LNLyadova@gmail.com
Sukhov.PSU@gmail.com

* Perm State National Researching University,
Perm, Russia

E_Zamyatina@mail.ru

An approach to integration of various modeling systems, based on use of the MetaLanguage DSM-platform facilities, is considered. The MetaLanguage tools allow to develop domain specific languages (DSL) and to describe their transformations. Domain specific modeling provides possibility to work in terms of subject domain for analysts at models creation. The transformation component is a basis for development of means of models export and import, for models integration. Ontologies formalize knowledge on domains and models. They are used by tools of DSL development automation, by means of model search and by means of model definition complete.

Introduction

The maximum flexibility of modeling tools is reached with use of the multilevel models describing modelled systems at the different points of view and with different detailing level. The tasks of DSL development and model transformations are important. Model search and model definition completing are actual too.

Main Part

The main part includes the description of MetaLanguage structure and using scheme. This DSM-platform is suggested as a central component of integration tools. These means allow realizing export and import of DSLs and models, integrating different information and analytical systems. Ontologies are the main elements of integration tools.

Conclusion

Now research prototypes of the described tools are developed. The MetaLanguage workbenches were approved, some domain specific languages were developed with this DSM-platform and some transformations are described.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8 + 004.9

АРХИТЕКТУРА СИСТЕМЫ ОБРАБОТКИ ПУБЛИКАЦИЙ ПО ТЕМАТИКЕ МОДЕЛИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

Ланин В.В.^{*}, Чугунов А.П.^{**}

^{} Национальный исследовательский университет «Высшая школа экономики»,
г. Пермь, Пермский край*

lanin@perm.ru

*^{**} Пермский национальный исследовательский политехнический университет,
г. Пермь, Пермский край*

chugunov@permedu.ru

В статье описывается реализация сервиса обработки научных публикаций. В рамках проекта планируется создание программных средств, предназначенных для решения задач поиска и обработки публикаций по тематике моделирования информационных систем на основе сочетания методов корпусной лингвистики и онтологического подхода.

Ключевые слова: интеллектуальная обработка документов, онтология.

Введение

Традиционные средства персонального, корпоративного и глобального поиска, имеющиеся в свободном доступе, предоставляют возможность поиска по вхождению строки в текст документа или его название или поиска на основе статистических методов. При этом не учитывается смысл текста. Однако, в большинстве случаев мы не знаем точных формулировок, а, зачастую, и слов, употребленных в целевом документе.

В настоящее время для решения задачи поиска применяются математико-статистические (латентный семантический поиск), графовые (набор документов представляется в виде ориентированного графа) и онтологические (поиск по готовым онтологиям) методы [Лапшин, 2010]. Каждый из перечисленных подходов имеет недостатки. Графовые методы неприменимы для поиска на локальном компьютере или в локальной сети из-за отсутствия явных ссылок между документами. Онтологические методы трудно применимы из-за отсутствия автоматических методов построения онтологий и значительных затрат на построение индексов и поддержание их актуального состояния. Латентный семантический поиск предполагает ведение поиска без учета семантики слов и выражений.

Несмотря на перечисленные недостатки,

интеграция латентного семантического поиска и поиска на графовых структурах дает хорошие результаты: большинство поисковых систем в сети Internet использует сочетание этих подходов [Гасанов, 2002]. Однако, как говорилось, графовые методы неприменимы ни в локальной сети, ни на локальном компьютере и оба подхода не учитывают семантику запроса и документов. Как следствие, задача поиска «по смыслу» остается не решенной, а новейшие алгоритмы поисковых систем в сети Internet, дающие хорошие результаты, остаются неприменимыми на локальном компьютере и в локальной сети.

При дополнении процесса поиска третьим – онтологическим – методом обеспечивается возможность решения проблемы построения ориентированного графа документов и проблемы учета семантики. Построение полных онтологий при этом не обязательно, что делает второй недостаток онтологического подхода не критичным.

Предлагаемый подход сочетает в себе латентный семантический метод поиска и использование онтологий. Это позволяет использовать данный подход в локальной сети и на локальном компьютере (в отличие от алгоритмов ранжирования, показывающих высокие результаты работы при поиске в сети Internet и [Гасанов, 2002]). Предложенный подход позволяет, таким образом, устранить основной недостаток других подходов, основывающихся исключительно на онтологиях:

избежать высоких затрат на построение самих онтологий документов.

Задача агрегации информации из разных источников и ее структуризации является чрезвычайно актуальной. Кроме того, требуют решения задачи устранения дублирования информации и поиска противоречий в результатах поиска. Слабоструктурированный характер информации и гетерогенность её источников предполагают применение средств и методов искусственного интеллекта для решения данной задачи (text mining, технологии Semantic Web и мультиагентные технологии).

1. Поиск информации

Модель, используемую для поиска документов, можно обобщённо представить в виде кортежа:

$$\langle D, Q, F, R(d, q) \rangle,$$

где D – множество представлений документа; Q – множество представлений информационной потребности (запроса); F – средства моделирования представлений документа, запросов и их отношений; $R(d, q)$ – функция ранжирования, которая ставит в соответствие d из D и q из Q вещественные числа, а также определяет порядок на множестве документов относительно запроса q .

Процесс поиска информации с помощью поисковой системы может быть описан следующим образом [Ланин, 2009b]. У пользователя возникает *информационная потребность* (необходимость найти сведения по какому-либо вопросу). Затем пользователь некоторым образом формализует свою информационную потребность в виде *запроса* (в традиционных системах это выделенное множество ключевых слов с зафиксированными отношениями между ними). На следующем этапе через интерфейс поисковой системы вводится запрос. Система на множестве документов, являющемся информационно-поисковым пространством, осуществляет *выборку документов*, которые по внесенным в систему критериям соответствуют запросу пользователя, и *формирует результат* (отклик). Найденные документы по своему содержанию делятся на две группы: документы, соответствующие информационной потребности пользователя (релевантные), и документы, не соответствующие его информационной потребности, но соответствующие запросу пользователя с точки зрения информационно-поисковой системы (информационный шум).

Учитывая специфику решаемой задачи, процесс поиска информации может быть улучшен по двум направлениям: релевантности результата и представлению отклика. Обе задачи предлагается решать с помощью онтологического подхода, завоевывающего все большую популярность.

Основная особенность предлагаемого подхода – использование репозитария онтологий на этапах преобразования запроса и документа. Откликом

является структурированный документ, т.е. документ, в котором выделены понятия онтологий. О структуре репозитария онтологий рассказано в следующем разделе.

2. Описание документа с помощью онтологии

Методы искусственного интеллекта, как правило, используются для решения трудно формализуемых задач, постановка которых проста и понятна для человека, но при разработке алгоритмов их решения возникают трудности. Одна из таких задач – работа с документами в информационных системах: их поиск и каталогизация, анализ и извлечение информации.

В настоящее время существуют различные подходы, модели и языки, ориентированные на интегрированное описание данных и знаний. Наиболее перспективным и универсальным, по мнению авторов, представляется онтологический подход.

Согласно общепринятому определению, под *онтологией* (в широком смысле) понимается база знаний специального типа, которая может «читаться» и пониматься, отчуждаться от разработчика и/или физически разделяться ее пользователями. Учитывая специфику решаемых в данной работе задач, можно конкретизировать понятие онтологии: онтология – это спецификация некоторой предметной области, которая включает в себя словарь терминов (понятий) предметной области и множество связей между ними, которые описывают, как эти термины соотносятся между собой [Ланин, 2009b].

Для построения иерархии понятий онтологии используются следующие базовые типы отношений: “*is_a*” («класс – подкласс», гипонимия); “*part_of*” («часть – целое», меронимия); “*synonym_of*” (синонимия). Следует учесть, что данные типы отношений являются базовыми и не зависят от онтологии, но необходимо предоставить пользователю возможность добавления новых отношений, которые учитывали бы специфику описываемой предметной области.

В представленном подходе выделяются три *типа онтологий*:

- онтология *предметной области* конкретной информационной системы (ИС);
- онтология как *база знаний* (БЗ) интеллектуального агента;
- онтология как *описание документа*.

Рассмотрим назначение каждого из перечисленных типов онтологий.

Онтологии предметной области имеют наиболее типичное применение, они используются для описания понятий предметной области ИС, например: моделирование систем и процессов,

школьное образование, социальная помощь гражданам или инновационное развитие регионов. В онтологии этого типа описывается связь понятий, языковые единицы для их выражения, аксиомы предметной области. Онтология предметной области используется для семантического индексирования и анализа всех документов системы.

Для анализа документов используется *мультиагентный* подход. Интеллектуальные агенты, руководствуясь онтологией как базой знаний (второй тип онтологий), производят поиск и анализ конкретных понятий документа. Каждая из вершин такой онтологии имеет определенный прототип, интерпретация которого известна агенту. Таким образом, агент использует онтологию как определенную программу своих действий. Вершинами онтологии данного типа могут являться понятия из онтологии предметной области.

Третий тип онтологий используется для описания структуры и содержания документов. Этот тип онтологий включает в себя *два класса* (две «плоскости») вершин. К первому классу относятся вершины, описывающие структуру документа, например: таблица, дата, должность и т.д. (они представляют собой общие понятия, не зависящие от конкретной предметной области). Другим типом будут являться вершины, содержащие понятия документа. Первый тип вершин будем называть *структурными вершинами*, второй тип – *семантическими*. Благодаря такому подходу из документа можно получить требуемые данные: известно, где искать данные и как они могут быть интерпретированы.

Если представлять документ с использованием онтологий, то задача сопоставления онтологии и анализируемого документа сводится к задаче *поиска понятий онтологии в документе*. Как следствие, системе необходимо ответить на вопрос: описывает ли данная онтология документ или нет. На последний вопрос можно ответить утвердительно, если в процессе сопоставления в документе были найдены все понятия, включенные в онтологию. Таким образом, исходная задача сводится к задаче *поиска в тексте документа общих понятий на основе формальных описаний*. На основе онтологии может быть получен *фрейм*, слоты которого заполняются в процессе анализа документа. В качестве слотов фрейма выступают понятия онтологии, а значения этих фреймов заполняются данными анализируемого документа. Таким образом из найденного неструктурированного документа может быть получен структурированный документ-фрейм.

Онтологии располагаются на *трех уровнях репозитария*. На первом уровне расположены онтологии, описывающие объекты, используемые в конкретной системе и учитывающие ее особенности. На втором уровне описываются объекты, инвариантные к предметной области. Объекты третьего уровня описывают наиболее

общие понятия и аксиомы, с помощью которых описываются объекты нижележащих уровней.

3. Архитектура сервиса

Общая архитектура разработанного сервиса представлена на рис. 1.

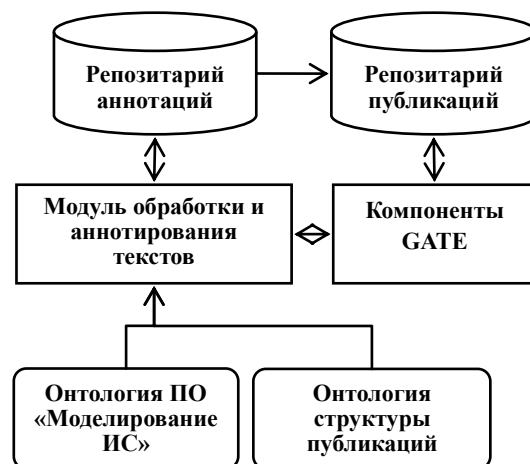


Рисунок 1 – Архитектура сервиса

При работе сервиса используются два онтологических ресурса: *онтология предметной области «Моделирование ИС»* и *онтология структуры публикаций*. Первая онтология содержит базовые понятия, относящиеся к различным аспектам моделирования информационных систем. Во втором онтологическом ресурсе представлены концепты, отражающие структурные элементы научной публикации и связанные с ними лексические маркеры.

Полученные в результате обработки документов аннотации хранятся в репозитории аннотаций, а сами ресурсы хранятся в репозитории публикаций. Модуль обработки и аннотирования текстов непосредственно выполняет обработку документов и записывает результаты в репозиторий аннотаций. Компоненты системы GATE [Cunningham, 2011] используются для выполнения базовых процедур по обработке текста.

4. Используемые программные и инструментальные средства

Программные компоненты разрабатываются в средах Microsoft Visual Studio 2012 и Eclipse на языках C# и Java соответственно. Онтологические ресурсы разрабатываются в редакторе онтологий Protégé 4.2.

Согласно опубликованному отчету агентства Gartner [Weintraub, 2011] по ECM системам в знаменитый магический квадрант попала единственная Open Source из всех представленных система Alfresco. Gartner отметили инновационный подход [Gilbert, 2012] – именно поэтому Alfresco располагается в правом нижнем углу. Также Alfresco Software попала и в отчет агентства Forrester.

Немаловажным факторами при выборе системы хранения документов стали открытость исходного кода и использование открытых стандартов. Кроме того, система Alfresco реализована на языке Java, что значительно упрощает интеграцию с компонентами обработки онтологий и инструментами Semantic Web, также реализованными на данном языке.

Заключение

Применение описанных подходов должно существенно снизить трудоёмкость поиска и анализа информации, обеспечить оперативность её использования в исследованиях, расширить доступный для обработки, изучения объем информации, извлекаемой из различных источников. Полученная в результате анализа документов информация, в свою очередь, может использоваться исследователями для усовершенствования моделей предметных областей, построенных ими. Таким образом, появляется основа для создания интеллектуальной системы поддержки научных исследований с высокой степенью обратной связи.

Ориентация на знания является базовым механизмом функционирования системы, что позволяет комплексно решать поставленные задачи.

Работа выполнена при поддержке Научного фонда НИУ ВШЭ по программе финансирования грантов РФФИ (проект № 13-09-0143).

Библиографический список

- [Segaran, 2009] Segaran T., Evans C., Taylor J. Programming the Semantic Web. – O'Reilly Media, 2009.
- [Гасанов, 2002] Гасанов Э.Э. Теория Хранения и поиска информации / Э.Э. Гасанов, В.Б. Кудрявцев // М.: Физматлит, 2002.
- [Лапшин, 2010] Лапшин В.А. Онтологии в компьютерных системах / В.А. Лапшин // СПб: Научный мир, 2010.
- [Никоненко, 2009] Никоненко А.А. Обзор знаний онтологического типа / А.А. Никоненко // Научно-теоретический журнал «Искусственный интеллект». – 2009. – №4. – С.208-219.
- [Ланин, 2009а] Ланин В.В. Методы и средства решения задач информационного поиска для системы поддержки научных исследований // Инновационное развитие регионов: методы оценки и поддержка исследований: межвуз. сб. науч. статей / Перм. гос. ун-т. – Пермь, 2009. С. 80-88.
- [Ланин, 2009б] Ланин В.В. Решение задач информационного поиска для исследовательского портала на основе агентного и онтологического подходов // Инновационное развитие регионов: методы оценки и поддержка исследований: межвуз. сб. науч. статей / Перм. гос. ун-т. – Пермь, 2009. С. 89-96.
- [Cunningham, 2011] Cunningham H., Maynard D., Bontcheva K. Text Processing with GATE. – Gateway Press CA, 2011.
- [Potts, 2008] Potts J. Alfresco Developer Book. Customizing Alfresco with actions, web scripts, web forms, workflows, and more. – Packt publishing, 2008.
- [Gilbert, 2012] Gilbert M. R., Shegda K. M., Chin K., Tay G., Koehler-Kruener H. Gartner's Magic Quadrant for Enterprise Content Management. 18 October 2012.
- [Weintraub, 2011] Weintraub A. The Forrester Wave™: Enterprise Content Management, Q4 2011 Alan Weintraub November 1, 2011.

AN ARCHITECTURE OF THE SCIENTIFIC PUBLICATIONS PROCESSING SYSTEM

Lanin V.V.^{*}, Chugunov A.P.^{**}

^{*} National Research University «Higher School of Economics» - Perm

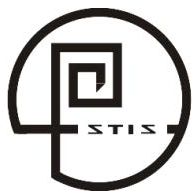
lanin@perm.ru

^{**} Perm National Research Polytechnic University, Perm, Russia

chugunov@permedu.ru

The purpose of the project is the creation of "self-developing" resource, which provides intelligent search and automatic processing of the results (documents and sources), easy navigation on the found resources. Implementation is based on the ontologies approach.

The main feature of suggested methods is an integrated approach to development. The approach bases on a multi-level ontology repository. The portal allows searching and analyzing information, creating and researching model, publishing research results. Software gives an opportunity of a flexible customizing. The main topic of this paper is an intelligent information search means based on semantic indexation, automatic document classification, tracking of semantic links between documents and automatic summarization.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ОНТОЛОГИЧЕСКИ-ОРИЕНТИРОВАННАЯ МОДЕЛЬ КЛАССИФИКАЦИЙ ТЕКСТОВЫХ ДОКУМЕНТОВ

Наместников А.М., Субхангулов Р.А.*

** Ульяновский государственный технический университет,
г. Ульяновск, Россия*

nam@ulstu.ru,

subkhangulov-ruslan@yandex.ru

В статье рассматривается специфика информационных потребностей специалистов занятых в процессе проектирования технических средств. Представлено описание пользовательских профилей, их применение в процессе полнотекстового поиска текстовых документов. Рассматривается применение байесовского классификатора в терминах онтологических моделей информационного поиска. Приведены результаты экспериментальных исследований, доказывающие эффективность разработанной программной системы.

Ключевые слова: интеллектуальная система; онтология; полнотекстовый поиск, классификация.

Введение

Внедрение систем автоматизированного проектирования (САПР) CAD/CAM/CAE в современное производство - это пройденный этап автоматизации, который повысил качество и скорость разработки продукции. На современном этапе автоматизации проектирования стоят новые задачи к ним относятся использование средств PLM (Product life Cycle) поддержка жизненного цикла изделия, формирования систем электронного архива проектно-технической документации.

Основными задачами электронного архива является обеспечение коллективной работы проектно-конструкторских отделов над проектом, добавление, хранение и поиск данных в электронном архиве. Поиск знаний в таких архивах выполняется с использованием классических моделей поиска, к ним относятся поиск по реквизитам документов, поиск по ключевым словам. Для профессионального, в том числе научно-технического поиска информации требуется обеспечение поиска, основанного на знаниях, – использование синонимов, возможности автоматического расширения запроса, возможностей автоматического анализа результатов запроса и помощь в интерактивном поиске. Для решения подобного рода проблем применяются интеллектуальные модели поиска, функционирование которых основано на предметно-ориентированных знаниях. Эти знания могут быть представлены в виде онтологии предметной области

[Добров и др., 2006]. В настоящее время наблюдается рост интереса к разработке новых методов интеллектуального анализа данных, основанных на применении онтологии [Гаврилова и др., 2000].

В данной статье предлагается использовать пользовательские профили, в основе которых лежит онтология предметной области. Профили содержат информационные предпочтения специалистов-проектировщиков. Они играют вспомогательную роль в задачах интерактивного поиска слабоструктурированных данных, что позволяет максимально удовлетворить информационную потребность пользователя электронного архива проектно-технических документов.

1.1. Информационная неопределенность проектировщика

Прежде чем рассмотреть пользовательские профили проектировщика и использование их в поддержке информационного поиска рассмотрим такой важный аспект, как информационную потребность специалиста-проектировщика, выделим специфику, которым обладает данная характеристика.

Дадим определение информационной потребности: «Информационная потребность – это информационная неопределенность, которую пользователь хочет уменьшить посредством получения информации из системы информационного поиска».

Для понимания специфики информационной потребности проектировщика необходимо детально рассмотреть процесс проектирования. Используем определение, которое дано Норенковым И.П. в работе [Норенков, 2009]: «Проектирование технического объекта – это создание, преобразование и представление в принятой форме образа этого еще не существующего объекта». В нашем случае проектирование необходимо рассматривать как информационный процесс, в котором выполняется преобразование входной информации о проектируемом объекте в выходную информацию в виде проектных документов.

Проектная деятельность имеет ряд специфических особенностей [Титов, 2009]:

1. Продуктом проектной деятельности является упорядоченная совокупность сведений, служащих знаковой моделью объекта, в момент проектирования реально еще не существующего.

2. Процедуры проектирования реального объекта соответствуют преобразованию его исходного описания в некотором конечном пространстве.

3. Способы преобразования информации при проектировании нельзя отразить в виде математических соотношений, т. е. невозможно построить строгую математическую модель такого процесса преобразования.

4. Ввиду сложности проектируемых объектов на каждом этапе разработки вовлекаются различные специалисты, что придает проектированию характер коллективной деятельности.

5. Проектируемый объект входит в упорядоченную иерархию объектов и, с одной стороны, выступает как часть системы более высокого уровня, а с другой — как система для объектов более низкого уровня. В соответствии с этим процесс проектирования можно разделить на два этапа: внешнего (объект — часть системы более высокого ранга) и внутреннего проектирования (объект — совокупность компонентов).

6. Проектирование, как правило, имеет итерационный многовариантный характер, для принятия проектных решений используются различные научно-технические знания.

1.2. Профили проектировщика

В традиционных поисковых системах поиск данных выполняется следующим образом: пользователь имеет информационную потребность, которая затем преобразуется в виде набора терминов в поисковый запрос к подсистеме информационного поиска. Подсистема информационного поиска, используя традиционные модели и алгоритмы, находит документы, которые должны уменьшить информационную потребность пользователя. Однако данная подсистема не имеет полного представления об информационных потребностях пользователя и тем самым всегда присутствует вероятность того, что документы,

которые были отобраны, не уменьшат информационную неопределенность.

Для решения задачи максимального удовлетворения информационной потребности специалистов-проектировщиков в работе [Филиппов и др., 2013] предлагается использовать модели кластеризации и информационного поиска, в основе которых лежит предметно-ориентированная онтология. В данной работе не будем рассматривать онтологические методы поиска и кластеризации. Отметим лишь, что формально предметная онтология имеет следующий вид [Наместников, 2009, Филиппов и др., 2013]:

$$\Theta = \langle r, S, C, W, R \rangle,$$

где r – корневая вершина онтологии, соответствующая классу проектных документов; S – множество структур документов, $C = \{c_1, c_2, \dots, c_k\}$ – множество понятий предметной области электронного архива, $W = \{w_1, w_2, \dots, w_l\}$ – множество терминов предметной области электронного архива, R – множество отношений онтологии.

В онтологии каждый термин $w_i \in W$ связан с понятием $c_j \in C$ отношением ассоциации $R_A^a: w_i R_A^a c_j$. Понятия C связаны друг с другом различными видами отношений (отношение обобщения, отношение включения и т.д.), образуя таксономию понятий предметной области. Понятийный уровень онтологии представим в виде ориентированного графа:

$$G = (C, E), \quad (1)$$

где C – множество вершин графа, каждая вершина – это понятие онтологии; E – множество дуг вида

$$E = \{ \langle c_i, c_k \rangle \}$$

для всех $c_i, c_k \in C$, для которых имеет место отношение $c_i R_G c_k$.

Онтологическое представление oV_j^d является вершинным подграфом графа G , определяемого выражением (1). Поскольку, в общем случае, в онтологическом представлении могут отсутствовать любые понятия из состава понятий предметной онтологии, результирующий граф онтологического представления oV_j^d может состоять из несвязанных деревьев и/или изолированных понятий.

В результате получаем, что проектно-технический документ в электронном архиве представляется не в лексическом пространстве терминов, которые удастся выделить в документе, а в пространстве понятий предметной области, которые зафиксированы в онтологии электронного архива.

В данной работе предлагается дальнейшее развитие темы использования предметной онтологии в задачах информационного поиска. В частности использования пользовательских профилей, в основе которых лежит онтология предметной области электронного архива. Пользовательский профиль объединяет различных пользователей по следующим признакам:

- Пользователи одного проектно-конструкторского отдела;
- По должностям;
- Пользователи, которые работают над одним проектом.

Такие профили формируют своеобразный «портрет» группы пользователей, объединенные по определенному признаку. Структура профиля состоит из двух категорий, формально представляется в следующем виде:

$$Pr = \{Tr_+, Tr_-\}$$

где Tr_+ – множество, содержащее информацию о документах, которые принадлежат пользовательскому профилю, Tr_- – множество, содержащее информацию о документах, которые не принадлежат профилю.

Множества представляются следующим образом:

$$Tr_i = \{c_{ij}, d_{il}\}$$

где $i \in \{+, -\}$, c_{ij} – j -ый доминирующий концепт из предметной онтологии в i -ой категории, d_{il} – технический документ из электронного архива.

Рассмотрим детально, каким образом формируются пользовательские предпочтения. В работе [Филиппов и др., 2013] выдвигается гипотеза о том, что любой текстовый документ можно разделить на множество непересекающихся фрагментов, в каждом из которых будет доминировать тот или иной концепт предметной области. Для нахождения значения доминирования концептов применяется метод сравнения текстового входа каждого понятия в онтологии предметной области с анализируемым текстом.

Алгоритм вычисления степени доминирования понятия в текстовом фрагменте состоит из следующих шагов [Филиппов и др., 2013]:

Шаг 1. Определение максимальной степени выраженности концептов в текстовом фрагменте:

$$\hat{\mu}_{S_p^d}(c) = \max_c(\mu_{S_p^d}(c)).$$

Шаг 2. Определение среднего значения степени выраженности концептов онтологии, исключая концепт с максимальной степенью выраженности (определенный на предыдущем шаге):

$$\tilde{\mu}_{S_p^d}(c) = \frac{1}{n-1} \sum_{i=1}^{n-1} \mu_{S_p^d}(c_i),$$

где $c_i \in c - c_k$, $c_k = \arg \max_c(\mu_{S_p^d}(c))$, n – количество концептов с ненулевой степенью выраженности для текстового фрагмента S_p^d .

Шаг 3. Определение степени детерминированности понятия в текстовом фрагменте S_p^d :

$$\Delta_{S_p^d}(c) = \hat{\mu}_{S_p^d}(c) - \tilde{\mu}_{S_p^d}(c), \quad (2)$$

Выражение (2) фактически определяет качество выделения текстового фрагмента в проектно-технических документах с целью ограничения в тексте определенного понятия предметной области, которое зафиксировано в онтологии электронного архива.

Опираясь на эту гипотезу, а также воспользуемся алгоритмом «Вычисления степени доминирования концепта в текстовом фрагменте», разработаем метод, который добавляет доминирующие концепты в категории пользовательского профиля. Рассмотрим данный алгоритм более подробно по этапам:

Шаг 1. При входе пользователя в поисковую систему активируется профиль, к которому принадлежит пользователь.

Шаг 2. Поисковая система, в ответ на пользовательский запрос, выдает список релевантных документов. В основе информационного поиска лежит онтологически-ориентированная модель [Филиппов и др., 2013].

Шаг 3. Специалист-проектировщик, просматривая отобранный поисковой системой документ, отмечает, удовлетворяет или не удовлетворяет документ его информационным потребностям.

Шаг 4. Подсистема находит доминирующие концепты в отмеченном документе. Алгоритм вычисления степени доминирования понятия в документе рассмотрен выше.

Шаг 5. Если документ отмечен положительно, то есть удовлетворяет потребностям проектировщика, тогда этот документ рассматривается как документ, который принадлежит пользовательскому профилю и, следовательно, доминирующие концепты и наименование документа добавляются подсистемой информационного поиска в категорию Tr_+ . Если документ отмечен отрицательно, то есть не удовлетворяет потребностям пользователя, тогда данный документ рассматривается как документ, который не принадлежит пользовательскому профилю и, следовательно, доминирующие концепты и наименование документа добавляются

подсистемой информационного поиска в категорию $Tr_{\text{пользовательского профиля}}$.

Таким образом, происходит формирование обучающей выборки пользовательского профиля, который в дальнейшем будет использоваться в задачах классификации и информационного поиска.

1.3. Модель классификации документов

Рассмотренный профиль пользователя содержит профессиональную информацию о пользовательских предпочтениях. Таким образом, подсистема информационного поиска имеет представление об информационных потребностях пользователя. Тем самым предполагается, что снизится неопределенность о том, какую информацию хочет получить пользователь. Это позволит улучшить качество информационного поиска в условиях неопределенности.

Будем применять пользовательские профили в рамках решения задач классификации в условиях неопределенности с использованием математического аппарата теории вероятностей.

Применительно к данной работе постановка задачи классификации документов выглядит следующим образом. Пусть даны: множество проектно-технических документов электронного архива $d \in \{d_1, d_2, \dots, d_n\}$ и множество пользовательских профилей $pr \in \{pr_1, pr_2, \dots, pr_m\}$. Множество документов имеет большую размерность, чем множество профилей. Количество профилей задается экспертом, по каким критериям могут формироваться пользовательские профили, рассмотрены выше в данной статье. Кроме того, имеется обучающее множество:

$$< d, pr > \in D \times Pr$$

Информация в профиле является динамической, то есть изменяется в процессе работы пользователей с подсистемой информационного поиска. Следовательно, результаты классификации проектных документов также будут изменяться в процессе работы всех пользователей с подсистемой информационного поиска, которые состоят в одном профиле. Используя метод обучения, получим функцию классификации, которая отображает документы в пользовательские профили:

$$\gamma : D \rightarrow Pr$$

В качестве метода обучения будет использоваться наивный метод Байеса, который имеет следующий вид:

$$P(pr|d) = P(pr) \prod_{1 \leq i \leq n} P(c_i | pr)$$

где $P(c_i | pr)$ – условная вероятность того, что концепт c_i из профиля будет доминирующим в

документе, $P(pr)$ – априорная вероятность того, что документ принадлежит профилю.

Так как необходимо рассчитать максимальную апостериорную вероятность величины $P(pr|d)$, то получим следующее выражение:

$$c_{\text{map}} = \arg \max_{pr \in Pr} (P(pr) \prod_{1 \leq i \leq n} P(c_i | pr))$$

Величины $P(d)$ и $P(d | pr)$ вычисляются следующими выражениями:

$$P(d) = \frac{N_{Tr}}{N}$$

где N_{Tr} – количество документов в категории профиля, N – общее количество документов в пользовательском профиле pr .

Как было отмечено ранее, в данной работе документы имеют концептуальное представление, в виде множеств концептов и степени выраженности концептов в документе. Следовательно, условную вероятность будем вычислять как относительную частоту концепта в документах обучающей выборки, принадлежащих профилю pr .

$$P(d | pr) = \frac{k_i}{\sum_{i \in V} k_i}$$

где k_i – количество появлений концепта c в категории профиля, V – список всех уникальных концептов в профиле.

Может оказаться так, что в обучающей выборке отсутствует концепт, который является доминирующим в анализируемом документе. Для того чтобы, решить данную проблему воспользуемся сглаживанием Лапласа. К каждой частоте добавляется единица. Таким образом, окончательно формула для вычисления условной вероятности будет выглядеть следующим образом:

$$P(d | pr) = \frac{k_i + 1}{\sum_{i \in V} k_i + B}, \quad (2)$$

где k_i – количество появлений концепта c в категории профиля, $B = |V|$ – список всех уникальных концептов в профиле.

Рассмотрим алгоритм поисковой системы, основанный на пользовательских профилях по этапам:

Шаг 1. При входе пользователя в поисковую систему активируется профиль, к которому принадлежит пользователь.

Шаг 2. Классификационная модель, обладая информацией о предпочтениях пользователя, в реальном времени формирует дополнительный список релевантных документов, которые возможно

удовлетворяют информационную потребность пользователя, следующим образом:

Шаг 2.1. На вход классификационной модели поступает концептуальные представления документов электронного архива.

Шаг 2.2. Для каждого документа с помощью выражения (2) и категории Tr_+ активного профиля вычисляется вероятность принадлежности документа к профилю, затем с помощью выражения (2) и Tr_- вычисляется вероятность того, что документ не принадлежит профилю.

Шаг 2.3. Полученные на этапе (2.2) вероятности сравниваются между собой, если значение вероятности принадлежности к профилю оказывается выше, то анализируемый документ отбирается в дополнительный список.

Шаг 3. Сформированный на втором этапе список дополнительных документов выводятся пользователю.

2. Результаты экспериментов

В ходе работы разработан прототип подсистемы информационного поиска, в котором были реализованы пользовательские профили, классификационная модель текстовых документов, в основе которой лежат онтология предметной области и профили пользователей.

В процессе работы были проведены эксперименты с разработанной подсистемой. Для оценки качества информационного поиска использовались следующие характеристики [Маннинг и др., 2011]:

Полнота (P) – доля релевантных документов в выборке, по отношению ко всем релевантным документам коллекции.

$$P = \frac{\text{кол-во релевантн\textsubscript{ых} найденных документов}}{\text{кол-во релевантн\textsubscript{ых} документов}}$$

Точность (T) – доля релевантных документов выборки, по отношению ко всем документам выборки.

$$T = \frac{\text{кол-во релевантн\textsubscript{ых} найденных документов}}{\text{кол-во найденных документов}}$$

Для удовлетворения баланса между двумя этими параметрами использовался параметр F-мера (F measure) [Маннинг и др., 2011]. Данный параметр представляет собой среднее гармоническое взвешенное:

$$F = \frac{1}{\alpha \frac{1}{P} + (1-\alpha) \frac{1}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

где $\beta^2 = \frac{1-\alpha}{\alpha}$, $\alpha \in [0,1]$, т.е. $\beta^2 \in [0,\infty]$. По

умолчанию сбалансированная F-мера присваивает точности и полноте одинаковые веса, т.е. $\alpha = 1/2$, или $\beta = 1$. Если $\beta < 1$ предпочтение отдают точности поиска, при $\beta > 1$ полноте поиска. При $\beta = 1$ формула принимает вид:

$$F_{\beta} = \frac{2PR}{P+R}$$

В данной работе использовалась сбалансированная F-мера.

Результаты работы разработанной подсистемы информационного поиска сравнивались со следующими поисковыми системами:

- 1) Яндекс. Персональный поиск (ЯПП).
- 2) Архивариус 3000 (A300).
- 3) AOL Desktop Search (AOL).
- 4) Copernic Desktop Search (CDS).

Для анализа результатов работы подсистемы информационного поиска использовалась выборка, состоящая из 100 текстовых документов, 20 из которых являются техническими заданиями, а 80 – статьями из области построения информационных систем. Результаты экспериментов представлены в таблице 1.

Таблица 1 – Результаты экспериментов с поисковыми системами (для первых 10 документов).

Оценка	Новая модель	ЯПП	A3000	AOL	CDS
Запрос «Microsoft Solution Framework»					
T	0.4	0.4	0.4	0.4	0.4
P	1	1	1	1	1
F-мера	0.6	0.6	0.6	0.6	0.6
Запрос «Rational Unified Process»					
T	0.4	0.4	0.4	0.4	0.4
P	1	1	1	1	1
F-мера	0.6	0.6	0.6	0.6	0.6
Запрос «хранимые процедуры»					
T	0.5	0.4	0.4	0.6	0.6
P	0.5	0.3	0.3	0.4	0.4
F-мера	0.5	0.3	0.3	0.5	0.5
Запрос «вариантов использования»					
T	0.5	0.3	0.2	0.2	0.2
P	0.7	0.4	0.3	0.3	0.3
F-мера	0.6	0.3	0.2	0.2	0.2

Результаты эксперимента проранжированы в порядке семантической неопределенности запроса. Семантическое значение запросов, состоящие из терминов «Microsoft Solution Framework» и «Rational Unified Process», явно указывают на конкретную предметную область, и как видно из результатов эксперимента все поисковые системы хорошо справились с поставленной задачей. Запрос, состоящий из терминов «вариантов использования», несет более слабое семантическое значение, для того чтобы понять, что он указывает на предметную область «Унифицированный язык моделирования (UML)» и, как видно из результата эксперимента, наиболее качественный результат показывает разработанная поисковая модель. Следовательно, можно сделать вывод, что поисковые и классификационные модели, обладающие информацией о пользовательских потребностях показывают более качественный результат в условиях неопределенности, чем традиционные модели.

Заключение

В данной работе рассматривается специфика информационной потребности специалистов занятых в процессе проектирования технических средств. Предлагается использовать информационную потребность пользователей в процессе информационного поиска в условиях неопределенности. Рассматривается новая модель информационного поиска, основанная на использовании пользовательских профилей, которые содержат информационную потребность и предметно-ориентированной онтологии. Приведенная в статье модель легла в основу разработанной программной системы информационного поиска проектных документов, с которой проведены вычислительные эксперименты.

Библиографический список

- [Добров и др., 2006] Добров Б.В., Лукашевич Н.В., Лингвистическая онтология по естественным наукам и технологиям: основные принципы разработки и текущее состояние // Десятая национальная конференция по искусственному интеллекту с международным участием (Обнинск, 25-28 сентября 2006 г.) – М.: Физматлит, 2006.
- [Гаврилова и др., 2000] Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. –СПб.: Питер, 2000. – 384 с.
- [Маннинг и др., 2011] Маннинг К., Рагхаван П., Шютце Х. Введение в информационный поиск. М: Вильямс, 2011
- [Наместников 2009] Интеллектуальные проектные репозитории. – Ульяновск: УлГТУ, 2009. Наместников А.М. Интеллектуальные проектные репозитории. – Ульяновск: УлГТУ, 2009.
- [Наместников и др., 2010] Наместников А.М., Филиппов А.А. Концептуальная индексация проектных документов // Автоматизация процессов управления. – 2010. – №2(20). – С. 34-39.
- [Норенков, 2009] Норенков И.П. Основы автоматизированного проектирования. М: МГТУ имени Баумана, 2009.
- [Титов, 2009] Титов Ю. А. САПР технологических процессов. Ульяновск, 2009.
- [Филиппов др, 2013] Филиппов А.А., Наместников А.М., Субхангулов Р.А. Применение нечетких моделей в задачах

кластеризации и информационного поиска текстовых проектных документов // Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов VII-й Международной научно-практической конференции (Коломна, 22-22 мая 2013 г.) В 3-х томах. Т.3. – М.:Физматлит, 2013. С. 1278-1289.

ONTOLOGICALLY-ORIENTED MODEL OF CLASSIFICATIONS OF TEXT DOCUMENTS

Namestnikov A.M. *, Subkhangulov R.A. *

** Ulyanovsk State Technical University,
Ulyanovsk, Russia*

nam@ulstu.ru

subkhangulov-ruslan@yandex.ru

In article the description of specifics of information needs of experts occupied in design process the technical means. We considered the user profiles, their application in the course of full-text query search in text documents. We considered to use of the bayesian qualifier in terms of ontological models of information search. The results of the experiments proving efficiency of developed program system are given.

Introduction

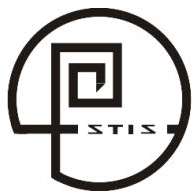
In CAD archive the designer has no possibility to solve semi-structured problems of search. We suggest to use the user profiles and to apply them in the course of information search. Such profiles use ontology of data domain and contain information on the user preferences.

Main Part

In article it is considered features information need of a designer. During operation we developed ontology of data domain. The domain ontology includes two levels: conceptual and terminological. Created a row of the user profiles for designers, using ontology of data domain. Developed the information retrieval system. This system uses the user profiles and classification model on a basis Bayes's rule in the course of information search. Were carried out a row of experiments.

Conclusion

Experiments showed that in the conditions of uncertainty our system shows good results. Results of researches showed that method of Ontologically-oriented model of classifications of text documents is more effective than a tradition method.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 33:518/519

СОСТОЯНИЕ, ТЕНДЕНЦИИ И КОНЦЕПЦИЯ РАЗВИТИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ТЕХНОЛОГИЙ В ЗАЩИТЕ ИНФОРМАЦИИ

Вишняков В.А.

*Минский институт управления,
г. Минск, Республика Беларусь*

vish2002@list.ru

В докладе представлены две проблемы использования интеллектуальных технологий в защите информации (ИТвЗИ) – создание специализированных БЗ с моделированием угроз и повышение уровня безопасности в корпоративных сетях и облачных вычислениях. Дан анализ двух направлений из второй проблемы ИТвЗИ: интеллектуальные поддержки принятия решений и использование многоагентных систем. В качестве тенденций развития рассмотрены совершенствования методов, моделей, архитектур, аппаратно-программных решений ИТвЗИ в КИС. В качестве концепции предложено развития ИТвЗИ для облачной инструментальной платформы проектирования интеллектуальных систем на основе семантических технологий.

Ключевые слова: интеллектуальные технологии, защита информации, многоагентные системы, защита в облачной инструментальной платформе

Введение

Для современного этапа развития теории и практики обеспечения защиты информации (ЗИ) характерна такая ситуация: с одной стороны, усиленное внимание к безопасности информационных объектов, повышение требований по ЗИ, принятие международных стандартов в области информационной безопасности (ИБ), растущие расходы на обеспечение защиты, с другой – возрастающий ущерб, причиняемый владельцам информационных ресурсов, о чем свидетельствуют публикуемые данные об ущербе мировой экономике от компьютерных атак [Машкина И.В., 2008].

Выходом является внедрение на всех этапах защиты интеллектуальных технологий, приобретающих все большее распространение в системах ЗИ. С одной стороны, сбор и обработка информации из Интернета о состоянии, направлении развития и уровне угроз тех или иных процессов в мировом сообществе и синтез знаний, отраженных в тех или иных источниках, осуществленный на основе их интеллектуальной обработки, дает новое интегративное качество, позволяющее спрогнозировать, смоделировать и предупредить развитие тех или иных угроз безопасности. С другой стороны, применение интеллектуальных технологий обработки данных дает возможность повысить уровень безопасности

различных корпоративных информационных систем (КИС) [Электр. ресурс, 2013] .

1. Направления интеллектуализации в защите информации

Основные задачи, которые должны решать интеллектуальные системы ЗИ (ИСЗИ):

- обеспечение обнаружения неизвестных вторжений;
- обеспечение автоматической поддержки принятия решения о перераспределении ресурсов СЗИ КИС;
- обеспечение возможности автоматического изменения своих свойств и параметров в зависимости от изменения условий среды функционирования;
- обеспечение дезинформации нападающей стороны об истинных свойствах и параметрах КИС.

ИСЗИ, обеспечивающие обнаружения атак, в качестве интеллектуального инструмента используют нейронные сети (НС), системы нечеткой логики и основанные на правилах экспертные системы (ЭС).

В ИСЗИ ЭС в базе знаний содержат описание классификационных правил, соответствующим профилям легальных пользователей и сценариям атак на КИС. Недостатки ИСЗИ на базе ЭС: система не является адаптивной; не всегда обнаруживаются неизвестные атаки [Калач А.В., 2011]. Если НС представлена в виде отдельной системы

обнаружения атак, при обработке трафика происходит анализ информации на наличие злоупотреблений. Случаи с указанием на атаку перенаправляются к администратору безопасности. Подход быстросействующий, поскольку используется один уровень анализа. Одним из основных недостатков нейронной сети является "непрозрачность" формирования результатов анализа.

В системах обнаружения атак можно выделить применение нейронных сетей, дополненных ЭС. Чувствительность системы возрастает, так как экспертная система получает данные только о событиях, которые рассматриваются в качестве подозрительных. Если нейронная сеть за счет обучения стала идентифицировать новые атаки, то экспертную систему следует обновить [Калач А.В., 2011].

Использование гибридных нейро-экспертных или нейро-нечетких систем позволяет отразить в структуре системы нечеткие предикатные правила, которые автоматически корректируются в процессе обучения нейронной сети. Свойство адаптивности нечетких нейронных сетей позволяет решать отдельно взятые задачи идентификации угроз, сопоставления поведения пользователей с имеющимися в системе шаблонами, автоматически формировать новые правила при изменении поля угроз [Калач А.В., 2011].

Недостатками этих систем являются: необходимость наличия экспертов высокой квалификации; трудности, возникающие при адаптации методов к потребностям конкретной организации; невозможность оценить эффективность конкретного комплекса средств защиты, применяемого на объекте защиты; требование наличия на предприятии достоверной статистики по инцидентам информационной безопасности.

2. Поддержка принятия решений в интеллектуальной системе защиты информации

Одна на сегодняшний день научная проблема в области построения ИСЗИ - это обеспечение интеллектуальной поддержки принятия решений (ИППР) по всему комплексу задач, решаемых ИСЗИ. В работе [Рахимов, Е. А. 2008] сделаны отдельные предложения по данной проблеме:

- предложено рассматривать множество угроз как множество каналов несанкционированного доступа, утечки информации и деструктивных воздействий (НСДУВ), реализуемых злоумышленником или нарушителем. Угроза рассматривается, с одной стороны, как сложная последовательность компонентов угроз при манипулировании злоумышленником информационными потоками, с другой - в виде графа структуризации на

множестве элементов физической среды распространения носителя информации. Подход позволяет использовать как статистические оценки уровней компонентов угроз, так и экспертные оценки: уровни компонентов угроз вычисляются с использованием аппарата нечеткой логики;

- разработана методика численной оценки уровня защищенности информации, в которой используются данные интегральной структурной вербальной модели каналов НСДУВ, позволяющая сравнивать различные комплексы средств защиты по уровню защищенности, проводить количественный анализ состояния информационной безопасности ОИ с целью выработки решений по усилению или ослаблению функций защиты.

- предложен метод синтеза рациональных наборов средств защиты, состоящих из совместимых программно-аппаратных продуктов, по целевой функции, максимизирующей отношение суммарного показателя защищенности к сумме показателей издержек, включающих стоимость, причем численные значения показателей защищенности и издержек определяются с использованием метода анализа иерархии. Метод позволяет осуществить синтез рациональных наборов средств защиты, состоящих из совместимых программно-аппаратных продуктов, с априорно заданными свойствами, удовлетворяющими требованиям к защищенности информации на ОИ.

- разработано алгоритмическое обеспечение подсистемы поддержки принятия решений (ПППР) по оперативному управлению защитой информации, позволяющее с одной стороны минимизировать влияние угроз на защищаемую информацию, с другой - уменьшить вероятность того, что ответные действия повлияют на нормальное функционирование защищаемого ОИ.

- предложена архитектура построения интеллектуальной СЗИ, позволяющей обеспечить автоматизированную поддержку принятия решений по выбору рационального ее состава и изменению его в процессе эксплуатации, по выбору варианта оперативного реагирования при возникновении потенциально опасных ситуаций в условиях неопределенности информационных воздействий.

По проблеме ИППР соединяющие результаты получены в работах [Машкина И.В., 2008, 2009]:

- модель противодействия угрозам нарушения информационной безопасности, базирующихся на использовании адаптированного для выбора рационального варианта реагирования метода принятия решений, заключается в том, что решение о выборе варианта реагирования принимается в зависимости от вероятности атаки, которая оценивается с использованием механизма нечеткого логического вывода, на основе оперативных данных о событиях безопасности от различных обнаружителей

- метод формирования рационального комплекса средств защиты заключающийся в том,

что на основе трехрубежной модели защиты разрабатываются морфологические матрицы для каждого из рубежей, генерируются варианты наборов средств защиты с использованием вспомогательных матриц совместимости программно-аппаратных средств, разрабатывается система иерархических критериев качества средств защиты на основе их технических характеристик, выбирается рациональный вариант набора для каждого рубежа защиты по целевой функции, максимизирующей отношение суммарного показателя «защищенность информации» к суммарному показателю «издержки», в состав системы защиты информации включаются рациональные наборы, суммарная стоимость которых не превышает выделенных на защиту ресурсов, что позволяет получить комплекс средств защиты, сертифицированных по заданному классу защищенности, удовлетворяющий требованиям к допустимым затратам на его реализацию.

3. Защита информационных ресурсов предприятия на основе многоагентной технологии

В работе [Kotenko et al., 2005] предложен подход к созданию команд агентов, участвующих в информационном. Подход позволяет моделировать атаки, направленные на нарушение доступности информационных ресурсов, и механизмы защиты от них. Проведено большое количество разнообразных экспериментов, в которых исследовались параметры эффективности механизмов защиты от топологии и конфигурации сети, структуры и конфигурации команд атаки и защиты, которые показали, что использования кооперации команд защиты приводит к повышению эффективности защиты. Планируется разработка формальных моделей поведения сложных систем в Интернет, совершенствование среды моделирования, более глубокое исследование эффективности механизмов кооперации различных команд и внутрикомандного взаимодействия агентов, реализация механизмов адаптации и самообучения агентов.

В работе [Погорелов Д. Н., 2008] предложена новая концепция построения интеллектуальной системы защиты информации предприятия, основанная на сочетании принципов функциональной интеграции, иерархической организации, комплексирования моделей, методов и алгоритмов, стандартизации систем защиты информации, построения информационных систем, что позволяет обеспечить согласованную работу всех подсистем защиты информации. Это позволило построить архитектуру автоматизированной системы защиты информации, основанная на многоагентном подходе.

Для решения задачи обнаружения вирусных атак в сети Интернет предлагается архитектура на

основе продукционной системы с многоуровневой вертикальной моделью агентов [Берестов А.А, 2011].

Данная архитектура включает базу знаний в виде правил продукций, механизма логического вывода, рецепторов и эффекторов агента, модуль коммуникации с другими агентами. Применительно к задаче обнаружения вирусных атак, рецепторы передают факты о внешних воздействиях в базу знаний. В результате логического вывода вырабатывается решение, которое передается эффектору об изменениях внешней среды.

Для распределенного решения задач могут быть использованы разные типы агентов: агент-субординатор, множество агентов исполнителей, агент-интегратор. Агенты могут быть связаны между собой в виде многоуровневой архитектуры, которая может быть горизонтальной или вертикальной. Для решения задачи обнаружения вирусных атак подходит вертикальная многоуровневая архитектура агентов. С учетом специфики решаемой задачи проектируемая многоагентная система должна включать несколько агентов, которые выполняют в системе различные функции. В результате анализа информационного процесса обнаружения вирусных атак в сетях КИС можно рассматривать агентов, разграничивающих права доступа пользователей сети, агентов обнаружения вторжений, то есть изменения состояния окружающей среды в сети, агентов обнаружения типа атаки, агентов, строящих сценарий поведения для отражения вирусной атаки, агентов, являющийся посредником-координатором всей многоагентной системы.

В работе [Никишева А.В, 2013] проанализированы основные распространяемые системы обнаружения атак (COA): Snort, Bro, Prelude, OSSEC, Suricata и рассмотрены основные тенденции их развития. В результате этого был определен перечень критериев и их значений, которым должна удовлетворять COA:

- многоуровневость наблюдения за системой. COA должна собирать сведения о состоянии ИС из различных источников на различных уровнях наблюдения – уровень сети, сервера и хоста;
- адаптивность, т.е. способность COA обнаруживать модифицированные реализации известных атак и новые виды атак.
- проактивность, COA должна обладать встроенными механизмами реакции на атаку
- открытость, COA должна обладать возможностью добавления новых анализируемых ресурсов информационной системы.
- тип управления. COA должна совмещать как централизованное, так и распределенное управление.
- защищенность. COA должна обладать средствами защиты своих компонентов.

В результате представлены следующие решения по многоагентной системе обнаружения атак на КИС:

– структура и состав многоагентной системы обнаружения атак, включающая в себя агентов рабочих станций, серверов, маршрутизаторов и сетей и позволяющая делать вывод о атаках, состоянии КИС и перспективах ее защиты;

– метод принятия агентами совместного решения, позволяющий сформировать круглый стол агентов и на основании их результатов анализа сведений, полученных из различных источников, оценить состояние КИС в целом;

– методика обнаружения атак с использованием многоагентных технологий, позволяющая обучить многоагентную систему обнаружению атак и использовать ее для дальнейшего обнаружения неизвестных воздействий.

– оценка эффективности всех предложенных методов, используя разработанные программные решения.

4. Тенденции и концепция развития

В качестве тенденций и концепции развития по использованию ИТвЗИ можно представить следующее [Машкина И.В., 2009, Вишняков В.А., 2013]:

– совершенствование архитектур систем ЗИ в КИС, обеспечивающих эффективное управление в условиях неопределенности состояния информационной среды;

– разработка новых моделей противодействия угрозам нарушения ИБ в КИС на основе выбора оптимального варианта реагирования на события безопасности;

– совершенствование инструментальных программных комплексов с интеллектуальной поддержкой принятия решений с исследованием эффективности методов, моделей и алгоритмов;

– развитие технологий многоагентных систем для обнаружения атак, противодействия угрозам нарушения ИБ, оценки уровня защищенности информации в КИС.

– разработка теоретических основ, моделей и средств защиты облачной инструментальной платформы проектирования интеллектуальных систем на основе семантических технологий.

Заключение

Первым направлением в СЗИ является дальнейшая разработка моделей, методов, архитектур и аппаратно-программных средств управления ЗИ для решения проблемы защиты КИС и облачной инструментальной платформы проектирования интеллектуальных систем на основе семантических технологий.

Другим направлением СЗИ является разработка моделей, методов, архитектур и аппаратно-программных средств сбора, структуризации информации из Интернете, формирования специализированных баз знаний и поддержки

принятия решений (на базе ИТ) по всему накопленному аспекту задач ИБ.

Библиографический список

[Машкина И.В., 2008] Машкина, И.В. Идентификация угроз на основе построения семантической модели информационной системы / И.В. Машкина // Вестник УГАТУ: Науч. журн. Уфимск. гос. авиац. техн. ун-та. Сер. Управление, вычислительная техника и информатика. 2008. №11. С. 208 – 214.

[Электр. ресурс, 2013] Современные технологии обеспечения информационной безопасности. Электронный ресурс. Код доступа: <http://ipb.mos.ru/ttb> Время доступа 9.12.2013.

[Калач А.В., 2011] Калач, А.В., Немтина Е.С. Интеллектуальные средства и моделирование систем защиты информации Интернет-журнал "Технологии техносферной безопасности" (<http://ipb.mos.ru/ttb>) Выпуск № 3 (37) – 2011. – С.3-11.

[Рахимов, Е. А. 2006] Рахимов, Е. А. Модели и методы поддержки принятия решений в интеллектуальной системе защиты информации. /Е.А. Рахимов/ Автореферат канд. дисс. по спец. 05.13.19. Уфа, 2006. – С.18.

[Kotenko et al., 2005] Kotenko, I., Ulanov A. Multiagent modeling and simulation of agents' competition for network resources availability // Second International Workshop on Safety and Security in Multiagent Systems. Utrecht, The Netherlands. 2005.

[Погорелов Д.Н., 2006] Погорелов, Д. Н. Защита информационных ресурсов предприятия на основе многоагентной технологии / Д. Н. Погорелов // Автореферат канд. дисс. по спец. 05.13.19. Уфа, 2007. – С.16.

[Берестов А.А., 2011] Берестов, А.А. Архитектура интеллектуальных агентов на основе продукционной системы для защиты от вирусных атак в сети Интернет / А.А. Берестов // Материалы . XV Всероссийской научной конференции «Проблемы информационной безопасности в системе высшей школы» М.: МИФИ, 2011. – С.24-25.

[Никишева А.В., 2013] Никишева, А.В. Многоагентная система обнаружения атак на информационную систему предприятия /А.В. Никишева // Автореферат канд. дисс. по спец. 05.13.19. Волгоград, 2013. – С.19.

[Машкина И.В., 2009] Машкина, И.В. Модели и метод принятия решений по оперативному управлению защитой информации / И.В. Машкина // Системы управления и информационные технологии. Москва - Воронеж, 2008. №2 (32). С. 98 – 104

[Вишняков, В.А., 2013] Вишняков, В.А. Анализ методов и средств защиты информации и использование интеллектуальных агентов для ее совершенствования / В.А. Вишняков // Материалы межд. научной конференции ИСиТ. Мн.: БГУИР, 2013. – С.130-131.

STATE, TRENDS AND CONCEPTION DEVELOPMENT OF INTELLIGENCE TECHNOLOGIES IN INFORMATION DEFENSE

Vishniakou U.A.

*Minsk Management Institute,
Minsk, Republic of Belarus*

vish2002@list.ru

Two problems the use of intelligence technologies in information defense (ITID) – creating specialized knowledge bases with threats simulation and high the security level in corporative nets and cloud computing are presented. The analysis of two directions of the second ITID problem: the intelligence decision support systems and the malty agent system use are given. As trends and conception development of intelligence technologies are the perfection of methods. models, architectures, and hard-sotware tools for ITID in corporative systems and cloud computing.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.032.26

РАЗРАБОТКА ГИБРИДНОЙ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ОЦЕНКИ ЗНАНИЙ СТУДЕНТОВ

Козлов О.А., Михайлов Ю.Ф.

*Учреждение Российской Академии Образования «Институт Информатизации Образования»
г. Москва, Россия*

ole-kozlov@yandex.ru

mikhayurij@yandex.ru

В статье приводится описание подхода по формированию оценки индивидуальной траектории изучения студентом некоторой предметной области знаний с помощью гибридной искусственной нейронной сети.

Ключевые слова: вектор параметров оценки качества усвоения знаний, нечеткие нейронные сети, интеллектуальная обучающая система.

Введение

Формирование индивидуальной траектории изучения является задачей со множеством неопределенных параметров, поэтому предлагается для оценки результатов ее решения применять искусственную нейронную сеть. Предлагается для оценки результатов обучения применять гибридные нейронные сети, построенные на основе нечеткой логики, описывается процесс функционирования системы. Ядром предлагаемой информационной обучающей системы, влияющим на построение траектории обучения, является индивидуальная траектория изучения студентом некоторой предметной области знаний, которую можно представить в виде декартового произведения векторов. Введение в структуру информационной обучающей системы модели индивидуальной траектории изучения студентом некоторой предметной области знаний, которая включает личностные и профессиональные качества студента, а также нечетких нейронных сетей, которые оценивают результаты обучаемых и определяют очередной этап траектории обучения, позволяют индивидуализировать и дифференцировать процесс обучения.

Интеллектуализация контроля индивидуальной траектории изучения предметной области знаний на основе гибридной нейронной сети

Проблема количественной оценки уровня знаний студентом, курсантом является актуальной.

Ввиду того, что при решении поставленной проблемы применяется метод экспертных оценок, в которых всегда присутствуют неточность и субъективность, целесообразно будет применение методов нечеткого моделирования: нечетких нейронных сетей. Предлагается использовать метод решения неформализованных задач, при помощи которого задачу можно представить в виде иерархического дерева подзадач и указать для каждой из них свой метод решения [Рутковская, 2006].

Средству обучения, функционирующему на базе информационных и коммуникационных технологий, при необходимости (по Роберт И.В.) можно частично передать функции обучающего: контроль результатов обучения; предоставления заданий, адекватных уровню обучающегося.

Индивидуальная траектория изучения студента выражает цель обучения и содержит информацию о состоянии знаний обучаемого.

С целью оценки начальных знаний по выбранному или заданному учебному материалу применяют ряд тестовых заданий, формирующих его рейтинг и другие характеристики. При этом задание разбивается на подзадачи (для конкретизации), а возникающая точка n-мерного шара ошибок анализируется системой. Такая организация позволяет избежать случайного совпадения ответа пользователя с правильным ответом.

Для формирования критерия качества обучения студента дидактической единице собирают точки зрения экспертов – какие наборы показателей в

зависимости от выбранной стратегии необходимо включать в модель. Далее происходит обучение сети и генерирование вариантов обучения в зависимости от той или иной стратегии. Точки зрения экспертов являются входными данными (векторами) для нейросети. По мере ввода данных осуществляется обучение сети (корректировка весовых коэффициентов).

Результат работы сети – стратегия обучения студента, которой соответствует определенный набор показателей. После того как сеть обучена, система приобретает возможность генерировать варианты наборов показателей в зависимости от того, какую стратегию обучения выбрал студент.

Программа дисциплины строится по модульному принципу, разработанному и представленному в монографии доктора педагогических наук О. А. Козлова «Теоретико-методологические основы информационной подготовки курсантов военных учебных заведений» [Козлов, 2001].

В каждом модуле выделяются элементы знаний, представленные в учебной программе, устанавливаются структурные и семантические связи между элементами и разрабатываются семантические модели знаний для этих модулей. Связи между элементами знаний реализуются по гипертекстовой технологии, в виде ссылок. В качестве ссылки берется ключевое слово, терм, понятие, которое присутствует в структурно связанных между собой элементах знаний [Михайлов, 2001].

Знание оценивается по трем уровням, которые задаются в государственной программе [Козлов, 2001], [Роберт, 2009]. Первый уровень усвоения означает наличие у студентов, курсантов знаний – знакомств, умение выделить и отличить среди нескольких понятий то, которое им предъявлено для опознания. Это уровень проблемной ситуации типа «выбор», вес ситуации, вопроса и ответа, которые генерируются такой ситуацией, равен единице. Второй уровень усвоения означает наличие у курсантов, студентов знаний – копий, то есть умение решать типовые задачи по типовому алгоритму, или применить типовой алгоритм в новых условиях. Это уровень проблемной ситуации типа «неопределенность решения», вес ситуации определяется количеством продуктивных операций, которые надо выполнить для ответа на контрольный вопрос, либо числом термов контролируемого модуля знаний. Третий уровень предполагает наличие у курсанта знаний – умений, знаний – навыков, то есть умение решать типовые задачи в условиях неопределенности постановочной части задачи. Это уровень проблемной ситуации типа «поиск выхода», вес ситуации определяется количеством продуктивных операций, которые надо было выполнить для ответа на контрольный вопрос.

Для рубежного контроля предлагается использовать систему рейтинговых оценок [Козлов,

1999], [Ширшов, 2002]. Рейтинговое число по теме определяется следующим образом. Вес одного существенного действия принят за единицу. Число баллов за вопрос определено как произведение весов на число действий. Весовая характеристика вопроса определяет долю знания вопроса в общем объеме знания (теста), и может быть представлена в виде модели вывода о качестве усвоения знания студентом. Однако, классические нейронные сети, такие как многослойные персептроны, имеют важный недостаток – трудности при объяснении полученных результатов.

Для реализации предложенного алгоритма контроля предлагается использовать нечеткие нейронные сети Такаги-Сугено-Канга (ИНС TSK), в которых модель вывода (функция заключения) определяется в виде совокупности M нечетких правил.

Анализ литературы [Круглов, 2001], [Рутковская, 2006] позволяет предложить для решения задачи оценки качества обучения по предложенной методике ИНС TSK, которая состоит из 5 слоев, среди которых только 2 настраиваемых – фаззификации и заключения (рисунок 1).

Слой 1 выполняет фаззификацию входных переменных, определяя для k -го правила и входной переменной X_i значение функции принадлежности. Обычно на практике используется обобщенная функция Гаусса либо сигмоидальная функция.

Слой агрегации определяет уровень активации k -го правила как результирующее значение функции принадлежности W_k для входного вектора \bar{X} .

Операции пересечения нечетких множеств соответствуют операции произведения или минимизации.

Слой 3 представляет собой реализацию функции заключения $Y=F(\bar{X})$, в качестве которой на практике часто используется обычный полином первого порядка.

Последние 2 слоя не содержат настраиваемых параметров и не нуждаются в обучении. В модели TSK выход системы определяется как средневзвешенное по всем M правилам.

Подстройка параметров нейронов на каждой итерации должна производиться в сторону уменьшения отклонения между вычисленными значениями функции принадлежности и эталонными.

Обучение слоя фаззификации следует проводить с помощью градиентных методов (наискорейшего спуска, k -Partan) либо генетических алгоритмов [Рутковская, 2006].

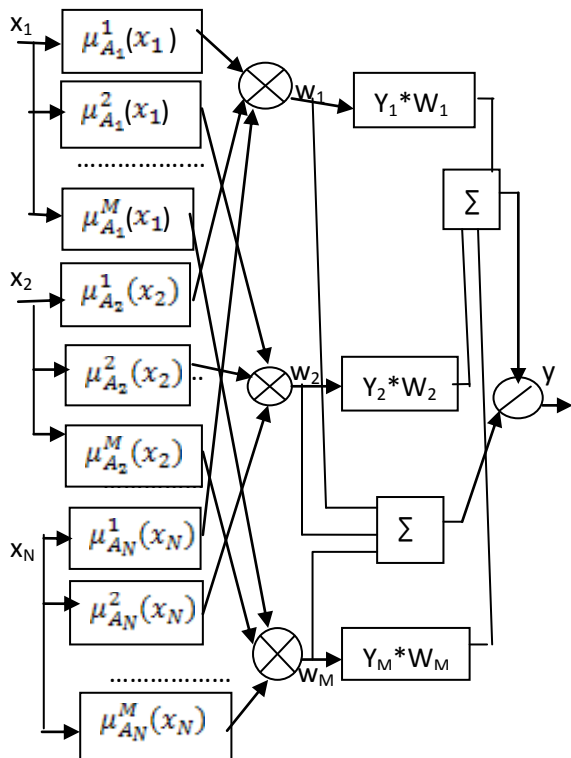


Рисунок 1 – Структура нечеткой нейронной сети Такаги-Сугено-Канга

где P_k — количество вопросов в обучающей выборке;

$\mu_y^k(\bar{x}^q)$ — желаемое (эталонное) значение функции принадлежности ответа для входного вектора \bar{x}^q

$\mu_A^k(\bar{x}^q)$ — вычисленное сетью значение уровня активации k-го правила (результатирующее значение функции принадлежности ответа для входного вектора \bar{x}^q).

В зависимости от текущего уровня знаний по изучаемой дисциплине движение по траектории изучения дидактической единицы можно разбить на три основных направления: возврат назад по траектории к предыдущему дидактическому элементу изучаемой дидактической единице учебного курса, если обучаемый неудовлетворительно справился с предлагаемыми тестовыми заданиями по изучаемой теме или разделу; движение вперед к новому дидактическому элементу, следующей порции учебного материала, определенной программой курса. В случае если предыдущая тема или раздел учебного курса были освоены на хорошем или отличном уровне выполняется движение вперед к новой дидактической единице учебного материала. Каждое выделенное направление движения по индивидуальной траектории изучения включает

несколько вариантов, позволяющих индивидуализировать и дифференцировать процесс обучения.

Таким образом, интеллектуальную информационную обучающую систему можно использовать для управления процессом обучения; с целью формирования индивидуальной траектории обучения; автоматизации процессов контроля и коррекции результатов учебной деятельности.

Заключение

Интеллектуальная информационная система организация учебного процесса на основе гибридной искусственной нейронной сети позволяет формировать и управлять индивидуальной траекторией изучения студентом, курсантом некоторой предметной области знаний, сохраняя, анализируя и объединяя результаты контроля, сформированные на основе разрозненных, субъективных мнений экспертов с помощью гибридной нейронной сети.

Библиографический список

- [Козлов, 1999] Козлов О.А. Развитие методической системы обучения информатике курсантов военно-учебных заведений Министерства обороны Российской Федерации. Дисс.... докт. пед. наук. – Серпухов. 1999.
- [Козлов, 2001] Козлов О.А. Теоретико-методологические основы информационной подготовки курсантов военно-учебных заведений. Монография. - М.: МО, 2001. – 328 с
- [Круглов, 2001] Круглов В.В., Дли М.И., Голунов Р.Ю. Нечеткая логика и искусственные нейронные сети. – М.: Горячая линия – Телеком, 2001.
- [Михайлов, 2001] Михайлов Ю.Ф. Технология информационной подготовки курсантов в условиях моделирования экстремальных ситуаций профессиональной деятельности. Дисс.... канд. пед. наук. – Москва. 2001.
- [Роберт, 2009] Роберт И.В. Теория и методика информатизации образования (психолого-педагогический и технологический аспекты), 2-е издание, дополненное. – М.: ИИО РАО, 2009.
- [Рутковская, 2006] Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы //Д. Рутковская, М. Пилиньский, Л. Рутковский; пер. с польск. И.Д. Рудинского - М.: Горячая линия - Телеком, 2006. - 452с.: ил..
- [Ширшов, 2002] Ширшов Е.В. Применение технологий искусственного интеллекта в учебном процессе вуза. Образовательные технологии для новой экономики: Сб. материалов международной конференции – М.: Изд-во МЭСИ, 2002. – с.340 – 348.

DEVELOPMENT OF A HYBRID INTELLECTUAL SYSTEMS FOR SOLVING THE PROBLEM OF ASSESSING KNOWLEDGE STUDENTS

Kozlov O.A., Mikhailov J.F.

*Institution Of The Russian Academy Of Education
"Institute Of Informatization Of Education",
Moscow, Russia*

ole-kozlov@yandex.ru

mikhayurij@yandex.ru

The article describes the approach to the formation of the assessment of individual trajectory of studying a

certain subject area knowledge through a hybrid artificial neural network. Key words: the vector of the parameters of assessment of quality of learning, fuzzy neural networks, intelligent training system.

Introduction

Proposed for the assessment of learning outcomes, use of hybrid neural network built on the basis of fuzzy logic describes the process of functioning of the system.

Introduction in structure of information learning system model of the individual trajectory of studying a certain subject area knowledge, which includes the personal and professional qualities of the student, and fuzzy neural networks, which evaluate the results of trainees and define the next stage of learning paths, allow you to customize and differentiate the process of learning.

Main Part

The problem of quantitative estimation of the level of knowledge of a student, a cadet is urgent.

Due to the fact that in solving this problem the method of expert estimates, which are always present inaccuracy and subjectivity, it is the application of methods of fuzzy modeling: fuzzy neural networks.

To estimate the initial knowledge in the chosen or given training material, apply a series of test tasks, forming its rating and other characteristics. The job is broken down into subtasks (to specify), and the appearing point n-dimensional ball errors analyzes system.

The program of the discipline is based on modular principle, developed and presented in the monograph by doctor of pedagogical Sciences O. A. Kozlova «Theoretical-methodological bases of information training of cadets of military educational institutions».

In each module are allocated elements of knowledge presented in the curriculum, are structural and semantic connections between the elements and develops semantic models of knowledge for these modules. The connections between the elements of knowledge are implemented on the hypertext technology in the form of links. As a reference is taken keyword, term, a concept which is present in structurally related elements of knowledge [Mikhailov, 2001].

Knowledge is assessed on three levels that are set in the state program [Kozlov, 2001], [Robert, 2009]. The first level of learning means that the MSU students, cadets knowledge - Dating identify and distinguish among several concepts that which they presented for identification. This level of problem situations such as «choice», the weight of the situation, questions and answers that are generated in such a situation, is equal to unity. The second level of learning means that the cadets, students knowledge copies, that is, the ability to solve typical tasks on a standard algorithm, or apply a model algorithm in the new conditions. This level of problem situations such as «uncertainty of decisions,» the weight of the situation is determined by the number of productive operations, which need to be fulfilled to answer the security question or the number of terms

monitored module knowledge. The third level implies cadet knowledge - skills, knowledge, skills, i.e. the ability to solve typical tasks in conditions of uncertainty staging of tasks. This level of problem situations such as «search of the way out,» the weight of the situation is determined by the number of productive operations that had to be done to answer the security question.

Rating number on the topic is defined as follows. The weight of one of significant actions taken as a unit. The number of points for the question defined as the product of the weights on the number of actions. Weight characteristics of the issue, is the proportion of knowledge of the question in the total volume of knowledge (testing), and can be represented as a model of conclusion on the quality of mastering of student's knowledge.

For realization of the proposed algorithm of control is offered to use fuzzy neural networks Takagi-Sugeno-Kang (Ann TSK), in which the output function (conclusion) is defined as a set M of fuzzy rules. In the model TSK output of the system is defined as the weighted average for all M rules.

Analysis of the literature [Kruglov, 2001], [Rutkovskaya, 2006] allows us to offer to solving the problem of assessing the quality of education in the proposed method Ann TSK, which consist of 5 layers, of which only 2 custom - фаззификации and conclusions.

Layer 1 performs фаззификацию input variables, defining for the kth rules and input variable X_i value of the membership function. Usually the practice is to use a generic function of Gauss or сигмоидальная function. Layer aggregation determines the level of activation of the k-th rules as the resulting value of the membership function W_k for input vector . The operation of intersection of fuzzy sets correspond to the operations of product or minimization.

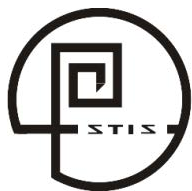
Layer 3 is an implementation of the functions of the conclusion of $Y=F()$, which in practice often use a regular polynomial first order.

The last 2 layer does not contain custom parameters and does not need training. In the model TSK output of the system is defined as the weighted average for all M rules.

Adjustment of parameters of neurons at each iteration must be made towards reducing the deviation between calculated values of function facilities and reference. Training layer фаззификации should be carried out with the help of gradient methods (quickest descent, k-Partan) or genetic algorithms [Rutkovskaya, 2006].

Conclusion

Intellectual information system of organization of educational process on the basis of a hybrid artificial neural network allows you to create and manage an individual trajectory of student, a cadet of a certain subject area knowledge, preserving, analyzing and aggregating the results of control formed on the basis of disparate, subjective opinions of experts using a hybrid neural network.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.855.3

ИСПОЛЬЗОВАНИЕ АДАПТИВНЫХ СЕМАНТИЧЕСКИХ МОДЕЛЕЙ ДЛЯ ПРЕДСТАВЛЕНИЯ И СТРУКТУРИЗАЦИИ ЗНАНИЙ В СИСТЕМАХ ОБРАЗОВАТЕЛЬНОГО НАЗНАЧЕНИЯ

Шихнабиева Т.Ш.

*Институт информатизации образования
Российской академии образования, г. Москва, Россия
shetoma@mail.ru*

Для эффективного функционирования информационных систем, используемых в системах обучения, необходимо решить ряд проблем, связанных с представлением, обработкой и использованием знаний. Для хранения и эффективного использования знаний необходима их структуризация и модульное представление. В работе предлагаются некоторые подходы к решению указанных проблем в интеллектуальных системах образовательного назначения.

Ключевые слова: системы образовательного назначения, структуризация знаний, модульное представление, адаптивные семантические модели.

Интенсивное развитие аппаратной, программной составляющих персональных компьютеров, проникновение информационных и коммуникационных технологий во все сферы науки, общественной жизни, образование, их использование для решения все более усложняющихся задач, постепенно пришли в противоречие с технологией фон – неймановских компьютеров. В рамках этой технологии необходимо качественно описывать проблему, формализовать ее, составлять алгоритм, разрабатывать программу, анализировать результаты, видоизменять в случае необходимости постановку проблемы и последующие компоненты. В итоге появляются лишь новые процедурные знания, т.е. факты, являющиеся результатом алгоритмического преобразования других фактов. В связи с этим возникает необходимость в поиске решения сложных проблем в человеко-машинной системе, каковым является процесс обучения с использованием информационных и коммуникационных технологий. Главной особенностью систем, основанных на знаниях, является наличие у них базы знаний, в которых они описаны и механизмы задач, использующие эти знания. Оба этих компонента систем обучения находятся в тесной взаимосвязи между собой. Для разработки систем, основанных на знаниях, необходимо их соответствующее представление. Причём это одна из важных проблем, которая влияет на характеристики обучающей системы и в

конечном итоге, на качество знаний обучаемых. Выбор способа представления знаний во многом зависит от характера и сложности решаемых задач. Форма представления знаний оказывает существенное влияние на характеристики и свойства систем обучения [1]. В данной статье речь идёт о представлении, хранении, обработке больших объёмов информации, а также о приобретении и использовании её для получения новых знаний.

Процесс обучения на основе информационных и коммуникационных технологий мы рассматриваем как один из видов информирования, основой которого является семантический диалог [2]. Для рассматриваемой системы возникает необходимость решения проблемы формализации семантического диалога, обеспечивающая повышение эффективности ее функционирования.

В свою очередь, для успешного решения указанной проблемы необходимо решить две задачи: формализация процесса информирования и формализация семантической информации. Формализация семантической информации предполагает решение одной из важных задач теории информационных семантических систем, а также искусственного интеллекта - представление знаний. На основе анализа существующих подходов инженерии знаний, в качестве основного способа структуризации знаний в системе обучения информатике нами выбраны адаптивные семантические модели (АСМ) [3]. Основой данной модели является система знаний, имеющая

определённый смысл в виде целостного образа сети, узлы которой соответствуют понятиям и объектам определённой предметной области, а дуги – отношениям между объектами. Адаптивные семантические модели используются также нами и для представления непосредственно самого процесса обучения. Предлагаемый нами подход основан на структуре человеческих знаний, принципах разработки систем искусственного интеллекта и информационных семантических систем, каковым является процесс обучения. Он объединяет процедурный и декларативный подход к представлению знаний, базируется на теории семантических сетей и продукционных правил. Следует подчеркнуть, что семантическая модель, изображая логическую структуру учебного материала в соответствии с существующими связями между его понятиями, одновременно показывает все основные понятия изучаемой темы и связи между ними, что облегчает её восприятие. Как показывает авторский опыт разработки семантических моделей по некоторым учебным дисциплинам, сам процесс построения моделей способствует эффективному приобретению знаний. Поэтому обучение студентов можно вести не только по разработанным преподавателем семантическим моделям, но и задавать студентам задания по их разработке, что способствует лучшему усвоению учебного материала.

На основе предложенных подходов проведена структуризация учебного материала по некоторым профильным дисциплинам подготовки будущих учителей информатики (“Программирование”, “Программное обеспечение ЭВМ”, “Архитектура компьютера”, “Компьютерное моделирование”, “Теоретические основы информатики”, “Математическая логика”, “Компьютерные сети”, “Основы искусственного интеллекта”), на основе которого разработан структурированный и построенный в виде справочника электронный учебник.

Преимуществом семантических сетей как модели представления знаний и непосредственно самого процесса обучения является наглядность описания предметной области, гибкость, адаптивность к цели обучаемого. Однако, свойство наглядности с увеличением размеров и усложнением связей базы знаний предметной области теряется. Кроме того, возникают значительные сложности по обработке различного рода исключений. Для преодоления указанных проблем используют метод иерархического описания сетей (выделение на них локальных подсетей, расположенных на разных уровнях) (рис.1).

Для построения модели предметной области в виде семантической сети разработан следующий алгоритм:

1. Классификация всех понятий рассматриваемой предметной области на макропонятия (класс понятий), метапонятия

(обобщенные понятия) и микропонятия (элементарные понятия).

2. Выделение общих свойств, признаков, присущих каждому уровню понятий.

3. Выделение отличительных признаков каждого уровня понятий.

4. Установка связей между понятиями, относящимися к одному уровню.

5. Выделение связей между понятиями, принадлежащими различным уровням семантической модели.

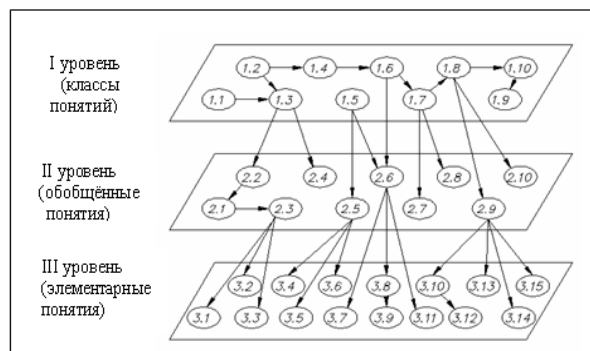


Рис.1. Общая многоуровневая модель представления знаний.

На основе установленных типов понятий и видов связей разработана модель предметной области “Информатика” в виде многоуровневой иерархической семантической модели. Необходимо отметить, что процесс подобной структуризации знаний при разработке модели их представления и контроля достаточно трудоемкий. Однако, это наглядная и более выразительная модель, отображающая логическую структуру учебного материала, которая позволяет одновременно видеть все понятия и их взаимосвязи по изучаемой теме, что имеет немаловажное значение для систем обучения на основе информационных и коммуникационных технологий.

Предложенные подходы к представлению и контролю знаний заложены в основу интеллектуальной обучающей системы (ИОС) “КАСПИЙ”, структурная схема которой приведена на рис.2. Программная оболочка ИОС “КАСПИЙ” реализована в объектно-ориентированной среде программирования Delphi. Для работы с базами данных имеется множество компонентов.

Благодаря компонентному подходу к программированию, программу можно собирать как конструктор, настраивая каждый компонент для решения той или иной задачи. Аналогично тому, как программист использует компоненты Delphi, так и компоненты Delphi используют системные функции для выполнения определенных функций.

Структура, принципы построения и пользовательский интерфейс ИОС “КАСПИЙ” предусматривает её использование в процессе обучения в следующих режимах: “Редактирование”, “Обучение”, “Проверка знаний”.

На этапе режима “Редактирование” формируются проблемно – ориентированные базы знаний учебных дисциплин на основе адаптивных семантических моделей.

Кроме того, по мере пополнения новыми понятиями содержание учебных дисциплин методика использования ИОС “КАСПИЙ” в учебном процессе предусматривает редактирование адаптивных семантических моделей учебного материала.

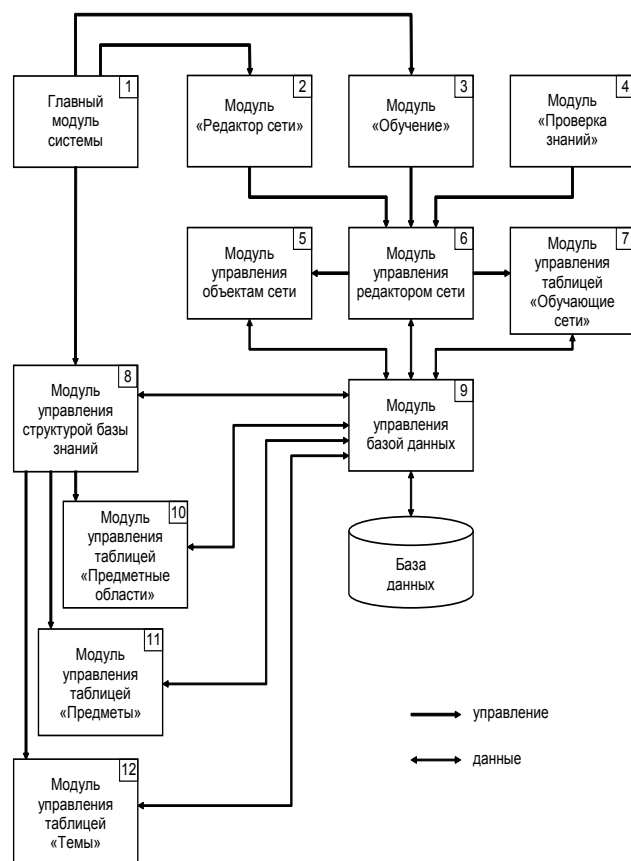


Рис. 2. Структурная схема системы “КАСПИЙ”.

Режим “Обучение” предъявляет пользователю учебный материал с учётом уровня его знаний, т.е. система “КАСПИЙ” является адаптивной. Режим “Проверка знаний” предполагает генерацию контрольных заданий различной сложности. Ответы обучаемых на контрольные задания в виде адаптивных семантических моделей сравниваются с находящимися в базе знаний моделями системы “КАСПИЙ” и выдаётся соответствующая оценка. В данной обучающей системе предусмотрена панель истории навигации, предназначенной для отображения пути, пройденного пользователем в структуре базы знаний и документирование результатов знаний.

Следует отметить, что предусмотрена как автономная, так и сетевая версия использования системы обучения “КАСПИЙ” и она инвариантна по отношению к конкретным учебным дисциплинам.

Как видно из структурной схемы системы

“КАСПИЙ” все операции с базой данных (БД) выполняются через модуль управления БД. Модуль содержит множество процедур и функций, обеспечивающих взаимодействие с БД без использования инструкций языка SQL и без непосредственного обращения к БД. Модуль управления базой данных является одним из основных модулей системы “КАСПИЙ” [4].

Система “КАСПИЙ” отличается тем, что выявляет базовый уровень знаний каждого студента и предоставляет ему соответствующий учебный материал. Два других основных модуля системы – это модуль управления редактором сети и модуль управления объектами сети. Данные три модуля составляют ядро системы (на рисунке выделено пунктиром). Все остальные модули являются надстройкой ядра и обеспечивают удобный интерфейс взаимодействия с пользователем. Модуль управления редактором сети содержит в себе полный набор функций для управления сетью. Модули «Обучение», «Редактор сети» и «Контроль знаний» используют только свою часть данного набора функций.

Итак, использование адаптивных семантических моделей в системах образовательного назначения позволяет: обеспечивать индивидуальный темп обучения при реализации обратной связи: деятельностный подход при выборе решения задачи с учетом учебных ситуаций; установить связи новых понятий с существующими понятиями и представлениями, что улучшает понимание; осуществление глубокой обработки знаний, что повышает способность применять знания в новых ситуациях. Предложенная модель учебной дисциплины показывает последовательность изложения учебного материала, что очень важно для начинающих учителей. Кроме того, последовательность изложения учебного материала может варьироваться. С помощью АСМ можно выбрать ту или иную последовательность изложения учебного материала, по усмотрению педагога. Причём, можно выбрать наиболее короткий путь достижения учебной цели, что позволяет сократить время обучения.

Приведенные выше сведения, соображения и рекомендации позволили систематизировать и обобщить основные методологические положения по представлению и контролю знаний в системах образовательного назначения с использованием адаптивных семантических моделей.

1. Для представления и контроля знаний в системах образовательного назначения в качестве основных обеспечивающих эти задачи моделей целесообразно использовать адаптивные семантические модели, учитывая их возможности по более адекватному описанию взаимодействия различных понятий и разделов учебных дисциплин и их адаптации к быстрому изменению содержания этих дисциплин и уровню знаний обучаемых.

2. Адаптивные семантические модели обеспечивают глубокую структуризацию изучаемых понятий и явлений, предметная область может быть представлена наглядно в виде сложных иерархических моделей, которые могут идентифицировать знания обучаемых и их способности, в полной мере использовать достижения современных систем искусственного интеллекта.

3. Основными этапами структуризации знаний в адаптивных семантических моделях в большинстве случаев можно считать:

- определение входных и выходных данных;
- составление словаря терминов;
- выявление объектов и понятий;
- выявление связей между понятиями;
- выявление метапонятий и детализация понятий;
- построение пирамид знаний;
- определение отношений между понятиями;
- определение стратегии принятия решений.

4. При построении пирамид знаний должны использоваться наглядные материалы: рисунки, схемы, диаграммы, графики и др.; уровни пирамиды знаний чаще всего возникают в сознании обучаемых в виде наглядных образов.

5. Одним из основных методов (аппаратов) структуризации знаний является психосемантика, которая позволяет исследовать структуры сознания через реконструкцию индивидуальной системы знаний, выявляя категориальные структуры сознания экспертов.

6. В связи с тем, что свойство наглядности АСМ при использовании многомерных баз знаний и усложнением связей между её объектами теряется целесообразно использовать многоуровневые АСМ, в которых понятия и объекты предметной области располагаются на нескольких уровнях.

7. Основными источниками содержательной информации для образовательных АСМ должны являться:

- государственные образовательные стандарты;
- типовые (примерные) учебные планы;
- рекомендуемая Минобрнаукой России учебная и учебно – методическая литература;
- материалы научно – методических, научно – практических конференций и выставок, обеспечивающих опережающее обучение студентов.

8. Разработку образовательных АСМ по конкретным учебным дисциплинам рекомендуется проводить в следующей последовательности:

- классификация понятий в предметной области;
- выделение общих свойств и признаков присущих каждому уровню учебных понятий;

- выделение отличительных признаков каждого уровня понятий;
- установление связей между понятиями, относящимися к одному уровню;
- выделение межуровневых и межпредметных связей.

9. Контроль знаний обучаемых на основе АСМ должен предполагать смысловую обработку их ответов и сравнение знаний обучаемых с данными образовательной АСМ, при этом может использоваться сеть запроса учебной информации и должна обеспечиваться активизация учебной деятельности студентов и повышение объективности контроля их знаний.

10. Процедура синтеза тестов для контроля знаний обучаемых должна обеспечивать максимально возможную информацию о предметной области в ответах обучаемых при минимально возможном числе тестов.

Библиографический список

[Строгалов, 1998] Строгалов А.С. Компьютерные обучающие системы: некоторые проблемы их разработок. Вузская подготовка в информационном обществе / А.С. Строгалов // М.: РГГУ, 1998.- С. 68 -72.

[Соломатин, 1989] Соломатин Н.М. Информационные семантические системы/ Н.М. Соломатин/-М.: Высшая школа, 1989. - 179 с.

[Шихнабиева, 2008] Шихнабиева Т.Ш. О представлении и контроле знаний в автоматизированных обучающих системах/ Т.Ш. Шихнабиева// Журнал Информатика и образование”, № 10, 2008. - С.55 – 59.

[Шихнабиева, 2008] Шихнабиева Т.Ш. Представление и контроль знаний в автоматизированной обучающей системе “КАСПИЙ”/ Т.Ш. Шихнабиева // Труды V Всероссийского научно – методического симпозиума «Информатизация сельских школ», Анапа, 2008. - С. 66 – 72.

USING ADAPTIVE SEMANTIC MODEL FOR THE PRESENTATION AND THE STRUCTURING OF KNOWLEDGE AMONG THE EDUCATIONAL APPOINTMENTS

Shicknabieva T.SH.

*Institute for Information Technology Education
Russian Academy of Education, Moscow, Russia*

shetoma@mail.ru

For the effective functioning of information systems used in learning systems , it is necessary to solve a number of problems related to the submission , processing and use of knowledge. For storage and effective use of knowledge they need to be modular structuring and representation . The paper suggests some approaches to solving these problems in intelligent systems educational purposes.

Keywords: system of educational purpose, structuring knowledge , modular representation, adaptive semantic models.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

КОНЦЕПЦИЯ ГЕНЕЗИСА САМОРАЗВИВАЮЩЕГОСЯ ИНТЕЛЛЕКТА Ж. ПИАЖЕ: МЕТОДОЛОГИЯ И ПРИНЦИПЫ МОДЕЛИРОВАНИЯ

Старжинский В.П.

*Белорусский национальный технический университет
г. Минск, Республика Беларусь*

Vstarzhinskij@yandex.by

Реконструируются концептуальная и инструментальная модели становления интеллекта ребенка в генетической психологии Ж. Пиаже. Использование методологии точных наук для исследования генезиса саморазвивающегося интеллекта позволило Ж. Пиаже построить теоретическое объяснение этапов развития интеллекта и получить средства их эмпирической верификации.

Ключевые слова: генетическая психология, конструктивная методология, когнитивное развитие, формальная логика, схема действия, конструирование мира культуры.

Введение

При построении объяснительной модели - теории становления интеллекта ребенка, наилучшие результаты в современной науке получены в генетической психологии Ж. Пиаже. В отличие от дескриптивного изложения принятого в гуманитарном знании, Ж.Пиаже поставил задачу вооружить психологию точными методами, свойственными экспериментальному естествознанию. Более пятидесяти лет научных исследований, проведенных Пиаже, показали, что он частично решил эту задачу. После работ Пиаже облик психологии стал другим. Она избрала в качестве своих методов эксперимент, диагностику, с одной стороны, с другой – метод конструктивного обоснования гипотезы и идеальные теоретические схемы, присущие методологии точного естествознания. В психологии стали использоваться новые парадигмы, присущие неклассической и постнеклассической науке, в основе которых лежат новые способы решения когнитивных задач. Психология перешла на этап развития, присущий постнеклассической науке в целом и, в частности, общей теории систем, кибернетике, синергетике.

Вместе с тем отметим, что операциональная концепция интеллекта Ж.Пиаже построена при весьма сильных допущениях: она рассматривала умственное развитие как спонтанный, независимый от обучения процесс, который подчиняется биологическим законам. Кроме того, логика избрана в качестве единственного и основного критерия интеллекта, а уровень

формальных операций репрезентирует высший уровень когнитивного развития. Как справедливо показал Л. С. Выготский Пиаже в своих исследованиях подходил к анализу развития высших психических функций на основе очень сильной абстракции «саморазвивающегося интеллекта», в которой не учитывалась роль социальной и культурной среды. Тем не менее, гениальность Ж.Пиаже состояла в принципиальном решении проблемы генетической эпистемологии посредством верифицируемых и научно обоснованных стадий когнитивного развития. Более того конструктивная методология, принципы и модели исследования, изобретенные Ж.Пиаже имеют непреходящее значение для решения проблемы создания искусственного интеллекта.

1. Конструктивная методология

Конструктивная методология в отличие от когнитивной не просто объясняет наличную ситуацию, но за счет разработки средств и инструментов регламентирует преобразование ее в интересах субъекта. Данный процесс можно представить как проектирование - построение концептуальной и инструментальной моделей. Концептуальная модель представляет собой дескриптивное описание сферы, подлежащей преобразованиям. Это описание в виде построения концептуальной модели является бинарным, ибо фиксирует два состояния — сущее и должное. Сущее — то, что представляет собой объективная, наличная реальность, существующее положение дел. Должное — состояние, которое должно быть

построено по замыслу проектанта. Третьим элементом является построение инструментальной модели, которая призвана устранить расхождение между сущим и должным. Концептуальная модель является теоретическим обоснованием инструментальной в смысле объяснения адекватности применения средств и ресурсов для реализации проекта. Установление расхождения между «сущим» и «должным» есть не что иное, как процедура проблематизации, дальнейшего целеполагания, формулировки возможных способов и приоритетов решения задач. Как правило, система принципов и приоритетов содержит потенциальный способ решения. Концептуальная модель — не просто совокупность понятий, объясняющих какую-либо сущность, существующую сама по себе, как объект. Она также рассматривает и объясняет эту сущность функционально, как устройство, как потенциальный инструмент, средства, и содержит в себе возможность конструктивного решения проблем. Конструктивная методология разработана для решения практико-ориентированных проблем, связанных с изменением наличной ситуации. Если объяснительный подход основывается на понятии и процедуре определения, то в качестве процедуры, обобщающей, синтезирующей объяснительный и деятельностный подход, избирается процедура построения модели. Причем модель рассматривается расширительно как когнитивный артефакт – не только отражение или копия некоторого состояния дел, но и репрезентация будущей практики. По существу, под расширительно рассматриваемой моделью понимается проект.

2. Инструментальная модель

Вначале остановимся на инструментальном моделировании в генетической психологии, затем рассмотрим становление мыслительных структур ребенка с позиций концептуального моделирования. Данный порядок рассмотрения объясняется не только чисто дидактическими причинами. Ж. Пиаже использовал циклический подход и конструктивную методологию одновременно на двух уровнях моделирования. При этом отметим, что исследование (изобретение) и объяснение идут в противоположных направлениях. Аналогично применение конструктивной методологии у ребенка идет в противоположном направлении инженерному проектированию, которое суть объяснение реализуемых идей. Если в инженерном проектировании процесс характеризуется экстериоризацией и направлен «от идеи к внедрению», то становление мыслительной деятельности ребенка связано с интериоризацией и является «внедрением деятельности в интеллект».

1. В объяснении механизмов познания следует ориентироваться на генезис когнитивного развития ребенка. Одним из концептуальных положений генетической психологии является тезис о том, что

когнитивное развитие детей представляет собой не стихийный, а упорядоченный процесс, в котором можно выделить ряд стадий. Основанием для измерения стадийного интеллектуального развития детской психики Пиаже считает принцип изоморфизма онтогенеза и филогенеза.

2. Познание на эмпирическом уровне изоморфно адаптации организма к окружающей среде, на теоретическом – логике формализованных операций. В качестве объяснительных принципов закономерностей мышления ребенка могут быть использованы, с одной стороны, биологические понятия, законы и принципы, объясняющие механизмы адаптации организма к окружающей среде. С другой стороны, с этой целью привлекается методология в виде инструментов решения определенного рода интеллектуальных задач в сфере логики, математики и других формализованных теорий.

3. Основой психического развития является становление интеллекта. Затем было экспериментально доказано, что речь, восприятие и память зависят от уровня развития интеллекта.

4. Этапы развития интеллекта определяются способностью ребенка формировать все более адекватную концептуальную схему проблемной ситуации, которая складывается в процессе адаптации организма к окружающей среде. В процессе когнитивного развития происходит изменение концептуальной схемы. При этом схема вырабатывается в процессе активного взаимодействия субъекта и объекта. Как писал Пиаже, схемы нет ни в субъекте, ни в объекте, она строится в процессе освоения субъектом объекта.

5. Построение схемы осуществляется за счет механизмов ассимиляции и аккомодации. Ассимиляция нового опыта выглядит как использование схемы как шаблона, подгонка под нее проблемной ситуации. Аккомодация, напротив, подвергает изменению имеющуюся схему для того, чтобы она была адекватна новому опыту.

6. От действия к схеме действия. В узком смысле схема – это сенсомоторный эквивалент понятия, инвариант, то общее, что сохраняется в действии при его многократном повторении в разных обстоятельствах. В широком смысле схема действия – это структура на определенном этапе интеллектуального развития. Схема носит динамический характер, поскольку структура – это саморегулирующаяся система, целостность, принцип активности которой отличен от принципов активности частей.

7. Обратимость операций как основание перехода к логике. Принцип динамического равновесия может быть методологически использован для объяснения механизма становления логического мышления, в частности появления чувствительности к противоречиям (формально логическим). Динамическое равновесие в мышлении достигается при

избавлении от противоречия. Критерием устойчивого равновесия Пиаже считал появление обратимости мысли, т.е. такого свойства мышления, при котором, отправляясь от первого действия, ребенок выполняет умственное действие, симметричное по отношению к нему и приходит к исходному состоянию. Симметричное умственное действие или обратимость операций не может возникнуть как обобщение наблюдений за явлениями природы.

В рамках исследований в области искусственного интеллекта существенно то, что Пиаже развернул плоскость исследования интеллекта с социокультурного измерения на формально-логическое моделирование физиологических оснований его становления. В частности, классическая для отечественной психологии точка зрения Л.С. Выготского и А.Р. Лурия утверждала о том, что способность к формально-логическому мышлению зависит от тех общественных отношений, в которые включен человек в процессе своей жизнедеятельности, что изменение этих отношений со временем перестраивает и мышление человека. Почему тогда если изменить условия общественных отношений «ребенка маугли» в период не соответствующий узкому «окну» становления языковых способностей, он не усвоит ни логику, ни язык.

Основная способность к формальным операциям проявляющаяся примерно на стадии от 11 до 15 лет состоит в способности иметь дело с *возможным*, с гипотетическим, а внешнюю действительность воспринимать как частный случай возможного. Познание становится *гипотетико-дедуктивным*. Ребёнок приобретает способность мыслить предложениями и устанавливать формальные отношения между ними (конъюнкция, дизъюнкция, импликация, эквиваленция) и множествами (пересечение, включение, объединение и дополнение). Ребёнок на этой стадии также способен систематически выделить все переменные, существенные для решения задачи, и систематически перебрать все возможные *комбинации* этих переменных.

Исходя из этого, в рамках исследований по искусственному интеллекту, эвристично исследуя механизмы кодовых переходов во внутренней речи ребенка, можно выявить некоторые мета-структуры формирования теоретико-множественных и логических структур.

Можно сказать, что удобным инструментом описания подобных структур является модальная логика. Если в классической логике, вслед за Г.Фреге, значениями повествовательных предложений принято считать абстрактные объекты – истину и ложь (идеализация лежащая в основе так называемой «референциальной» семантики), то в основе семантики модальной логики лежит понятие «возможных миров». Как было показано, в процессе онтогенеза ребенка это понятие предвещает становление классических формально-логических операций. В частности,

логический атоанизм, описывающий только наличные факты, модальные, временные и контрфактические суждения рассматривает как бессмысленные. Однако, большая часть нашей модели мира основана на таких суждениях и искусственная интеллектуальная система решающая референциальную неоднозначность и снимающая эллиптичность высказываний языка должна будет также иметь в основании такую модель мира, основанную на модальной семантике.

8. Конструктивизм как альтернатива отражению. Классическая философия утверждала, что в основе теории познания лежит теория отражения. При этом делалась оговорка, что субъект не просто отражает мир, но и творит его (активность субъекта). Теория Пиаже утверждает противоположное: 1) внешний мир не действует непосредственно на ум субъекта; 2) наши знания о мире – это не просто отражение внешнего мира. Идеи субъекта во многом являются результатом его собственной конструктивной деятельности. Установки и презумпции различного рода влияют и даже искажают «объективный ход событий». Более того, в неклассической философии и эпистемологии существование «объекта самого по себе» является довольно сильной абстракцией и не может быть описано вне познавательных процедур деятельности.

9. Преодоление эгоцентризма – принцип генезиса объективности. Согласно Пиаже, преодолеть эгоцентризм как установку мышления возможно за счет построения системы отношений между различными позициями «я» и «другой» в различных отношениях – гносеологическом и социальном. Построение «различных систем координат» - на языке физики означает способ преодоления абсолютной точки зрения (гносеологические отношения). Осознание собственного «я» представимо как построение двух пространств – как отделение субъекта от объекта и координация собственной точки зрения с другими (социальное отношение).

3. Концептуальная модель в исследовании становления интеллекта ребенка

Прежде всего, отметим весьма заметную условность деления моделирования на концептуальную и инструментальную, поскольку понятия обладают инструментальным смыслом. Известно, что концептуальная модель – совокупность, прежде всего понятий, задающих проблемное поле в аспекте сущего и должного.

1. Понятия целого и части. Данное взаимоотношение в работах Пиаже рассматривается диалектически, поскольку никакое целое нельзя считать изолированной системой. Взаимоотношение целого и части зависит от степени развития структуры, в которую они включены. Если структура целого и части совпадает (общая), то отношения целого и части

уравновешены. Причем состояние равновесия может быть описано понятием гомеостаз и меняется от менее устойчивого к более устойчивому. В логике устойчивое равновесие фиксируется в виде формально-логических отношений, в разуме воспринимается в виде «логической необходимости». Данная модель объясняет механизм развития логических структур за счет формирования устойчивых психических процессов.

2. Понятие операциональности как источника знания. Познание органически связано с операциями, конструктивными действиями по преобразованию объекта. Вначале границы между субъектом и объектом не существует, она образуется вместе с интеллектуальным развитием субъекта. Основная «эпистемологическая проблема» не сводится к тому, как субъект отражает объект. Напротив, она заключается в ответе на вопрос, каким образом субъект становится способным познавать объект все более адекватно. Другими словами, каким образом субъект познает объект в предметных формах практики, т.е. все более объективно.

3. Субъект у Пиаже – это любой живой организм, наделенный функциональной активностью приспособления, на основе которой происходит структурирование окружения. Субъекту присущи две основные функции: организация и адаптация. Организация выступает в форме структуры, адаптация выражает динамический аспект структуры и состоит из равновесия процессов ассимиляции и аккомодации.

4. Идеализации детского эгоцентризма. Эгоцентризм ребенка связан с отсутствием обратимости мысли (операций). Эгоцентризм ребенка проявляется по – разному, и, в частности, в эгоцентричности речи, когда ребенок говорит лишь «со своей точки зрения». Ребенок работает с абстракцией собеседника – «другого». У ребенка нет заботы о том, слышат ли его, понимают ли. Идеализация «другого» у ребенка основывается на предположении «говорить означает быть услышанным». Кроме того, мышление ребенка содержит другие установки, презумпции, которые он не осознает. Их общая характеристика – спонтанность, отсутствие рефлексии, абсолютность его позиции, а также отсутствие у вещей и мира в целом второго плана, скрытой, не видимой сущности вещей. Основной опыт ребенка – визуальный. Ему присуще отсутствие понимания относительности познания мира и координации точек зрения.

5. Понятие социальность. Два способа конструирования социального мира. Пиаже показывает, что развитие самосознания (т.е. знаний о себе) осуществляется за счет смены умственных позиций, которые, в свою очередь, развиваются под влиянием социальных взаимоотношений субъектов. Социальные отношения характеризуются двумя крайними типами: отношения конкуренции и отношения кооперации,

которые формируют у ребенка систему нормативных правил соответственно обязательного и творческого плана. Лишь последние способствуют установлению кооперации среди детей. Пиаже обосновывает связь социального и логического через формирование потребности приспосабливаться к другому. Столкновение собственной мысли и чужой вызывает сомнение и необходимость доказательства.

6. Понятие структуры определяется через соотношение логики и операциональных схем деятельности. Одним из методологических принципов методологии Пиаже является анализ операций деятельности, инвариантные свойства которых и позволяют выявить необходимые структуры мышления. Речь идет, прежде всего, об обратимости операций деятельности, в которых результаты деятельности решения задач затем используются в решении других задач как средства. На наш взгляд, речь идет о методологическом эффекте, который получил название «оборачивание метода». Общие результаты, которые выявлены в процессе деятельности, приобретают нормативные свойства регулятивов деятельности, т.е. выступают в функции методов, регламентирующих деятельность. Пиаже рассматривает обратимость операций по отношению к эффекту сохранения материала (вещества), веса, а затем объема. Ребенок использует, по выражению Ж.Пиаже, «пре-логику», которая, основывается на логике операционной деятельности инвариантов мышления.

7. Понятие объективности. Реализм. Вначале в деятельности ребенка 1-2 лет нет процесса объективизации, т.е. различения и отделения внешнего мира и себя, нет субъект-объектного отношения. Представление о мире, как объективной реальности, существующей независимо от сознания человека, ребенок усваивает через интериоризацию социального опыта. Инвариантные структуры предметной деятельности одновременно подкрепляются у него инвариантами, тем общим, что есть в разных точках зрения. Он дает онтологическую интерпретацию вопросу: в чем инвариантны разные сознания-представления-образы объективного мира. И наоборот, абсолютность знания, синкретичность субъект-объектного отношения закрепляется в непосредственном восприятии как единственно возможным.

8. Понятие структура и механизм ее формирования. В концепции становления интеллекта ребенка роль основного конструктивного элемента играют структуры, как узловые точки и стадии развития когнитивных функций интеллекта-мышления. При этом структура у Пиаже обладает одновременно взаимно противоречивыми свойствами. С одной стороны, она результат развития когнитивных функций интеллекта и в этом смысле представляет

собой законченную целостность. С другой стороны, она содержит в себе возможности дальнейшего развития в виде создания новых структур. Таким образом, развитие интеллекта предстает как реализация потенциала самой структуры в виде системы постоянных преобразований, которые также подвержены преобразованиям.

При моделировании генезиса, развития, становления когнитивных функций интеллекта основную задачу составляют модели (описания) процесса создания новых структур, ибо он и содержит «тайну» развития мышления. Развитие поглощает, снимает всю предварительную работу в своем результате. И если иметь дело только с результатом, то остается только восхищаться, либо недоумевать.

9. Базовыми понятиями Ж.Пиаже в его концепции являются такие понятия генезиса и функции. Генезис предполагает структуру, поскольку он никогда не начинается с бесструктурной точки. Генезис – это развитие структуры от простой к сложной. Одновременно развитие структур осуществляется за счет формирования функций. Структура, развиваясь через формирование функций, осуществляет генезис. Оформление структуры представляет собой организацию.

Пиаже приводит онтологическое обоснование формирования новой структуры, которая имеет логическую природу. Ребенок испытывает, переживает чувство необходимости, связанное с рождением новой структуры. Аналогичную роль у взрослых играет эмпирическая интерпретация, где данные наблюдения можно проверить опытным путем. При этом вектор эмпирической интерпретации направлен вниз на эмпирически воспринимаемый мир. Напротив, появление новых структур «обосновывается» внутри субъекта посредством чувства «необходимости».

10. Пиаже интерпретирует динамические свойства структуры, используя понятие равновесия, которое, возникнув в физике, стало подвергаться межкультурной трансляции и использоваться в биологии (гомеостаз), общей теории систем, кибернетике, синергетике и т.д. Функционирование структуры можно рассматривать как стадию равновесия в генезисе. Такое функционирование будет вести дальше к строительству других структур.

11. Уравновешенная система – это еще одно понятие, составляющее концептуальную модель генезиса интеллектуального развития. Пиаже показывает, что генезис мышления, как адаптация ребенка к новым ситуациям идет через попытки ассимиляции и аккомодации, которые могут быть иногда ошибочными; однако в конечном итоге заканчивающиеся рождением новой структуры. Уравновешенная система характеризуется не статическим, а динамическим балансом, в основе которого лежит регуляция поведения. Данное

состояние можно описать на языке физики как «смещенное равновесие», балансирующее и учитывающее положительные и отрицательные отклонения от линии равновесия (статического). Данное состояние равновесия можно назвать «возбужденным». Другими словами, «уравновешенная система» – это не только результат, но и процесс, ведущий к равновесию. Пиаже также рассматривает ситуацию, противоположную генезису – регресс, когда не создаются новые структуры, не восстанавливается равновесие.

12. Отметим также еще одно понятие, играющее фундаментальную роль в концепции Пиаже – группа. Как вытекает из контекста исследований, а также определения данного самим автором, группа – абстрактная структура, состоящая из совокупности элементов, включающая операции с этими элементами так, что при ее осуществлении не нарушаются свойства композиции, ассоциативности, тождественности и обратимости.

13. Следует определить одно из фундаментальных понятий концепции Ж.Пиаже – понятие генезис, которое задает общую направленность исследования и обозначается как подход. Понятие генезиса используется в широком и узком смысле слова. Узкий или собственный смысл понятия генезис используется в биологии и означает саморазвитие особи. В узком смысле термин «генетическая психология» относится к индивидуальному развитию или онтогенезу. В широком смысле генезис означает формирование каких-либо функций по отношению к любой онтологии.

14. Соотношение генетической психологии и генетической эпистемологии. По мнению Пиаже, генетическая эпистемология объясняет процесс научного познания на основе его истории, социогенеза. Второй объяснительный принцип – истоки понятий и операций, на которые научное познание опирается. Эти истоки могут быть выявлены в детской психологии, психологии интеллектуального развития ребенка. Генезис представляет собой модель развития интеллекта в проблемной ситуации субъекта через «самопорождение» структур, которые эксплицируются как процесс строительства их субъектом.

15. Обратимся к понятию интеллект, интеллектуальное развитие. Пиаже полагает его в качестве презумпции и определяет конструктивно, т.е. не описательно, посредством набора понятий, а операционально, т.е. как средство решения определенных задач. Класс таких задач определяется процессом адаптации ребенка к новым ситуациям, через непрерывное строительство структур.

4. Анализ способов построения теоретической и экспериментальной моделей в генетической психологии

Базовые компоненты культуры, согласно Пиаже, представляют собой конструктивные элементы, фактически «несущие конструкции здания», которое называется детская картина мира. При этом базовые компоненты культуры носят амбивалентный характер. С одной стороны, это реперные точки культуротворчества ребенка по конструированию онтологии или картины мира. С другой стороны, они представляют собой принципы онтологии, выражающиеся в фундаментальных категориях бытия: мир, вещь, причинность, пространство, время и др. Естественно, что для обеспечения единства и целостности такой амбивалентности необходимо одно условие: базовые компоненты культуры должны быть принципами культуротворчества или, другими словами, иметь конструктивную, операциональную природу. Например, категория мир, вещь существующая «сама по себе» объективно формируется у ребенка особым образом, через систему операций и процедур деятельности по установлению принципа объективности.

В каком смысле исследование Ж.Пиаже является теорией? Теория, как правило, включает в себя: 1) эмпирическую основу в виде совокупности эмпирических фактов, полученных в эксперименте и нуждающихся в теоретическом объяснении; 2) исходную теоретическую основу в виде допущений, гипотез, описывающих идеализированный объект; 3) логическую основу в виде правил вывода и доказательств; 4) совокупность теоретических утверждений с их доказательствами. [В.Степин, 2003]

Методологический центр теории – идеализированный объект, теоретическая модель – конструкция, в которой фиксируются основные свойства объекта в виде гипотетических допущений и идеализаций. Идеализированный объект теории выступает как конструктивное средство развертывания всей системы теории. Развертывание происходит в соответствии с методом восхождения от абстрактного к конкретному.

Следует описать свойства такого абстрактного идеального объекта, как саморазвивающийся интеллект, который является ядром теории. Отметим, что, в отличие от упоминавшихся ранее психологов, которые усматривают в абстрактном объекте недостатки концепции Ж.Пиаже, мы рассматриваем этот объект через призму ограничений, т.е. свойств, которыми наделяется идеальный объект, и от которых он абстрагируется.

Во-первых, саморазвивающийся интеллект – это не целостное психофизиологическое существо, а только познающий ум, который абстрагируется от процессов, связанных с биологическими факторами.

Во-вторых, саморазвивающийся интеллект, это абстракция чисто рациональная, исключая мотивы, аффекты, а также специальное научение. Имеет место спонтанное обучение, или самообучение.

Во-третьих, саморазвивающийся интеллект присущ отдельному индивиду по отношению к окружающему миру. Его социализация как операциональный конструктивный механизм вводится как развитие теории из объекта (абстрактного). Другими словами, саморазвивающийся интеллект понятие тождественное становлению сознания.

Напомним что, концептуальная модель выступает в качестве теоретического обоснования инструментальной, ибо объясняет основной замысел или идею как переход от сущего к должному. При этом, как уже говорилось в становлении детского мышления имеет место инверсия понятий и действий в отличие от взрослого интеллекта и проектирования.

Ж.Пиаже разработал учение, в котором органически сочетается теория, как описание общих закономерностей развития интеллекта, а также опытно-экспериментальная деятельность. Благодаря этому, с одной стороны, теория получила возможность верификации, с другой, эксперимент мог быть использован в качестве диагностического для исследования феноменов психологической реальности, в частности, диагностики уровней интеллектуального развития. Именно ему принадлежит заслуга в разработке проблемы интеллектуального развития ребенка в аспекте истоков научного познания. Другими словами, великий психолог смог ответить на вопросы: как формируется мышление человека, каков механизм познания человеком мира, в чем секрет, «тайна» человеческого сознания. При этом решение этой сверхзадачи требует переформулировки вопросов, ибо правильно сформулированный вопрос уже содержит половину ответа. Пиаже одним из первых понял в исследовании генезиса интеллекта два обстоятельства. Во-первых, задавать вопросы и исследовать необходимо процесс, мыслительную деятельность, а не ее результат – продукт. Во-вторых, истоки научного познания кроются в схемах деятельности, операций, которые вначале являются внешними по отношению к психике, а затем становятся принадлежащими самой психике в виде ее атрибута.

Заметим, что предметом настоящей статьи не является интеллектуальное развитие ребенка, а конкретная методология, вернее ее применение для решения определенных задач, то есть экспликация элементов концептуального и инструментального моделирования, осуществленного Пиаже. Тем более, по мнению многих исследователей, эта задача – анализ механизмов построения теории, выявление методологии научного исследования применяемой Пиаже, не решена в современной

психолого-педагогической и философской литературе.

Несомненной заслугой Пиаже является разработка не только проблем психологических, но и методологических, а также трансграничных. Имеется в виду трансляция методов исследования, разработанных в биологии, философии, логики, математике в психологию при помощи которых были построены эффективные теоретические модели, объясняющие механизмы мыслительной деятельности. Одной из конструктивных идей Пиаже было утверждение о том, что логика, как образец правильного (истинного) мышления не является некой константой мышления, кантовской *a-priori*, а есть результат постепенного развития мышления, начиная с примитивных дологических форм и заканчивая логического формализмами. Онтологическое развитие логики есть не только задача психологии, но и ключ к объяснению и построению эпистемологии и теории искусственного интеллекта. Пиаже предположил, что существует соответствие между психическими процессами и логическими операциями и поставил задачу исследовать логические операции посредством выявления механизмов психических процессов им соответствующих. Возникновение стабильных целостных структур интеллекта должно соответствовать определенным логическим формам. Такова была первоначальная идея-гипотеза, которую следовало обосновать конструктивно. Метафора «эмбриология» интеллекта и отражает основную идею автора.

Исследования Пиаже генезиса числа, количества, пространства, времени, движения и др. позволили увидеть операционально целостные структуры интеллекта в виде группировок. Данное понятие Пиаже вводит в психологию, придав ему расширительный смысл, в отличие от понятия группа, которое использовалось в математической логике. Группировка – это любая классификация в своей целостности, будь то серия объектов, определенным образом упорядоченных, система родословных связей, либо шкала ценностей. Согласно Пиаже, группировка – это операциональное определение понятия или суждения как единицы мысли. Процесс построения группировок непосредственно не наблюдаем (как сущность), «наблюдаем» только результат – группа. Именно поэтому Ж.Пиаже строит теоретическую модель в виде дедуктивной теории группировок, при помощи которой производит интерпретацию эмпирических фактов. Следует заметить, что в данном контексте термины эмпирический и теоретический используются в смысле предметного языка и метаязыка. Группировка как абстрактный теоретический объект представляет собой закрытую обратимую систему, в которой все операции объединены в одно целое и подчиняются пяти формальным законам (отношениям), которые можно использовать в качестве критериев сформированности группировки. Таким способом

было построено теоретическое объяснение сформированности определенных логических операций на основе установления их соответствия (интерпретации) с помощью аксиоматических моделей, взятых из логики. Интеллектуальные операции осуществляются в форме образования целостных структур. Целостность достигается за счет установления равновесия, к которому стремится эволюция.

И наконец, рассмотрим проблему психологии как точной науки. Пиаже использует два критерия точности: эмпирический – верифицируемость, опытная подтверждаемость фактов и теоретический, как способность быть описанной при помощи формально-логического, математизированного языка и объясненной на уровне онтологической интерпретации. Психологическое исследование Пиаже удовлетворяет образцам точного естествознания. С одной стороны, он осуществляет эмпирическую деятельность – устанавливает и классифицирует факты, затем выводит эмпирические закономерности. С другой стороны, он конструирует теорию, при помощи которой объясняет причины и механизмы этих закономерностей. Метод конструктивного обоснования гипотезы Пиаже применяет следующим образом. Операторная логика, разработанная как формализованная теория – ключ к пониманию операций мышления. Абстрактный, идеальный объект, который имеет теоретический статус – группировка может рассматриваться как теоретическая схема – модель этапов интеллектуального развития. Данный подход дает возможность реализовать программу «точного» исследования. С одной стороны, процесс развития психики можно исследовать эмпирически с помощью наблюдения или мысленного эксперимента, фиксирующего этапы сформированности группировок. С другой стороны, этот процесс можно описать теоретически, путем дедуктивного, конструктивного обоснования этапов формирования логических операций. Группировка как единица мысли является основанием формирования логических операций классификации, сериации и др.

Таким образом, дедуктивная теория группировок Пиаже соответствует идеальным фундаментальным схемам физической теории, которая используется для интерпретации эмпирических фактов. Методология – это не просто определенный слой знаний, но и нормативная составляющая, которая регламентирует и обосновывает деятельность через рефлексию ее структуры. Классическая наука и образование основывались на методологии, которая ограничивалась в регламентации только когнитивной, познавательной сферой в рамках субъект-объектного отношения. Неклассическая, или конструктивная методология, своим объектом имеет не только познавательную деятельность, но

весь процесс культуротворчества – этический, эстетический, технический, эмоционально-волевой и др. способы освоения человеком мира.

Ребенок в своем генезисе сознания пользуется конструктивной методологией как наиболее адекватной процессу культуротворчества – интегративному способу освоения ребенком мира. Как известно, культура может рассматриваться амбивалентно: виде результата – духовная культура и материальные артефакты, а также в виде процесса, «живой культуры», или культуротворчества. Подобная амбивалентность культуры требует выстраивать процесс формирования базовых компонентов культуры относительно застывшей культуры – продукта, а также относительно культуротворчества. Базовые компоненты культуры, строго говоря, не относятся ни к объекту, ни к субъекту, а представляют собой принципиальные схемы, операции деятельности по конструированию детской картины мира и ее онтологических принципов (принципы объективности, причинности, пространства, времени и др.). Именно эти принципы создают целостность картины мира, т.к. с одной стороны, регламентируют операции деятельности, а с другой – характеризуют «мир сам по себе», т.е. в предметных структурах практики.

«Объективность» означает, что содержание знания, если оно претендует на истинность, не зависит от субъекта. Понятие «рациональность» (научная рациональность) возникает в результате рефлексии над средствами, методами, операциями познавательной деятельности. Научная рациональность – это способ осуществления познавательной деятельности, который основан на рефлексии средств познания, а также идеалов и норм познания.

Понятие объективности существования мира формируется через операциональные процедуры, которые подчиняются (а затем преодолеваются) установкам, презумпциям мышления ребенка: наивный реализм, эгоцентризм, абсолютность и др. Результатом операциональных процедур являются структуры как узловые точки, теоретические модели, идеализированные объекты формирования когнитивных функций ребенка. В определенном смысле формирование этих структур можно рассматривать в качестве теоретических моделей формирования базовых компонентов культуры у дошкольников и мл. школьников. Механизм формирования базовых компонентов культуротворчества ребенка может быть построен в виде теоретической модели, включающей в себя следующие понятия и принципы: реализм, объективность, структура, механизм формирования структур (ассимиляция, аккомодация, адаптация), генезис и функция, обратимость операций, уравновешенная система и др.

Заключение

1. На языке деятельностной методологии, которая наиболее адекватна когнитивной

деятельности, субъектом формирования базовых компонентов культуры является ребенок. Он одновременно является субъектом культуротворчества, т.к. доминирующим в процессе формирования базовых и вторичных (последующих) компонентов культуры является конструирование мира культуры.

2. Теоретической моделью функционирования базовых компонентов культуры в генетической психологии Ж.Пиаже являются абстрактные идеализированные объекты – концептуальная схема, обратимость операций, группировка и др.

3. Данные теоретические конструкторы выражают операциональный смысл понятий, суждений, силлогизмов как единиц мысли, находящихся выражение в соответствующей завершенной, целостной классификации.

4. Использование методологии точного знания для исследования генезиса саморазвивающегося интеллекта дало возможность построить теоретическое объяснение этапов развития интеллекта и получить средство их эмпирической диагностики.

Библиографический список

- Башляр, 1987] Г. Башляр. Новый рационализм / Г. Башляр // М., 1987,
[Кун, 2001] Кун, Т. Структура научных революций / Т. Кун. // М., АСТ, 2001.
[Пиаже, 1984] Ж. Пиаже. Генетический аспект языка и мышления / Ж. Пиаже. // М., 1984.
[Пиаже, 1994] Пиаже Ж. Избранные психологические труды. / Ж. Пиаже. // М., 1994.
[Пиаже, 1994] Пиаже Ж. Речь и мышление ребенка. / Ж. Пиаже. // М., 1994.
[Пиаже, 1983] Пиаже Ж. Схемы действия и усвоение языка / Ж. Пиаже. // М., 1983.
[Пиаже, 2004] Пиаже Ж. Генетическая эпистемология / Ж. Пиаже. // СПб.: Питер, 2004.
[Поппер, 2005] Поппер, К. Логика научного исследования / К. Поппер. // М., «Республика», 2005.
[Саймон, 1972] Саймон, Г. Науки об искусственном / Г. Саймон. // М., 1972
[Старжинский, 2008] Старжинский, В.П. Конструктивная методология как основа для экспликации базовых компонентов культуротворчества в генетической психологии Ж. Пиаже. Образование и педагогическая наука. / Труды национального института образования. Серия 5. Дошкольное и общее начальное образование. Выпуск 2. Концепции, модели, методики. Старжинский, В.П. // Минск, 2008
[Степин, 2003] Степин, В.С. Теоретическое знание / В.С. Степин. // М.: Прогресс-Традиция, 2003.

CONCEPT GENESIS SPONTANEOUS INTELLIGENCE J. PIAGET'S METHODOLOGY AND PRINCIPLES of MODELING

VP Starzhinsky

*Belarusian national technical University,
Minsk, Republic of Belarus
Vstarzhinskij@yandex.by*

Abstract: the Reconstructed conceptual and instrumental model of the formation of a child's intellect in the genetic psychology of Jean piaget. Using the methodology of the exact Sciences for the research of the Genesis of the self-developing intelligence allowed J. piaget build a theoretical explanation of the stages of development of the intellect and get their means of empirical verification.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.432. 4

ПРОЕКТИРОВАНИЕ ЭЛЕКТРОННЫХ ОБУЧАЮЩИХ СИСТЕМ

Ниязова Р.С. *, Алсеитова А.Т. *

** Евразийский национальный университет им. Л.Н.Гумилева,
г. Астана, Республика Казахстан*

Rozamgul@list.ru

ainura1604@mail.ru

В работе описан процесс создания электронных учебных изданий, построены алгоритмы их работы, а также приведены программы их реализации.

Ключевые слова: информатизация образование; электронные учебные издания; компьютерное тестирование.

Введение

Информатизация образования предполагает наличие электронных учебных изданий (ЭУИ) по всем изучаемым дисциплинам. При этом разработчики ЭУИ используют различные технологии, создают разные интерфейсы для пользователей, предлагают разные требования к составу, структуре, эргономике и другим элементам. Это порождает все новые несовершенные ЭУИ, которые не всегда пригодны к использованию в учебном процессе.

Электронное учебное издание: Электронное издание, предназначенное для автоматизации обучения и контроля знаний, и соответствующее учебному курсу или отдельным его частям, а также позволяющее выбрать траекторию обучения и обеспечивающее различные виды учебных работ [СТ РК 34.017-2005, 2005].

1. Проектирование электронных учебных изданий

Достоинствами электронных учебных изданий являются: во-первых, их мобильность, во-вторых, доступность связи с развитием компьютерных сетей, в-третьих, адекватность уровню развития современных научных знаний. С другой стороны, создание учебных изданий способствует также решению и такой проблемы, как постоянное обновление информационного материала. В них также может содержаться большое количество упражнений и примеров, подробно иллюстрироваться в динамике различные виды информации. Кроме того, при помощи электронных

учебников осуществляется контроль знаний - компьютерное тестирование.

На основании проведенного обзора мирового опыта создания ЭУИ были сделаны следующие выводы. В общем случае любой ЭУИ должно включать в себя два основных элемента: средства получения информации и средства контроля знания по электронным изданиям.

Таким образом, общую структуру работы ЭУИ можно представить в виде алгоритма, приведенного на рис. 1.

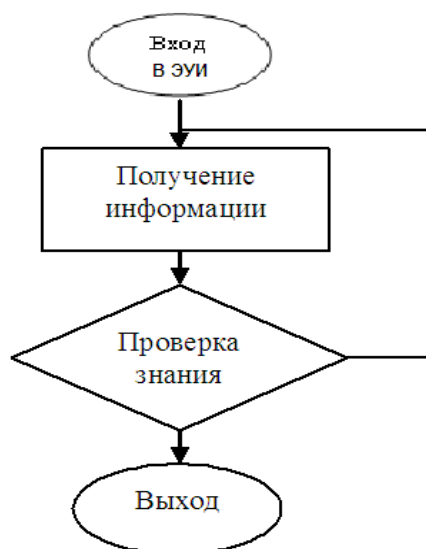


Рисунок 1. Алгоритм работы электронных учебных изданий

В работе ЭУИ должны быть реализованы следующие функции: регистрация пользователей; защита данных; навигация; организация просмотра

содержания; определение траектории обучения; обучение и контроль знаний; тестирование; статистический учет.

При получении информация используется система навигации по электронному изданию, она обычно состоит из частей, изображенных на рис. 2. Рекомендуется снабжать каждую страницу электронного издания удобными для пользователя средствами навигации. Эти команды также размещаются в системном меню.

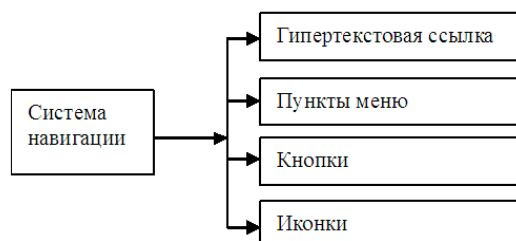


Рисунок 2. Система навигации

Пункт «получение информации». Объектом ЭУИ должна быть цифровая, текстовая, графическая, аудио, видео и другая обучающая информация, представляющая собой совокупность научно обоснованных фактов, утверждений и правил, а также свойств и отношений объектов, явлений и процессов, изучаемых в рамках данного учебного курса. В состав ЭУИ должны входить пункты: титул, оглавление, контент, утилиты, помощник и документация. Сама концепция мультимедиа предусматривает подразделения этого пункта. Каждый из этих пунктов алгоритма может распасться на сложные подалгоритмы.

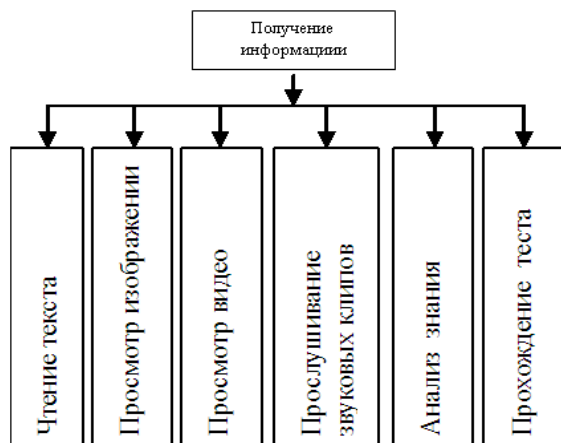


Рисунок 3. Способы получения информации в электронных учебных изданиях

2. Компьютерное тестирование

Компьютерное тестирование является одним из основных направлений внедрения информатизации образования и является важным элементом ЭУИ, поэтому рассмотрим вопрос программирования

тестирующего блока в электронных учебных изданиях. Обычная тестирующая программа состоит из базы данных, содержащей вопросы (текст, рисунки, видео и др. формы представления информации в мультимедиа среде) и эталоны правильных ответов, и процедур анализа ответов тестируемого и выставления оценок.

Рассмотрим разновидности компьютерных тестов: без ограничения времени, с ограничением времени на весь тест, с ограничением времени на каждый вопрос в отдельности. Также могут встречаться тесты, обладающие сразу несколькими этими свойствами. Кроме того, способы ответов на вопросы теста бывают следующих видов: выбрать один ответ из нескольких вариантов, выбрать несколько ответов из нескольких вариантов, ввести ответ вручную.

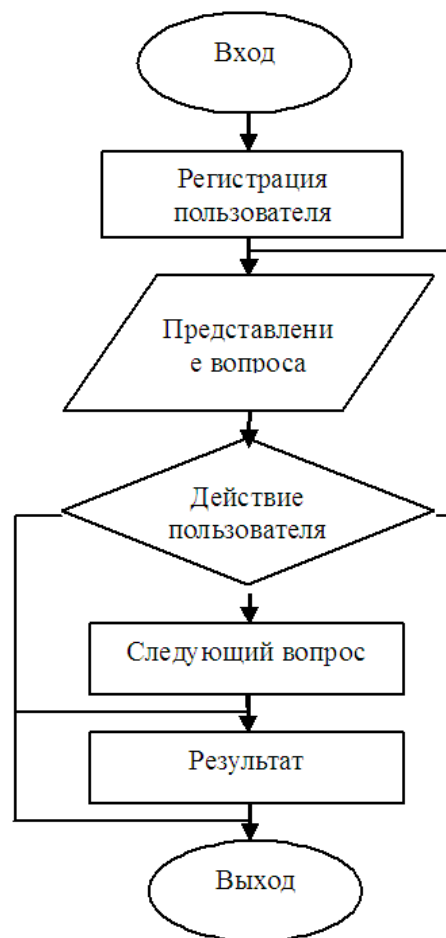


Рисунок 4. Алгоритм работы тестирующей системы электронных учебных изданий а) с неограниченным временем тестирования.

Функция обучения и контроля знаний должна обеспечить возможности изучения теории, интерактивного выполнения задания, отвечать на вопросы и пройти текущий, промежуточный, рубежный и итоговый контроль знаний в заданной траектории обучения. При этом в случае недостаточного количества правильных ответов на вопросы, обучаемый должен повторить прохождение текущей единицы обучения в ЭУИ. Обучение может включать функцию имитационного

компьютерного моделирования. Обучение может прерываться и продолжаться с места прерывания.

А также функция тестирования должна обеспечить средствами автоматизированного тестирования по всему ЭУИ с возможностями фиксации или не фиксации результатов и предусмотреть настраиваемое ограничение времени прохождения тестов. Вопросы с вариантами ответов предоставляются посредством случайной выборки из базы данных вопросов по учебному курсу. Результат тестирования выводится на экран. При неудовлетворительном результате тестирования должен быть предусмотрен свободный переход к любому вопросу, ответ к которому дан неверно.

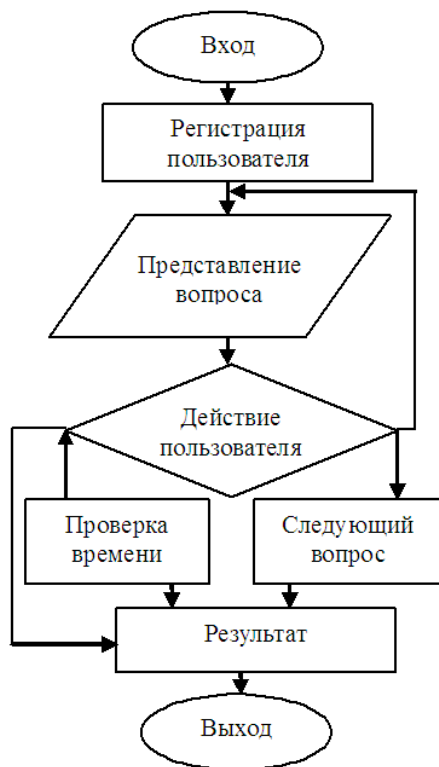


Рисунок 4. Алгоритм работы тестирующей системы электронных учебных изданиях б) с ограничением времени на вес тест.

Опыт создания ЭУИ показывает, что необходим переход к более производительной технологии создания ЭУИ различных типов. Причем, преимущественно, ЭУИ должны создаваться с помощью какой-либо инструментальной системы.

Применение инструментальной системы создания ЭУИ позволяет сократить трудоемкость и сроки разработки, а также дает возможность непосредственной работы с компьютером для преподавателей, которые не являются

3. Требования к структуре содержания электронных учебных изданий

Контент ЭУИ должно быть разбито на

трехуровневые семантические единицы обучения: уровень 1 - модули, уровень 2 – блоки, уровень 3 - уроки.

высококвалифицированными программистами [Omarbekova, 2012 a].

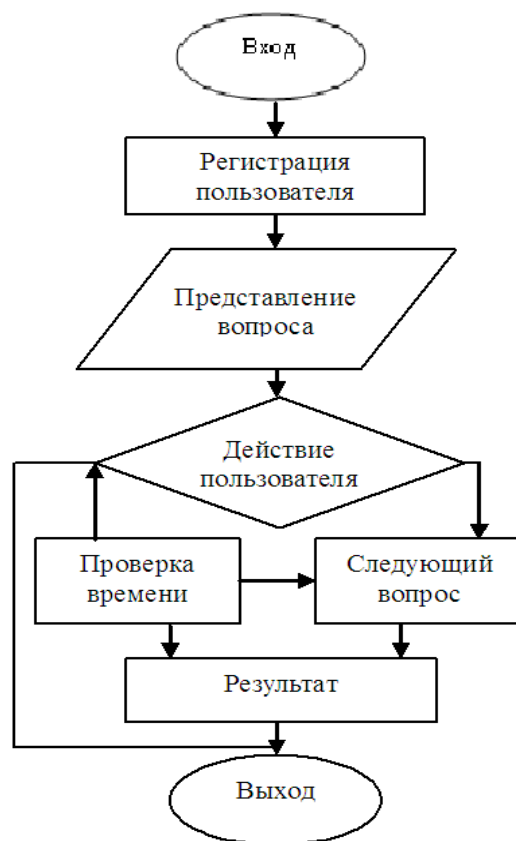


Рисунок 4. Алгоритм работы тестирующей системы электронных учебных изданиях в) с ограничением времени на каждый вопрос в отдельности

На рис. 4 представлены алгоритмы работы тестирующих систем различных типов. Пункт «Регистрация» заключается в запросе данных тестируемого для последующего генерирования базы данных тестируемых и аттестационного листа. Пункт «Представление вопроса» предназначен для ознакомления тестируемого с вопросом (вывод текстовой информации, рисунков и т.д.). «Действие пользователя» - это ожидание события. «Результат» - вывод на экран, в файл или на принтер аттестационного листа тестируемого.

Практика использования электронных учебных изданий показывают, что студенты качественно усваивают изложенный материал, о чем свидетельствуют результаты тестирования.

- Модуль является крупной синтаксической, семантической и прагматической единицей обучения и состоит из последовательности логически связанных блоков, имеющих смысловые связи по нарастанию объема и содержания информации от блока к блоку.

- Блок является средней синтаксической, семантической и прагматической единицей

обучения и состоит из последовательности логически связанных уроков, имеющих смысловые связи по нарастанию объема и содержания информации от урока к уроку.

Урок является минимальной синтаксической, семантической и прагматической единицей обучения и состоит из нескольких элементов обучения. Обязательными элементами обучения в уроке должны быть теоретический материал, примеры, задания, вопросы-ответы и тесты. Необязательными элементами обучения может быть справочник, графика, аудио и видео, которые позволяют осознать, осмыслить, запомнить учебную информацию и обеспечить информационную плотность [Шарипбаев, 2005].

Заключение

Применение электронных учебных изданий позволит студенту качественно усваивать изложенный материал. Электронное издание, записывается на носитель информации, рассчитан на использование с помощью электронных технических устройств, представляет собой электронный документ (группа электронных документов), проходит через редакционно-издательскую обработку, предназначен для распространения в неизменном виде, имеет выходные сведения. Структура и содержание электронных изданий зависит от целей его использования. Он и репетитор, тренажер и самоучитель.

Таким образом, развитие информационных технологий дает широкую возможность для изобретения новых методов методик в образовании и тем самым повысить его качество.

Библиографический список

[Петренко А.И., 1994] Петренко А.И. Мультимедиа. Киев: Торгово-издательское бюро BVH, 1994.

[СТ РК 34.017-2005, 2005] Информационные технологии. Электронное издание. Электронное учебное издание// Государственный стандарт Республики Казахстан, 2005. С 5-8.

[Шарипбаев А.А., 2005] Шарипбаев А.А. Автоматизация создания электронных учебных изданий / Шарипбаев А.А., Омарбекова А.С. // Вестник ЕНУ им. Л. Гумилева, 2005, С.119-125.

[Шарипбаев А.А., 2012] Шарипбаев А.А. Семантическая модель электронного учебного издания / Шарипбаев А.А., Омарбекова А.С. // Спецвыпуск Вестник ЕНУ им. Л. Гумилева, 2012, С. 432-435.

[Шрейдер Ю.А., 1987] Шрейдер Ю.А. Экспертные системы: их возможности в обучении// Вестник высшей школы. 1987. №2. С. 14.

[Gruber T.R., 1991] Gruber T.R. The role of common ontology in achieving sharable, reusable knowledge bases // Principles of Knowledge Representation and Reasoning. Proceedings of the Second International// Conference. J.A. Allen, R. Fikes, E. Sandewell – eds. Morgan Kaufmann– 1991– P.601-602.

[Omarbekova A.C., 2012a] A.S.Omarbekova. Automatization create electronic learning. Journal of International Scientific Publications Education Alternatives / A.S. Omarbekova , A.Seifullina. // Volume 10, Part 1, ISSN 1313-2571. Bulgaria, 2012, P.242-250.

DESIGNING E-LEARNING SYSTEMS

Niyazova R.S. *, Alseitova A.A. *

* L.N. Gumilyov Eurasian University, Astana,
Republic of Kazakhstan

Rozamgul@list.ru

Ainura1604@mail.ru

Introduction

This paper describes the process of creating electronic textbooks, built algorithms for their work, and provides programs for their implementation.

Main Part

Application development system allows the creation Eui reduce the complexity and development time, and enables direct work with the computer for teachers who are not highly qualified programmers.

Conclusion

The use of e-books will allow the student to assimilate the material described qualitatively. The electronic edition is written to the storage medium , designed for use with electronic technical devices , is an electronic document (the group of electronic documents), passes through the editorial and publishing process , be distributed in an unchanged form , has output information. Thus, the development of information technology give sample opportunity for the invention of new methods and techniques in education thus improving its quality.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822

СЕМАНТИКА И ОНТОЛОГИИ В ЕСТЕСТВЕННЫХ И ИСКУССТВЕННЫХ СИСТЕМАХ

Сторож В.В.

Донецкий государственный технический университет,

г. Донецк, Украина

ws52@mail.ru

В работе рассмотрено влияние моделей семантики и созданных на их базе онтологий на возможности современных интеллектуальных систем. Показано, что широко распространенные в настоящее время модели семантики имеют критически важные ограничения по поиску, интеграции и анализу информации. Предложено использовать интеллектуальные системы с моделями семантики, в большей степени учитывающими специфику анализа информации человеком. Сформулирован ряд требований к составу и содержанию таких систем.

Ключевые слова: требования к моделям семантики; сущность онтологий; учет принципов человеческого мышления.

Введение

Большие надежды в области поиска, интеграции и анализа информации в последние годы связаны с онтологиями, как одним из направлений искусственного интеллекта (ИИ). При этом особенности реализации и возможности онтологий существенно зависят от модели семантики, принятой в данной онтологии [Colomb, 2007]. В то же время используемые в ИИ модели семантики существенно беднее семантики человека [Поспелов, 1989; Кузнецов, 2012; Васильев, 1990; Попова и др., 2007]. Очевидно, что неадекватность моделей семантики семантике человека должна привести к каким-то ограничениям на использование онтологий. Каковы же эти ограничения? По-видимому, этот вопрос никем систематически не исследовался.

Существует не только теоретическая, но и настоятельная практическая потребность в понимании того, какие ограничения накладывают используемые семантические модели на онтологии, и как можно снять, либо хотя бы ослабить эти ограничения. В качестве примера приведем Интернет-технологии Semantic Web, основанную на онтологиях, и широко разрекламированную Тимом Тернерсом-Ли в 2001 году [Terners-Lee et al., 2001]. Интернет сообщество и ведущие разработчики программного обеспечения с энтузиазмом подхватили эту технологию и начали внедрять ее в работу Интернет и в программные продукты. Но достижения пока довольно скромные, что признал и сам автор Semantic Web [Shadbolt et al., 2006].

Радикальные идеи, как исправить ситуацию, в настоящее время отсутствуют.

Было бы наивно полагать, что поднятые выше вопросы можно решить и осветить в одной публикации. Целью данной работы является скорее не развернутый ответ на поставленные вопросы, а междисциплинарное рассмотрение использования семантики в онтологиях, и на этой основе более четкая формулировка исходной ситуации, направлений исследований и области возможных применений этих исследований.

1. О семантике в различных дисциплинах

Слово «семантика» является понятием (синоним – концепт), и по общему свойству всех понятий обладает объемом и содержанием [Попова и др., 2007]. Рассмотрим ключевые составляющие этого объема и содержания в системах ИИ и у человека (такой порядок рассмотрения выбран для большей наглядности и уменьшения повторов).

Общие принципы формальной семантики в системах ИИ приведены в [Голенков и др., 2012]. Они сводятся к следующему:

«В качестве формальной основы проектируемых интеллектуальных систем, в качестве основы абстрактных логико-семантических моделей интеллектуальных систем, использовать графодинамические модели специального вида – семантические модели представления и обработки знаний, в основе которых лежат семантические сети.

...все элементы (атомарные фрагменты) семантической сети являются знаками различных сущностей (объектов). Такими сущностями могут быть всевозможные внешние описываемые объекты, а также различные множества, состоящие из элементов (атомарных фрагментов) этой же семантической сети.

...Таким образом, семантическая сеть – это абстрактная знаковая конструкция «рафинированного вида», в которой нет ничего кроме знаков и инцидентности этих знаков».

Имеется целый ряд более конкретных трактовок формальных семантических моделей. В частности [Colomb, 2007, p.29] выделяет:

- отношения сущность-связь (Entity-Relationship model), где используются понятия сущности, экземпляра, отношения, атрибута и подтипа;
- унифицированный язык моделирования (Unified Modeling Language – UML), использует терминологию класса, экземпляра, ассоциации, атрибута и подкласса;
- объектно-ролевое моделирование (Object-Role Modeling), использует терминологию типа объекта, объекта, типа факта, роли и подтипа;
- дескриптивная логика (Description logics), использует терминологию концепта, индивидуального, роли и включения;
- язык веб онтологий (Web ontology Language – OWL), использует терминологию класса, индивидуального, свойства и подкласса;
- общая логика (Common Logic – CL), основана на синтаксисе исчисления предикатов первого порядка и использует терминологию выражение, атомарный факт, предикат и функция.

Несмотря на существенные различия перечисленных семантических моделей, в них можно выделить и общие черты:

- понятия, а если точнее, модели понятий, являются одним из ключевых элементов во всех типах семантических моделей. В некоторых типах моделей, в частности, логических, модели понятий являются единственным типом ключевых элементов;
- модели понятий могут содержать в себе более мелкие информационные единицы: атрибут, свойство, значение и др. Причем максимальное количество уровней детализации не превышает двух (сущность-атрибут-значение);
- возможны различные типы взаимоотношений между различными моделями понятий, а также их компонентами, в частности: включение в класс, роль, тип объекта, следование и др.;
- включение ключевого элемента в некоторую общность более высокого уровня не приводит к автоматическому изменению характеристик этого ключевого элемента (это утверждение не включает представление ключевых элементов с помощью объектно-ориентированного программирования, а также как агентов). Этот тезис можно

сформулировать и по принципу «сверху-вниз»: возможна декомпозиция некоторого информационного объекта, более крупного, чем модель понятия, на более мелкие элементы без изменения информационного содержания как системы в целом, так и составляющих ее элементов. Еще одна трактовка – информационные объекты являются линейно независимыми.

Рассмотрим понятие семантики в лингвистике. Сразу отметим, что в лингвистике мы также имеем дело с моделями. Однако эти модели получены в результате лингвистического анализа носителей семантики (людей) и поэтому они являются более адекватными, чем модели семантики в ИИ. К недостаткам этих моделей относится низкий уровень формализации, в результате чего они только описывают ситуацию, но не пригодны без дальнейшей формализации к использованию в автоматизированных системах анализа информации (если только не «вложить» эти модели в носителя семантики – человека).

С точки зрения лингвистов [Васильев, 1990, с.67] «Многочисленные определения значения, предлагавшиеся представителями различных направлений языкознания, можно свести к двум основным:

- 1) К определению значения как психической, отражательной сущности, соотносимой с такими психическими явлениями, как представления, эмоции, понятия и т.п.
- 2) К определению значения как реляционной сущности, как отношения языкового знака к предмету (денотативное значение), к понятию (сигнификативное значение), к условиям акта речи и его участникам (прагматическое значение), к сфере или ситуации его употребления (стилистическое значение), к другим языковым знакам (структурное значение) и т.д.».

Очевидно, что все без исключения модели семантики в ИИ не содержат в себе психических переживаний типа эмоций, представлений, сознания и т.п. – т.е., всего, что описывается лингвистами в п.1.

В то же время некоторые типы значений (семантики), приведенные в п.2 содержатся, или могут содержаться при определенных условиях, и в моделях семантики, используемых в ИИ. В частности, структурное значение семантики в ИИ и у человека практически совпадают. Для реализации прагматического значения в интеллектуальной системе необходимо реализовать модель мира или окружающей среды, а также цели и задачи субъекта порождения семантики. Такого рода условия моделируются, в частности, в экспертных системах и некоторых СППР. Поэтому можно сказать, что прагматическая семантика может частично содержаться в системах ИИ.

Для отображения стилистических значений необходимо моделирование текущей ситуации,

субъектов влияния, их действий, намерений и т.д. По-видимому, такая задача в настоящее время ИИ недоступна (если рассматривать уровень людей, а не примитивных искусственных агентов), хотя принципиальных препятствий для ее реализации нет, и она может быть решена в обозримом будущем.

В [Попова и др., 2007, С.94-100] различается три вида значения понятия или концепта:

1) Лексикографическое – это то, что фиксируется в словарях. Именно это значение используют специалисты по ИИ, фиксируя его в базах знаний и в онтологиях.

2) Психолингвистическое – это весь набор значений, который могут зафиксировать лингвисты у носителей языка в результате углубленных исследований. Психолингвистическое значение гораздо шире и объемней, нежели его лексикографический вариант, который составляет лишь небольшую часть психологически реального значения.

3) Но содержание понятия (концепта) еще шире психолингвистического значения, поскольку оно включает в себя смысл как неформализуемую, т.е. не выражаемую словесно часть восприятия.

Выше были рассмотрены концепции семантики естественного языка, в которых ключевые единицы семантики либо не выделяются [Васильев, 1990], либо в качестве таких единиц используются понятия (концепты) [Попова и др., 2007]. Но объектами семантики являются не только концепты! Так, в Большом энциклопедическом словаре [Ярцева, 1989. – С.438-439] семантика представляется в виде многоуровневой иерархической системы семантических значений, включая:

- уровень морфем (семантика морфемы);
- уровень понятий или концептов (семантика концепта);
- уровень предложений (семантика предложения);
- уровень текста (семантика текста).

И что самое существенное, семантика, т.е. содержание каждого уровня не является просто суммой (или частью) нижестоящих (вышестоящих) уровней [Васильев, 1990, с.70-71.].

Нагляднее всего это положение можно проиллюстрировать на примерах. Так, выражение «сидеть на солнце» (идея примера взята из [Кобозева, 2000], вовсе не означает сидеть, свесив ноги, на центральном объекте Солнечной системы. Оно предполагает субъекта или субъектов, солнечную погоду, отдых и т.д., т.е., довольно развитую модель ситуации, которая в ряде своих составляющих никаким образом не содержится в определении исходных понятий.

Второй пример: пусть некто в шутку или всерьез заявляет, что у них в организации монголо-татарское иго. Если следовать буквальному смыслу

входящих в выражение понятий, то в организации имеются принадлежащие монголам и татарам хомуты. При выходе на первый вышестоящий уровень мы обращаемся к истории, и понимаем это как захват Руси монголо-татарами. На втором уровне мы понимаем, что угнетение, по тяжести сравнимое с монголо-татарским игом, происходит в данной организации. Таким образом, входящие в состав выражения понятия переосмысливаются коренным образом дважды, и вывод из данного выражения делается не на основании значения входящих в него понятий, а на основании вновь сконструированного смысла.

Из проведенного сравнительного обзора формальной и естественной семантики можно сделать следующие выводы:

- формальная семантика использует лишь небольшую часть значения понятий по сравнению с носителями естественного языка;
- в формальной семантике значения вышестоящих уровней формируются из нижестоящих по неким типовым законам. При этом содержание единиц нижестоящего уровня остается практически неизменным;
- в семантике естественных языков содержание вышестоящих уровней формируется каждый раз индивидуальным образом с учетом истории, традиций, культуры вообще, а также с учетом психологических особенностей субъекта восприятия, - его намерений, эмоционального состояния и др. При этом значение единиц нижестоящего уровня может существенно изменяться;
- в формальных системах для получения выводов и принятия решений фактически используется информация одного-единственного уровня. Получение вывода и семантика есть почти одно и то же;
- человек вначале создает на основании речи или языка содержание верхних уровней семантики, а затем использует его для получения выводов и принятия решений. Конструирование семантики и получение на ее основе выводов, - это разные (но последовательные) психические процессы.

В таблице 1 суммированы сравнительные характеристики объема семантики в естественном и искусственном интеллектах.

Таблица 1 – Сравнительные характеристики объема семантики

Компонента семантики	ИИ	Человек
Прагматика	+-	+
Стилистика	+-	+
Структурное значение	+	+
Психологическое	-	+
Лексикографическое	+	+
Психолингвистическое	+-	+
Смысл	-	+
Нелинейность	-	+
Иерархия уровней	-	+

Обозначения: +- - качество частично присутствует в некоторых современных разработках;

+- - качество не реализовано в современных разработках вследствие высокой трудоемкости, но принципиальных ограничений для его реализации нет;

- - качество принципиально невозможно реализовать в системах ИИ на основании используемых принципов.

Сравнение объема семантики в ИИ и естественном интеллекте показывает следующее:

- из целого ряда семантических компонент ИИ полноценно использует лишь одну – структурную, которую с равным успехом можно назвать не семантической, а синтаксической;
- в ИИ в принципе не присутствуют и не создаются значения вышестоящих уровней, которые, собственно и используются для получения выводов.

В качестве комментария: такое тяжелое психическое заболевание как аутизм с когнитивной точки зрения заключается в буквальном понимании слов [Питерс, 1992, с.29-30], т.е. в неспособности строить семантические значения как для понятий, так и для высших уровней. И такого посредника мы создаем между собой и необозримыми базами знаний. На основании проведенного рассмотрения оптимизм по поводу творческих способностей такого «интеллекта» представляется не просто необоснованным, а очевидно неверным.

И в заключение данного вопроса о нейрофизиологических «корнях» семантики и сущности формализации.

Мозг является самой сложной системой во Вселенной, известной человеку. Нейрофизиологи выделяют не менее девяти уровней его структурной организации [Shepherd, 2004, p.6]. Поскольку каждый структурный уровень задает какие-то свои специфические информационные процессы, то число уровней представления информации (и соответствующих этим уровням информационных сущностей) в мозгу также не менее девяти.

На рис.1. представлены примерные соответствия между уровнями организации мозга и компонентами рассмотренных выше типов семантики.

Самые дробные информационные единицы, с которыми работают формальные методы, - это признаки или значения. С точки зрения нейрофизиологии этому соответствует уровень никак не меньше нейрона (а скорее всего даже на один-два уровня выше). Поскольку же формальные методы не способны к построению семантики на вышестоящих уровнях информационного содержания, то они неспособны и к генерации целостных информационных объектов вышестоящих уровней. Т.е., их уровень

использования целостности близок нулю. Эта ситуация и представлена на рис.1.

С другой стороны, смысловое информационное содержание интегрирует в некоторую целостность всю (или значительную часть) доступной человеку информации, используя для этого все структурные и процессные уровни представления информации (что также представлено на рис.1.).

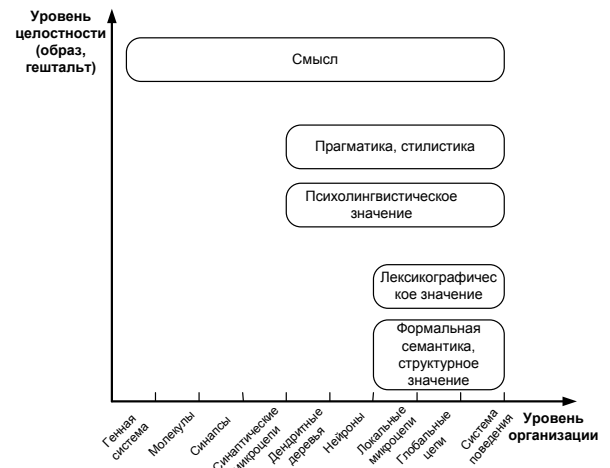


Рис.1. Взаимосвязь типов семантики с уровнями организации мозга

Опираясь на предыдущее рассмотрение, и наглядное представление на рис.1, можно сделать чрезвычайно важный для последующих рассуждений вывод: сущность формализации заключается в отбрасывании ряда нижележащих уровней обработки информации. Результаты их, вообще говоря, сложной индивидуальной деятельности, заменяются существенно более простыми и однотипными отношениями (логическими выражениями или математическими функциями). Как показывают оценки [Сторож, 2013], такая процедура позволяет уменьшить информационные потоки в ИИ на 20 и более порядков. Но неизбежно возникает вопрос: а что же при этом теряется? Несмотря на исключительную важность ответа на этот вопрос, специалисты по ИИ его, по-видимому, не исследовали. Проведенные же в данной работе исследования объема потерь «по факту» показывают, что теряется смысл, целостность и компоненты семантики, участвующие в понимании и мышлении.

2. Онтологии и формальные семантики

Что такое онтология (или онтологии) в ИИ? Дадим несколько определений, в которых формулируются различные взгляды на сущность этого понятия. Наиболее известно определение, восходящее к Груберу, и затем уточненное Студером и др. [Studer, 1998]: «Онтология является формальной, явно выраженной спецификацией совместной концептуализации».

Широко известно и цитируется определение Нариньяни А.С. [Нариньяни, 2001]: «Онтология, это модель предметной области, использующая все доступные средства представления знаний, релевантные для данной области».

В связи с развитием Semantic Web возникло еще одно уточняющее определение [Ландэ, 2005]: «Онтология – это «спецификация концептуализации предметной области», или упрощенно документ или файл, формально задающий отношения между терминами. Это своего рода словарь понятий предметной области и совокупность явным образом выраженных предположений относительно смысла этих понятий».

Прокомментируем эти определения. Первое из них выражает позицию разработчиков программного обеспечения и содержит целый ряд спорных, узких или недостаточно определенных положений.

Так, в [Guarino, 2009, p.2] отмечается, что термин «концепт» в данном случае является существенно неадекватным, поскольку он не несет в себе смысла сущности, существования чего-либо, и предлагают заменить его более уместным, в частности, термином «универсалия».

Вызывает вопросы и использование словосочетания «спецификация совместно разделяемых знаний», поскольку под спецификацией принято обычно понимать некоторый перечень, список. Но к формальным совместно разделяемым знаниям могут относиться также и программы, схемы, графические и процессные модели. Более того, эти типы представления знаний широко используются в настоящее время для создания онтологий. Поэтому термин «спецификация» необходимо заменить на термин с более широким значением, включающий все типы формального представления знаний.

И третье замечание к определению Грубера-Студера рассмотрим в контексте определения, данного А.С. Нариньяни. Онтология является не просто совместным представлением чего-либо, а таким представлением, в котором выделяются наиболее существенные свойства предметной области, - по определению, это модель. Но эту мысль можно выразить еще более определенно: создавая модель, мы выделяем инварианты, описывающие предметную область и позволяющие наиболее экономным и оптимальным образом описывать ее с точки зрения задач, которые мы ставим по отношению к данной предметной области. В частности, эта идеология и терминология широко используется в таком направлении кибернетики, как идентификация систем.

И относительно определения, связанного с Semantic Web. В нем подчеркивается, что онтология задает не только список понятий, связанных с какой-либо предметной областью, но и отношения между понятиями, смысл. Иначе говоря,

неотъемлемой, центральной частью любой онтологии является та или иная модель семантики, принятая данной онтологией. И именно принятая в онтологии модель семантики определяет тип языков представления знаний, используемых моделей, и, в конечном счете, область использования и возможности данной онтологии.

С учетом приведенных определений и обсуждения их содержания, дадим свое определение онтологии:

Онтология, - это инвариантное, наиболее экономное и последовательное с точки зрения выполняемых задач представление содержания некоторой предметной области или вида деятельности, основанное на использовании определенной модели семантики, и одного или нескольких стандартизированных языков представления знаний.

В это определение не включено требование разделяемости онтологии участниками, поскольку на этапе разработки онтологии оно просто невозможно, а в случае успеха онтологии обеспечивается автоматически.

Тезис об использовании в онтологии стандартизированных языков представления знаний включен в определение на основании следующих соображений:

- во-первых, все наиболее известные онтологии основаны на ранее созданных или разработанных для данной онтологии языках представления знаний [Colomb, 2007, p.29];
- во-вторых, использование в онтологии стандартизированного языка представления знаний существенно облегчает задачу изучения данной онтологии сообществом пользователей, т.е. ее «совместную разделяемость».

В нашем определении онтология основывается на содержании не только предметной области, но и вида деятельности. Это связано с тем, что понятие «предметная область» связывается в основном с текущим положением, статикой. А реальные онтологии во многих случаях в той или иной мере предназначены для описания действий, т.е. динамики. В частности, это относится к Semantic Web, идеологии онтологического проектирования, и вообще всем типам моделей с участием процессов, работ, действий и т.п.

3. Требования к моделям семантики и онтологиям

Переформулируем приведенное выше определение онтологий с учетом их практических функций:

Входящие в онтологии информационные объекты представляют собой своеобразные интеллектуальные точки отсчета, при использовании которых некоторый класс задач, входящих в сферу действия данной онтологии,

решается максимально быстрым и экономным образом.

Каким образом это происходит? Для иллюстрации приведем несколько широко известных интеллектуальных объектов, выполняющих те же функции, что и онтологии.

В механике для упрощения решения уравнений движения используются преобразования координат фазового пространства. Фактически ищется некоторая новая система координат, в которой уравнения решаются максимально просто [Ланцош, 1965, с.227].

В многомерной статистике существует сходный принцип в методе выделения главных компонент, приводящий к уменьшению числа переменных. Он заключается в последовательном вращении переменных с целью максимизации дисперсии исходного пространства переменных, а полученные направления используются как главные компоненты [Айвазян, 1998, с.565].

Т.е., ключевой принцип, заложенный в онтологиях, не является для человечества совершенно новым, он широко использовался и ранее. Но в отличие от рассмотренных примеров, онтологии работают с областями знаний, для которых процедуры формальной оптимизации заключаются не во вращении координат, а в выборе ключевых точек в концептуальном пространстве, и связей между ними. Причем такая процедура, по-видимому, не является полностью формализуемой, хотя и включает использование математики, например, для определения частоты связей между понятиями.

Полученные с использованием вышеописанных принципов онтологии могут использоваться в двух ключевых режимах:

1) В режиме выполнения стандартных операций или процедур. Очевидно, что для организации такой деятельности достаточно формальной семантики, заложенной в онтологиях (конечно, если она адекватна выполняемым задачам).

2) В режиме, где недостаточно стандартных знаний, и требуются новые идеи, инновации, разработки и т.п. Онтологии также выполняют полезную роль, позволяя сформулировать оптимальную для поиска исходную интеллектуальную позицию. Но самих по себе онтологий для получения правильного решения, как правило, недостаточно, что подтверждается на опыте.

Почему это происходит? С самых общих позиций можно сказать, что в онтологиях не хватает гибкости, «разрешающей способности». Новую идею, видение, понимание, можно получить, используя неожиданные, ранее не формализованные связи между понятиями и группами понятий, которых в онтологиях нет по определению, нет в них и генерации новых значений на уровне групп

понятий (выражений, предложений), и более крупных участков текста. А именно на этом уровне происходит понимание конкретных ситуаций, рождение новых идей, и вообще всех интеллектуальных процессов, недоступных компьютеру.

Кто-то может сказать, что онтологии и не ориентированы на автоматический вывод новых идей, но вот для поиска информации они подходят превосходно. Но любой специалист прекрасно знает, что это не так. На большинство вопросов поисковики выдают огромное количество ссылок (обычно от десятков тысяч до миллионов). Выделить среди них нужную информацию средствами Интернет невозможно, вот и приходится ограничиваться первыми десятками или сотнями, пропуская, может быть критически важную информацию. Почему формальные методы не могут отобрать нужную информацию? Можно выделить целый ряд причин:

- новые идеи используют и новые связи между понятиями. Естественно, этих новых связей статистически немного по сравнению с традиционными, и они просто не попадают в онтологии. Их появление может заметить лишь человек в результате углубленного изучения информации;

- одни и те же понятия различными людьми понимаются существенно различным образом; и, наоборот, называемые разными словами представления могут быть близки между собой. Поэтому человек при углубленном анализе информации «индивидуализирует» понимание ключевых понятий в тексте в зависимости от их происхождения;

- при поиске информации в Интернет компьютерными методами неявно предполагается, что вся информация правда. Нет нужды доказывать, что это далеко не так. Информация может быть искажена самыми различными способами: прямая ложь, преувеличение в рекламных целях, умолчание в той или иной степени, подмена понятий; пересказы известного, в той или иной мере неточные (просто информационный мусор) и др.

Не подходят современные формальные методы и для аналитики уже готовых отобранных текстов (например, базы знаний предприятия). Как уже говорилось, многие (если не подавляющее большинство) ключевые идеи выражаются не перешедшими в штампы словосочетаниями, а на этом уровне формальные методы семантику просто не улавливают.

На основании вышеизложенного можно сделать вывод, что повышение эффективности интеллектуальных систем на основе онтологий напрямую зависит от уровня семантики этих интеллектуальных систем. Причем уровень семантики определяется не мощностью формальных методов, а соответствием человеческому мышлению, тем принципам, которые использует человек при генерации новых идей,

оценке ситуаций, построении моделей конкретных объектов и действий и т.д. Человек не просто использует стандартные модели семантики, он в каждом конкретном случае «строит» семантику, создавая смыслы верхних уровней. И эти уровни сами по себе никаким образом в тексте не содержатся. Для их генерации необходим мощный интеллект, владеющий культурой, и способный понимать ситуации, к которым относится текст или любая другая не полностью формализованная информация.

Под интеллектом в данном случае понимается программная конструкция типа экспертной системы, но с существенно более широкими возможностями. Эта конструкция должна включать в себя систему порождения смысла высших уровней, и получения выводов с использованием этих смыслов. Для этого необходимы:

- база знаний самого интеллекта (в отличие от анализируемой информации), включающая хотя бы в некоторой степени культуру, историю, и интерпретацию (смысл, семантику) стандартных словосочетаний;
- способность к порождению смыслов высших уровней, используя, в частности, механизмы построения метафор и аналогий;
- механизмы интеграции информации не на основе слов, а на основе порождаемых смыслов;
- механизмы оценки качества информационного содержания и на этой основе присваивание рейтинга тексту;
- человек не использует при мышлении сложных функций, а обработка информации происходит существенно параллельным образом, и всего за несколько циклов. В связи с этим необходимо искать и использовать простые и в высокой степени параллельные механизмы порождения семантики.

4. Перспективные направления использования онтологий

Интеллектуальные системы с использованием онтологий можно улучшать двумя принципиально различными способами:

1) Разработка некоторого формального языка, в семантике которого максимально возможным образом заложены и закономерности предметной области.

К такого типа формальным языкам относятся, в частности, SADT (Structured Analysis and Design Technique – методология структурного анализа и проектирования) и возникшее на основе SADT семейство языков IDEF, широко используемое разработчиками программного обеспечения в описании и управлении бизнесом, и вообще во всех случаях работы со сложными системами. Большим достоинством этих языков является то, что они основаны на использовании графики, что позволяет человеку, с одной стороны, легко излагать свои

представления, в том числе и образные, а с другой, понимать чужие.

К такому типу разработок следует, по-видимому, отнести онтологию проектирования, развиваемую Н.М. Боргестом и коллегами.

Достоинством идеологии, основанной на использовании формальных языков (в случае их удачной разработки) является возможность описания ситуации с минимальными затратами, и последующего эффективного управления ситуацией.

Недостаток, – затруднения или даже невозможность нахождения эффективного решения в случае, когда ситуация не описывается первоначальными упрощающими представлениями. В таком случае решение ищется человеком путем выхода за пределы первоначальной модели и другими методами. И для поддержки интеллектуальных усилий человека в этих случаях необходимо разрабатывать онтологии второго типа.

2) Разработка формальных семантик, в максимальной (на современном уровне) степени учитывающей особенности понимания и мышления человека. Такая семантика должна разрабатываться в комплексе с:

а) базой знаний, содержащих как знания о мире, так и знания о семантике в целом, и семантике информационных единиц различных уровней;

б) интеллектом, способным генерировать новые смыслы. Возможно, для этого недостаточно использования символьных методов, и нужно привлекать иные методы, в частности, нейросети.

В контексте необходимости более мощной семантики невозможно не упомянуть работы Д.А. Поспелова [Поспелов, 1986] и его последователей, которые на несколько десятилетий опередили время. Их представления о семантике как модели ситуации очень актуальны и в наше время, и, конечно, их нужно использовать и развивать.

Предлагаемое направление разработок весьма трудозатратно. Возникает вопрос: а где же брать деньги?

Представляется, что для этого необходимо организовать бизнес-проект в области Semantic Web, объединяющий усилия специалистов из Белоруссии, России и Украины. Ежегодный рынок услуг в этой области составляет многие миллиарды. Необходимо создание собственной поисковой машины, основанной на семантике нового поколения, и раскрутка сервера под эту поисковую машину, сравнимого по масштабу с Yahoo, Altavista, Google...

Заключение

В работе рассмотрены особенности реализации семантики в искусственных системах и у человека. Рассмотрены ключевые компоненты, составляющие

семантику. Показано, что лишь незначительная и очень простая их часть реализуется в искусственном интеллекте. Наиболее важным недостатком формальной семантики является ее неспособность к построению высших уровней смысла сообщения, а именно на этом уровне понимается информация, происходит мышление и принимаются решения во всех случаях, когда ситуация отличается от типовой.

На основании анализа литературных данных о сущности онтологий и областей их применения дано определение онтологии как инвариантного, наиболее экономного и последовательного представления содержания некоторой предметной области или вида деятельности, основанного на определенной модели семантики, а также одним или несколькими стандартизированными языками представления данных.

Разработаны требования к моделям семантики и онтологиям, основанные прежде всего на использовании принципов понимания информации человеком. Сделан вывод, что реализации семантики нового типа необходимо совместно реализовывать и использовать интеллект, онтологию и базы знаний о мире и о семантике.

Предложено реализовать разработку новой семантики для поиска информации как бизнес-проект в рамках развития Semantic Web.

Библиографический список

- [Berners-Lee et al., 2001] Berners-Lee T., Hendler J., Lassila O. The Semantic Web // Scientific American -2001. – V.284, N 5. – P.29-37.
- [Colomb, 2007] Colomb R.M. Ontology and the Semantic Web – IOS Press – 2007. – 272 p.
- [Guarino et al., 2009] Guarino N., Oberle D., Staab S. What Is an Ontology / Eds. Staab S., Studer R. – Springer-Verlag – Berlin – 2009. – p.1-17.
- [Studer et al., 1998] Studer R., Benjamins R., Fensel D. Knowledge engineering: Principles and methods // Data and Knowledge Engineering – 1998/ - V.25, N 1-2. – P.161-198.
- [Shadbolt et al., 2006] Shadbolt N., Hall W., Berners-Lee T. The Semantic Web Revisited // IEEE Intelligent Systems – 2006. – V.21, Issue 3 – P.96-101.
- [Shepherd, 2004] The Synaptic Organization of the Brain – Ed. Shepherd G.M. – Oxford – University Press – 2004. – 736 p. (P.6)
- [Айвазян и др., 1998] Айвазян С.А., Мхитарян В.С. Прикладная статистика и основы эконометрики – М.: ЮНИТИ – 1998. – 1022 с. (с.565).
- [Васильев, 1990] Васильев Л.М. Современная лингвистическая семантика: Учеб. пособие для вузов – М.: Высш. шк. – 1990. – 176 с.
- [Голенков и др., 2012] Голенков В.В., Гулякина Н.А. Графодинамические модели параллельной обработки знаний: Принципы построения, реализации и проектирования // Минск – БГУИР – 2012. – С.23-52.
- [Кобозева, 2000] Кобозева И.М. Лингвистическая семантика: Уч. пособие – М.: Эдиториал УРСС. – 2000. – 352 с.
- [Кудрявцев, 2010] Кудрявцев Д.В. Системы управления знаниями и применение онтологий: Учеб. пособие –СПб – Изд-во Политехн. ун-та – 2010. – 344 с.
- [Кузнецов, 2012] Кузнецов О.П. Когнитивная семантика: новые подходы к старым проблемам // XII Международная научная конференция имени Т.А. Таран «Интеллектуальный анализ информации» - Киев – 2012 – С.77-88.
- [Ландэ, 2005] Ландэ Д. Семантический веб: от идеи – к технологии // Телеком – 2005. - № 6.
- [Ланцош, 1965] Ланцош К. Вариационные принципы механики – М.: Мир – 1965. – 406 с.

[Нариньяни, 2001] Нариньяни А.С. Кентавр по имени Теон: тезаурус + онтология / / Международный семинар по компьютерной лингвистике и ее приложениям «Диалог 2001» - 2001. – Т.1. – С.184-188.

[Питерс, 1999] Питерс Т. Аутизм: От теоретического понимания к педагогическому воздействию – СПб.: Институт специальной педагогики и психологии – 1999. – 192 с. (с.29-30)

[Попова и др., 2007] Попова З.Д., Стернин И.А. Когнитивная лингвистика – М.: АСТ: Восток-Запад – 2007. – 314 с.

[Поспелов, 1986] Поспелов Д.А. Ситуационное управление: теория и практика – М.: Гл. ред. физ.-мат. лит. – 1986. – 288 с.

[Поспелов, 1989] Поспелов Д.А. Моделирование рассуждений. Опыт анализа мыслительных актов – М.: Радио и связь – 1989. – 184 с.

[Сторож, 2013] Сторож В.В. Актуальные направления исследований мышления и интеллекта // Искусственный интеллект – 2013. - № 3. – С.30-37.

[Шведин, 2011] Шведин Б.Я. Онтология проектирования – Terra Incognita? // Онтология проектирования – 2011. - № 1. – С.9-21.

[Ярцева, 1989] Большой энциклопедический словарь. Языкознание – Под ред. В.Н. Ярцевой – Советская энциклопедия – 1989. – 687 с.

SEMANTICS AND ONTOLOGY IN NATURAL AND ARTIFICIAL SYSTEMS

Storozh V.V. *

*Donetsk State Technical University

ws52@mail.ru

In work features of realisation of semantics in artificial systems and at the human are considered. The key components making semantics are considered. It is shown that only insignificant and their very simple part is realised in artificial intelligence. The most important lack of formal semantics is its inability to construction of the highest levels of sense of the message. However at this level the information is understood, there is a thinking and decisions in all cases when the situation differs from the typical are made.

On the basis of the analysis of the literary data definition ontology as invariant, most economical and consecutive representation of the maintenance of some subject domain or activity kind is made. Thus ontology as a basis should use certain model of semantics, and also one or several standardised languages of data presentation.

Requirements to models of semantics and ontology are developed. These requirements are based first of all on use of principles of understanding of the information by the human. The conclusion is drawn that it is necessary for realisation of semantics of new type to realise and use in common intelligence, ontology and knowledge bases about the world and about semantics.

It is offered to realise working out of new semantics for information search as the business project within the limits of development Semantic Web.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.891.2

ИСПОЛЬЗОВАНИЕ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ ПРОГНОЗИРОВАНИЯ СУДЕБНЫХ РЕШЕНИЙ

Климов В.С. *, Новикова А.О. *, Данко А.И. *

* ФГБОУ ВПО «Тольяттинский государственный университет»,
г. Тольятти, Россия

KlimovV@gmail.com

nelenivec777@mail.ru

east1941@gmail.com

В настоящем исследовании предложен способ прогнозирования судебных решений, основанный на выделении из материалов дела критериев, прямо или косвенно влияющих на исход дела, и анализа их с помощью нейронных сетей. Путем изменения диапазона дел, используемых при обучении нейронной сети, возможна подстройка алгоритма на оценку прогнозов исхода дела в зависимости от личности конкретного судьи.

Ключевые слова: искусственный интеллект, нейронные сети, прогнозирование судебных решений.

Введение

Существующие научные разработки по применению информационных технологий в судебной деятельности РФ направлены лишь на автоматизацию документооборота и архивирование судебных решений. К таким системам относятся ГАС Правосудие (разработчик ФГУП НИИ «Восход»), КонсультантПлюс (разработчик «КонсультантПлюс» НПО ВМИ), Гарант (разработчик НПП «Гарант-сервис-университет») и др. Однако данные системы не обеспечивают участников судебного процесса инструментарием для анализа материалов дела и прогнозирования исхода судебного разбирательства. Из-за этого снижается уровень эффективности выбора тактики выступления как у стороны обвинения, так и у стороны защиты.

Поэтому актуальной можно признать цель данного исследования – повышение эффективности судебного делопроизводства в РФ путем разработки технологии прогнозирования судебных решений, основанной на применении искусственного интеллекта.

Алгоритм прогнозирования

Для преодоления описанных выше проблем в настоящем исследовании предложена оригинальная идея о возможности синтеза нейронной сети (как технологии искусственного интеллекта) в алгоритм

анализа материалов судебных дел. Так как на современном уровне развития нейронные сети обладают возможностью обучения и обобщения предыдущих прецедентов на неизвестные ранее случаи [Wasserman, 1989; Skitt et al., 1993], то предполагается, что синтез сети в алгоритм анализа обеспечит высокую точность прогнозирования результатов судебного решения.



Рисунок 1 – Схема функционирования алгоритма прогнозирования судебных решений

Предложен следующий алгоритм функционирования системы прогнозирования судебных решений (рисунок 1) [Данко и др., 2013]:

- сначала из материалов дела выделяются критерии (например, состояние здоровья обвиняемого, наличие судимости), которые прямо или косвенно могут повлиять на исход дела;
- полученные критерии преобразуются в сигналы (в числовые или биполярные) которые подаются на вход обученной нейронной сети;

- выработанный нейронной сетью выходной сигнал преобразуется в понятный для конечного пользователя вид – текст прогноза судебного решения.

Так как юридический текст является строго формализованным и использует специализированные термины и определения, то для автоматизированного извлечения необходимых критериев из материалов дела удобно использование технологии Text Mining, являющейся сочетанием базовой технологии Data Mining с технологиями поиска информации Information Retrieval и извлечения информации Information Extraction.

Материалы судебных разбирательств Российской Федерации после вступления решения суда в законную силу становятся общедоступными. Поэтому посредством Text Mining возможно осуществление сбора необходимой для обучения нейронной сети базы данных судебных разбирательств.

Выбор типа нейронной сети, используемой для прогнозирования судебного решения, зависит от категории анализируемых правонарушений. Для правонарушений с малым количеством критериев, влияющих на исход дела, целесообразно использования нейронных сетей Хэмминга и Хопфилда. К таким судебным делам относятся административные правонарушения и малая часть уголовных правонарушений (например, дорожно-транспортные происшествия, повлекшие тяжкий вред здоровью). Для анализа дел из других категорий требуется проведение дополнительных исследований по определению в каждом конкретном случае оптимального (с точки зрения точности функционирования) типа и структуры нейронной сети.

Нейронные сети Хэмминга и Хопфилда умеют работать с данными, представленными только в виде биполярного вектора, компоненты которого принимают значения либо «-1», либо «+1», поэтому в данном исследовании используется алгоритм кодирования информации о критериях материалов дела в биполярные сигналы, основанный на проведенных ранее исследованиях [Климов и др., 2013]. Для каждого критерия определяют список допустимых значений, а затем для каждого значения набор биполярных сигналов, пример представлен в таблице 1.

Таблица 1 – Пример возможного варианта биполярных сигналов для критерия «Состояние водителя в момент дорожно-транспортного происшествия»

Значение критерия	Биполярный вектор значения критерия
Состояние не определялось	(+1 -1 -1 -1 -1)
В сознании, трезв	(-1 +1 -1 -1 -1)
В состоянии опьянения	(-1 -1 +1 -1 -1)
В состоянии аффекта	(-1 -1 -1 +1 -1)
...	...

Объединяя сбора и объединения данных о различных критериях можно получить биполярную матрицу значений, описывающих особенности материалов судебного дела (рисунок 2).

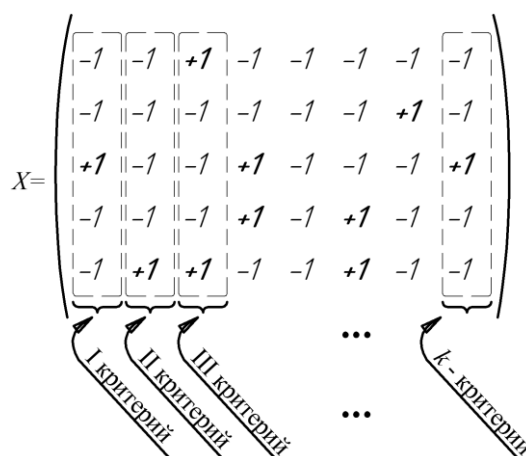


Рисунок 2 – Матрица входных X сигналов нейронной сети описывающих k критериев материалов судебного дела

Фактически значения полученной матрицы являются входными сигналами нейронной сети (рисунок 3). Для удобства зрительного восприятия состояния входов сети активированные нейроны представлены в виде квадратов, закрашенных в черный цвет, а не активированные нейроны – в виде белых квадратов.

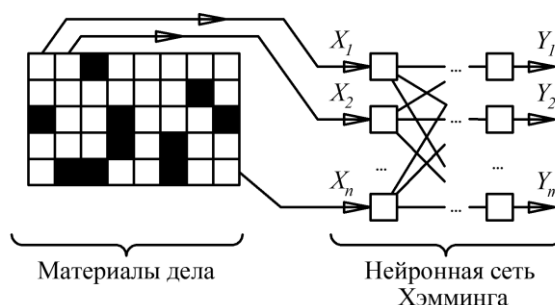


Рисунок 3 – Подача данных на вход нейронной сети

Процесс прогнозирования судебных решений (на основе нейронной сети Хэмминга) представлен далее. Сначала на вход нейронной сети подается неизвестный сигнал X_i , описывающий критерии анализируемого дела, исходя из которого,

рассчитывается состояние аксонов нейронов первого слоя $y_j^{(1)}$ (1):

$$y_j^{(1)} = \sum_{i=1}^n \omega_{ij} X_i + \frac{n}{2}, \quad j = 1..m \quad (1)$$

Далее полученные значения $y_j^{(1)}$ инициализируют значения $y_j^{(2)}$ аксонов второго слоя (2):

$$y_j^{(2)} = y_j^{(1)}, \quad j = 1..m \quad (2)$$

Затем проводится расчет состояния синапсов $s_j^{(1)}$ нейронов второго слоя (3):

$$s_j^{(2)}(p+1) = y_j^{(2)}(p) - \varepsilon \sum_{i=1}^m y_i^{(2)}(p), \quad (3)$$

$$i \neq j, i = 1..n, j = 1..m$$

где p – номер итерации функционирования нейронной сети Хэмминга.

Выходной сигнал с каждого нейрона второго слоя определяется через активационную функцию f (4):

$$y_j^{(2)}(p+1) = f[s_j^{(2)}(p+1)], j = 1..m \quad (4)$$

где $f(x)$ – активационная функция с порогом F (5):

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & 0 \leq x < F \\ F, & x \geq F \end{cases} \quad (5)$$

Порог F выбирается равный количеству m шаблонов судебных дел.

При дальнейшем функционировании сети выходные сигналы со второго слоя нейронов проверяются на наличия в них изменений за последнюю итерацию. Если выходные сигналы изменились, то они подаются с помощью обратной связи на входы второго слоя нейронов. С этого момента начинается следующая итерация функционирования сети и в соответствии с равенствами (1) и (2) рассчитываются новые выходные сигналы. Если выходные сигналы второго слоя нейронов за последнюю итерацию не изменились, то они передаются на выход Y нейронной сети Хэмминга (6):

$$Y_j = y_j^{(2)}, j = 1..m \quad (6)$$

Номер нейрона в выходном слое Y , значение сигнала которого отлично от нуля, соответствует номеру шаблона дела наиболее близкого к анализируемым материалам дела.

Перед началом функционирования нейронной сети, необходимо поместить в ее память информацию об известных шаблонах материалов дел. Для этого проводится обучение сети, которое заключается в расчете весовых коэффициентов ω_{ij} первого слоя (7):

$$\omega_{i,j} = -X_i^j, i = 1..n, j = 1..m, \quad (7)$$

Особенности предложенного алгоритма прогнозирования:

- Алгоритм прогнозирования можно использовать на любом этапе судебного разбирательства, тем самым наблюдая за динамикой изменения прогнозов исхода дела;
- Добавляя критерии материалов дела, подаваемые на вход нейронной сети, можно определить рычаги влияния на благоприятность исхода дела;
- Путем изменения диапазона дел, используемых при обучении нейронной сети, можно настроить алгоритм на оценку прогнозов исхода дела в зависимости личности конкретного судьи.

Практическая значимость

Предложенный алгоритм предсказания судебных решений может быть полезен:

- гражданам Российской Федерации для оценки собственного уровня юридической ответственности;
- юридическим лицам любых организационно-правовых форм для оценки вероятности положительного исхода судебного разбирательства;
- прокурорам и адвокатам в качестве инструмента для выбора тактики выступления;
- суды могут использовать предложенную разработку в качестве системы поддержки принятия решений.

Заключение

Совершенствование информационных систем, применяемых в судебной деятельности, возможно путем внедрения в них нейросетевых технологий. С помощью нейронных сетей возможно прогнозирование судебных решений с определенной точностью.

Юридический язык является строго формализованным, поэтому для автоматизированного извлечения данных из материалов дела может быть успешно применена технология Text Mining.

Выбор типа и структуры нейронной сети зависит от категории анализируемых правонарушений. Для анализа административных правонарушений с малым набором критериев влияющих на исход дела целесообразно использование нейронных сетей Хэмминга и Хопфилда.

Финансирование исследований осуществляется при поддержке федеральным государственным учреждением «Фонд содействия развитию малых форм предприятий в научно-технической сфере» по программе «Участник молодежного научно-инновационного конкурса»

Библиографический список

[Wasserman, 1989] Wasserman, P. D. Neural computing: Theory and Practice / P. D. Wasserman // New York : Van Nostrand Reinhold Co, 1989. – 230 p.

[Skitt et al., 1993] Skitt, P.J. Process monitoring using auto-associative, feed-forward artificial neural networks / P.J. Skitt, M.A. Javed, S.A. Sanders, A.M. Higginson // Journal of Intelligent Manufacturing, 1993, №4, P. 79-94.

[Данко и др., 2013] Данко, А. И. Прогнозирование судебных решений искусственным интеллектом на основе анализа материалов дела с помощью нейронных сетей / А. И. Данко [и др.] // Межуниверситетские осенние инновационные чтения УМНИК 2013: материалы доклада конкурса УМНИК – Самара: Изд-во ООО «Инсома-пресс», 2013. – С. 71-73.

[Климов и др., 2013] Применение нейросетевых технологий распознавания образов для диагностики контактной сварки в производственной среде / В.С. Климов [и др.] // Сварка и диагностика, 2013, № 2, С. 40-44.

[Baldi et al., 1989] Baldi, P. Neural Networks and Principal component Analysis: Learning from Examples Without Local Minima / P. Baldi, K. Hornik // Neural Networks, 1989, №2, P. 53-58

USE OF ARTIFICIAL INTELLIGENCE FOR PREDICTION OF JUDGMENTS

Klimov V.S. *, Novikova A.O. *, Danko A.I. *

** Togliatti State University, Togliatti, Russia*

KlimovV@gmail.com

nelenivec777@mail.ru

east1941@gmail.com

In research the method of prediction of judgments is offered. For this purpose from materials of judicial proceedings to select criteria directly or indirectly influencing an outcome of judicial proceedings. These criteria are analyzed by means of neural networks.

Introduction

Existing scientific research on the application of information technologies in the Russian judicial activities aimed only at automating workflow and archiving of court decisions. Such information systems do not provide trial participants tools for the analysis of the case and predict the outcome of the trial. Because of this reduced level of effectiveness of tactical appearance in court as the prosecution and the defense.

Therefore can be considered an actual purpose of this study - improving the efficiency of judicial proceedings in the Russian Federation by developing technology forecasting judgments based on the application of artificial intelligence.

Main Part

Following algorithm functioning forecasting system of judicial decisions:

- the materials of the case highlighted the criteria that may directly or indirectly affect the outcome of the case ;
- obtained criteria are converted into signals (in numerical or bipolar) which is input to the trained neural network;

- the output of the neural network is converted into a form suitable for the end user view - text prediction judgment.

Features of the proposed prediction algorithm:

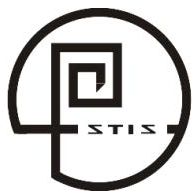
- prediction algorithm can be used at any stage of the proceedings , thereby observing the dynamics of change forecasts outcome of the case;
- adding the criteria of the case, signals input to the neural network , we can determine leverage on the favorable outcome of the case;
- by changing the range of cases used in the training of the neural network, we can adjust the algorithm to evaluate forecasts outcome of the case according to the personality of the judge .

Conclusion

Improvement of information systems used in judicial activities, possibly through implementations of neural network technologies in them. With the help of neural networks forecasting possible outcomes of judicial decisions with certain accuracy.

Legal language is strictly formalized, so for automated data extraction from the case can be successfully applied technology of Text Mining.

Selecting the type and structure of the neural network depends on the category of offenses analyzed. For the analysis of administrative offenses with a small set of criteria affecting the outcome of the case is reasonable to use neural networks Hamming and Hopfield.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ОПЫТ ЭКСПЕРИМЕНТАЛЬНОЙ РАЗРАБОТКИ ПРОГРАММНОЙ СРЕДЫ С БАЗОЙ ЗНАНИЙ НА СЕМАНТИЧЕСКИХ СЕТЯХ

Катаев В.А.

*ООО «Центр интеллектуальных технологий»,
г. Пермь, Российская Федерация*

bravo666666@yandex.ru

В работе содержится краткое описание программной среды Multi Studio, которая реализует командный язык Multi на универсальных семантических Multi-сетях.

При разработке MS в качестве образца использовалась информационная среда человека, которая базируется на естественном языке и включает конечный синтаксис, бесконечную расширяющуюся Базу знаний человека и хранилище этой Базы – мозг, состоящий из нейронов и связей между ними.

Аналогично Multi Studio включает высокоуровневый Язык знаний Multi, Базу знаний на семантических Multi-сетях и Машину работы с Базой.

Ключевые слова: семантические сети, язык знаний, база знаний, интеллектуальная система, Multi Studio.

Введение

Потребности в интеллектуальных системах различного назначения стремительно растут. Необходима разработка универсальной технологии создания указанных систем, базирующейся на универсальном языке супервысокого уровня и унифицированных информационных структурах.

Разработка среды Multi Studio (MS) является попыткой экспериментально проверить некоторые идеи, ведущие к решению указанной проблемы.

В этом же направлении весьма масштабно работает Белорусский государственный институт информатики и радиоэлектроники (проект OSTIS, руководитель проекта Голенков В.В.).

В его статьях [Голенков, 2012] и [Голенков, 2013] поставлена задача разработки Метасистемы IMS.OSTIS, предназначенная для комплексной (в том числе организационной) поддержки проектирования интеллектуальных систем.

1. Основные принципы разработки Среды

При разработке MS в качестве образца использовалась информационная среда человека, которая базируется на естественном языке и включает конечный синтаксис, бесконечную расширяющуюся Базу знаний человека и хранилище этой Базы – мозг, состоящий из нейронов и связей между ними.

1.1. Язык знаний

Язык знаний должен быть:

- максимально простым;
- формально однозначным в границах заданных контекстов.

1.2. База знаний

База знаний Среды – это хранилище знаний в электронной памяти, обладающее следующими основными свойствами:

- на физическом уровне – это набор файлов, состоящих из записей (ячеек) весьма сложной структуры. Запись – малый аналог нейрона;
- на логическом уровне – это набор абстрактных узлов с однородной унифицированной структурой, в которых могут располагаться любые одиночные информационные объекты (слова) Языка Среды (числа, даты, команды Языка, имена, отношения ...);
- соединение одиночных информационных объектов в некоторые информационные агрегаты по правилам базового синтаксиса порождает однородные синтаксические сети;
- в сетях объекты-узлы связываются друг с другом направленными переходами, которые формально представляют «родственные» отношения между узлами: «предок-потомок» и «потомок-потомок»;
- синтаксические сети являются основой для семантических сетей, которые представляют собой

различные неоднородные информационные агрегаты: одиночные показатели, списки, множества, матрицы, кортежи, таблицы, деревья, фреймы, правила, онтологии, базы данных, базы знаний, программы;

- семантические сети являются фрагментами единой Базы знаний.

1.3. Машина

Машина работы с Базой знаний состоит из набора программ.

Машина имеет 2 основных цикла работы (возможно разделенных во времени):

- перевод (компиляция) текстов программ и данных в унифицированные внутренние структуры и запись их в Базу знаний;
- чтение программ (и данных) из Базы и их исполнение (интерпретация).

Внутренние интерфейсы между локальными и распределенными фрагментами Баз знаний обеспечивает Язык знаний.

Внешние интерфейсы:

- интерфейс с непосредственным пользователем Среды (обозначения команд Языка знаний, комментарии программ в процессе работы, тексты в Базе знаний);

- осуществляется на текстовых фрагментах естественного языка (Язык пользователя);

- в качестве Языка пользователя может применяться русский язык, английский и другие;

- при этом Машина должна обеспечивать конкретного пользователя простыми средствами создания своих национальных синонимов стандартных обозначений Языка пользователя;

- интерфейс с внешними программными системами, работающими на иных языках и структурах:

- минимально требуется перевод внешних баз знаний в Базу Среды и/или обратно;

- в принципе, программы указанных переводов можно делать на любой стороне. Предпочтительней - на стороне Среды.

Непосредственное использование Среды в качестве Вычислителя (Решателя) производится преимущественно с использованием Языка знаний. Информационные системы, созданные на базе Среды, напротив, общаются с оператором в основном на Языке пользователя.

2. Среда Multi Studio

Multi Studio (MS) – это универсальный многооконный диалоговый интерпретатор команд языка Multi, работающий под управлением грамматики, написанной на самом языке Multi.

Сеанс работы с MS называется беседой и состоит из серии заданий. Задание на языке Multi (в текстовом варианте) записывается в окно ввода.

Результат выполнения задания выдается, как правило, в окно вывода.

Язык пользователя – русский язык. В эксперименте опробован английский.

Язык знаний - Multi – это высокоуровневый командный (алгоритмический) язык с функциональной парадигмой и префиксной формой записи.

2.1. Базовый синтаксис Языка знаний

Описание базового синтаксиса Языка (во внешнем текстовом варианте):

- multi-текст задания – это непустой список выражений, разделенных пробелами;

- выражение [семья] – это предок [объект] и потомство, которого может не быть;

- потомство – это список выражений, заключенный в круглые скобки. Таким образом, выражения могут быть многоуровневыми;

- в конце выражения может присутствовать символ точка с запятой, который явно обозначает конец выражения и позволяет не указывать последние закрывающие скобки выражения;

- объект: первичный (терминальный) или вторичный (составной);

- к первичным объектам относятся: число (-5.667), литерал («Расчет плана»), дата (25.12.2014), имя (Расчет_плана), понятия (`человек, `любит), команды языка;

- команды Языка: Сумма (+), музыкальная студия (MIDI.), другие (более 500); Команды представлены в Базе номерами. Обозначения команд представлены в таблицах команд вне Базы. Пользователь может ввести в таблицу свои национальные обозначения команд в качестве синонимов к стандарту;

- к составным объектам относится: пара (25!метров), тройка (2!40!метров), другие;

- пара состоит из двух первичных объектов, разделенных символом ! :

- +!доллар [вычислить сумму долларов];

- 3!4 [музыкальный размер «вальс»].

Пример Multi-текста из двух выражений:

- *P# ("сумма=" +(X 123 ; X (5000) (1)*

- где P# [вывод на экран объектов потомства через пробел];

- вывод результата: 'сумма= 5123'.

2.2. Multi-сеть представления знаний

2.2.1. Бункер

На физическом уровне База знаний среды MS, называемая Бункером, является файлом. Первоначально в Бункере находится единственная синтаксическая сеть Куча в виде списка, в котором находятся все пустые узлы Бункера.

Бункер подразделяется на множество Сфер. Каждая Сфера имеет свой каталог имен. В каталоге Сферы не допускается дублирование имен. Но в

разных Сферах можно иметь имена-омонимы. Благодаря этому Сферу можно использовать для хранения знаний определенной предметной области, не заботясь о совпадении обозначений объектов с разной семантикой из соседних областей (например, класс в объектном программировании, в образовании и в политике).

2.2.2. Компиляция

При запуске задания исходный Multi-текст компилируется в синтаксическую сеть задания и записывается в Бункер.

На первом этапе для каждого выражения создается отдельная синтаксическая сеть.

Копия объекта (или ссылка на него) помещается в свободный узел. Между узлами устанавливаются переходы:

- открывающая скобка (исходного текста) – переход "предок→потомок";
- пробел между объектами – "потомок→потомок";
- закрывающая скобка – "потомок→пусто".

Каждый переход между двумя узлами предусматривает встречную запись адресов связываемых узлов. Кроме того узлы содержат дополнительные поля, которые используются при работе с узлами. Например, для накопления суммы командой +.

На втором этапе сети выражений связываются в общую синтаксическую сеть задания:

- сети именованных выражений (у которых предок - имя) связываются переходами "предок→потомок";
- Сети неименованных выражений - переходами «потомок→потомок».

При отсутствии в сети синтаксических ошибок Машина запускает задание.

Пример: на рисунке 1 представлен графический вид сети задания (1). Переходы между узлами представлены непрерывными стрелками. Исполнение – пунктирными стрелками.

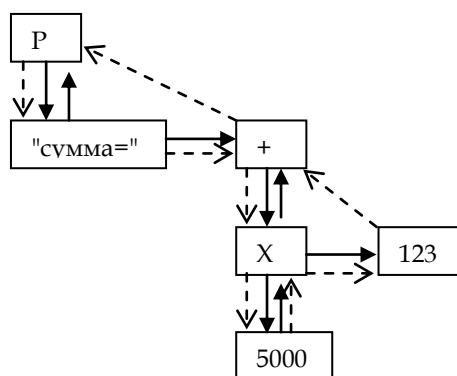


Рисунок. 1 – Сеть задания (1) и маршрут исполнения

2.2.3. Исполнение задания

Машина производит выполнение задания, начиная с первого узла задания путем перехода от узла к узлу преимущественно по принципу "сверху вниз [предок→потомок] и слева направо [потомок→потомок]". Некоторые команды (.. [цикл]) могут изменять маршрут исполнения.

Синтаксически правильное задание может быть прервано из-за семантических ошибок.

2.2.4. Семантика

Базовый синтаксис локален и предельно прост. Его правила распространяются только на выражения одного уровня. Семантика глобальна.

Семантика выражений анализируется не в фазе компиляции, а в процессе их исполнения. Причина: в статической фазе компиляции нет динамических контекстов, необходимых семантике.

По времени и области действия (контексту) различаются семантики Бункера, беседы, задания и выражения. Каждая из 500 команд Языка имеет свою семантику и выступает в качестве параллельно-последовательного агента вместе с другими при выполнении задания.

Имеются глобальные команды, которые действуют на все уровни выражения любого задания через глобальные параметры. Например, команда *сфера.*, которая задает идентификатор текущей сферы Бункера. Семантическая структура выражения этой команды: *сфера.(имя_сферы)*.

Первоначальные значения глобальных параметров задает Машина. Она же сохраняет последние изменения.

Большинство команд являются локальными и действуют только на потомков своего выражения.

Например, команда + имеет следующую семантику: + (потомство). Область ее действия - все потомки-числа всех уровней, являющиеся константами или полученными в результате вычислений другими командами. Другие объекты для нее нейтральны. Если среди нейтралов встречается команда, она прерывает действие + и работает в пределах своего потомства, а по окончании возвращает вычисленное значение и управление старшей команде.

Пример:

P# (+ (111 P# («Привет!») + (5 5 (10))))
 Результат выводится на экран: Привет! 131

Многоагентная работа команд Среды позволяет иметь в одном задании программу оркестровки песни, в текст которой вставлены команды многоуровневой сортировки и вывода таблиц базы данных, а также предикаты универсума Эмбрана, представленные на рис. 4.3 в [Осипов,2012] и все это выполнить: создать и запустить песню, выдать отсортированную таблицу и узнать, имеет ли Петя собаку. Проверено: работает.

Язык Multi обладает несколькими полезными свойствами, облегчающими семантическую отладку программ:

- Повышенный уровень Языка.
- Простейший синтаксис.
- Простая всепроникающая декомпозиция Multi-программ посредством имен. При этом отладку отдельных поименованных фрагментов программы можно вести последовательно и/или даже независимо параллельно. Отсюда существенно уменьшается общая сложность и сроки отладки программ.
- Наличие команды *test*. [пошаговая динамическая диагностика действий программы с любого узла исполнительного маршрута сети].
- Наличие команды *##* [вывод статической структуры сети любого узла любой сложности с заданным числом уровней потомков].
- Визуальное хождение пользователя по Базе знаний в окне Среды.

Заключение

Получен опыт применения Среды для создания различных интеллектуальных систем: экспертная система промышленного назначения, учебный программный имитатор управления самоходной артустановкой, виртуальная звуковая студия с использованием музыкального расширения языка Multi вместо классической нотации. А также имеется опыт использования среды при обучении школьников информатике.

На основе опыта разработок Multi Studio, OSTIS, ИМПЗ [Загоруйко, 2013] и некоторых других рекомендуется открыть международный проект по развитию инструментальных средств разработки интеллектуальных систем.

Библиографический список

[Голенков, 2012] Голенков, В.В. Графодинамические модели параллельной обработки знаний: принципы построения, реализации и проектирования / В.В. Голенков, Н.А. Гулякина // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2012): материалы Междунар. научн.-техн. конф. (Минск, 16–18 февраля 2012 г.); – Минск: БГУИР, 2012, С. 23–52.

[Голенков, 2013] Голенков, В.В. Открытый проект, направленный на создание технологии компонентного проектирования интеллектуальных систем / В.В. Голенков, Н.А. Гулякина // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2013): материалы Междунар. научн.-техн. конф. (Минск, 16–18 февраля 2013 г.); – Минск: БГУИР, 2013, С. 55–78.

[Загоруйко, 2013] Загоруйко Ю.А. Технологии разработки интеллектуальных систем, основанные на интегрированной модели представления знаний // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2013): материалы Междунар. научн.-техн. конф. (Минск, 16–18 февраля 2013 г.); – Минск: БГУИР, 2013, С. 31–42.

[Осипов, 2012] Осипов Г.С. Лекции по искусственному интеллекту. Изд. 2-е, испр. и доп. М.: Книжный дом «ЛИБРОКОМ», 2013. – 272с. (Науки об искусственном.)

EXPERIENCE OF EXPERIMENTAL DEVELOPMENT OF SOFTWARE ENVIRONMENT WITH THE KNOWLEDGE BASE ON SEMANTIC NETWORKS

Kataev V.A.

*Center of Intelligent Technologies, Ltd, Perm,
Russian Federation*

Bravo666666@yandex.ru

The paper presents a brief description of the Multi Studio environment which realizes the Multi command language on universal semantic Multi-networks.

Introduction

Needs for intelligent systems of various purpose grow promptly. That is why it is necessary to develop a universal technology of creation of these systems, based on a universal language of superhigh level and unified information structures.

Development of Multi Studio (MS) is an attempt to test experimentally some of ideas leading directly to the solution of the specified problem.

Main Part

When developing MS, as an example we used the information environment of a human based on a natural language; this environment includes finite-state syntax, the infinite extending Knowledge base of a human and a human brain (consisting of neurons and connections between them) as a repository of this Base.

Similar to this, our Multi Studio includes Multi (the high-level Knowledge Language), the Knowledge base on semantic Multi-networks and the Software to work with the Base.

The Multi-network consists of uniform abstract knots which are connected by directed transitions: "ancestor-descendant", "descendant-descendant". Various information units can be based in Multi-networks, such as single units, lists, sets, matrixes, tuples, tables, trees, frames, rules, ontologies, databases, knowledge bases, programs.

All those features of the environment along with simplicity of the source language allow a user to apply Multi Studio both for its direct purposes as a "calculator", and as a tool environment for developing intelligent information systems.

We have an experimental experience of application of the environment for developing an industrial-purpose expert system, training program simulator of self-propelled assault gun operation, virtual sound studio.

Conclusion

On the basis of experience of development of MS, OSTIS and the others, it is recommended to start the international project on elaboration of Tools for development of intelligent systems.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

РАСПОЗНАВАНИЕ БОТОВ В СОЦИАЛЬНЫХ СЕТЯХ

Кипаева Е.В., Кириченко М.И., Орлова Ю.А., Заболеева-Зотова А.В.

Волгоградский государственный технический университет,

г. Волгоград, Россия

M9iy.aht@gmail.com

Nemozy@list.ru

yulia.orlova@gmail.com

zabzot@gmail.com

В работе изучается описание понятия "Бот", рассматривается применение ботов в различных сферах жизни. Проводится исследование методов защиты от роботизированного внесения информации и общения. В статье также приведены существующие системы распознавания ботов, а также рассматривается бот, как интеллектуальная диалоговая система.

Ключевые слова: бот; обратный тест Тьюринга; методы распознавания; интеллектуальная диалоговая система.

Введение

Интеллектуальные диалоговые системы стали очень распространены в наше время. Данные системы могут использоваться в различных сферах жизни, например, для автоматической проверки знаний людей, для ведения диалогов с пользователями глобальной сети Интернет, для диагностики неисправностей автомобилей, а также для развлечений.

Работа авторов статьи направлена на изучение так называемых виртуальных собеседников или, как их еще часто называют – чат-ботах. В статье описывается понятие сущности бота, его применение и методы распознавания. Также рассматривается бот, как интеллектуальная диалоговая система.

1. Описание понятия бот

1.1. Бот

Бот (англ. bot, сокр. от robot) - программа, автоматически выполняющая действия на компьютере вместо людей. Впервые термин «робот» использовал писатель Карел Чапек в своей пьесе «Россумские универсальные роботы». В семидесятых годах был употреблен впервые термин «бот».

1.2. Применение ботов

Боты могут использоваться в различных сферах жизни, например:

- имитация живого человека при общении в чате;
- автопилот для участия в Интернет-аукционах, купки билетов и прочих ценностей;
- автоматизация рутинных действий, таких как рассылка разнообразных рекламных акций по электронной почте;
- массовые вычисления, в частности взлом паролей и индексация сетевых ресурсов, получение и предоставление приватной информации, включая пароли и номера кредитных карт.

2. Методы защиты от ботов. Обратный тест Тьюринга

Бороться с современными ботами средствами IT-технологий довольно сложно. Какая бы не была придумана защита, через несколько дней она обходится хакерами, и решения борьбы с данной защитой перепродаётся среди взломщиков.

Для защиты от подобных агентов, системы начинают формировать нечто вроде «иммунной системы», а именно на входе производится тестирование интеллекта, где пользователь должен убедить систему, что представляет собой человека (рисунок 1).

Поскольку в данной системе компьютер верифицирует наличие интеллекта у исследуемого объекта, такой тест называется обратным тестом Тьюринга, поскольку прямой тест Тьюринга производится человеком, для определения наличия

у машины интеллекта. Институт Карнеги Меллон ввел специальный термин CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart), что переводится как «Полностью Автоматический Открытый Тест для Отделения Компьютеров от Людей» [Коробкова и др., 2012].

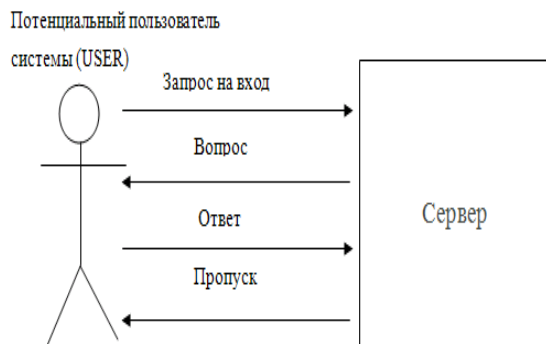


Рисунок 1 – Иммунная система для защиты от интеллектуальных агентов (веб-роботов)

Обратные тесты Тьюринга, могут быть как зрительными так и слуховыми. Типовой пример внешнего вида такого теста на веб-странице показан на рисунке 2.

Введите символы, показанные на картинке.



Буквы вводятся без Слушайте и вводите услышанные числа

Рисунок 2 – Внешний вид обратного теста Тьюринга

В данном примере, сложность теста заключается в искажении символов, затрудняющем применение технологии OCR (Optical Characters Recognition). Однако, как показывают исследования Microsoft, подобные тесты могут быть решены без участия человека с применением нейросетевых алгоритмов [Розалиев и др., 2010].

Более сложные обратные тесты требуют смыслового анализа изображений, фраз, решения задач (рисунок 3).



Рисунок 3 – Обратный тест Тьюринга, требующий от пользователя указать три изображения, представляющих собой «Природу»

Такие тесты не сводятся к решению «усложненной» проблемы OCR, а требуют совершенно иного уровня искусственного интеллекта, для их взлома, пока еще не созданного. К проблемам подобных тестов можно отнести громоздкость, ограниченный запас терминов, ограниченный запас изображений.

3. Существующие системы распознавания ботов

В наше время существует достаточное количество систем для распознавания ботов. Примером такой системы может являться программный продукт Akismet, который автоматически обнаруживает спам (рассылка коммерческой и иной рекламы или иных видов сообщений лицам, не выразившим желания их получать). Данная система постоянно развивается и обучается, что позволяет обнаруживать новых сложных ботов, с которыми другие сервисы не могут справиться. Для личного пользования разработчики предоставляют ключ. Объясняется бесплатное использование тем, что сервису надо обучаться, чтобы более крупные компании и коммерческие проекты могли уверенно использовать на платной основе и не беспокоиться за спам. Положительными сторонами является то, что боты распознаются абсолютно на любых социальных сетях или сервисах общения. Всё зависит от клиентов, от того, как они будут использовать доступ к этому сервису. Минусами для русских разработчиков является то, что разработчик-иностранец и цены на использование сервиса очень высокие [Котенко и др., 2012].

Также примером такой системы является программный комплекс HumanPresent, который блокирует действия, выполняемые автоматизированными системами. По словам главы Pramana Санджая Сегала, компания планирует запустить HumanPresent как в виде SaaS-сервиса, так и в виде программы, предотвращающей активность ботов внутри веб-приложений. Программный комплекс может быть использован для защиты таких онлайн-форм, как бланки регистрации почтовых аккаунтов и транзакций интернет-магазинов, а также для обнаружения автоматизированных кликов по рекламным баннерам. Сегал сообщил о том, что Pramana использует 32 параметра для выявления активности бота на веб-странице. В их числе – особенности и частота нажатия на клавиши мыши и клавиатуры, а также некоторые другие вещи. Например, если человек нажмет на ссылку и ничего не произойдет, он, скорее всего, попытается нажать на нее снова, тогда как бот не станет этого делать.

«Vkontakte Antispam» - система распознавания ботов, предназначенная для социальной сети ВКонтакте. С его помощью можно сократить в разы количество спама в группах и публичных страницах, которыми вы управляете. На данный момент в системе реализован только функционал

выявления и добавления ботов в чёрный список. Программный продукт был создан в 2012 году Дмитрием Москниным. Достоинствами данной системы является усиленная направленность на обнаружение спама с дальнейшей ликвидацией и отправки автора в чёрный список. Хочется отметить, что программа распространяется бесплатно и является довольно эффективной. К недостаткам данной разработки можно отнести то, что она не может проверять ссылки, содержащие пробелы.

4. Критерии для распознавания ботов в социальной сети "ВКонтакте"

Методы распознавания ботов разделены на два вида - на проверку получаемой информации во время присутствия пользователя на ресурсе (онлайн) и на проверку информации, которая уже была ранее залита на онлайн-сервис (оффлайн). Во втором случае информацией могут являться фотографии, которые пользователь выкладывал с последней встречи выпускников, публичные сообщения, информация о зарегистрированном человеке, например, дата рождения, место проживания. По всем этим данным можно определить реальный ли это человек. Сто процентного результата, конечно, получить не сможет даже сам человек, не говоря уже про то, что анализировать данную информацию, может компьютер.

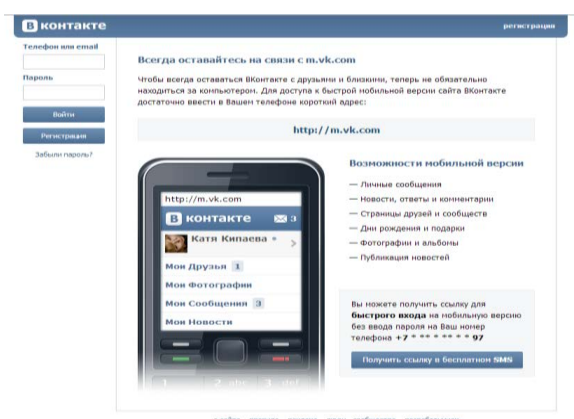


Рисунок 4 – Главное окно при авторизации в социальной сети "ВКонтакте".

Чтобы помочь определить бот это или нет, на сегодняшний день выделяют ряд критериев, которые, как уже писалось ранее, не дают сто процентного результата, но совокупность этих критериев повышает шанс обнаружить бота. Чтобы привести пример такого критерия рассмотрим социальную сеть "ВКонтакте" (Рисунок 4). Если на аватарке (графическое представление пользователя, двумерное изображение (иконка) или трёхмерная модель) стоит изображение с явной рекламой стороннего контента и другие фотографии отсутствуют, то, скорее всего, это будет бот, созданный для рекламных акций какого-либо продукта.

В своей работе мы предлагаем следующий ряд критериев, выделенный для распознавания ботов в социальной сети "ВКонтакте".

Таблица 1 – Список критериев выявления ботов

Критерий	Комментарий
Заблокирован ли пользователь.	Блокировка пользователей в «ВКонтакте» в 99 из 100 случаев происходит из-за вмешательства сторонних программ, которые осуществляют спам-деятельность.
Количество подписчиков.	Количество людей, которые подписаны на пользователя, не должно быть больше 300.
Полнота заполнения полей при регистрации.	Когда регистрируют ботов, то обычно стараются заполнить минимум данных, чтобы избежать лишних затрат времени. Если ботов регистрируют с помощью программных средств, то поля заполняются по максимуму.
Количество комментариев от друзей.	Если большинство комментариев написано не от друзей то, возможно, это бот.
Орфография в комментариях.	Человеку свойственно делать орфографические ошибки. Если за все время существования страницы все сообщения написаны без единой ошибки – это признак автоматизированной системы.
Ведения диалога	Современные автоматизированные системы еще не обучены поддерживать беседу на заданную тему. Поэтому если при общении с пользователем разговор перерастает в форму "вопрос – ответ" то, скорее всего это бот.

5. Бот, как интеллектуальная диалоговая система

Одним из примеров ботов являются чат-боты. Это специальные виртуальные собеседники, созданные для общения в различных Интернет-каналах. В научных исследованиях такие боты называются интеллектуальными диалоговыми системами [Заболеева-Зотова и др., 2010].

Интеллектуальные диалоговые системы работают по общей схеме: воспринимают сообщение пользователя, как запрос и формирует соответствующий ответ на основе имеющихся знаний. Систем, учитывающих в совокупности морфологию, синтаксис и семантику запросов на естественном языке - нет. Несмотря на большой объем работ, проведенных в области обработки естественно-языковых текстов, к настоящему

моменту не предложено механизмов практического их применения для поиска информации [Möller, 2005].

В России довольно много разработано интеллектуальных диалоговых систем, но подавляющее большинство очень примитивны. Данные системы могут только поддерживать беседу на заданную тему по типу вопрос-ответ. Умение поддерживать беседу в таких разработках не реализовано.

В данной статье хотелось бы предложить определенный минимум, которым должна обладать интеллектуальная диалоговая система. Такая система должна обладать следующими функциональными возможностями:

- умение поддерживать беседу на заданную тему, то есть система должна не просто отвечать на вопросы, но и задавать их, спорить, отстаивать свою точку зрения;
- механизм самообучения;
- механизм "запоминания" и "узнавания", то есть система должна запоминать собеседника, чтобы в новом диалоге с ним не пришлось начинать беседу сначала;
- обладать индивидуальностью и эмоциональностью;
- умения выполнять какие-либо действия, например, если собеседник скидывает какую-либо ссылку, то система должна ее открыть.

Интеллектуальная диалоговая система может быть внедрена в различные социальные сети для рекламных акций. Если система будет обладать функциональными возможностями, описанными выше, то и распознать ее будет гораздо тяжелее.

Заключение

Обработка информации и принятие решений при человеко-компьютерном взаимодействии является актуальными задачами, требующими пристального внимания. [Бобков и др., 2011] На современном этапе развития информационных технологий разработка методов автоматического распознавания бота является актуальной задачей, позволяющей решить ряд экономических, социальных, бытовых проблем и играющей важную роль в вопросах безопасности.

Разрабатываемые системы распознавания ботов могут применяться в бизнес-компаниях, где требуется работать с клиентами через Интернет. Использование системы позволит увеличить скорость определения ботов, тем самым, повышая эффективность работы аналитиков. Интерес к применению в своих целях может найти не только организация, но и частное лицо. Например, при общении с пользователем в Интернете, можно в любой момент запустить систему распознавания ботов и убедиться, что ведешь беседу с реальным человеком. С помощью данных программ можно проводить социальные исследования, например, на

количество ботов в той или иной социальной сети. Также такие системы могут применяться в сфере обеспечения безопасности для повышения эффективности работы соответствующих служб на крупных объектах.

Работа частично поддержана Российским фондом фундаментальных исследований (проекты 12-07-00266, 12-07-00270, 13-07-00459, 13-07-97042).

Библиографический список

- [Бобков и др., 2011] Развитие системы автоматизированного определения эмоций и возможные сферы применения / Бобков А.С., Заболеева-Зотова А.В., Орлова Ю.А., Розалиев В.Л. // Открытое образование. - 2011. - № 2. - С. 59-62.
- [Заболеева-Зотова и др., 2010] Автоматизация семантического анализа текста технического задания: монография. / Заболеева-Зотова А.В., Орлова Ю.А. - Волгоград, ИУНЛ. 2010. - 155 с.
- [Коробкова и др., 2012] Обратный тест Тьюринга - противостояние естественного и искусственного интеллекта / Коробкова С. В. [и др.] // Нейрокомпьютерная обработка сигналов и изображений. - 2012. - 8 с.
- [Котенко и др., 2012] Исследование бот-сетей и механизмов противодействия им на основе имитационного моделирования / Котенко И. В. [и др.] // СПИИРАН. - 2012. - 26 с.
- [Розалиев и др., 2010] В.Л. Розалиев, А.С. Бобков, О.С. Федоров Применение нейронных сетей и грануляции при построении автоматизированной системы определения эмоциональных реакций человека / Изв. ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолгГТУ. - Волгоград, 2010. - Вып. 9, № 11. - С. 63-68.
- [Möller, 2005] Möller, S. Quality of Telephone-based Spoken Dialogue Systems / S. Möller, Springer Science + Business Media, Inc. Boston, 2005

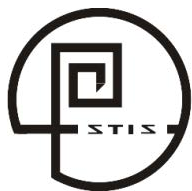
RECOGNITION BOTS IN SOCIAL NETWORKS

Kipaeva E.V., Kirichenko M.I., Orlova Yu.A.,
Zaboleeva-Zotova A.V.

*Volgograd State Technical University,
Volgograd, Russia*
M9i.aht@gmail.com
Nemozy@list.ru
yulia.orlova@gmail.com
zabzot@gmail.com

In this articles the description of the concept of "bot", considered the use of bots in various spheres of life. A study of methods of protection against bots. The article also shows the existing recognition systems bots and bot regarded as intellectual dialogue system.

Intelligent dialogue systems have become very common nowadays. These systems can be used in various spheres of life, such as automatic test people's knowledge, for dialogue with the users of the Internet global network for fault diagnosis of vehicles, as well as for entertainment. Work the authors focused on the study of the so-called virtual companions, or, as they are often called - chat bots. The article describes the concept of the essence bot, its use and recognition methods. Also considered bot as intellectual dialogue system.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.912

АВТОМАТИЗАЦИЯ СЕМАНТИЧЕСКОГО АНАЛИЗА НОВОСТНЫХ ИНТЕРНЕТ-ТЕКСТОВ

Солошенко А.Н., Розалиев В.Л., Заболевая-Зотова А.В.

*Волгоградский государственный технический университет,
г. Волгоград, Россия*

nastyasolan@gmail.com

vladimir.rozaliev@gmail.com

zabzot@gmail.com

Данная статья посвящена проблеме представления неструктурированных новостных сюжетов в сжатом виде с сохранением их смысла. Особое внимание уделено интеграции систем с новостными сайтами и социальными сетями. В работе скомбинированы статистические алгоритмы извлечения ключевых слов и алгоритмы формирования семантической связности блоков текста.

Ключевые слова: семантика, синтаксический анализ, новостной интернет-текст, интеграция с новостными сервисами.

Введение

Развитие информационных ресурсов Internet многократно усилило проблему информационной перегрузки. Еще в начале XXI века американская исследовательская служба Cyveillance сообщила о том, что количество страниц в Internet превысило 4 млрд, и с каждым днем увеличивается на 7 млн. «Сырые» неструктурированные данные составляют большую часть информации, с которой имеют дело пользователи, поэтому многие организации (службы рассылки новостей, информационно-библиотечные системы и др.) и частные лица заинтересованы в эффективных технологиях автоматизированного семантического анализа информации, представленной на естественном языке [Ландэ Д.В., 2005; Розалиев и др., 2010].

При этом необходимо отметить, что темпы роста аудитории онлайн-новостных ресурсов практически вдвое превышают темпы роста общей численности пользователей интернета, и к сегодняшнему дню данная аудитория составляет 43,2% российских интернет-пользователей (согласно исследованию, проведенному сотрудниками Nielsen//NetRatings). Так, недельная аудитория составляет в среднем 6,3 миллиона пользователей: новостных сайтов – 3,9 млн., газет – 3,7 млн., информационных агентств – 2,2 млн., радиостанций и телеканалов – 1,3 миллиона. Число

ежедневных сообщений в Twitter также приблизилось к отметке 400 млн. записей в день, тогда как в апреле 2012 года этот показатель составлял 340 млн. в сутки.

Данные факты говорят о том, что необходимость использования разрабатываемой системы, позволяющей проанализировать и представить информацию в сжатом виде, но с сохранением смысла, с каждым годом будет возрастать. Существующие программные системы полностью не решают данную проблему. Причиной этому является отсутствие моделей и методов, обеспечивающих формализацию и адекватный семантический анализ текстов на естественном языке. Объяснением этого является сложность и неоднозначность решения задачи семантического анализа для различного вида текстов. Кроме того, большинство систем не направлены на обработку новостных текстов.

Основная идея заключается в разработке нового подхода к обработке текстовой информации, основанного на комбинировании нескольких типов подходов: традиционного статистического и более сложного лингвистического.

1. Обзор существующих систем семантического анализа новостных текстов

На международном рынке представлено множество программных продуктов, которые позволяют проанализировать текст с точки зрения семантики.

Среди отечественных стоит выделить АОТ и Semantic Analyzer Group, из зарубежных – мощный инструмент анализа текстов IBM Text Miner. Технология Semantic Analyzer Group, как и АОТ (Автоматическая Обработка Текста) позволяет строить синтактико-семантическую сеть, включает графематический, морфологический и семантический модули. Обе системы работают со словарями и тезаурусами. IBM Text Miner содержит утилиты классификации, кластеризации, поиска ключевых слов и составления аннотации текстов. Однако на обработку новостных статей программы не направлены. [Заболеева-Зотова и др., 2010а]

Российская система Яндекс Новости позволяет автоматически группировать данные в новостные сюжеты и составлять аннотации статей на основе кластера новостных документов. Сервис InfoStream, обеспечивает доступ к оперативной информации в поисковом режиме с учетом семантической близости документов.

Прямым аналогом системы является мобильный агрегатор новостей Summly, купленный в марте 2013 компанией года Yahoo!. Summly – приложение под iPhone, которое позволяет сжимать произвольную статью в резюме до 400 знаков и подбирать подходящие картинки для оформления на экране мобильного устройства. Однако приложение абсолютно неприменимо для обработки текстов на русском языке.

Таким образом, разрабатываемая система призвана устранить недостатки существующих систем, модифицировать существующие решения семантического анализа применительно к новостным текстам на русском языке.

2. Интеграция с новостными сервисами

Для удобства пользователей в разрабатываемой системе заложена возможность получения необходимой статьи с новостного сайта по ее URL, т.е. будет реализована интеграция с наиболее популярными новостными сайтами (например, rbc.ru, newsru.com, expert.ru, kommersant.ru, lenta.ru). Для этого разрабатывается лексический анализатор новостных сайтов, при помощи которого из разметки HTML можно получить необходимый текст.

Далее рассмотрим решение задачи семантического анализа новостного текста по этапам.

3. Методика семантического анализа новостного текста

Решение задачи семантического анализа можно разбить на несколько этапов: предварительная обработка текста (графематический и морфологический анализ), синтаксический анализ и затем непосредственно семантическая обработка.

В основе структуры новости положен принцип «перевернутой пирамиды»: заголовок отражает тему и содержит не более 10 слов, основные факты, касающиеся события, отражены в 1-2 абзацах (лид), 3-й и последующие абзацы составляют бэкграунд (контекст) [Солошенко А.Н. и др., 2013]. В задачу графематического анализа входит внутреннее представление структуры новости: $T = \langle P, S, W \rangle$, где P – абзацы, S – предложения, W – слова. При этом необходимо корректно выделить заголовок и первое предложение абзаца, содержащее основные факты статьи. Для морфоанализа целесообразно использовать словарные методы, за основу можно взять, например, словарь А.А. Зализняка.

В простейшем случае структуру совокупности знаний S текста новости можно определить следующим образом: $S = \{M, F\}$, где M – множество всех понятий данной совокупности знаний, F – отношение «смысловая связь». В качестве формальной модели структуры знаний можно использовать семантическую сеть, определяемую в виде ориентированного графа $G = (E, V)$, где E – множество вершин, поставленное во взаимно однозначное соответствие с множеством понятий; V – множество ориентированных дуг; дуга выходит из вершины, соответствующей основному понятию A , и входит в вершину, соответствующую понятию, которое сочетается по смыслу в тексте с понятием A [Заболеева-Зотова А.В. и др., 2010b; Михайлов Д.В. и др., 2009].

Для построения вышеописанного семантического графа, или семантической сети, необходимо проведение синтаксического анализа, задача которого – выделение синтаксических конструкций, определение связности и подчинения фрагментов. Для поиска фрагментов потребуется шаблон поиска – кортеж $\langle N, S, P \rangle$, где N – нормальная форма искомого слова, S – часть речи и $P = \{p\}$ – множество искомых параметров искомого слова.

После нахождения фрагмента необходимо его преобразование.

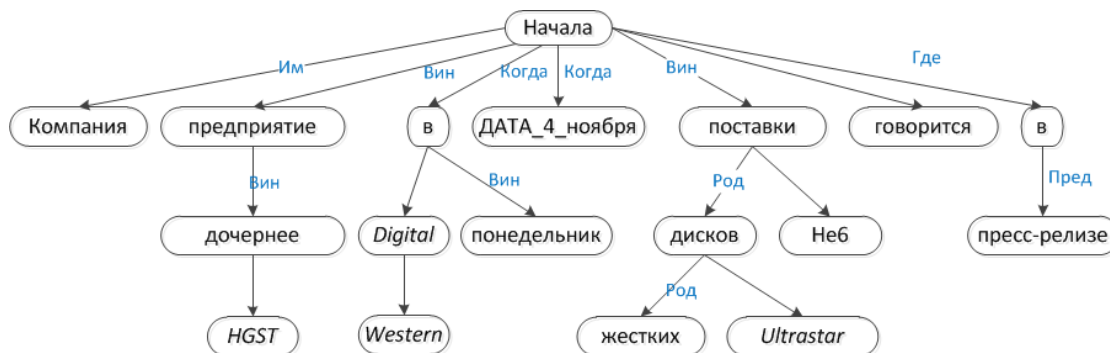


Рисунок 1 – Дерево зависимостей предложения (семантический граф)

Таким образом, правило синтаксической сегментации будет состоять из шаблона поиска фрагмента, шаблона формирования фрагмента и списка исключений. В результате синтаксического анализа получается граф, подобный представленному на рисунке 1 [Большакова Е.И. и др., 2011].

После построения графа, можем выделить наиболее часто встречающиеся в тексте новости субъекты и объекты повествования, построить наглядное облако новости (рисунок 2, размер шрифта отражает важность ключевой фразы), аннотацию.

he6 hgst ultrastar винчестер воздух гелий гермозона говорится
ДИСК люйм жесткий заполнить изображение использоваться клиент
компания поставка пробный раз решение снизить температура трение частность
четырёхтерабайтных

Рисунок 2 – Облако текста

При размещении статей в новостной ленте требуется определять рубрику, к которой относится анализируемая статья. Для этого целесообразно использовать метод латентно-семантического анализ [Машечкин И.В. и др., 2011; Тарасов С.Д., 2008]. На первом шаге требуется составить частотную матрицу индексируемых слов.

Следующим шагом мы проводим сингулярное разложение полученной матрицы. Сингулярное разложение – это математическая операция раскладывающая матрицу на три составляющих. Т.е. исходную матрицу M мы представляем в виде:

$$M = U \cdot W \cdot V^t \quad (1)$$

Где U и V^t – ортогональные матрицы, а W – диагональная матрица. Причем диагональные элементы матрицы W упорядочены в порядке убывания. Диагональные элементы матрицы W называются сингулярными числами.

Достоинство сингулярного разложения состоит в том, что оно выделяет ключевые составляющие матрицы, позволяя игнорировать шумы. Согласно простым правилам произведения матриц, столбцы и

строки, соответствующие меньшим сингулярным значениям, дают наименьший вклад в итоговое произведение. Важно, что при этом гарантируется оптимальность полученного произведения. Таким образом гарантируется достаточное точное определение тематики новостной статьи.

4. Пример применения системы

Рассмотрим пример работы системы на примере новостного сообщения, взятого из Интернет, заглавием которого является строка «Пользователи из США предпочли карты Apple картам Google» (рисунок 4).

Предполагается возможность настройки агрегации новостей по темам или ключевым словам. Для подобранных текстов, кроме аннотирования и построения деревьев зависимости (описанных выше), доступна функция комплексного анализа текста (определение темы, ключевых сущностей, синтаксических и морфологических характеристик), цитирования новостей, сокращенных до определенного формата, в социальных сетях (например, Твиттере). Возможно сохранить понравившиеся обзоры.

Был проведен ряд экспериментов, в ходе которых определено, что время, затраченное на обработку текста вручную составляет 2,5 минут, время обработки с помощью программы – порядка 10 секунд (рисунок 3).

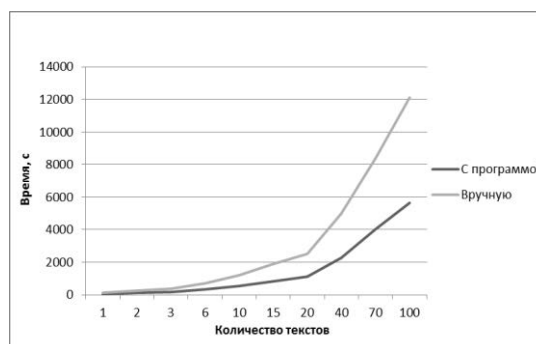


Рисунок 3 – Зависимость времени обработки от количества текстов

Пользователи из США предпочли карты Apple картам Google



Приложение «Карты» для iPhone, выпущенное компанией Apple в сентябре 2012 года, за год обошло Google Maps по популярности в США. Об этом пишет The Guardian, ссылаясь на отчет исследовательской компании comScore.

В течение года после выхода карт Apple аудитория Google Maps на iPhone сократилась на 25 миллионов пользователей. Сейчас она составляет 6 миллионов пользователей, причем у двух миллионов из них на смартфонах установлена iOS 5 и ниже.

Анализ текста

Сохраненные обзоры

Наука и техника

Спорт

Экономика

Политика

Результаты анализа:

1) Предложение, отражающее тему:

Приложение «Карты» для iPhone за год обошло Google Maps по популярности в США.

2) Ключевые сущности, их вес:

Apple, пользователи, iOS, Google, приложение, США, карты, аудитория

3) Синтаксические и морфологические характеристики (на примере заголовка)

Простое предложение, объект: пользователи (сущ. Мн. ч., И.п., подлежащее), субъект: карты, действие: предпочли (глагол. Пр. вр.).

Рисунок 4 – Пример работы системы

Заключение

Таким образом, совмещение статистических и лингвистических алгоритмов семантического анализа позволит улучшить качество обработки новостных статей для последующей публикации в новостных лентах. В условиях нынешнего информационного века, с огромным количеством новостных сообщений в сети интернет, применение таких технологий является необходимостью, так как значительно ускоряет обработку повседневной информации.

Работа частично поддержана Российским фондом фундаментальных исследований (проекты 12-07-00266, 12-07-00270, 13-07-00459, 13-07-97042).

Библиографический список

[Большакова Е.И. и др., 2011] Автоматическая обработка текстов на естественном языке и компьютерная лингвистика : учеб. пособие. / Большакова Е.И. [и др.]. – М. : МИЭМ, 2011.

[Заболеева-Зотова и др., 2010a] Автоматизация семантического анализа текста технического задания: монография. / Заболеева-Зотова А.В., Орлова Ю.А. - Волгоград, ИУНЛ. 2010. - 155 с.

[Заболеева-Зотова и др., 2010b] Автоматизация начальных этапов проектирования программного обеспечения / Заболеева-Зотова А.В., Орлова Ю.А. // Изв. ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. - Волгоград, ВолгГТУ. 2010. - Вып. 8, № 6. - С. 121-124.

[Ландэ Д.В., 2005] Ландэ, Д. В. Поиск знаний в INTERNET. Профессиональная работа. : Пер. с англ. – М. : Диалектика, 2005. – 272 с. : ил.

[Машечкин И.В. и др., 2011] Машечкин, И.В. Латентно-семантический анализ в задаче автоматического аннотирования / Машечкин И.В., Петровский М.И. // Программирование. – 2011. – Т. 37, № 6. – 67-77.

[Михайлов Д.В. и др., 2009] Михайлов, Д. В. Морфология и синтаксис в задаче семантической кластеризации / Михайлов Д. В., Емельянов Г. М. // Математические методы распознавания образов (ММО-14), Владимирская область, Суздаль, 21-26 сентября 2009 г. – 2009.

[Розалиев и др., 2010] Моделирование эмоционального состояния человека на основе гибридных методов / Розалиев

В.Л., Заболеева-Зотова А.В. // Программные продукты и системы: международный науч.-практ. журнал. – Тверь, 2010 – Вып.2 (90). – С.141-146.

[Солошенко А.Н. и др., 2013] Солошенко, А.Н. Автоматизированное составление обзорных рефератов новостных Интернет-текстов / Солошенко А.Н., Орлова Ю.А., Розалиев В.Л., Заболеева-Зотова А.В. // Труды Конгресса по интеллектуальным системам и информационным технологиям "IS&IT'13", п. Дивноморское, 3-9 сент. 2013 г. В 4 т. Т. 1 / Рос. ассоциация искусственного интеллекта, ФГАОУ ВПО «Южный федеральный ун-т» [и др.]. - М., 2013. - С. 233-238.

[Тарасов С.Д., 2008] Тарасов, С. Д. Алгоритм ранжирования связанных структур в задачах автоматического составления обзорных рефератов новостных сюжетов.// RuSSIR'2008, труды Второй Российской конференции молодых ученых по информационному поиску. – Таганрог: Изд-во ТТИ ЮФУ, 2008. – С. 90-100.

AUTOMATION OF SEMANTIC ANALYSIS OF INTERNET NEWS TEXTS

Soloshenko A.N., Rozaliev V.L.,
Zaboleeva-Zotova A.V.

Volgograd State Technical University,
Volgograd, Russia

nastyasolan@gmail.com

vladimir.rozaliev@gmail.com

zabzot@gmail.com

Semantic analysis of the text received in recent years a considerable urgency in connection with development of Internet and catalogs of information resources. This article is devoted to a problem of unstructured news stories representation in compressed form. Particular attention is paid to the integration of systems with news services and social networks, text parsing, developing of models and methods of news texts semantic analysis based on a combination of statistical algorithms for extracting keywords and algorithms forming the semantic coherence of text parts.

Keywords: semantics, syntactic analysis, Internet news texts, integration with news services.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.89:004.4

АГЕНТНЫЙ ПОДХОД К РАЗРЕШЕНИЮ НЕ-ФАКТОРОВ В ЗАДАЧЕ СЕМАНТИЧЕСКОГО СРАВНЕНИЯ ИМЕН СУЩНОСТИ ПРИКЛАДНОГО ПРОГРАММНОГО РЕШЕНИЯ

Бердник В.Л., Заболеева-Зотова А.В.

*Волгоградский государственный технический университет,
г. Волгоград, Россия*

bwlg@inbox.ru
zabzot@gmail.com

В работе рассматриваются вопросы автоматизации семантического сравнения символьных имен на эквивалентность. При эксплуатации широкого класса программ, как простейших учетных систем, так и систем управления жизненным циклом изделия (PLM), остается актуальной задача поддержания перечней материалов, полуфабрикатов и т.п. в актуальном, внутренне непротиворечивом состоянии, а также, нахождение их отражения в прейскурантах поставщиков для автоматизации деятельности служб снабжения.

Ключевые слова: моделирование принципа интуиции; многоагентные системы; семантическое сравнение символьных имен; НЕ-факторы.

Введение

Под сравнением символьных имен в статье будем понимать определение эквивалентности денотатов (кореферентности) имен для последующего прикладного использования в различных системах автоматизации производства, торговли и учета. Пусть S – множество строк символов (символьных названий), D – множество документов d . Каждый документ:

$$d = \langle e, S^d \rangle, \quad (1)$$

где e – денотат, соответствующий сущности реального мира (товар, физическое лицо, услуга, подразделение организации и т.п.);

S^d – подмножество строк S , для которых известен денотат e .

Таким образом

$$S = S^i \cup \left(\bigcup S^d \right), \quad (2)$$

где S^i – подмножество строк S , для которых денотат не определен.

Пример символьного имени:

«Растворитель технический 646 0,5л»

Определение денотата e для строк S^i является практически востребованной задачей в системах автоматизации учета предприятий. Разработанные авторами модели и методы позволяют существенно снизить затраты предприятий в долгосрочном плане. В неавтоматизированном виде решение задачи (ввод в систему приходных накладных, заказов покупателей, созданию прайс-листов, поддержания в актуальном состоянии электронных справочников и т.п.) требует существенных затрат труда оператора ЭВМ. Практически всегда имеется возможность получения электронной версии вводимого документа по электронной почте или сканированием печатной формы, но последующий автоматический ввод в систему учета предприятия не возможен в силу разных способов именования объектов учета (разных строк S^i денотата e).

Рассмотрим пример НЕ-фактора в задаче семантического сравнения символьных названий. Пусть имеется некий документ описывающий товар «Материнская плата ЗРЕ-А». В какой-то момент времени производитель материнской платы модифицирует изделие и появляется в продаже «Материнская плата ЗРЕ-А Green». На уровне модели знаний программной системы неизвестно, какую семантику несет терм «Green» так как этот терм встретился впервые. Возможно это добавление новой подсистемы энергосбережения в изделие, или другой цвет окраски корпуса. Фактор не полноты знаний приводит к нечеткости отношения между

символьным обозначением и денотатом документа. Более подробный анализ вопросов семантического сравнения был представлен в работе [Бердник, 2012a].

Разрешение очередного НЕ-фактора это добавление нового знания о терме, его семантике, семантических взаимосвязях. Каждый новый терм несет новое семантическое значение, которое не выводимо на основе правил из семантики БЗ. Учитывая рассуждения, приведенные выше, применение какого – либо логического аппарата представляется не эффективным.

Как известно, интуиция — непосредственное постижение истины без логического анализа, основанное на воображении, эмпатии и предшествующем опыте. В нашей задаче, интуиция – это совокупность гипотез, построенных на системе закономерностей достоверных знаний системы, а также взаимосвязей внутри текущего фактически существующего комплекса НЕ-факторов.

Интуиция оперирует латентными (скрытыми) знаниями и ассоциациями. Как известно [Чанышев, 2011], слова одного предложения ассоциативно связаны. В нашем случае в качестве предложения выступает строка символьного обозначения, совокупность термов отражает закономерность строения изделия или маркетинга. В символьных обозначениях термы имеет скрытые ассоциации между собой, которые могут быть экстраполированы на новые символьные обозначения. Авторам представляется наиболее удачным моделирование интуиции описанным ранее методом "семантического пятна" [Бердник, 2012b].

Критерии эффективности сравнения

Методы семантического сравнения, основаны на разбиении множества D на подмножества D^i на основе род-видовых отношений таким образом, что каждый род сущностей $g \in R$ отличается присущим только ему набором типов существенных признаков. Для каждого типа существенного признака (например, цвет) задается семантическое поле — совокупность семантических признаков взаимно отделяющие виды сущности друг от друга в пределах рода [Кобозева, 2007] (например, белый, черный, серебристый). Соотнесение строки s^i и рода сущностей g является простой задачей, так как признак рода в строке указывается в явном виде.

Таким образом, задача семантического сравнения включает в себя уточнение цепочки НЕ-факторов [Нариньяни, 1994] за счет сокращения не полноты информации о семантике термов, к разрешению всего комплекса НЕ-факторов, например, между «Материнская плата ЗРЕ-А» и «Материнская плата ЗРЕ-А Green», и выбора сингулярного значения оператором системы.

Эффективность автоматизации семантического сравнения зависит от качества базы знаний, ее полноты, непротиворечивости, нечеткости и т.д. Количественным показателем эффективности автоматизации является значение обратное зависимое от мощности множества D^i .

$$D^i(s^i) = \{d | \mu(d, s^i) > 0\}, \quad \forall s^i \in S^i \quad (3)$$

Ключевыми источниками знаний в нашей задаче являются операторы ЭВМ. Помимо вопросов к оператору, гипотезы для НЕ-факторов можно извлекать из структуры коллекции документов D , из корпусов текстов сети Интернет и т.п. Эффективность автоматизации задачи тем выше чем меньше количество вопросов от системы к оператору. Количество вопросов к оператору ЭВМ должно быть значительно меньше количества вопросов при непосредственном сравнении символьных обозначений за длительный период времени эксплуатации системы.

Требуется построить такую систему автоматизации, которая при ограниченном объеме получения знаний от пользователя ЭВМ обеспечивала фиксацию не менее заданного количества пар $\langle s^i, d \rangle$ в одном сеансе взаимодействия с пользователем, и минимизировала среднюю за длительный период времени ее эксплуатации мощность множества:

$$D^i(s^j) = \{d | \mu(d, s^j) > 0\}, \quad \forall s^j \in S^j, \quad (4)$$

где S^j – наиболее востребованные в будущем операторами ЭВМ символьные обозначения. Под $D^i(s^j)$ будем понимать активные в производственном процессе (например, в торговле - список продаваемых товаров) документы $\forall d \in D^i$ в ограниченный период времени предшествующий текущему, (например, 3 месяца).

Агентный метод разрешения НЕ-факторов

Как правило, в беседе двух людей каждый задаваемый вопрос имеет интуитивную предпосылку. Собеседники могли бы задавать большое число вопросов, восполняя каждую мелочь своего незнания. Однако, каждая из сторон имеет своё виденье ситуации, это видение является гипотезой, дающей ответы на множество примитивных вопросов. В начале беседы собеседники ненавязчиво сверяют свои взгляды – идет корректировка гипотезы. Затем собеседники вопросами и ответами взаимно дополняют свое представление о предмете разговора. Такой способ получения информации продуктивнее списка простых вопросов, позволяет существенно меньше обращаться к собеседнику и более комфортный для человека. Сверенная точка зрения собеседников позволяет экстраполировать ответы на вновь возникающие вопросы. Именно такая концепция

положена в основу разрабатываемого интеллектуального интерфейса.

Как свойственно любой гипотезе, ее достоверность окончательно не известна. Целесообразно в ходе беседы иметь несколько различных гипотез, из которой выбирается более правдоподобная. Кроме того, мы имеем дело с множеством сравниваемых пар $\langle s^i, d \rangle$. В данной постановке задачи отслеживается наличие конкуренции между множеством гипотез для каждой из сравниваемых пар $\langle s^i, d \rangle$. Ограничивающим ресурсом задачи является право задать вопрос пользователю ЭВМ. Средой конкуренции является лабиринт (граф) из семантических элементов (комнат) и взаимосвязями между ними (проходов). НЕ-факторы в таком пространстве являются потайными проходами и комнатами, а также ложными связями (проходами).

Столь нетрадиционная постановка задачи дана здесь для пояснения применения многоагентного подхода в разрешении НЕ-факторов при построении интеллектуального пользовательского интерфейса на основе интуиции.

Архитектура агента

Как известно, агенты — это программы акторы в проблемной области, которые имеют взаимные обязательства, определяемые в процессе диалога, ведут переговоры и координируют передачу информации. [Рассел, 2006]

В разрабатываемой многоагентной среде для каждого d^j создадим виртуальный агент-гуманоид, который в зависимости от цели может создавать реактивные агенты-животные для решения локальной задачи [Braspenning, 1997]. Полученная информация пополняет базу знаний, а агент информирует других агентов. Если агенты выполняют свою цель, то они получают

дополнительное право на ошибку. Агенты являются дружественными, если разрешение НЕ-фактора одного агента с некоторой вероятностью может разрешить НЕ-фактор другого агента, даже если они созданы разными агентами-гуманоидами. Сила такой группы увеличивается в том агенте, от которого зависит разрешение всей цепочки НЕ-факторов.

Агент-гуманоид создается для каждого $d \in D^j$. При создании, ему предоставляется релевантная выборка

$$S^r(d) = \{s | \mu(d, s) > L\}, \quad \forall s \in S^i, \quad (5)$$

где L — пороговое значение, заданное пользователем в политике интеллектуальной системы, μ — рассчитывается по методу TF-IDF. Учитывая, что $d = \langle e, S^d \rangle$, в распоряжении агента имеется множество S^d — строки, для которых семантическая эквивалентность задана $\mu(d^j, s^d) = 1, \forall s^d \in S^d$.

Учитывая что строка есть множество термов, то представляется правдоподобным $\mu(d, s^d) = 1$, если

$$\bigcup (s^r / s^d) = \emptyset, \quad \forall s^d \in S^d, \quad (6)$$

где S^d — строки документа d , s^r — множество термов сравниваемой строки, исходя из вырожденной синтаксической структуры символического обозначения.

Выбор цели агента гуманоида происходит на графе методом «семантическое пятно». Граф предварительно настраивается агентами-животными. Вершинами графа являются термы, семантические поля, документы, роды сущности. Направленным дугам графа назначен вес, который рассчитывается как количество ассоциаций между вершинами.

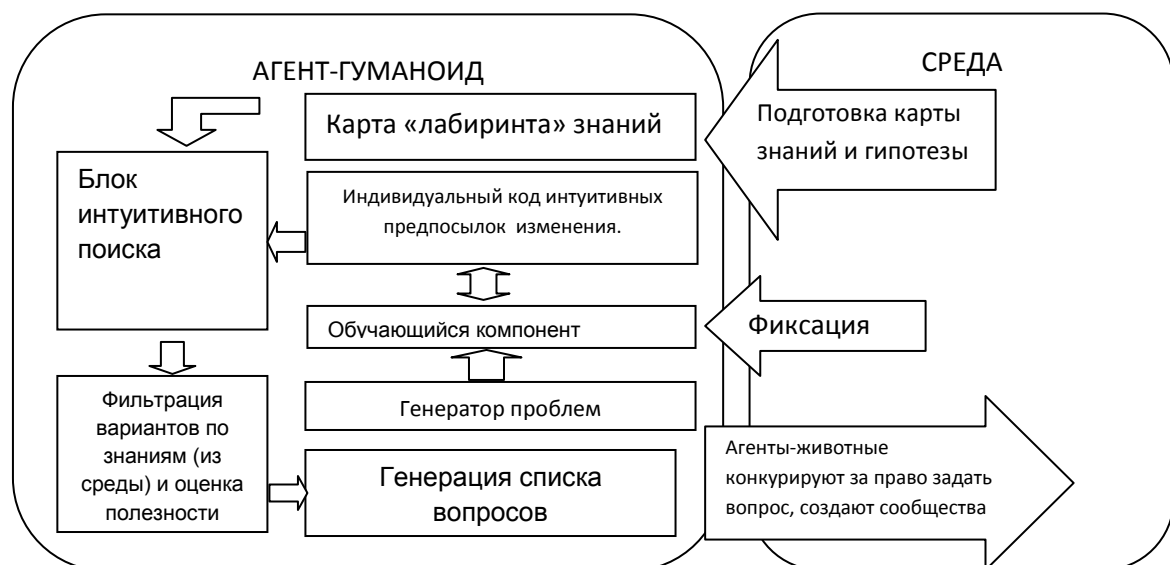


Рисунок 1 - Архитектура интеллектуального агента.

Отношение терм-терм складывается из частоты встречаемости термов в одном символическом обозначении, для терм – документ определяется как количество строк документа, содержащего терм, и т.д.

Далее веса графа нормируются, таким образом, что бы сумма весов всех исходящих вершин равнялась 1. Таким образом, вес ребра графа находится в интервале $[0..1]$. При анализе s^T – множество термов сравниваемой строки находится в вершинах графа. Для найденной вершины задается возбуждение = 1. Затем волновым методом возбуждение распространяется по графу. При прохождении ребра значение возбуждения умножается на вес ребра. Возбуждения, поступившие по разным маршрутам в одну вершину, складываются. Вершины, получившие наибольшее вторичное возбуждение воспринимаются агентом как гипотезы. Так, если не задана фрейм-модель, в качестве гипотезы выбирают наиболее возбужденную вершину рода сущности, для термов, семантика которых неизвестна, выдвигаются гипотезы об их семантических полях из соответствующих возбужденных вершин, относящихся к роду сущности, и т.д.

Полученные таким образом гипотезы сверяются с доской объявлений. Если какая либо гипотеза возникла у другого агента, то они обмениваются адресами. Если для гипотезы не найдена на доске объявлений аналогичная – эта гипотеза помещается на доску объявлений.

Для наиболее возбужденных вершин – документов отыскиваются агенты. Если такие агенты есть, то с ними так же происходит обмен адресов.

Каждая группа выбирает вопрос пользователю ЭВМ. Если гипотеза оказалась верной, каждый элемент группы получает дополнительный бал. Полученное знание фиксируется в БЗ. Если гипотеза неверна, очко снимается со всех агентов группы, и если у агента не было в запасе бала, агент умирает. Выжившие агенты запоминают гипотезы умерших агентов как ошибочные.

Заключение

Работа выполнена при поддержке гранта РФФИ № 13-07-00461

Библиографический список

[Бердник, 2012a] Бердник, В.Л. Семантический анализ символических обозначений в коллекции документов: Монография/ В.Л. Бердник, А.В. Заболеева-Зотова, Ю.А. Орлова; ВолГТУ. – Волгоград, 2012. – 124 с.

[Бердник, 2012b] Бердник В.Л. Модель «семантическое пятно» в сложноформализуемых задачах интеллектуальной обработки информации/ В.Л. Бердник, А.В. Заболеева-Зотова// Известия ЮФУ. Технические науки.-2012.-№ 1.-С. 116-121.

[Кобозева, 2007] Кобозева И.М. Лингвистическая семантика: Учебник. Изд.3-е, стереотипное. М.:КомКнига, 2007.-352с.

[Нариньяни, 1994] Нариньяни А.С. НЕ-факторы и инженерия знаний: от наивной формализации к естественной

прагматике В сб. Труды IV Национальной конференции Искусственный Интеллект94.v.1, Рыбинск 1994

[Рассел, 2006] Рассел, Стюарт, Норвиг, Питер Искусственный интеллект: современный подход, 2-е изд.: Пер. с англ.- М.: Издательский дом «Вильямс», 2006 .- 1408с.

[Чанышев, 2011] Чанышев О.Г. «Ассоциативные поля доминант и анализ текста», Материалы Всероссийской конференции с международным участием «Знания-Онтологии-Теории» (ЗОИТ-11), 3-5 октября 2011 г., Т.2, стр. 126-135, Новосибирск 2011.

[Braspenning, 1997] Braspenning, P. Plant-like, Animal-like and Humanoid Agents and Corresponding Multi-Agent Systems / P. Braspenning // Proceedings of the International Workshop "Distributed Artificial Intelligence and Multi-Agent Systems" (DAIMAS'97, St. Petersburg, Russia, June 15-18, 1997). - P. 64-77.

AN AGENT-BASED APPROACH FOR RESOLVING NON-FACTORS IN THE PROBLEM OF SEMANTIC COMPARISON OF ESSENCE NOTIONS IN APPLIED PROGRAM SOLUTION

Berdnik V.L., Zaboleeva-Zotova A.V.

Volgograd State Technical University

bwlg@inbox.ru

zabzot@gmail.com

Automation of semantic comparison of symbol names is an imminent task of practical application in automation accounting systems of companies. It is basically always possible to get an electronic copy of incoming document by, but subsequent incorporation into a company's accounting system may be difficult due to different ways of naming an accounted item. Architecture of intellectual agent and intuition modeling inference engine by the method of "semantic spot" for that task are reported here. Resolution of non-factor is adding new knowledge about the term of symbolic name, its semantics, semantic inter links. Each new term has a new semantic meaning which cannot be derived on rules of knowledge base. In view of experiments, conducted by the authors and above considerations, the use of any logical device is not deemed efficient. Intuition operates latent knowledge and associations is used here. Normally, agent's behavior is based on knowledge base and an inference engine. Intelligent agents might observable environments to achieve their goals. Possibility to build an agent behavior by intuitive are reported here.

Keywords: modeling of the principle of intuition; multiagent systems; semantic comparison of symbolic names; non-factors.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ОПРЕДЕЛЕНИЕ КЛЮЧЕВЫХ ПАРАМЕТРОВ ЛИЦА ЧЕЛОВЕКА ДЛЯ ПОСТРОЕНИЯ ОБЪЕМНОЙ МОДЕЛИ

Климов А.С., Фоменкова М.А., Орлова Ю.А., Розалиев В.Л.

*Волгоградский государственный технический университет,
г. Волгоград, Российская Федерация*

vladimir.rozaliev@gmail.com

В работе рассматривается задача определения ключевых параметров для построения объемной модели человеческой головы по изображению с произвольной ротацией и наклоном моделируемого объекта. Для восстановления фронтального изображения используется относительное положение характерных точек лица. Изменение положений характерных точек и их описание представляется нечеткими темпоральными высказываниями. 3D-модель строится на основе полученной стереопары изображений – восстановленного фронтального изображения и входного изображения с углом поворота не превышающем 40 градусов.

Ключевые слова: биометрическая идентификация, нечеткое темпоральное высказывание, распознавание лиц, 3D-модель, анализ изображений.

Введение

Для решения задачи построения объемной модели головы человека, необходимо решить ряд задач. В частности необходимо распознать лицо, определить характерные точки, описать их семантическую связность. [Заболеева-Зотова и др., 2010] В статье мы опишем подходы к распознаванию лиц, и метод нахождения характерных точек.

Разрабатываемые подходы и модели будут применяться в создаваемой системе определения эмоциональных реакций человека. [Бобков и др., 2011]

1. Подходы к распознаванию лиц

В настоящее время существует несколько подходов к распознаванию лиц. Однако все подходы имеют свои особенности и ограничения. Критериями эффективности этих методов являются процент распознавания, максимальное измерение ракурса в градусах, условия, препятствующие работе метода, чувствительность к изменению яркости и освещения, характерные точки лица, с которыми работают подходы и иные особенности. [Berk G'okberk и др., 2005], [Viola P и др., 2001], [Fattah Alizadeh и др., 2011].

Подход 1. Сравнение эластичных графов. В данном методе лицо представляется в виде графа.

Вершины графа находятся в характерных точках лица (контур губ, центр глаза, носа и т.д.). Каждой точке соответствует коэффициент разложения по функциям Габор. Набор точек называется джетом.

Джеты необходимы для нахождения определенных точек на двух разных изображениях (нахождение соответствия) и для сравнения соответствующих областей на двух изображениях. Метод способен распознавать лица при угле наклона не более 20 градусов.

Подход 2. Метод главных компонент. Применяется для сжатия информации. Суть метода состоит в линейном ортогональном преобразовании входного вектора в вектор меньшей размерности. Выходные вектора называются собственными лицами. Они могут быть представлены как изображения и выглядят как лица. При распознавании лиц первое собственное лицо является фильтром, который позволяет определить корреляцию изображения лица с самим собой.

Подход 3. Линейный дискриминантный анализ. В отличие от метода главных компонент, линейный дискриминантный анализ не ставит своей целью найти пространство меньшей размерности. Основная задача метода состоит в нахождении подпространства, где разница между объектами, принадлежащими разным классам, будет максимальна. Классы представляются как компактные кластеры, удаленные друг от друга на максимальные расстояния. Таким образом,

кластеры лиц и «не лиц» имеют минимальные пересечения.

Подход 4. Факторный анализ. Данный метод является обобщением метода главных компонент. Основная идея метода состоит в выявлении скрытых зависимостей между наблюдаемыми переменными с помощью определенной совокупности моделей и методов. В основе факторного анализа лежит гипотеза, что переменные являются проявлением небольшого числа факторов. В задаче распознавания наблюдаемые переменные являются признаки объектов.

Подход 5. Метод Виолы-Джонса. Основными принципами, которые используются в методе, являются: интегральное представление; признаки Хаара; бустинг; использование классификатора; построение каскадов классификаторов.

Алгоритм сканирующего окна. Выбирается окно сканирования и используемые признаки. Окно перемещается параллельно на одну ячейку окна. В каждом окне вычисляется около 200000 способов расположения признаков за счет масштабирования. Сканирование происходит последовательно для различных масштабов окна. Все признаки попадают на вход к классификатору, который определяет, лицо это или нет.

Подход 6. Метод сравнения с эталоном. На вход подается изображение с фронтальным расположением лица и определенным для данной базы данных набором масок. Каждая маска представляет собой регион лица (нос, рот, глаза и т.д.). Положения масок для каждого изображения в базе данных нормализованы одинаково. При распознавании изображений части входного изображения сравниваются с частями изображений, хранящихся в базе данных и на основе совпадений изображение классифицируется.

Подход 7. Нейронные сети. При идентификации лиц с помощью нейронных сетей необходимо сначала выбрать архитектуру сети. Существует несколько архитектур нейронных сетей (когнитрон, неокогнитрон, сверточные нейронные сети, радиально-базисные нейронные сети и т.д.). Принципы функционирования нейронных сетей построены на автоассоциативной памяти: на входную совокупность данных – ключ (при идентификации лиц ключом является лицо) – выдается наиболее близкая совокупность той же размерности.

Подход 8. Распознавание по цвету кожи. Существует 2 подхода к распознаванию лиц по цвету кожи. Первый подход базируется на обработке пикселей для всех частей, имеющих цвет человеческой кожи. Каждый пиксель обрабатывается отдельно, а затем определяется принадлежность пикселя к цвету кожи. Затем, опираясь на структуру лица или другие признаки, принимается решение, является ли данный набор

пикселей лицом или нет. Второй подход базируется на статусе региона изображения. При этом подходе необходимо определить регион, где может находиться лицо. И, учитывая данную информацию, решается вопрос, является ли этот регион лицом или нет.

Были проведены эксперименты для определения лучшего подхода. Выборка делалась на группе из 3 человек. Каждый человек тестировал программы, реализующие приведенных выше алгоритмы (программы использовали веб-камеру для получения изображений), следующим образом: поочередно закрывались области лица, и проверялось, будет ли программа распознавать лицо при закрытии данной области. Такими областями лица являлись нос, рот, глаза, лоб, щеки. В результате были выявлены достоинства и недостатки подходов, условия работы. Все подходы обеспечивают высокий процент распознавания. Однако угол поворота головы при распознавании у большинства подходов ограничен. Существенно влияют на работу – закрытие определенных участков лица. Также некоторые подходы не допускают изменения яркости изображения. Таким образом было предложено использовать комбинированный подход. [Розалиев и др., 2010]

2. Характеристики профильного и фронтального изображения лица

Криминалистическая литература содержит описание более 30 антропометрических точек головы человека (рис. 1), относящихся к профильному изображению. Взаимное расположение этих точек характеризует конкретного человека и может служить признаками при его распознавании [Кадомский и др., 2006]. Из данного множества точек можно выделить те, которые видны на изображении и могут быть локализованы достаточно точно (рис. 2) как на профильном так и фронтальном изображении. Рассмотрим их подробнее.

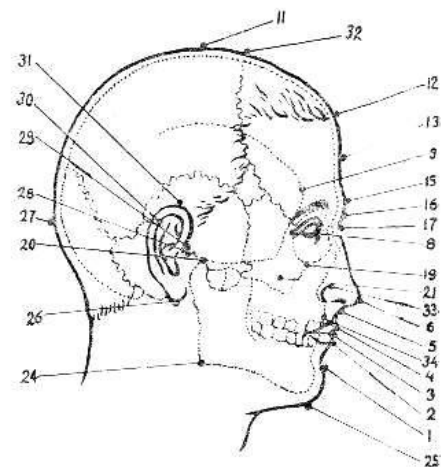


Рисунок 1 - Антропометрические точки головы человека, относящиеся к профильному изображению

Положение подбородочной точки (Е) отличается от антропометрического положения показанного на рис. 1. Данная точка является важной в процессе идентификации, однако она не может быть точно локализована по изображению. Именно поэтому её удобнее сместить из 25 (рис. 1) в Е (рис. 2). Теперь эта точка находится в точке перехода от вертикальной части контура подбородка к горизонтальной.

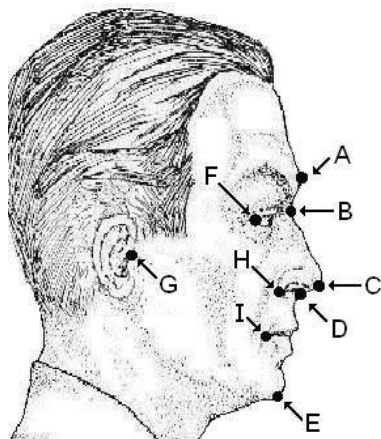


Рисунок 2 - Характерные точки профильного изображения лица

Признаками, характеризующими конкретного человека, служит взаимное положение этих точек. Они позволяют определить высоту носа (l_1), выступание носа (d_1), длину склона носа (спинка) (l_2) и глубину переноса (d_2) указанных на рисунке 3.

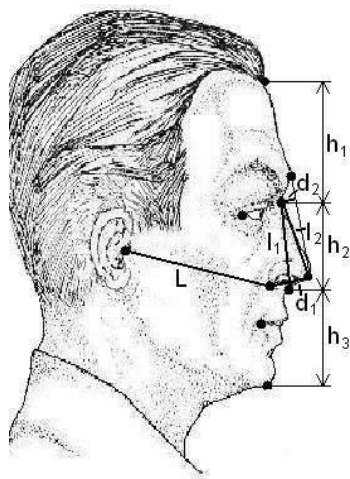


Рисунок 3 - Характеристики профиля лица человека

Поскольку масштаб лица на изображении, как правило, неизвестен, и, следовательно, определить реальные значения этих параметров невозможно, указанные параметры профиля принято относить либо к физиономической высоте лица, определяемой как расстояние от линии роста волос до подбородка.

В таблице 1 и 2 приведены рассматриваемые точки, разделенные на две группы. Точки первой группы принадлежат профильной линии лица. Точки второй группы расположены внутри контура

профиля головы. Такое разделение существенно, поскольку различие в их положении определяет способы их локализации на изображении.

Таблица 1 – Характерные точки фронтального изображения лица человека

Точка	Название	Группа принадлежности
A	Глабелла	Внутренняя
B	Верхненосовая	Внутренняя
C	Кончик носа	Внутренняя
D	Подносовая	Внутренняя
E	Подбородочная	Профильная
F	Наружный край глаза	Внутренняя
G	Надкозелковая точка	Внутренняя
H	Точка примыкания крыла носа к щеке	Внутренняя
I	Угол рта	Внутренняя

Таблица 2 – характерные точки профильного изображения лица человека

Точка	Название	Группа принадлежности
A	Глабелла	Профильная
B	Верхненосовая	Профильная
C	Кончик носа	Профильная
D	Подносовая	Профильная
E	Подбородочная	Профильная
F	Наружный край глаза	Внутренняя
G	Надкозелковая точка	Внутренняя
H	Точка примыкания крыла носа к щеке	Внутренняя
I	Угол рта	Внутренняя

3. Восстановление фронтального изображения лица

Для определения шаблонов характерных точек применяется корреляционный поиск, при этом набор шаблонов сравниваемых лиц должен включать изображения с разным углом поворота головы. Для учета наклона шаблоны поворачиваются по часовой стрелке и против неё на углы в 15 и 30 градусов. Точки локальных максимумов корреляции сохраняются как возможные кандидаты.

После нахождения кандидатов характерных точек необходимо построить из них «правильное» лицо. Для изображения «правильного» лица правый глаз находится слева, а левый – справа, верхненосовая точка располагается между ними. Эти три точки должны образовывать линию, близкую к прямой. То же самое относится и к краевым точкам и середине рта. Верхненосовая и подносовая точка, кончик носа и подбородочная точка образуют вертикальную линию также близкую к прямой. Надкозелковые точки должны находиться сбоку от краев глаз. Расстояния между точками должны определенным образом соответствовать масштабу шаблонов, а наклоны линий глаз и рта – их наклону. Кроме того, следует учесть, что переносица смещается от среднего

положения на линии, соединяющей края глаз, а центр рта смещается по линии, соединяющей края рта. Причем величина этого смещения зависит от поворота головы.

Перебирая все возможные комбинации найденных характерных точек, и учитывая суммарную величину их корреляций, а также оценивая выполнение указанных выше критериев «правильности» лица, можно выбрать такую комбинацию характерных точек, которая наилучшим образом соответствует лицу человека.

Взаимное расположение найденных характерных точек позволяет определить углы наклонов и поворота головы на изображении. Угол поворота головы α определяется исходя из положения точек переносицы и внешних краев глаз, при этом угол поворота не должен превышать 40 градусов. Алгоритм определения углов подробно рассмотрен в [Агарков и др., 2005]. Найденные углы поворота позволяют построить точечный скелет фронтального изображения лица.

Теперь имеется стереопара из фронтального и входного изображения с известным углом поворота и наклона головы. Этих данных достаточно для определения длины, ширины и высоты носа, выступления подбородка, глубины посадки глаз, ширины рта и носа и др. параметров необходимых для построения объемной модели лица человека.

Заключение

Использование комбинации подходов позволяет добиться большей точности определения лиц, а описанный метод нахождения характерных точек позволяет находить точки и восстанавливать фронтальное изображение в автоматическом режиме при углах поворота не превышающих 40 градусов. Получение характерных точек необходимо для построения семантически связанной сети и построения нечетких темпоральных высказываний, описывающих активность каждой такой точки. [Заболеева-Зотова и др., 2011] Полученная стереопара изображений позволит также в автоматическом режиме рассчитать ключевые параметры человеческого лица и построить его объемную модель.

Работа частично поддержана Российским фондом фундаментальных исследований (проекты 12-07-00266, 12-07-00270, 13-07-00459, 13-07-97042).

Библиографический список

- [Агарков и др., 2005] Агарков А.В., Нюнькин К.М. Восстановление фронтального вида лица человека по одному изображению // Искусственный интеллект. – 2005. – № 1. – С. 4-12.
- [Бобков и др., 2011] Развитие системы автоматизированного определения эмоций и возможные сферы применения / Бобков А.С., Заболеева-Зотова А.В., Орлова Ю.А., Розалиев В.Л. // Открытое образование. - 2011. - № 2. - С. 59-62.
- [Заболеева-Зотова и др., 2010] Автоматизация семантического анализа текста технического задания:

монография. / Заболеева-Зотова А.В., Орлова Ю.А. - Волгоград, ИУНЛ. 2010. - 155 с.

[Заболеева-Зотова и др., 2011] Применение нечетких темпоральных высказываний для описания движений при эмоциональных реакциях / Заболеева-Зотова А.В., Орлова Ю.А., Розалиев В.Л., Бобков А.С. // Изв. ВолгГТУ. Серия "Актуальные проблемы управления, вычислительной техники и информатики в технических системах". Вып. 10 : межвуз. сб. науч. ст. / ВолгГТУ. - Волгоград, 2011. - № 3. - С. 60-64.

[Кадомский и др., 2006] Кадомский К.К., Нюнькин К.М. Определение характеристик профильного изображения лица человека // Искусственный интеллект. – 2006. – №1. – С. 138-146.

[Нюнькин, 2004] Нюнькин К.М. Определение углов наклонов головы человека на изображениях // Искусственный интеллект. – 2004. – № 1. – С. 243-250.

[Розалиев и др., 2010] В.Л. Розалиев, А.С. Бобков, О.С. Федоров Применение нейронных сетей и грануляции при построении автоматизированной системы определения эмоциональных реакций человека / Изв. ВолгГТУ. Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»: межвуз. сб. науч. ст. / ВолгГТУ. - Волгоград, 2010. - Вып. 9, № 11. - С. 63-68.

[Снетков и др., 1984] Снетков В. А., Виниченко И. Ф., Житников В. С., Зинин А. М., Овсянникова М. Н. Криминалистическое описание внешности человека: учебное пособие/ Под общей редакцией профессора В. А. Снеткова. – М.: ВНИИ МВД СССР, 1984. – 128 с.

[Berk G'okberk и др., 2005] Berk G'okberk, M.O. 'Irfano'glu, Lale Akarun, and Ethem Alpaydin. Selection of Location, Frequency and Orientation Parameters of 2D Gabor Wavelets for Face Recognition // Advanced Studies in Biometrics, Vol. 3161, pp. 138-146., 2005.

[Viola P и др., 2001] Viola P., Jones M.J. R. Rapid object detection using a boosted cascade of simple features // IEEE Conf. on Computer Vision and Pattern Recognition. – Kauai, Hawaii, USA, 2001. – V. 1. – P. 511–518, 2001.

[Fattah Alizadeh и др., 2011] Fattah Alizadeh, Saeed Nalouisi, Chiman Savari. Face Detection in Color Images using Color Features of Skin // World Academy of Science, Engineering and Technology, 52, 2011.

DETERMINATION OF FACE KEY POINTS FOR 3D-MODEL CONSTRUCTION

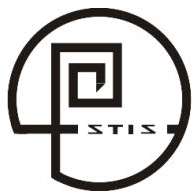
Klimov A.S., Fomenkova M.A., Orlova Yu.A.,
Rozaliev V.L.

*Volgograd State Technical University, Volgograd,
Russian Federation*

vladimir.rozaliev@gmail.com

This article considers the problem of face key parameters determination to construct 3D-model of human head, from image with arbitrary and tilts rotation. The frontal restoration uses relative position of face's key point. 3D-model is based on stereo pair of images, obtained frontal and input image with rotation angle not exceeding 40 degrees. Modern biometric identification systems are based on comparing methods of input images and already existing in database. It is necessary that the angles of these face images was identical. However, to comply with this condition is not possible, so you need to convert the input image to the same format, by restoration of frontal facial image. Another vulnerability of modern facial recognition systems is their problem with flat images.

Keywords: biometric identification, fuzzy temporal cognition statement, face recognition and 3D-model, image analysis



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8

ПРИНЦИПЫ ИНТЕГРАЦИИ СОДЕРЖИМОГО RDF-ХРАНИЛИЩ В ПРОЕКТ OSTIS

Каешко А.И.^{*}, Колб Д.Г.^{**}

^{} Институт пограничной службы Республики Беларусь,
г. Минск, Республика Беларусь
ondister@gmail.com*

*^{**} Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь
kolb@bsuir.by*

Рассмотрена общая идея трансляции содержимого RDF-хранилищ в семантические сети с базовой теоретико-множественной интерпретацией. Описаны основные этапы трансляции, подробно разобран один из этапов.

Ключевые слова: транслятор, N-triples, Semantic Web, RDF, SC-код, онтологии

Введение

Высокие темпы развития информационных технологий в области онтологий и онтологического инжиниринга сопровождаются развитием различных направлений, форм и средств представления онтологий [Загоруйко, 2011].

Для многих предметных областей уже созданы обширные онтологии, такие как SNOMED в медицине, инженерные онтологии PhysSys, EngMath, существуют и межпредметные онтологии, такие как NAICS или SUMO. Стоит отметить, что большинство онтологических ресурсов являются англоязычными. Некоторые онтологии являются международными, и активно используются, например международная классификация болезней 10 пересмотра (МКБ-10).

Наиболее популярным направлением, в рамках которого разрабатываются онтологии верхнего уровня, онтологии предметных областей и прикладные онтологии [Соловьев, 2006], является Semantic Web. С момента рождения направления в 2001 году создано огромное количество онтологий для различных предметных областей.

Одной из ключевых задач развития проекта мы считаем интеграцию уже накопленных другими разработчиками знаний в рамках баз знаний систем, построенных на базе технологии OSTIS.

Основными проблемами, стоящими на пути к решению данной задачи являются:

- разнородность систем метаданных,

использующихся в различных онтологиях;

- особенности описания предметной области присущие конкретному инструменту, к примеру, использование только бинарных отношений (OWL, RDF).

1. Системы метаданных.

С ростом количества информационных ресурсов, поддерживающих стандарты Semantic Web, появилась необходимость унификации представления хранящихся в них знаний. Решением этой проблемы стало использование для представления знаний в информационных ресурсах систем метаданных. В настоящее время существует несколько десятков проектов, связанных с разработкой систем метаданных [Антопольский, 2012]. Одним из наиболее популярных проектов, направленных на решение проблемы унификации представления знаний в виде семантических сетей, стал проект «Дублинское ядро» [DC, 2012]. Основными результатами проекта являются словари метаданных общего назначения, стандартизирующие описание ресурсов с помощью различных RDF-форматов.

Анализируя работы по развитию метаданных можно сделать вывод о том, что основными проблемами, связанными с использованием систем метаданных, являются:

- абстрагируемость большинства систем метаданных от технологий представления и обработки информации, что не позволяет формировать метаописания всего информационного

ресурса и приводит к проблемам “глубинного web” [Ландэ,2009];

- ориентация большинства систем метаданных на создание метаописаний на каком-то одном уровне представления информации, что в итоге создает проблемы при описании самого ресурса и его фрагментов или совокупности ресурсов в рамках одной системы метаданных;

- наличие систем метаданных, предназначенных для решения одной задачи, но построенных на разных понятийных базисах, которые могут частично пересекаться или не пересекаться вовсе. Ярким примером может служить универсальные системы метаданных GILS [GILS, 2012] и Dublin Core, в которых ключевые элементы пересекаются.

Таким образом, назревает необходимость использования инструментов, которые могли интегрировать и унифицировать существующее многообразие систем метаданных. Наиболее перспективным в этом отношении нам представляются семантические сети с базовой теоретико-множественной интерпретацией, развиваемые в проекте OSTIS.

С точки зрения проекта OSTIS система метаданных определяется как язык семантических сетей, имеющий определенную семантику в рамках некоторой предметной области или класса предметных областей.

Каждому классу предметных областей с заданной сигнатурой можно поставить в соответствие язык семантических сетей с заданным алфавитом и набором ключевых узлов [Голенков, 2011]. Каждому из языков семантических сетей можно поставить в соответствие множество конструкций SC-кода, в которых, кроме ключевых узлов самого SC-кода используются ключевые узлы, определяемые сигнатурными элементами соответствующего класса предметных областей. Каждое такое множество конструкций SC-кода будем называть sc-языком, ориентированным на представление структурных моделей предметных областей соответствующего класса.

Определим sc-язык, как язык семантических сетей, ориентированный на представление структурных моделей предметных областей определенного класса.

Каждый sc-язык задается алфавитом (который совпадает с алфавитом SC-кода) и семейством ключевых узлов, которое включает в себя семейство ключевых узлов SC-кода.

Таким образом, процесс интеграции систем метаданных в проект OSTIS заключается в определении SC-языка или SC-языков, ключевыми узлами, которых являются выделенные в рамках систем метаданных понятия.

В настоящий момент в рамках проекта OSTIS выделено пятнадцать SC-языков, которые постоянно дорабатываются и совершенствуются на основе ключевых элементов существующего теоретико-множественного аппарата и универсальных систем метаданных, разрабатываемых мировым научным сообществом.

2. Инструменты представления онтологий и их свойства.

В настоящее время определено несколько крупных научных проектов, в которых сформировался инструментарий, позволяющий создавать семантическое описание различных предметных областей. В первую очередь здесь необходимо отметить проект OpenCyc [Лапшин,2010] – старейшее хранилище онтологий со своим уникальным инструментарием, проект schema.org от крупнейших поисковых гигантов, определяющий свой инструмент для разметки web-ресурсов (microdata) и проект Semantic web, который на сегодняшний день является “законодателем моды” среди онтологического инструментария, на который ровняется большинство проектов.

Несомненным плюсом проекта Semantic web является ориентация на независимую распределенную разработку онтологий. Знания о предметной области могут накапливаться и уточняться постепенно, с участием большого числа людей, без постоянного согласования [Трофимов, 2011].

Однако инструменты Semantic web не дают ответа на вопрос, как избежать добавления в онтологию противоречивых утверждений и что делать, если противоречия возникнут.

Наиболее обсуждаемыми проблемами инструментария Semantic web и в частности языка OWL являются:

- что моделировать при помощи классов, а что при помощи экземпляров. В инструментах Semantic web существует неоднозначность при определении классов и их экземпляров;

- еще одним недостатком является отсутствие возможности естественным образом определять свойства у свойств. Это не позволяет моделировать атрибуты у предметных отношений, n-арные отношения и атрибуты у атрибутов;

- ориентация проекта на web и близкое к машинному представление семантических сетей;

- не развиты стандарты представления переменных во времени и нечетких предметных областей;

- слабо проработанный уровень верификации онтологий на противоречивость и полноту.

Очевидно, что, не смотря на свою популярность, инструменты Semantic web имеют ряд недостатков. В первую очередь они связаны с тем, что инструменты Semantic web изначально были нацелены на машиноориентированное описание информационных ресурсов в web-пространстве без учета комплексного решения проблем семантического представления с точки зрения теории искусственного интеллекта.

В противоположность инструментам Semantic web инструменты проекта OSTIS имеют строгую теоретико-множественную трактовку и не привязаны к конкретной прикладной области. Что

обеспечивает более компактное и формально точное представления информации [Голенков, 2011].

Это определяется рядом свойств, которые позволяют говорить о языковых средствах проекта OSTIS, как наиболее предпочтительном средстве интеграции знаний из различных источников:

- как и в языках Semantic web в технологии OSTIS отдается предпочтение бинарным отношениям, однако существует возможность представления отношений любой арности;
- отношения представляются в виде узлов семантической сети, что позволяет характеризовать их свойства;
- экземпляры отношений выделяются как отдельные узлы семантической сети, что дает возможность характеризовать каждый экземпляр отношения уникальным образом;
- в алфавите ключевых узлов и дуг имеются элементы для описания нечетких, негативных и нестационарных объектов.

3. Этапы трансляции содержимого RDF-хранилищ в тексты SC-кода

На основании частей 1 и 2 данной работы представим общую идею трансляции содержимого RDF-хранилищ в тексты SC-кода.

Предварительный этап: Преобразование наиболее популярных систем метаданных в sc-языки. В рамках такого преобразования учитывается, что различные системы метаданных могут пересекаться по ключевым элементам. Само преобразование осуществляется вручную, ввиду важности такого преобразования, и наличия, как правило, небольшого количества ключевых элементов в рамках систем метаданных. Пересекающиеся элементы различных систем метаданных оформляются одинаково.

Этап 1: Определение использования известных транслятору систем метаданных в рамках RDF-хранилища. Согласно документации Semantic web, используемая система метаданных оформляется в виде RDF-словаря и помещается в начале определения любого RDF-документа. В рамках этапа возможна верификация RDF-документа на предмет корректного использования системы метаданных в данном документе. Такая верификация может быть осуществлена на основе схем отношений, представленных в определении RDF-словаря конкретной системы метаданных.

Этап 2: Трансляция с использованием существующих описаний систем метаданных. На данном этапе осуществляется преобразование той части содержимого RDF-хранилища, которая соответствует выявленным в RDF-хранилищах системам метаданных.

Этап 3: Трансляция нераспознанных фрагментов RDF-хранилища. На этом этапе части содержимого RDF-хранилища, которые не соответствуют известным транслятору системам метаданных, преобразуются предопределенную форму (для простоты в бинарные ориентированные отношения). Таких форм можно

определить несколько в зависимости от типа хранилища или типа языковых средств используемых в хранилище.

4. Трансляция нераспознанных объектов RDF-хранилища

4.1. Характеристики RDF и SC-кода

Язык RDF и SC-код имеют несколько общих черт. Во-первых, оба они могут быть представлены в виде графа, имеющего узлы и дуги. Во-вторых, все элементы такого графа должны иметь уникальные идентификаторы.

Атомарным объектом в RDF является триплет: СУБЪЕКТ – ПРЕДИКАТ – ОБЪЕКТ (рисунок 1).

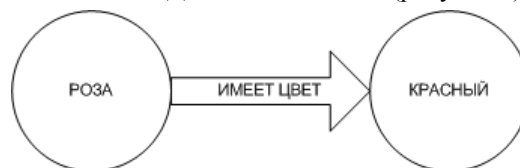


Рисунок 1 – Триплет RDF. «РОЗА», «ИМЕЕТ ЦВЕТ», «КРАСНЫЙ» – уникальные идентификаторы.

Субъект и объект являются узлами, а предикат направленной дугой, всегда указывающей от субъекта к объекту [RDF Concepts, 2013]. Уникальность идентификаторов должны обеспечивать международные идентификаторы ресурсов (IRI, international resource identifier) [RDF Semantics, 2013]. Считается, что посредством таких триплетов можно описать любой объект и отношения между этими объектами. Таким образом, множество триплетов образуют семантическую сеть. Для ограничения количества предикатов и исключения их противоречивости и дублирования используются онтологии верхнего уровня (Дублинское ядро), словарь RDFS, и описательную логику OWL. Однако технически ничто не мешает использовать произвольные предикаты в пространстве имен документа [RDF Freebase, 2013].

Простейшая трансляция представленного выше RDF графа в SC-код осуществляется с помощью введения неролевого отношения (рисунок 2).

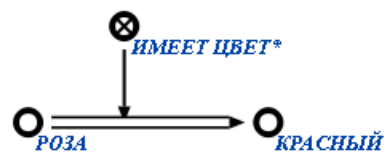


Рисунок 2 – Результат трансляции, представленный в виде SCg-конструкции.

Безусловно, трансляцию необходимо осуществлять между строковыми (линейными) представлениями RDF и SC-кода. При этом семантика RDF графа должна быть сохранена максимально. Для RDF форматом, наиболее удобным для понимания человеком и одновременно легким для машинного чтения, является N-triples, для SC-кода – SCs-код. В N-triples можно транслировать документ любого формата, выполненный на основе модели RDF.

Спецификация формата N-triples рекомендуется консорциумом Всемирной паутины (W3C) [RDF N-Triples, 2013]. Рекомендации от 5 ноября 2013 года предполагают, что каждый триплет содержится в одной строке и состоит из субъекта, предиката и объекта, разделенными символом табуляции или пробела. Каждая строка с триплетом заканчивается точкой. Субъект может быть представлен IRI ссылкой или пустым узлом (blank node), предикат всегда IRI ссылка, а объект может быть представлен IRI ссылкой, пустым узлом или литералом.

Как уже указывалось выше, IRI ссылки предназначены для обеспечения уникальной идентификации объектов. В N-triples они помещаются между символами «<» и «>». Например, `<http://example.org/#РОЗА>`.

Литералы используются для идентификации значений, таких как текст, даты, числа. Литералы заключаются в кавычки. Они подразделяются на литералы, относящиеся к определенному языку, литералы, имеющие тип данных, и простые литералы. Язык литерала указывается после символа «@», например, «Это литерал на русском языке»`@ru`. Тип данных литерала указывается после знаков «^^», и является IRI, например `"2"^^<http://www.w3.org/2001/XMLSchema#integer>`. Простые литералы не имеют явного указания типа, но, по сути, имеют тип данных `<http://www.w3.org/2001/XMLSchema#string>`.

Пустые узлы всегда начинаются с символов «_:» и должны иметь уникальные идентификаторы в пределах документа. Пустые узлы стоит отметить отдельно. В RDF они могут использоваться для обобщенных утверждений. Например, триплет `<РОЗА> <ИМЕЕТ ЦВЕТ> _:x`, означает, что роза имеет какой-то цвет, но при этом не уточняет, какой именно. И наоборот, триплет `_:x <ИМЕЕТ ЦВЕТ> <КРАСНЫЙ>`, указывает, что какой-то объект имеет красный цвет. Пустые узлы так же используются для описания реальных объектов классов. Описание знакомств пользователя «Алексей» может выглядеть так:

```
_:bn1 <ОБЪЕКТ_КЛАССА> <ЧЕЛОВЕК>
_:bn1 <ИМЯ> "Алексей"
_:bn1 <ДАТА_РОЖДЕНИЯ>
"30.01.1979"^^<ДЕНЬ_МЕСЯЦ_ГОД>
_:bn1 <ЗНАЕТ> _:bn2
_:bn1 <ЗНАЕТ> _:bn3
_:bn2 <ИМЯ> "Илья"
_:bn2 <ДАТА_РОЖДЕНИЯ>
"12.05.1981"^^<ДЕНЬ_МЕСЯЦ_ГОД>
_:bn3 <ИМЯ> "Мария"
_:bn3 <ДАТА_РОЖДЕНИЯ>
"01.01.1967"^^<ДЕНЬ_МЕСЯЦ_ГОД>
```

Таким образом, синтаксис N-triples довольно простой. Предикат всегда является IRI ссылкой и объясняет связь субъекта с объектом. Кроме того, разнообразие предикатов теоретически ограничено словарями и онтологиями представления.

4.2. Этапы трансляции RDF-графов в sc-конструкции

Транслятор RDF-графов в sc-конструкции должен:

1. обладать высокой скоростью работы;
2. быть платформенезависимым, то есть поддерживать все аппаратные архитектуры и операционные системы, поддерживаемые технологией OSTIS;
3. обеспечивать семантическую и логическую эквивалентность исходных RDF-графов и получаемых sc-конструкций;
4. быть автоматизированным.

Трансляция RDF-графов в sc-конструкции должна состоять из последовательных этапов:

1. Трансляция RDF-дампа в SCs-код. На этом этапе должен быть выполнен перевод текстов формата N-triples в тексты SCs-кода по заранее определенным правилам трансляции. Данный этап должен быть максимально быстрым и автоматизированным. Реализация транслятора будет зависеть от платформы, на котором он будет выполняться, поэтому алгоритмы его работы должны быть максимально простыми и легко воспроизводимыми в трансляторах для различных платформ и операционных систем в будущем.

2. Трансляция SCs-кода в базу знаний в виде нового компонента. Выполнение этапа будет осуществляться с использованием уже разработанных инструментов технологии OSTIS. Платформенная независимость таких инструментов обеспечивается и поддерживается командой разработчиков OSTIS. На этом этапе необходим контроль архитектора информационной системы, так как ошибки, допущенные при трансляции на предыдущем этапе, становятся видны. Как правило, это неправильная или некорректная идентификация узлов. Однако может встречаться и некорректное для SC-кода определение отношений.

3. Интеграция нового компонента с существующей базой знаний, с использованием метасистемы OSTIS. На данном этапе происходит оптимизация структуры sc-конструкций, согласование понятий новой базы знаний с уже существующими.

4.3. Подход к трансляции N-triples в SCs-код

Документы, представленные на языке N-triples, часто имеют большие размеры. Дамп freebase, например, имеет размер более 100 гигабайт, и хранится в архиве Gzip, который позволяет построчное считывание файла без его полной распаковки. По причине больших объемов файлов трансляция из N-triples в SCs-код должна быть однократной. На языке C++ (компилятор MinGW) был создан прототип транслятора (далее транслятор). Разберем основные реализованные принципы его работы.

По аналогии с рисунками 1 и 2 триплет N-triples, состоящий только из IRI

`<http://kaiko.getalp.org/kaiko/ontology/colors.owl>`

```
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://www.w3.org/2002/07/owl#Ontology> .
```

при трансляции в scs-предложение будет выглядеть следующим образом:

```
colors => rdfs_type : w3owl_Ontology;;
```

Как видно из примера, у IRI выделяется пространство имен, а все символы, кроме цифр, букв заменяются символом «_». Так, название онтологии

```
<http://kaiko.getalp.org/kaiko/ontology/colors.owl>
```

заменяется идентификатором «colors», а у предиката пространство имен

```
<http://www.w3.org/1999/02/22-rdf-syntax-ns>
```

заменяется префиксом «rdfs». Эти видоизмененные IRI будут являться системными идентификаторами в базе знаний OSTIS. Субъект и объект транслируются в scs-узлы, а предикат должен быть нерольевым отношением. Поэтому, транслятор создает файл _norole_rdfs_type.scs следующего содержания:

```
rdfs_type <- sc_node_norole_relation;;
rdfs_predicates -> rdfs_type;;
rdfs_type => nrel_main_idtf: [rdfs_type*_type the ru id
by manual] (* <- lang_ru;; *);;
rdfs_type => nrel_main_idtf: [rdfs_type*]( * <- lang_en;;
*);;
```

То есть, каждый новый предикат включается во множество нерольевых отношений и RDF предикатов, так же для него определяются глобальный русскоязычный и англоязычный идентификатор. Русскоязычный идентификатор предиката необходимо отредактировать вручную, задав значение на русском языке. Это неприятная особенность, но, даже для онтологии SUMO, содержащей более чем 500000 триплетов и использующей собственный словарь предикатов, их число не превышает 70. В примере выше отредактированное scs-предложение с русским идентификатором будет выглядеть так:

```
rdfs_type => nrel_main_idtf: [Объект класса*] (* <-
lang_ru;; *);;
```

И на SCn-коде транслированное в базу знаний scs-предложение

```
colors => rdfs_type : w3owl_Ontology;;
```

будет иметь вид:

```
colors
=> Объект класса*:
w3owl_Ontology
```

SCs-код имеет аналоги пустых узлов N-triples. Они могут быть локальными (уникальными в пределах одного SCs документа) и глобальными (уникальными в пределах одной транзакции трансляции SCs-кода в базу знаний). Идентификаторы таких узлов не используются как системные и начинаются с символов «_». Трансляция идентификатора пустого узла N-triples в пустой узел Scs-кода заключается в замене символов «_» на символы «.».

При трансляции литералов типы данных и идентификаторы языков относятся к множеству

типов литералов и определяются как узлы, не являющиеся отношением:

```
numeric <- sc_node_not_relation;;
lit_types -> numeric;;
numeric => nrel_main_idtf: [numeric_type the ru id by
manual] (* <- lang_ru;; *);;
numeric => nrel_main_idtf: [numeric]( * <- lang_en;;
*);;
```

Они так же, как и предикаты имеют описание в отдельном файле и число таких файлов, как правило, не большое. Значение литерала при трансляции помещается в scs-рамку. Триплет

```
<M83_3> <has_Description>
"Osteomalacia"^^<w3xml#string>
```

после трансляции будет иметь вид (рис. 3):

```
icd_10_M83_3 => icd_10_has_Description :
[Osteomalacia] (* <- w3xml_string;; *);;
```

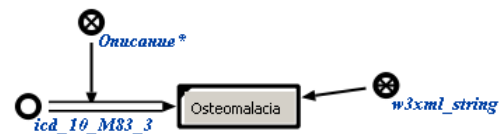


Рисунок 3 – SCg-код результата трансляции триплета с типизированным литералом.

Таким образом, технологически трансляция N-triples в SCs-код особых проблем не вызывает, однако ряд особенностей имеется.

Технология Semantic Web ориентирована в первую очередь на использование во Всемирной паутине. По этой причине часто IRI, содержат ссылку на RDF документ (то есть являются URL), описывающий данный IRI. Например, IRI <http://www.rdfabout.com/rdf/usgov/geo/us/ny> является идентификатором штата Нью-Йорк в Соединённых Штатах. Если перейти по указанному IRI, можно получить RDF-документ, описывающий Нью-Йорк. Такой подход, безусловно, хорош в сети, но в режиме оффлайн, при трансляции документа, такие IRI не будут иметь описание.

Определенной проблемой являются особенности синтаксиса N-triples в различных онтологиях и дампах. Так, идентификаторы пустых узлов могут быть заключены в символы «<» и «>», не смотря на то, что этот подход не описан в рекомендациях W3C. Часто в литералах используются символы форматирования текста, например символ переноса строки. Учет всех этих допущений влияет в основном на скорость работы транслятора.

Ничем не ограниченное, кроме рекомендаций, множество предикатов и типов литералов создает трудности для автоматического согласования метасистемы OSTIS и онтологий представления RDF. Такое согласование необходимо производить вручную на следующих этапах трансляции.

Если говорить не только о фактах, представленных в виде триплетов, то стоит упомянуть RIF (Rule Interchange Format). Онтологии, построенные с использованием модели RDF, имеют богатые возможности описания знаний, но обладают весьма ограниченными средствами вывода следствий из имеющихся знаний. В RDF все

знания должны храниться явно в виде триплетов (фактов, аксиом). RIF является языком правил, и создан для вывода следствий, однако он имеет ограниченные описательные возможности. Комбинация RDF и RIF позволяет использовать достоинства обоих и устранить их недостатки. Таким образом, для получения новых триплетов RDF используются документ с правилами RIF и факты RDF. Использование RIF не обязательно, если онтология изначально наиболее полная, более того перед трансляцией можно получить все новые факты на основе правил, то есть пополнить онтологию.

Заключение

В настоящее время практически реализована трансляция нераспознанных RDF-графов в сконструкции. Оптимизация алгоритмов трансляции будет продолжена, но уже сейчас можно говорить о реально существующей совместимости технологий Semantic Web и OSTIS на уровне линейных форматов представления знаний.

Библиографический список

- [Антопольский, 2012] Антопольский, А.Б. Исследование и разработка системы метаданных для электронных информационных ресурсов и сервисов в фундаментальной науке / А.Б. Антопольский, В.И. Ауссем, С. А. Блау, А.И. Жежель // Отчет о результатах работ по гранту РФФИ № 04-07-90087 [Электронный ресурс]. – 2012. – Режим доступа: <http://rd.feb-web.ru/antopolsky-04.htm#4>. – Дата доступа: 3.04.2012.
- [Голенков, 2011] Голенков, В. В. Принципы построения массовой семантической технологии компонентного проектирования интеллектуальных систем / В.В. Голенков, Н. А. Гулякина // Материалы международной научно-технической конференции OSTIS-2011. – 2011. – Минск: БГУИР. С. 21-58
- [Голенков, 2013] Голенков, В. В. Открытый проект, направленный на создание технологии компонентного проектирования интеллектуальных систем / В.В. Голенков, Н. А. Гулякина // Материалы международной научно-технической конференции (OSTIS-2013). – Минск: БГУИР, 2013, С. 55-77.
- [Загорюлько, 2011] Загорюлько, Ю. А. Подход к построению интеллектуальных информационных систем на основе семантических сетей. – В кн. Междунар. научн.-техн. конф. «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2011). Материалы конф. [Минск, 10-12 февр. 2011 г.]. – Минск: БГУИР, 2011, с. 15-20.
- [Ландэ, 2009] Ландэ, Д. В. Глубинный веб информационная среда для бизнес аналитика / Д. В. Ландэ // Информационные технологии для менеджмента, 2009, № 9, с. 28-32
- [Лапшин, 2010] Лапшин, В.А. Система Сус и её библиотека онтологий / В. А. Лапшин // Искусственный интеллект и принятие решений, 2010, № 2, с. 42-53
- [Соловьев, 2006] Соловьев, В.Д. Онтологии и тезаурусы / уч. пособие [Электронный ресурс], Режим доступа: <http://bookre.org/reader?file=786973&pg=0> // Д.В. Соловьев, Б. В. Добров, В.В. Иванов, Н.В. Лукашевич. – Казань, Москва. – 2006. – 157 С.
- [Трофимов, 2011] Трофимов, И. В. Эволюция выразительных способностей языка OWL / И. В. Трофимов // Программные системы: теория и приложения : электрон. научн. журн. 2011. № 4(8), с. 85–94. URL: http://psta.psiras.ru/read/psta2011_4_85-94.pdf
- [DC, 2012] The Dublin Core Metadata Initiative [Электронный ресурс]. – 2012. – Режим доступа: <http://dublincore.org/>. – Дата доступа: 30.04.2012
- [GILS, 2012] Global Information Locator Service (GILS). [Электронный ресурс]. – 2012. Режим доступа: <http://www.gils.net>. – Дата доступа: 3.04.2012.
- [GRDDL, 2007] Gleaning Resource Descriptions from Dialects of Languages (GRDDL). W3C Recommendation 11 September 2007 [Electronic resource]. – 2007. - URL: <http://www.w3.org/TR/grddl> – Дата доступа: 28.11.2013.
- [RDF Concepts, 2013] RDF 1.1 Concepts and Abstract Syntax. W3C Candidate Recommendation 05 November 2013 [Electronic

resource]. – 2013. - URL: <http://www.w3.org/TR/2013/CR-rdf11-concepts-20131105> – Дата доступа: 19.11.2013.

[RDF N-Triples, 2013] RDF 1.1 N-Triples. A line-based syntax for an RDF graph. W3C Candidate Recommendation 05 November 2013 [Electronic resource]. – 2013. - URL: <http://www.w3.org/TR/2013/CR-n-triples-20131105> – Дата доступа: 18.11.2013.

[RDF Freebase, 2013] RDF Overview. Freebase API. [Electronic resource]. – 2013. - URL: <https://developers.google.com/freebase/v1/rdf-overview> – Дата доступа: 18.11.2013.

[RDF Semantics, 2013] RDF 1.1 Semantics. W3C Candidate Recommendation 05 November 2013 [Electronic resource]. – 2013. - URL: <http://www.w3.org/TR/2013/CR-rdf11-mt-20131105> – Дата доступа: 18.11.2013.

BASIC PRINCIPLES OF INTEGRATION OF RDF-STORAGE CONTENT INTO PROJECT OSTIS

Kayeshko A.I.* , Kolb D.G.**

**Institute of the border service of the Republic of Belarus, Minsk, Republic of Belarus*

ondister@gmail.com

*** Belorussian state university of informatics and radioelectronics, Minsk, Republic of Belarus*

kolb@bsuir.by

In the article research the approaches and possibilities to translate N-Triples format of the RDF-model into SCs-code. Implementation of the possibility a RDF to SCs-code translation will reduce the time of the filling an OSTIS knowledge storages by a Semantic Web technology content.

Introduction

To ensure semantic interoperability between intelligent information systems use ontologies. One of the most promising technologies of intelligent information systems is the Open Semantic Technologies for Intelligent Systems (OSTIS). For fast filling OSTIS knowledge storages, necessary to use an already created ontology.

Main Part

Ontologies created on the basis of model RDF, ubiquitous. The format of the RDF-model, most convenient for human understanding and at the same time the machine-readable, is N-triples, for SC-code such format is SCs-code. All RDF-model based documents can be translated in N-triples document.

Due to the large size of some RDF-dumps, selected single-pass translation algorithm. During the process of translation separated and described in separate files predicates and literals defining entity classes. Due to an unlimited number of predicates and literals it is difficult to create the automatic translator.

Conclusion

The proposed approach to translation RDF N-triples in SCs-code is fairly primitive, mainly due to a single-pass translation algorithm. Translator optimization will continue, but now we can to talk about real compatibility between Semantic Web technologies and OSTIS at linear format of knowledge representation.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.934

ПРИМЕНЕНИЕ МЕТОДА СЕМАНТИЧЕСКОГО ДИФФЕРЕНЦИАЛА ДЛЯ ОЦЕНКИ ПОКАЗАТЕЛЕЙ КАЧЕСТВА КОНВЕРСИИ ГОЛОСА

Захарьев В.А., Петровский А.А.

Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь

zahariev@bsuir.by

palex@bsuir.by

В докладе рассмотрены возможности применения метода семантического дифференциала для оценки показателей качества конверсии голоса. Стандартные подходы, основанные на субъективных методах оценок MOS и ABX, в первую очередь предназначены для оценки степени разборчивости речи, и не могут быть напрямую применены для определения степени близости голосов дикторов. Представлена методика проведения оценки качества конверсии на основе построения семантического пространства слушателей.

Ключевые слова: семантический дифференциал, психолингвистика, конверсия голоса.

Введение

Речевой интерфейс является одной из форм организации взаимодействия с интеллектуальной системой. По эффективности и удобству данный способ является оптимальным для пользователя, поскольку сама по себе речь, как канал коммуникации между людьми, является наиболее естественной формой её организации. Поэтому в данный момент вопросам реализации речевых интерфейсов в рамках исследования и разработки интеллектуальных систем уделяется немалое внимание.

Важное место в структуре речевого интерфейса интеллектуальной системы занимает подсистема синтеза речи по тексту, обеспечивающая реализацию обратной связи от системы к человеку. Развитие такого типа систем на настоящем этапе достигло значительного прогресса. Перед ними ставятся задачи не просто обеспечения заданных показателей разборчивости синтезируемого речевого сигнала, но и предъявляются требования по натуральности речи, наличию широкого спектра просодических шаблонов, поддержки множества языков синтеза и различных голосов дикторов. Последний аспект является наиболее важным и интересным для исследования, особенно в контексте перехода от систем синтеза к системам клонирования голоса, конкретного диктора [Лобанов, 2008]. Для решения задачи по реализации свойств мультимодальности для синтезатора речи

применяются различные технологии обработки исходного речевого сигнала, в том числе конверсия голоса.

Конверсия голоса является технологией обработки речевого сигнала, позволяющей реализовать процесс трансформации параметров голоса, характеризующих речь исходного диктора, в параметры целевого. Объектами конверсии голоса, как технологии обработки сигналов, являются стабильные во времени свойства говорящего, проявляющиеся в речевом сигнале через изменение его акустических параметров [Stylianou, 2009].

Необходимо отметить, что очень важным этапом при разработке и реализации такой технологии является оценка результатов качества конверсии. Аудиторная степень “близости” или соответствия сконвертированного голоса голосу целевого диктора, положенная в основу многих методов оценки качества конверсии является очень обобщенной и нечёткой метрикой, к тому же сильно субъективированной. В работе предлагается рассмотреть возможность использования метода семантических дифференциалов для построения более объективной методики оценки качества конверсии голоса.

1. Стандартные методы оценки показателей качества конверсии голоса

Для оценки качества конверсии голоса в системах конверсии используются два подхода на

основе объективных и субъективных методов оценок. Суть объективных методов заключается в определении степени несоответствия акустических параметров сконвертированного голоса голосу целевого диктора. Другими словами, на определении расстояния между двумя векторами в пространствах акустических признаков дикторов. Например, на евклидовой метрике расстояния между параметрами результирующей и целевой спектральной огибающей закодированной с помощью коэффициентов линейного предсказания. Недостатком данного вида методов является невысокая корреляция полученных оценок с субъективными представлениями о качестве результирующего речевого сигнала для конкретного слушателя. Второй подход опирается на использование субъективных методов оценки, учитывающих мнения реальных субъектов выступающих в роли экспертов.

1.1. Метод на основе средней оценки мнений

Наиболее широко используемая методика субъективной оценки качества речевого сигнала описана в Рекомендации ITU-T P.800 и известна как методика средней оценки мнений (Mean Opinion Score – MOS), изначально применяющаяся для оценки качества передачи в телефонных сетях [ITU-T, 1996]. В соответствии с MOS качество речи, получаемое при прохождении сигнала от источника через систему связи к приемнику, оценивается как арифметическое среднее от всех оценок, выставленных экспертами после прослушивания речевого материала. Количественные результаты этих тестов отображают усредненное качество, уровень усилий слушателя, разборчивость, естественность звучания. Экспертные оценки определяются по пяти балльной шкале: 5 – отлично, 4 – хорошо, 3 – приемлемо, 2 – плохо, 1 – неприемлемо. В случае оценки качества конверсии голоса экспертам предоставляется запись сконвертированной и целевой фонограмм, и предлагается кроме основных оценок, определить меру соответствие голосов дикторов в речевом материале по дополнительной шкале “узнаваемости”.

1.2. Метод на основе слепого тестирования

Вторым распространённым субъективным методом определения оценок качества конверсии голоса, является подход на основе слепого тестирования или так называемый ABX-тест [Clark, 1982]. В области аудио так обычно называют метод определения слышимой разницы между двумя звуковыми фрагментами. Метод наиболее эффективен для определения потенциальных различий находящихся вблизи порога слышимости. Главным преимуществом теста является то, что эксперт не знает какой именно фрагмент (сконвертированной речи или речи целевого диктора) воспроизводится в данный момент. Другой важный момент – возможность многократно повторения теста, что значительно

уменьшает влияние случайности. Производится тест следующим образом. Эксперт присваивает кнопке А один фрагмент, кнопке В – второй, а программа проведения тестирования случайным образом присваивает кнопке Х один из них (эксперту неизвестно, какой именно). Далее, посредством нажатий на кнопки, эксперт может в любой последовательности неограниченно слушать А, В и Х, после чего должен определить чему соответствует Х: А или В. Затем тест повторяется. По мере выполнения каждого прохода, программа подсчитывает вероятность “заблуждения” эксперта, т.е. вероятность того что он не услышит разницу между фрагментами речи содержащими голос сконвертированный системой и голос целевого диктора. Например, если эксперт правильно ответил при первом проходе, вероятность будет 50 %, если и второй проход даст такой же результат – 25%, и т.д. Положительным (эксперт действительно слышит различия) результат считается после как минимум 13-ти правильных ответов в 16-ти проходах.

Оба из перечисленных методов успешно и широко используются для оценки качества систем конверсии голоса. Однако, они также не лишены определенных недостатков. Легко видеть, что данные методы не были разработаны специально для проведения тестирования систем конверсии голоса. Так, например, тест MOS использовался для оценки качества речи передающейся по различным каналам связи, а тест ABX является общей методикой слепого тестирования применяемого во многих областях статистических исследований. Следствием этого является не самая высокая степень адекватности оценок получаемых по данным методам, метрики которых не отражают в полной степени характеристики голоса говорящего с точки зрения auditors. Они опираются на такие общие плохо определённые, размытые и сильно интегрированные категории оценок отраженные в таких шкалах как “разборчивость”, “узнаваемость”, “соответствие”, слабо связанных с особенностями восприятия экспертами личностных черт голоса диктора. Еще одной проблемой является неустойчивость данных оценок, которая выражается в их вариабельности во времени в зависимости от различных условий и речевого материала. Например, для одной пары дикторов для различных фраз данные оценки могут быть различными. Для построения более обоснованного пространства оценок предлагается рассмотреть возможность применения метода семантического дифференциала.

2. Оценка на базе метода семантического дифференциала

2.1. Метод семантического дифференциала

Метод семантического дифференциала (СД) принадлежит к методам экспериментальной семантики и является одним из методов построения субъективных семантических пространств. Техника

СД предназначалась для измерения различий в интерпретации понятий испытуемыми. Этот метод был разработан в 1952 г. группой американских психологов во главе с Ч. Осгудом в ходе исследования механизмов синестезии [Osgood, 1952]. Как полагает Ч. Осгуд, метод СД позволяет измерять так называемое коннотативное значение – те состояния, которые следуют за восприятием символа-раздражителя и необходимо предшествуют осмысленным операциям с символами, тем самым позволяя выявить общую меру, на основе которой выносятся человеческие оценки. В методе СД измеряемые объекты (понятия, изображения) оцениваются по ряду биполярных градуированных (трех-, пяти-, семибалльных) шкал, полюса которых заданы с помощью вербальных антонимов. Оценки понятий по разным шкалам, вообще говоря, коррелируют друг с другом, и с помощью факторного анализа удается выделить пучки таких высококоррелирующих шкал, сгруппировать их в факторы. Психологическим механизмом, объясняющим взаимосвязь и группировку шкал в факторы, Осгуд считал явление синестезии – психологический феномен восприятия, заключающийся в том, что раздражение одного органа чувств, вызывает специфические ощущения в другом органе чувств, незадействованном в данном конкретном случае, например, переживание цветового образа в ответ на музыкальную фразу в цветомузыке. Механизмы синестезии признаются основой метафорических переносов в высказываниях типа “бархатный голос”, “кислая физиономия” и т.п.

Переход от признаков, заданных шкалами, к факторам фактически является построением семантического пространства. Фактор можно рассматривать как смысловой инвариант содержания входящих в него шкал, и в этом смысле факторы являются формой обобщения прилагательных-антонимов, на базе которых строится СД, а его факторная структура отражает структуру антонимии в лексике. Группировка шкал в факторы позволяет перейти от описания объектов с помощью признаков, заданных шкалами, к более емкому описанию с помощью меньшего набора категорий-факторов, представив содержание объекта в виде совокупности факторов (многочлен) с различными коэффициентами веса. Нагрузки объекта по каждому из выделенных факторов определяются как среднее арифметическое оценок объекта по шкалам, входящим в этот фактор.

Семантическое пространство является своеобразным метаязыком описания значений, позволяющим путем разложения их содержания в фиксированном алфавите категорий-факторов проводить семантический анализ этих значений, выносить суждения об их сходстве и различии путем вычисления расстояний между соответствующими значениями координатными точками в пространстве. Традиция использования метода семантического дифференциала, как перспективного метода изучения восприятия

слуховых сигналов, возникла в отечественной науке еще в начале 70-х годов по инициативе В.И.Галунова, и продолжена в работах В.Х.Манерова, В.Ф. Петренко и др [Галунов, 1978., Петренко, 1988].

2.2. Методика оценки на базе метода семантического дифференциала

Для оценки показателей качества конверсии голоса методика использования семантического дифференциала представляется совокупностью следующих основных шагов:

Шаг 1. Для группы экспертов путем поочередного прослушивания фонограмм исходного и целевого диктора производится формирование и тестирование списка прилагательных, положенных в основу шкал сравнения. От выбранных признаков зависит тот уровень осознанности, на котором аудитор будет оценивать измеряемый объект. Обязательное условие – они не должны содержать прямых характеристик объекта (голос – мужской, тембр – низкий), а обязаны иметь ассоциативные характеристики (голос – мягкий, тембр – звонкий). Возможно применение уже апробированного набора шкал. За основу факторного пространства предлагается взять предложенной в работе [Манеров, 2012] полученных при изучении восприятия мужских и женских голосов, протестированную на большом наборе из представителей разных социальных слоев, обладателей различной речевой культуры, обеспечивавшей большую репрезентативность выборки. Более подробно шкалы, сгруппированные по факторам представлены на шаге 2.

Шаг 2. Далее экспертам предлагается прослушать два разбитых, на несколько блоков, набора фраз, озвученных целевым диктором и сконвертированных системой. Система установок аудитора по отношению к значимым для него характеристикам голоса обнаруживается в его оценочных суждениях, которые классифицируются сознанием по схеме логических дихотомий. Пары противоположных эмоционально-оценочных прилагательных, распределенных по факторам представлены ниже

- Фактор силы (мужественный – женственный, басовитый – тонкий, сильный – слабый);
- Фактор активности (бодрый – вялый, активный – пассивный, быстрый – медленный);
- Оценка (теплый – холодный, хороший – плохой, нежный – грубый, мягкий – жесткий);
- Ненормативность (аспект оценки, связанный с качеством тембра и артикуляции: звучный – сиплый, звонкий – глухой, нехриплый – хриплый, четкий – расплывчатый, красивый – некрасивый);

Таким образом, даже при наличии в методике большого числа денотативных шкал была получена четкая структура, близкая к Осгудовской. Природа четвертого фактора получается смешанной, в нем присутствует и эстетическая оценка, и

характеристики денотативной природы, связанные речевой культурой, качеством звучания и артикуляции. Оценки усредняются по количеству блоков в тестовой выборке и количеству экспертов.

Шаг 3. Производится математическая обработка полученной матрицы данных. Для этого используется процедура корреляционного и факторного анализа, позволяющая выявить латентные (скрытые) критерии оценивания, в которые складываются первоначальные шкалы. Поскольку нами были взяты за основу уже разработанные шкалы, факторы по которым они группируются, для удобства сразу были отражены на предыдущем шаге. В результате, производится выделение факторов, формирующих семантическое пространство голосов слушателей-экспертов.

Шаг 4. С помощью методов линейной алгебры определяется расстояния между прослушанными голосами в семантическом пространстве и на основе его величины делается вывод о качестве конверсии голоса. Степень качества конверсии голоса, производимая той или иной системой определяется на основе метрики расстояния в данном семантическом пространстве, и является величиной обратно пропорциональной данному расстоянию.

Заключение

В докладе рассмотрены вопросы создания методики оценки на основе применения метода семантического дифференциала для определения показателей качества конверсии голоса. Стандартные подходы, основанные на субъективных методах оценок MOS и ABX, не всегда полностью удовлетворяют требованиям по адекватности оценок сравнения различного рода систем конверсии. Поскольку методики, которые они предлагают, а также категории оценок, не учитывающие всего феноменологического богатства смысловых, эмоциональных, мотивационных, осознаваемых и неосознаваемых составляющих субъективного образа – голоса диктора, будут, по крайней мере, неполными. При решении данной проблемы необходимо глубокое проникновение в смысловое пространство слушателей, причем не только как на уровне отдельных экспертов, но и на уровне их группы. Предложенная методика позволяет прийти к формализации представления о свойствах голосов и факторов, которые могли бы наиболее точно отразить степень восприятия близости одного диктора к другому, тем самым позволяя определить более объективную оценку соответствия целевого и сконвертированного голоса. Применение метода семантического дифференциала даёт возможность перейти от непосредственного сравнения голосов дикторов, к сравнению, условно выражаясь, инвариантного представления экспертов о самих дикторах в их семантическом пространстве.

Предложенная методика является еще одним шагом на пути устранения границ в областях междисциплинарных исследований на основе

подходов искусственного интеллекта. Подтверждая возможности использования психосемантики для решения вопросов связанных с оценкой качества обработки сигналов.

Библиографический список

- [Лобанов, 2008] Компьютерный синтез и распознавание речи / Б. М. Лобанов, Л. И. Цирульник — Минск : Белорусская наука, 2008. — 344 с.
- [Stylianou, 2009] Voice conversion: a survey / Y. Stylianou // Proc. of International Conference on Acoustics, Speech and Signal Processing. — Taipei, 2009. — P. 3585 - 3588.
- [ITU-T, 1996] Methods for subjective determination of transmission quality // ITU-T Recommendation P. 800: Telephone transmission quality. — 1996. — 37 p.
- [Clark, 1982] High-Resolution Subjective Testing Using a Double-Blind Comparator / D. L. Clark // Journal of the Audio Engineering Society. — Vol. 30 No. 5, May 1982. — P. 330-338.
- [Osgood, 1952] The nature and measurement of meaning, / C.E. Osgood // Psychological Bulletin. — vol. 49, 1952. — P.197-237
- [Галунов, 1978] Влияние индивидуальных и эмоциональных изменений параметров артикуляторного тракта на характеристики речевого сигнала / В.И. Галунов, С.Л. Коваль, И.Б. Тампель // Речь, эмоции и личность. — Л.: Наука, 1978. — С. 83-90.
- [Петренко, 1988] Психосемантика сознания / В. Ф. Петренко. — М.: Изд-во МГУ, 1988. — 207 с.
- [Манеров, 2012] Опыт использования метода семантического дифференциала при изучении слухового восприятия речевых сообщений / В.Х. Манеров // Международная научно-практическая конференция «Преемственность психологической науки в России: традиции и инновации» — Спб., 2012. — С. 452-458.

SEMANTIC DIFFERENTIAL METHOD FOR THE VOICE CONVERSION QUALITY TESTING

Zahariev V.A. , Petrovsky A.A.

*Belarusian State University of Informatics and
Radioelectronics, Minsk, Republic of Belarus*

zahariev@bsuir.by

palex@bsuir.by

The report examines the possibility of using the semantic differential method for assessing the quality of voice conversion. Standard approaches based on subjective methods of estimating the MOS and ABX, primarily designed to assess the degree of intelligibility, and can not be directly used to determine the degree of closeness of speakers voices. Presents a methodology for assessing the quality of the conversion on the basis of building a listeners semantic space.

Keywords: semantic differential, psycholinguistics, voice conversion.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

КАМПАНЕНТ ДЛЯ АНАЛІЗУ ЧАСЦІН МОВЫ БЕЛАРУСКІХ СЛОЎ ВА ЁМОВАХ АБМЕЖАВАНЫХ СІСТЕМНЫХ РЭСУРСАЎ

Шчурко М.Л. *, Гецэвіч Ю.С. **, Пакладок Д.А. **

** Беларускі дзяржаўны ўніверсітэт інфарматыкі і радыёэлектронікі, Мінск, Рэспубліка Беларусь*

maxe1.first@gmail.com

*** Аб'яднаны інстытут праблем інфарматыкі НАН Беларусі, Мінск, Рэспубліка Беларусь*

yury.hetsevich@gmail.com

dima.pokladok@gmail.com

Апісваецца стварэнне кампанента для аўтаматычнага вызначэння часціны мовы слова па фіксаванай даўжыні заканчэння слова для 3, 4, 5 сімвалаў. Вынікі тэставанняў паказалі максімальную дакладнасць 65,3% пры фіксаванай даўжыні 5 сімвалаў. Прыведзена паслядоўнасць крокаў для ўжывання гэтага алгарытму на платформе J2ME, якая з'яўляецца тыповай платформай з абмежаванымі сістэмнымі рэсурсамі.

Ключавыя словы: кампанент; сінтэз маўлення па тэксце; мабільны сінтэзатар маўлення па тэксце.

Уводзіны

У цяперашні час немагчыма ўявіць жыццё і працу без мабільных прылад, такіх як тэлефоны, планшэты, інтэрактыўныя тэле-, аўдыёсістэмы і розныя сістэмы кіравання тэхнікай. Дзякуючы вялікай колькасці задач, праграмае забеспячэнне для іх актыўна развіваецца.

Падчас распрацоўкі праграмнага забеспячэння для гэтых прылад трэба памятаць, што яны маюць абмежаваны набор рэсурсаў памяці і камандаў, таму стандартныя і выдавочныя рашэнні не заўсёды падыходзяць да гэтых платформаў.

Адной з заўсёды запатрабаваных задач з'яўляецца сінтэз маўлення. А адной з падзадач у ім з'яўляецца вызначэнне часцін моў слоў сказа. Гэта нескладана зрабіць, калі мець поўны слоўнік слоў з пазнакамі часцін мовы, але для прылад з абмежаванай колькасцю памяці гэты спосаб нязручны, таму што гэтыя слоўнікі маюць вялікі памер [Цирульник, 2012]. Выдавочна, што калі нельга выкарыстоўваць нейкі агульны слоўнік, то трэба ўжываць алгарытм, які зможa з высокай верагоднасцю вызначыць часціну мовы ўсіх слоў у тэксце.

Прыкладам такога алгарытму з'яўляецца вынаходжанне часціны мовы з дапамогай базы пар “заканчэнне – часціна мовы”. Сутнасць алгарытма заключаецца ў вызначэнні часціны мовы слова па

яго заканчэнню. Пад заканчэннем тут трэба разумець нейкую фіксаваную колькасць літар ад канца слова (ці ўсё слова, калі яго даўжыня менш за гэту колькасць).

У гэтым артыкуле даследаваная адпаведнасць гэтага алгарытму для беларускай мовы і прапанаваная рэалізацыя для платформы J2ME [Oracle, 2013] у выглядзе модуля для сістэмы сінтэзу маўлення па тэксце для мабільных тэлефонаў [Цирульник, 2012]. Гэта платформа выкарыстоўваецца зараз не вельмі шырока, але абмежаванасць тэлефонаў у сістэмных рэсурсах, якія яе ўжываюць, гарантуе, што алгарытм зможa працаваць на сучасных платформах, такіх як Google Android, Apple iOS, BlackBerry OS, Windows Phone і інш, а таксама проста ўбудовацца як кампанент у іншыя інтэлектуальныя сістэмы.

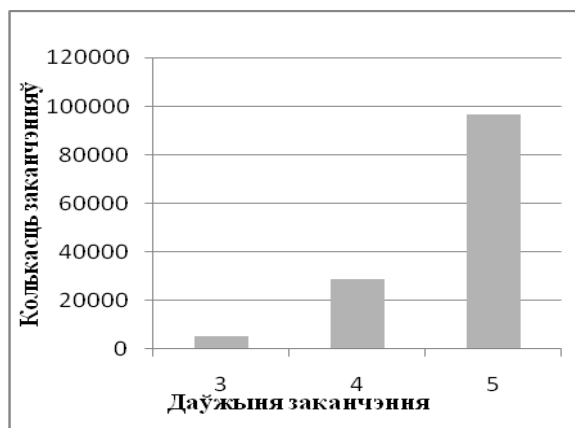
Праблема вызначэння часціны мовы слова не з'яўляецца новай: існуюць і іншыя алгарытмы. Частка з іх ужывае маркаўскія ланцугі [Paul Taylor, 2009], частка – нейронныя сеткі [Nakagawa, Kudoh, Matsumoto, 2001]. Але складанасць сучасных алгарытмаў можа стаць істотным абмежаваннем ці наогул спыніць працу сінтэзу маўлення на прыладзе, якая абмежаваная ў рэсурсах (вызначэнне часціны мовы – толькі адна з аперацый патрэбных для сінтэзу).

1. Даследаванне ідэнтыфікацыі часціны мовы слова па заканчэнні слова

Асноўным рэсурсам для алгарытма з’яўляецца слоўнік, у якім для кожнага слова была вызначана часціна мовы [Hetseвич, 2012]. Усяго ў гэтым слоўніку 211845 словаўжыванняў. На яго базе будзецца новы слоўнік S, які складаецца з пар “заканчэнне – часціна мовы”.

Для змяншэння аб’ёму слоўніка S было праведзена адмысловае даследаванне. Яго сутнасць складаецца ў вызначэнні аптымальных суадносін паміж аб’ёмам слоўніка S і дакладнасцю для выкарыстання слоўніка S у алгарытме вызначэння часцін мовы. У даследаванні разглядаюцца некалькі слоўнікаў S з заканчэннямі рознай даўжыні, а менавіта: не больш за 3, не больш за 4 і не больш за 5 літар. Яно складалася з трох этапаў:

1) Са слоўніка былі выбраныя ўсе магчымыя заканчэнні рознай даўжыні. Колькасць атрыманых заканчэнняў пэўнай даўжыні адлюстравана на малюнку 1.



Малюнак 1 – Колькасць заканчэнняў пэўнай даўжыні

2) Для кожнага заканчэння пэўнай даўжыні былі ўзятыя адпаведныя словы-прыклады, якія яго маюць, і часціны мовы, да якіх належаць гэтыя словы. У выніку былі атрыманы файлы наступнага выгляду для кожнай даўжыні заканчэнняў:

Заканчэнне1 (слова11 часціна мовы11) (слова12 часціна мовы12) ...

Заканчэнне2 (слова21 часціна мовы21)

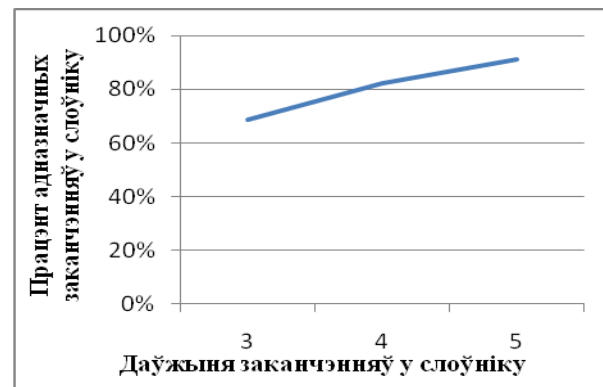
дзе першы індэкс гэта нумар радка, другі – нумар пары “слова – часціна мовы” ў гэтым радку.

3) У файлах, атрыманых на кроку 2, былі вызначаныя два віды заканчэнняў – адназначныя і шматзначныя.

Адназначнае заканчэнне – заканчэнне, якое характэрна толькі для адной часціны мовы. Напрыклад, заканчэнне „-фатар”, адназначна вызначае часціну мовы *назоўнік* (трыўмфатар). Шматзначныя заканчэнні, характэрныя для некалькіх часцін мовы. Напрыклад, заканчэнне „-чнага” можа сустракацца ў лічэбнікаў (*шасцітысячнага*) і назоўнікаў (*падапечнага*).

Праз шматзначнае заканчэнне немагчыма вызначыць часціну мовы адназначна без дадатковых рэсурсаў і алгарытмаў. Таму ўсе шматзначныя заканчэнні былі выдаленыя. Атрыманыя файлы з парамі “заканчэнне – часціна мовы” з’яўляюцца слоўнікамі, якія падыходзяць для адназначнага вызначэння часціны мовы слова ці для канстатацыі, што часціну мовы вызначыць нельга.

На малюнку 2 паказана залежнасць колькасці адназначных заканчэнняў у слоўніку ад іх даўжыні.



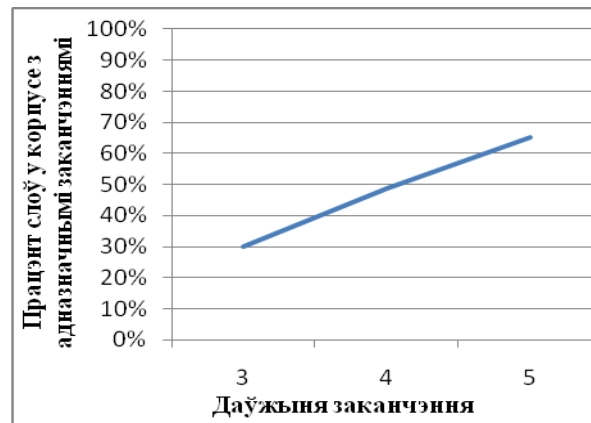
Малюнак 2 – Залежнасць колькасці адназначных заканчэнняў у слоўніку ад іх даўжыні

З малюнка 2 бачна, што пры павелічэнні даўжыні заканчэнняў слоў павялічваецца колькасць адназначных часцін мовы, якія могуць быць вызначаныя па заканчэннях. Гэта значыць, павялічваецца дакладнасць слоўніка, але разам з гэтым павялічваецца і яго аб’ём.

Такім чынам, на базе даследавання былі атрыманыя слоўнікі для трох-, чатырох- і пяцілітарных заканчэнняў, па якіх можна адназначна вызначыць часціну мовы.

2. Тэставанне дакладнасці слоўнікаў “заканчэнне слова – часціна мовы” на корпусе тэкстаў

Для тэставання быў узяты размечаны часцінамі мовы корпус памерам 284774 слоў з тэкстаў розных стыляў: навуковага, мастацкага, афіцыйна-дзелавога.



Малюнак 3 – Працэнт слоў у корпусе з адназначнымі заканчэннямі

На малюнку 3 адлюстраваны працэнт адназначных заканчэнняў у корпусе тэкстаў у залежнасці ад даўжыні гэтых заканчэнняў. Па іх можна меркаваць пра колькасныя паказчыкі магчымасцяў вызначаць часціны мовы з дапамогай распрацаваных слоўнікаў.

У табліцы 1 сабраныя ўсе лічбы і вынікі тэставанняў для аднолькавых тэкстаў з прымяненнем розным па нападзенні слоўнікаў “заканчэнне – часціна мовы”.

Табліца 1 – Вынікі тэставанняў на размечаным корпусе тэкстаў

Параметры тэставанняў	Тэст 1	Тэст 2	Тэст 3
Даўжыня заканчэнняў	3	4	5
Усяго слоў у корпусе тэкстаў	284774	284774	284774
Слоў з адназначнымі заканчэннямі	85076	138364	186035
Адносная колькасць слоў з адназначнымі заканчэннямі	29,9%	48,6%	65,3%

Вынікі тэставанняў паказалі максімальную дакладнасць 65,3%. Яе “неадпаведнасць 100% дакладнасці” абумоўлена папярэднім рашэннем адносна пастаўленай задачы: былі адкінутыя шматзначныя заканчэнні, што ўплывае на адкіданне шматзначных слоў. Вынікі тэставанняў паказалі паступовае памяншэнне дакладнасці да 29,9%, бо дакладнасць змяншаецца пры выкарыстанні слоўнікаў зменшаных даўжынь заканчэнняў у слоўніках ад 5 да 3.

3. Выкарыстанне слоўнікаў пад J2ME

Нагледзячы на прастату тэарэтычнага і эксперыментальнага выкарыстання (для стацыянарнага запуску праграмы на J2ME) пабудаваных слоўнікаў для вырашэння пастаўленай задачы вызначэння часцін мовы слоў застаецца праблема выкарыстання слоўнікаў у рэальных мабільных прыстасаваннях. Пажадана, каб аб’ём слоўніка быў меншы чым 900 кБ. Таму патрэбны нейкі дадатковы алгарытм, каб паменшыць колькасць месца, якое займае масіў дадзеных слоўніка (аднаго з трох распрацаваных). Ідэя наступнага алгарытму ў тым, каб прадставіць кожны сімвал заканчэння і адпаведнага заканчэнню кода не двума байтамі, як гэта робіцца ў Java, а нейкай мінімальнай колькасцю біт фіксаванай даўжыні.

Крокі алгарытму:

1) Выбраць слоўнік з фіксаванай даўжынёй заканчэння F (паводле другога раздзелу артыкула).

2) Стварыць алфавіт з розных сімвалаў, якія ўжываюцца ў заканчэннях і пранумараваць сімвалы, пачынаючы з нуля.

3) Пранумараваць розныя часціны мовы слоўніка, пачынаючы з нуля.

4) Вызначыць мінімальную колькасць біт, якая патрэбная, каб пазначыць нумар любога сімвалу альфабэту $N_{\text{альф}}$. Напрыклад, для таго каб абазначыць альфабэт з 29 сімвалаў, дастаткова 5 біт ($2^4 < 29 < 2^5$; 28 таму што адлік пачынаецца з нуля).

5) Аналагічна кроку 4 вызначыць такую колькасць біт $N_{\text{часц}}$, якая дазволіць вызначыць нумар любой часціны мовы.

6) Каб усе заканчэнні былі аднолькавай даўжыні, дапоўніць іх прабеламі, дзе патрэбна.

7) Закадаваць усе пары “заканчэнне – часціна мовы” паслядоўна ў масіў байтаў. Заканчэнне павінна адпавядаць фармату:

нумар_літары1нумар_літары2...нумар_літарыN

Выніковая паслядоўнасць павінна адпавядаць фармату:

заканчэнне1нумар_часціны_мовы1заканчэнне2нумар_часціны_мовы2...заканчэннеNнумар_часціны_мовыN.

Даўжыня аднаго заканчэння будзе $N_{\text{альф}} \times \text{даўжыня_канчатку} + N_{\text{часц}}$ біт замест стандартных $8 \times 2 \times \text{даўжыня_канчатку} + 8 \times 1$ біт, калі захоўваць заканчэнні ў радковым выглядзе. Дзякуючы таму, што $N_{\text{альф}}$ будзе меншым за 8×2 (для беларускай мовы $N_{\text{альф}} = 6$) і $N_{\text{часц}}$ таксама будзе менш за 8 (для 10 часцін мовы $N_{\text{часц}} = 4$), а даўжыня канчатку мінімум 3, то на выхадзе атрымаецца слоўнік SS меншых памераў, чым слоўнік да закодавання ўсіх пар “заканчэнне – часціна мовы”.

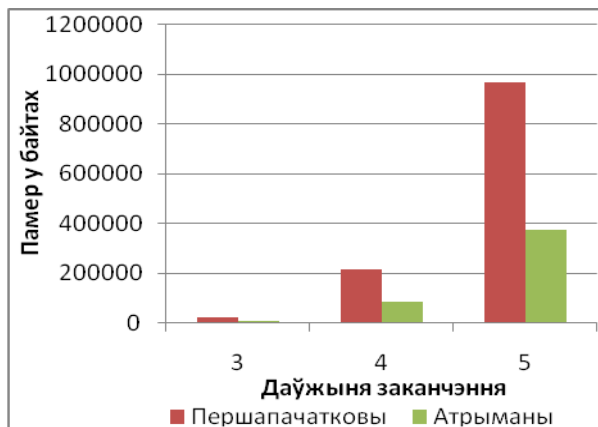
8) Модуль у тэлефоне прымае як дапаможны рэсурс слоўнік SS. З кожнага ўваходнага слова, для якога патрэбна ідэнтыфікаваць часціну мовы, выдзяляецца заканчэнне адпаведнай даўжыні F. (Калі не хапае літар для выдзялення заканчэння, то астатнія пазіцыі запаўняюцца прабеламі.) Далей гэтае заканчэнне правяраецца на наяўнасць па слоўніку SS, калі такое ёсць, то выдаецца за адказ адпаведная часціна мовы, калі такога няма, то выдаецца адказ UNKNOWN.

Параўнанне памераў першапачатковых (паводле раздзелу 2) і атрыманых слоўнікаў SS (толькі файлаў пар “заканчэнне – часціна мовы” без памеру альфабэту і іншай невялікай службовай інфармацыі) прыводзіцца на малюнку 4. Лічбы для гэтага малюнку прыведзены ў табліцы 2. На малюнку бачна, што алгарытм, які выкарыстоўваецца, значна змяншае памер слоўнікаў для любой фіксаванай даўжыні заканчэння.

Табліца 2 – Зыходныя і выніковыя памеры слоўнікаў

Даўжыня заканчэння	Зыходны памер слоўніка, байт	Выніковы памер слоўніка, байт	Размер зменшаны на, %
3	24710	9709	60,71
4	213156	82894	61,11
5	966614	373465	61,36

Найбольш прымальнай даўжынёй канчатку для мабільнай платформы з’яўляецца даўжыня не большая за 4 літары з-за аптымальных суадносін “аб’ём слоўніка – дакладнасць вырашэння пастаўленай задачы”.



Малюнак 4 – Параўнанне памеру атрыманага і першапачатковага масіву дадзеных

Заклучэнне

Такім чынам, была пастаўлена і вырашана з пэўнай дакладнасцю задача прадказання часціны мовы слоў на прыладзе з абмежаванымі рэсурсамі. Для гэтага быў прапанаваны алгарытм ідэнтыфікацыі часціны мовы па заканчэннях слоў вядомага электроннага граматычнага слоўніка. У залежнасці ад выкарыстання слоўніка для фіксавання даўжыні заканчэння слова ад 3 да 5 літар дакладнасць працы алгарытма мяняецца ад амаль 30% да 65%. Для выкарыстання ў рэальных мабільных прыстасаваннях быў прапанаваны ясны алгарытм скарачэння выкарыстання памяці для захавання слоўнікаў у залежнасці ад фіксавання даўжыні заканчэння слова для не больш за 3, 4, 5 літар. Алгарытм кампрэсіі даў магчымасць скараціць аб’ём слоўніка на 60%, 61%, 61% адносна першапачаткова атрыманых.

Для павышэння дакладнасці працы алгарытма па вызначэнні часціны мовы слоў трэба ўжываць у слоўніку таксама шматзначныя заканчэнні. Аўтарамі прапануецца ў будучым выбіраць сярод часцін мовы, характэрных для шматзначных заканчэнняў, часціну мовы, якая найбольш часта ўжываецца з гэтым заканчэннем у тэкстах.

Спіс літаратуры

- [Цирульник, 2012] Цирульник, Л.И. Алгоритмы создания и пополнения грамматического словаря русского языка для синтеза речи по тексту / Л.И. Цирульник, В.В. Веремей // Информатика. – 2012. – № 1 (31). – С. 27–38.
- [Oracle, 2013] Java ME Landing Page // Oracle [Electronic recourse]. – 2013. – Mode of access: <http://www.oracle.com/technetwork/java/javame/index.html>. – Date of access: 23.07.13.
- [Цирульник, 2012] Цирульник, Л.И. Система синтеза речи по тексту для мобильных телефонов / Л.И. Цирульник, Д.А. Покладок // Речевые технологии. – 2010. – № 1. – С. 81–90.
- [Paul Taylor, 2009] Paul Taylor Text-to-Speech Synthesis // Paul Taylor // Cambridge University Press – 2009. – P. 89 – 93.

[Nakagawa, Kudoh, Matsumoto, 2001] Tetsuji Nakagawa, Taku Kudoh and Yuji Matsumoto Unknown Word Guessing and Part-of-Speech Tagging Using Support Vector Machines // Tetsuji Nakagawa, Taku Kudoh and Yuji Matsumoto // Graduate School of Information Science, Nara Institute of Science and Technology. – 2001

[Hetseвич, 2012] Hetseвич, Y. Overview of Belarusian And Russian dictionaries and their adaptation for NooJ / Y. Hetseвич, S. Hetseвич // Automatic Processing of Various Levels of Linguistic Phenomena: Selected Papers from the NooJ 2011 Intern. Conf. / eds. Vučković Kristina, Bekavac Božo, Silberstein Max. – Newcastle : Cambridge Scholars Publishing, 2012. – P. 29 – 40.

COMPONENT FOR PART-OF-SPEECH TAGGING OF BELARUSIAN WORDS WITH LOW SYSTEM REQUIREMENTS

Shchurko M.L. *, Hetseвич Y.S. **, Pakladok D.A. **

*Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus
maxel.first@gmail.com

**United Institute of Informatics Problems,
National Academy of Sciences,
Minsk, Republic of Belarus
yury.hetseвич@gmail.com
dima.pokladok@gmail.com

Introduction

This article describes the problem of part-of-speech tagging on devices with architecture limitations of memory and processor power like mobile phones, tablet PC, etc.

The problem is that system requirements for modern algorithms don't fit these devices.

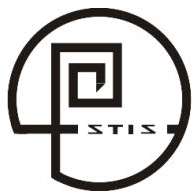
Main Part

In this article was researched algorithm of guessing part-of-speech by the ending of a word for endings that have unambiguous corresponded part-of-speech. Also was proposed algorithm of compression pairs “ending – part-of-speech”.

Mentioned algorithms were tested with different ending size (from 3 to 5). On tested text it has 30%, 45%, 65% accuracy for 3, 4, 5 ending size accordingly. It decreases size of pairs base by 60%, 61%, 61% for 3, 4, 5 ending size accordingly.

Conclusion

Researched algorithm is not accurate enough. Should be proposed another algorithm that statistically chooses part-of-speech for endings which have polysemantic corresponded parts-of-speech. Compression algorithm allows to store greater “ending – part-of-speech” base on a device.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 007:519.816

АНАЛИЗ И ВЕРИФИКАЦИЯ РАСКРАШЕННЫХ СЕТЕЙ ПЕТРИ РЕАЛЬНОГО ВРЕМЕНИ С ПОДДЕРЖКОЙ ЛОГИКИ АЛЛЕНА

Еремеев А.П., Королев Ю.И.

*Национальный исследовательский университет «МЭИ»,
г. Москва, Россия*

Eremeev@appmat.ru

KorolevYu@gmail.com

Рассматриваются вопросы анализа темпорального подкласса сетей Петри, предложенного для моделирования интеллектуальных систем. Обуславливается необходимость верификации моделей, созданных на основе таких сетей. Предлагается использование графовых инструментов анализа, отражающих смену состояний в подобных сетях. Приводится пример построения таких графов для модели простой системы управления. Работа выполнена при финансовой поддержке РФФИ (проект № 14-01-00427) и Фонда содействия инновациям.
Ключевые слова: модифицированные сети Петри; темпоральные модели; системы управления.

Введение

Проблема обеспечения правильности программных и аппаратных компонентов систем управления приобретает сегодня первостепенное значение. Надежность и предсказуемость поведения таких систем зачастую являются более важными свойствами, чем производительность, модифицируемость и т.п. На кафедре прикладной математики Национального исследовательского университета «Московский энергетический институт» (НИУ «МЭИ») уже более тридцати лет проводятся исследования по разработке математического и программного обеспечения интеллектуальных систем поддержки принятия решений (ИСППР) [Вагин, 1988]. При разработке таких систем, типичным представителем которых являются ИСППР реального времени (ИСППР РВ) для мониторинга и управления сложными техническими системами, используется предложенная Д.А. Поспеловым концепция семиотических систем управления [Поспелов, 1981] и аппарат нетрадиционных логик [Вагин и др., 2008]. В качестве одного из инструментов разработки предлагается использовать аппарат раскрашенных (цветных) сетей Петри реального времени с поддержкой логики Аллена (РСР РВ ЛА). Данный формализм позволяет корректно моделировать как количественные, так и качественные временные зависимости [Еремеев и др., 2013], упрощая процесс разработки моделей реальных систем. Однако использование сетей

Петри подразумевает достаточно высокий уровень параллелизма, что, как и учет темпоральных зависимостей, может привести к возникновению ошибок и неточностей при разработке. Поэтому при применении аппарата РСР РВ с поддержкой логики Аллена актуален вопрос анализа и верификации моделей, созданных на его основе.

1. Повышение качества моделей

Параллельные, распределенные и многопоточные программы, характерные для многих систем управления, в том числе, реального времени типа ИСППР РВ, крайне подвержены ошибкам. Хорошо известно, что даже в тех случаях, когда функционирование каждой из параллельных взаимодействующих компонент системы абсолютно ясно, человеку трудно понять работу всей параллельной системы в целом, процессы в которой взаимозависимы. Такие системы, которые работают правильно «почти всегда», годами могут сохранять «тонкие» ошибки, проявляющиеся в исключительных ситуациях. Их непосредственными причинами являются и некорректные спецификации, и неправильное понимание спецификации разработчиками, несогласованность параллельных ветвей процессов и многое другое. Наиболее очевидным и широко распространенным методом проверки правильности программных систем является тестирование - проверка работы построенной системы в различных ситуациях, при различных исходных данных. Однако в случае с параллельными системами обычно невозможно

заранее определить все возможные траектории функционирования. Поэтому в качестве основного метода повышения качества разработки применяется верификация – формальная проверка того, что система (модель) удовлетворяет сформулированным заранее требованиям [Карпов, 2010]. Методы верификации различаются в зависимости от того, какой аппарат лежит в основе проверяемой системы. Рассматриваемый подкласс РСП РВ ЛА представляет собой визуальный язык программирования [Еремеев и др., 2013] с формально определенным синтаксисом. Модели, разработанные с помощью этого аппарата, кажутся полностью формализованными. Однако с точки зрения семантики это не так. Из самой модели не следует непосредственно полное формальное описание ее поведения. Параллелизм, присущий сетям Петри в целом, и учет темпоральных зависимостей, введенный для упрощения процесса разработки, позволяют элегантно решить многие проблемы создания моделей систем (протекающих в них процессов), но зачастую делают целостное восприятие сложным. Поэтому для анализа поведения и верификации моделей, построенных с помощью РСП РВ ЛА, необходимо использовать дополнительные инструменты. Известны три основных группы методов анализа сетей Петри [Мурата, 1989]: основанные на построении графов изменения состояний; матричные методы, использующие уравнения сети и инварианты; методы редукции. При работе с раскрашенными и темпоральными сетями Петри последние две группы методов используются редко из-за высокой (по сравнению с классическими сетями Петри [Котов, 1984]) сложности формальных определений подобных подклассов. Чаще в качестве основного инструмента анализа рассматриваются графы достижимости и покрытия [Szpyrka, 2008].

2. Инструменты анализа

2.1. Смена состояний сети

Для работы с инструментальными средствами анализа РСП РВ ЛА, определим формально ключевые понятия состояния и перехода между состояниями. В качестве примера рассмотрим модель системы управления экстренным торможением поезда [Еремеев и др., 2013] (рисунок 1). Состояние сети представляет собой пару (M, S) , где M - маркировка, функция на множестве мест P , а S - временной вектор, ставящий в соответствие каждому месту сети число - временную метку. Для сети, изображенной на рисунке 1, множество мест зададим следующим образом: $P = \{Timer1, LightSig, SoundSig, Brake, Timer2, Driver\}$. Тогда начальное состояние сети (M_0, S_0) будет выглядеть так:

$$\begin{aligned} M_0 &= (on, off, off, off, on, active), \\ S_0 &= (0, 0, 0, 0, 0, 0). \end{aligned} \quad (1)$$

Переход от одного состояния сети к другому может быть обусловлен двумя причинами:

- срабатыванием перехода $t \in T$ в подстановке b (подстановка - функция, которая замещает каждую переменную в защитной функции $G(t)$ и функциях весовых и временных значений дуг E_M, E_S , влияющих на переход t , значением соответствующего типа);

- течением времени (постепенное уменьшение каждой временной метки на фиксированную величину, пока не появится переход, который может сработать).

Следует отметить, что безусловный приоритет при смене состояний сети имеет событие срабатывание перехода. Течение времени позволяет только дожидаться момента, когда может сработать очередной переход. Для рассматриваемого примера в начальном состоянии могут сработать два перехода: *TurnOnLS* и *Activity*. Рассмотрим изменение состояния сети при срабатывании первого. На переходе *TurnOnLS* защитная функция всегда принимает значение *true*, поэтому переход срабатывает в т.н. тривиальной подстановке $b = ()$. Результатом срабатывания перехода *TurnOnLS* в начальном состоянии будет являться состояние (M_1, S_1) :

$$\begin{aligned} M_1 &= (on, on, off, off, on, active), \\ S_1 &= (60, 0, 0, 0, 0, 0). \end{aligned} \quad (2)$$

При срабатывании перехода фишки-токены извлекаются и помещаются в места, связанные с переходом, в соответствии со значениями весовых выражений дуг, временные метки входных мест обнуляются, а временные метки выходных мест определяются в соответствии со значением временных выражений дуг, идущих из перехода к этим местам. В соответствии с условиями, накладываемыми защитной функцией, переход *Activity* может сработать в трех различных подстановках: $b_1 = (x/n)$, $b_2 = (y/n)$ и $b_3 = (z/n)$, где $x \in (0; 6)$, $y \in (6; 9)$, $z > 9$. Для удобства анализа примем $x = 5$, $y = 8$, $z = 10$. Результатом срабатывания перехода *Activity* в подстановке b_2 будет состояние (M_2, S_2) :

$$\begin{aligned} M_2 &= (on, on, off, off, on, active), \\ S_2 &= (60, 0, 0, 0, 60, 8). \end{aligned} \quad (3)$$

Ни один переход не может сработать в этом состоянии. Необходимо подождать $\tau = 6$ секунд, чтобы временная метка в месте *Console* позволила сработать переходу *TurnOnSS* (машинист не реагирует на световой сигнал):

$$S'_2 = (54, -6, -6, -6, 54, 2). \quad (4)$$

После срабатывания перехода *TurnOnSS* в тривиальной подстановке сеть перейдет в новое состояние (M_3, S_3) :

$$\begin{aligned} M_3 &= (on, on, on, off, on, active), \\ S_3 &= (54, 0, 0, -6, 54, 2). \end{aligned} \quad (5)$$

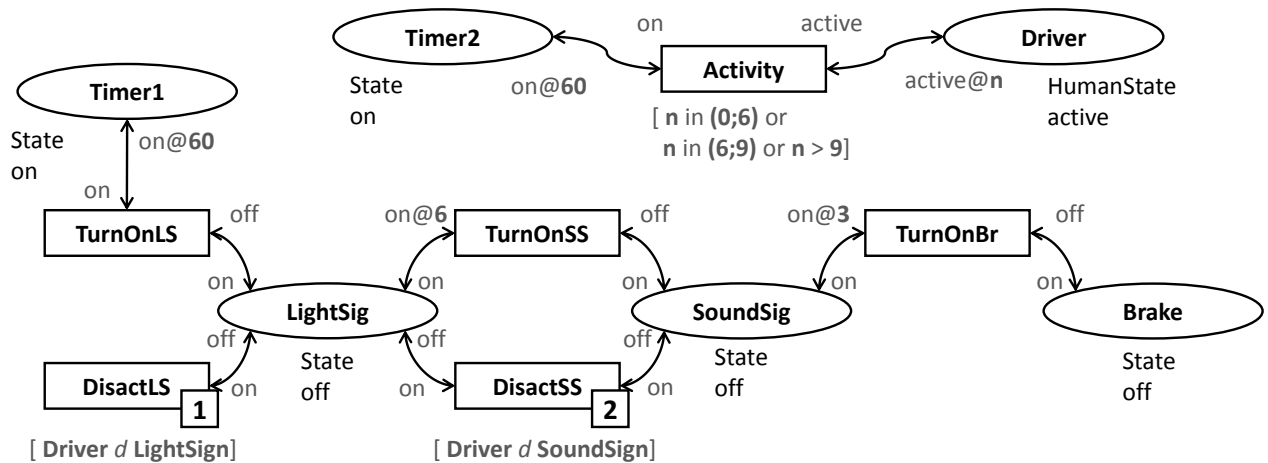


Рисунок 1 - Пример РСП РВ с поддержкой логики Аллена

Рассмотренную последовательность смены состояний сети (1-5) графически можно представить в следующем виде:

$$\begin{aligned}
 (M_0, S_0) &\xrightarrow{(TurnOnLS())} (M_1, S_1) \rightarrow \\
 &\xrightarrow{(Activity, (8/n))} (M_2, S_2) \xrightarrow{\tau=6} (M_2, S_2') \rightarrow (6) \\
 &\xrightarrow{(TurnOnSS())} (M_3, S_3) \rightarrow \dots
 \end{aligned}$$

2.2. Граф достижимости

Будем считать, что состояние (M', S') достижимо из состояния (M, S) , если существует конечная последовательность переходов, начинающаяся с состояния (M, S) и оканчивающаяся состоянием (M', S') . Обозначим за $R(M, S)$ множество всех состояний, достижимых из состояния (M, S) . Анализ РСП РВ ЛА можно проводить, используя граф достижимости (ГД), вершинами которого являются элементы множества $R(M_0, S_0)$, а каждая дуга отображает изменение состояния (M_i, S_i) на (M_j, S_j) по прошествии времени $\tau \geq 0$ и срабатывания перехода t в подстановке b .

Граф, построенный по этим правилам для цепочки (6), представлен на рисунке 2.

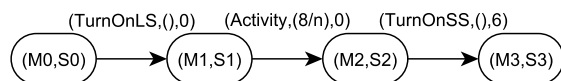


Рисунок 2 - Фрагмент графа достижимости

Анализ свойств сети может осуществляться с помощью маркировки узлов ГД и меток дуг. Каждая метка дуги представляет собой тройку, состоящую из перехода, его подстановки и значения временного промежутка перед его срабатыванием. Последний параметр, таким образом, позволяет определить время, затраченное на переход от одного состояния к другому. Используя стандартные алгоритмы поиска кратчайшего или длиннейшего пути между двумя узлами мультиграфа, можно найти минимальное и максимальное время перехода из одного состояния в другое.

Если продолжить строить ГД для анализируемой сети, можно убедиться, что из-за непрерывного уменьшения значений временных меток он оказывается бесконечным. Подобная ситуация возникает при анализе практически любой РСП РВ, в том числе, РСП РВ ЛА, причем степень связанности сети не влияет на конечность ГД. Очевидно, что в этом случае данный формализм неудобно использовать для анализа сетей. Введение дополнительных условий на временные метки позволяет трансформировать бесконечный граф в конечную структуру.

2.3. Граф покрытия

Одним из главных преимуществ РСП РВ ЛА является возможность представления множества достижимых состояний с помощью конечного графа покрытия (ГП). Отношение покрытия позволяет определять эквивалентные по своим характеристикам состояния сети Петри. Считаем, что два состояния покрывают друг друга, если их маркировки совпадают, а временные метки либо совпадают, либо не превышают максимального возраста доступа места $p \in P$, то есть такого значения временной метки, когда фишки-токены становятся доступными для всех выходных переходов места p .

Графы достижимости и покрытия строятся одинаково. Различие заключается только в способе добавления новой вершины в графы. Для ГП после определения нового состояния сети необходимо проверить, есть ли в графе вершина, которая отображает состояние, покрывающее новое. Если есть, то необходимо добавить только новую дугу, которая идет к найденной вершине. В противном случае вершина нового состояния добавляется в ГП вместе с соответствующей дугой. ГП содержит только одну вершину для каждого класса эквивалентности отношения покрытия состояний. ГП, построенный для анализируемой сети, приведен на рисунке 3.

ГП для РСП РВ ЛА предоставляет такие же возможности анализа сетевых свойств, как и

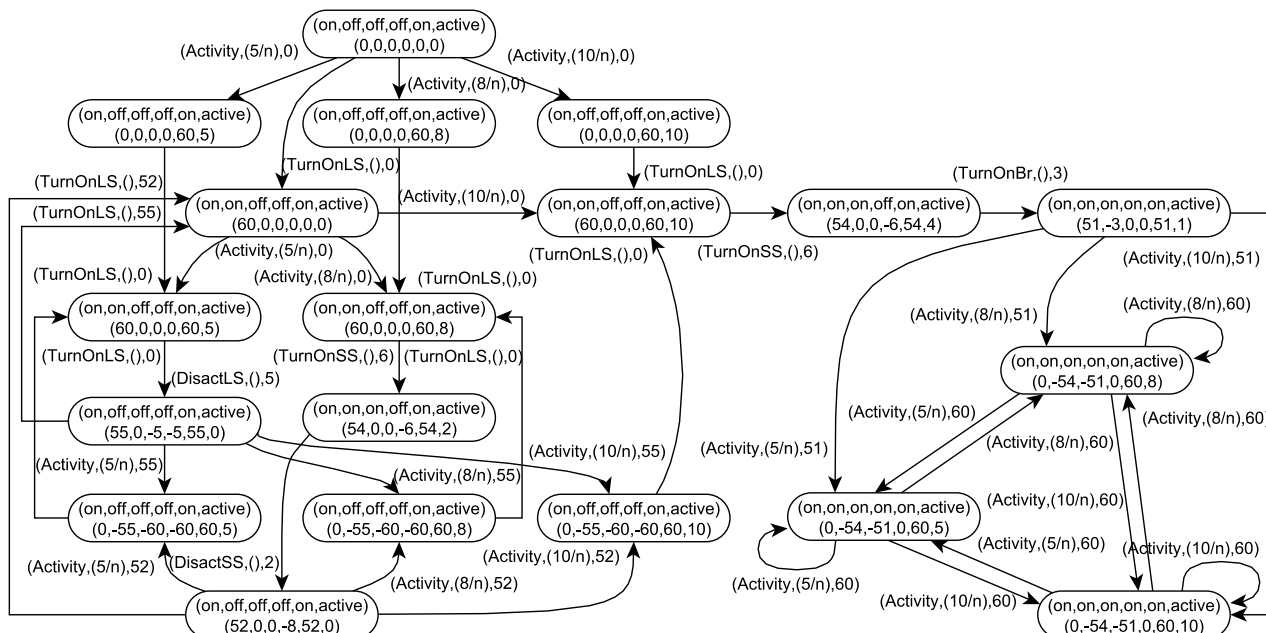


Рисунок 3 - Пример полного графа покрытия для РСРП РВ с поддержкой логики Аллена

полный ГД. Он содержит все достижимые маркировки. Чтобы найти минимальное и максимальное время перехода из одного состояния в другое, можно использовать те же алгоритмы, что и для графов достижимости.

ГП сети позволяет увидеть все состояния с точностью до временных меток. Анализируя его, разработчик оценивает корректность выполнения поставленной задачи, т.е. верифицирует созданную модель. Отметим, что ГП и для сравнительно малых РСРП РВ ЛА может достигать достаточно больших размеров. Поэтому прямые исследования сетей путем их компьютерного моделирования могут упростить задачу разработчика. Для проведения подобных исследований и разрабатывается базовый инструментальный [Еремеев и др., 2013].

Заключение

В работе рассматривается проблема анализа и верификации моделей, построенных на основе РСРП РВ с поддержкой логики Аллена. В качестве основного инструмента анализа предложен модифицированный для РСРП РВ граф покрытия.

Библиографический список

- [Вагин, 1988] Вагин В.Н. Дедукция и обобщение в системах принятия решений. – М.: Наука, 1988. – 384 с.
- [Вагин и др., 2008] Вагин В.Н., Головина Е.Ю., Загорянская Н.А., Фомина М.Б. Достоверный и правдоподобный вывод в интеллектуальных системах. – М.: Физматлит, 2008. – 712 с.
- [Еремеев и др., 2013] Еремеев А.П., Королев Ю.И. Реализация интеллектуальных систем реального времени на основе сетей Петри с поддержкой темпоральных зависимостей // Программные продукты и системы. – 2013. – №3. – С. 88-94
- [Карпов, 2010] Карпов Ю.Г. Model Cheking. Верификация параллельных и распределенных программных систем. – СПб.: БХВ-Петербург, 2010. – 560 с.
- [Котов, 1984] Котов В.Е. Сети Петри. – М.: Наука, 1984. – 160 с.
- [Мурата, 1989] Мурата Т. Сети Петри: Свойства, анализ, приложения // ТИИЭР. – 1989. – т. 77, №4. – С. 41-85.

[Поспелов, 1981] Поспелов Д.А. Логико-лингвистические модели в системах управления. – М.: Энергоиздат, 1981. – 232 с.

[Szpyrka, 2008] Szpyrka M. Modelling and Analysis of Real-Time Systems with RTCP-Nets // Petri Net, Theory and Applications. I-Tech Education and Publishing. – 2008. – P. 17-40.

ANALYSIS AND VERIFICATION OF REAL-TIME COLORED PETRI NETS WITH SUPPORT OF ALLEN'S LOGIC

Eremeev A.P., Korolev Y.I.

National Research University «Moscow Power Engineering Institute», Moscow, Russia

Eremeev@appmat.ru

KorolevYu@gmail.com

In work analysis of temporal Petri nets' subclass, proposed by research group for intelligent systems' modeling, is considered.

Introduction

Formalism of real-time colored Petri nets with support of Allen's logic allows efficient simulation of the system, but the methods of analysis are needed to be defined to reduce the number of errors.

Main Part

Reachability graph is proposed to verify models. A set of places for such graphs is often infinite, as considered formalism is temporal. The concept of coverability is introduced. Coverability graph is proposed for models' analysis and verification.

Conclusion

Coverability graph modified for this type of network is proposed for models' analysis.



УДК 004.832.3

МЕТОДЫ И АЛГОРИТМЫ НАХОЖДЕНИЯ СТЕПЕНЕЙ ОБОСНОВАНИЯ В СИСТЕМАХ АРГУМЕНТАЦИИ

Моросин О.Л.*

** Национальный Исследовательский Университет «МЭИ»,
г. Москва, Россия*

omorsik@gmail.com

В данной работе приводится краткий обзор подходов к формализации аргументации и рассматриваются возможности использования степеней обоснования в пересматриваемых рассуждениях. Степени обоснования позволяют эффективнее моделировать различные задачи аргументации, и позволяют не только отвечать на вопросы о правдоподобности того или иного утверждения, но и давать ей числовую оценку. В качестве формальной системы аргументации используется теория пересматриваемых рассуждений.

Ключевые слова: Пересматриваемые рассуждения; аргументация; степени обоснования; немонотонный вывод.

Введение

Очень часто база знаний, используемая в системах поддержки принятия решений содержит противоречивую информацию. Классические методы логического вывода не применимы для таких баз знаний. Одним из способов обнаружения и разрешения внутренних противоречий является применение аргументации. Под аргументацией обычно понимают процесс построения предположений, относительно некоторой анализируемой проблемы. Как правило этот процесс включает в себя обнаружение конфликтов и поиск путей их решения. В отличие от классической логики, аргументация предполагает, что могут быть доводы как «за», так и «против» некоего предположения. Введение степеней обоснования позволяет оперировать не только терминами «за» и «против», но и давать числовую оценку аргументам и контраргументам. Для подтверждения некоторого предположения, необходимо доказать, что существует больше доводов «за» и степень их обоснования выше, чем у доводов «против». Для целостности описания разрабатываемой модели, в разделе 1 будут коротко приведены основные определения и идеи, используемые в реализуемой системе аргументации, а обзор теоретических основ применения степеней обоснования будет приведен в разделе 2.

1. Теоретические основы аргументации

1.1. Теоретические основы аргументации

Необходимость теории аргументации возникает

из-за неполноты и недостоверности данных и знаний. Обычно, говоря об аргументации, выделяют три типа информации [Besnard, 2008].

1) Объективная информация – информация, полученная из надежных источников, или которая может быть напрямую измерена или подтверждена. Например, утверждение «В центральной части России весной продолжительность светового дня увеличивается» является объективной информацией, подтверждаемое наукой и нашими наблюдениями. Такая информация, как правило, используется как неоспоримые аргументы.

2) Субъективная информация – информация полученная из менее надежных источников. Это могут быть некоторые предположения, суждения. Часто формулируются с помощью фраз «как правило», «обычно», «еюе всего». Такая информация и служит «источником» противоречий и конфликтов.

3) Гипотетическая информация. Она необходима для построения гипотез. Очень часто она является ложной информацией, и более того, даже может быть заранее неверной. Однако, построенные аргументы для ее опровержения могут быть полезны для других рассуждений. Например, маловероятно, что уровень мирового океана поднимется на метр в течении следующих 10 лет. Однако, такое предположение может быть полезным при планировании застройки прибрежных территорий, с учетом возможности их затопления. Часто при недостатке информации строятся те или иные гипотезы, и производится попытка доказать или опровергнуть их.

Все приведенные типы информации, могут быть

использованы для аргументации. Объективная информация служит фактами и исходными посылками, субъективная информация является источником пересматриваемых выводов, а гипотетическая информация – помогает строить предположения.

Существуют несколько формализаций теории аргументации. Например, системы абстрактной аргументации, предложенные Дангом (Dung P.M.) [Bondarenko et al., 1997], аргументационная система Лина и Шоэма (Lin F., Shoham Y.) [Lin et al., 1989], система Вресвийка (G.A.W. Vreeswijk [Vreeswijk, 1989], система аргументации Поллока (John L. Pollock) [Pollock, 1992] и некоторые другие.

Все эти подходы можно условно разделить на три типа [Besnard, 2008].

1) Абстрактные системы, предложенные Дангом и позднее развиваемые Праккеном и Сартором. В этих системах аргументы представляются как элементы множества, в котором задано бинарное отношение «атака». В этих системах авторы полностью абстрагируются от внутренней структуры аргументов и природы множества аргументов. В таких системах отсутствует механизм получения новых аргументов, и задача сводится к поиску неконфликтующих аргументов на заданном множестве аргументов.

2) Согласованные системы (coherence systems). В таких системах основной стратегией обработки противоречий в базе знаний является выделение согласованных подмножеств из всего объема информации, имеющейся в базе знаний. Обычно такие системы базируются на классической логике, хотя возможны применения и модальных, временных или дескриптивных логик.

3) Системы пересматриваемых рассуждений. В таких системах обычно происходит введение пересматриваемого следствия в качестве элемента языка. То есть кроме импликации в обычном смысле вводится ее пересматриваемый аналог. Аргументы в таких системах представляются как последовательность рассуждений, ведущих к заключению, каждый шаг которых может подвергнуться поражению.

В данной работе будет подробно рассмотрена система аргументации, основанная на пересматриваемых рассуждениях, предложенных Джоном Поллоком [Pollock, 1992].

1.2. Пересматриваемые рассуждения

Прежде чем переходить к изложению основного материала, дадим необходимые определения и введем обозначения [Вагин и др., 2008].

Определение 1. Аргумент - пара, состоящая из множества посылок и заключения. Записывать такие пары будем в следующем виде p/X , где p заключение, а X множество посылок.

Например, аргумент $(p \rightarrow q)/\{\sim A, B\}$ означает, что из посылок $\sim A, B$ следует $p \rightarrow q$. На всех

иллюстрациях будем обозначать аргументы овалами. Для аргументов с пустым множеством посылок (такие аргументы называют фактами), будем писать только заключение. Например, фактом является утверждение, что Земля вращается вокруг Солнца.

Определение 2. Интерес – аргумент, который мы хотим обосновать в ходе монотонного и/или пересматриваемого вывода. На графе вывода будем обозначать интересы прямоугольниками.

Определение 3. Граф вывода – граф, показывающий взаимосвязи между аргументами и интересами. Он отображает, из каких аргументов порождается новый аргумент. Аналогично в нем показывается, из каких интересов получаются новые интересы. А также граф вывода отображает конфликты между аргументами при пересматриваемых рассуждениях.

Определение 4. Дедуктивное следствие – простые, дедуктивные правила вывода, означающие, что если истинно P , то истинно и Q . Такие правила не являются пересматриваемыми. Записывать такие правила будем так: $P \Rightarrow Q$. На графе вывода будем отображать их обыкновенными стрелками (см. аргументы P и Q на рис 1).

Определение 5. Пересматриваемое следствие. Это пересматриваемые правила вывода, которые могут быть получены, например, в результате индукции или абдукции. В данной работе нас не интересует конкретный механизм получения таких выводов, поэтому такие правила подаются декларативным образом на вход программы. Аргументы, полученные в результате таких выводов, будем называть пересматриваемыми. Записывать такие правила будем так: $M \Rightarrow N$. На графе вывода такие связи будем отображать пунктирными стрелками, а пересматриваемые аргументы – двойным овалом (см. аргументы M и N на рис. 1).

Понятие конфликта – основа системы аргументации. Будем рассматривать два типа конфликтов – опровержение и подрыв [Вагин и др., 2008].

Определение 6. Опровержение (rebutting) – ситуация, когда некоторые аргументы опровергают заключения других аргументов. Иными словами, аргумент $A_1 = p_1/X_1$ опровергает аргумент $A_2 = p_2/X_2$, когда заключение p_1 опровергает заключение p_2 . Опровержение является симметричной формой атаки.

Определение 7. Подрыв (undercutting) – несимметричная форма атаки, когда один аргумент отрицает связь между посылками и заключением другого аргумента.

Определение 8. Подрывающие доводы. Это аргументы, поражающие связь между двумя другими аргументами, соединенными пересматриваемым следствием. Например, имеется аргумент E , подрывающий пересматриваемую связь $C \Rightarrow D$ между аргументами C и D . Такие правила подрыва будем записывать в виде $E \Rightarrow (C @ D)$. На графе вывода подрывающие аргументы и пораженные ими аргументы будем соединять

жирной пунктирной стрелкой. Пораженные аргументы будем помечать темно-серым цветом (см. аргумент D и на рис. 1).

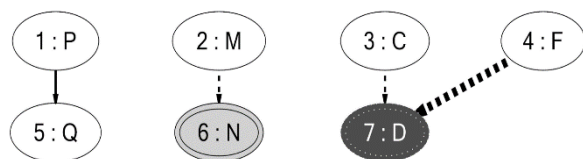


Рисунок 1 - Граф вывода

1.3. Обнаружение конфликтов

Обнаружение конфликтов является наиболее сложной частью системы аргументации. Основные сложности появляются из-за необходимости поддержки логики предикатов первого порядка. Если говорить про логику высказываний, то конфликт типа «опровержение» это ситуация, когда одновременно существует два аргумента – A и $\sim A$. Однако, для логики предикатов первого порядка все несколько сложнее. Основная идея — применение механизма унификации для обнаружения конфликтов. Подстановка $U = \{t_1/x_1, t_2/x_2, \dots, t_n/x_n\}$, где t_i — термы и x_i — переменные называется унификатором для выражений W_1 and W_2 , если $W_1 * U = W_2 * U$, где $W_1 * U$ результат замены переменных x_1, x_2, \dots, x_n на термы t_1, t_2, \dots, t_n и предполагается, что W_1 и W_2 не содержат переменные с одинаковыми именами (если это так их необходимо переименовать) [Вагин и др., 2008]. Например выражения $P(x_1) \& G(x_2) \rightarrow H(x_1, x_2)$ и $P(f(a)) \& G(b) \rightarrow H(f(a), b)$ унифицируются унификатором $U = \{f(a)/x_1, b/x_2\}$.

Будем говорить, что два аргумента $A_1 = p_1 / X_1$ и $A_2 = p_2 / X_2$ вступают в конфликт типа «опровержение», если существует унификатор U , такой что:

1. $\sim p_1 * U = p_2 * U$.
2. $X_1 * U \in X_2 * U$ или $X_2 * U \in X_1 * U$.

Аргумент A_1 подрывает пересматриваемую связь между аргументами A_2 и A_3 :

1. Существует пересматриваемая связь между A_2 и A_3 .
2. Существует подрывающее правило $E \Rightarrow (C @ D)$ и существуют такие унификаторы U_1 и U_2 , такие что:
 - a. $E * U_1 = A_1 * U_1$
 - b. $(C * U_1) * U_2 = (A_2) * U_2$
 - c. $(D * U_1) * U_2 = (A_3) * U_2$

Определив все возникающие конфликты, можно определять статусы аргументов. На каждом шаге работы системы определение статусов каждого аргумента (поражается он или нет) играет ключевую роль. Введем необходимые для определения статусов поражения определения.

2. Степени обоснования в пересматриваемых рассуждениях.

Прежде чем переходить к способам и алгоритмам вычисления степеней обоснования,

рассмотрим как они могут задаваться и откуда получены. В данной статье для задания степеней обоснования используется числовая шкала $[0,1]$, где 0 соответствует пораженному аргументу, 1 наиболее обоснованному аргументу. Степени обоснования могут быть двух типов:

- 1) Степени обоснования исходных аргументов;
- 2) Степени обоснования пересматриваемых правил.

Первый тип степеней обоснования присваивается каждому исходному аргументу, и представляет собой некую оценку достоверности источника, из которого получен данный аргумент. Например, по телевизору сказали, что вероятность осадков 70%. Соответственно мы можем построить аргумент $A1: \text{Завтра}(\text{дождь})$ со степенью обоснования 0.7. Степени обоснования будем записывать функцией $Jus(A)$. То есть для приведенного примера $Jus(\text{Завтра}(\text{дождь})) = 0.7$. Конкретные механизмы получения степеней обоснования зависят прежде всего от предметной области. Например, это могут быть статистические данные (в 90% этот источник дает верные данные) или экспертные оценки (вероятность роста акций 60%).

Второй тип степеней обоснования связан с пересматриваемыми правилами. Как уже говорилось выше, часто пересматриваемые правила появляются в результате формализации знаний эксперта вида «Если A , то чаще всего B ». Такие правила так же могут нести в себе некоторую числовую оценку.

Например, применение анальгина в 85% приводит к снижению температуры тела пациента (формально $R1$:

$$(\forall x) \text{ прием(анальгин, } x) \mid \Rightarrow \text{понижение_температуры}(x).$$

Одновременное использование обоих типов степеней обоснования является довольно сложной задачей и требует дополнительных исследований. В данной статье ограничимся рассмотрением степеней обоснования первого типа – для изначально заданных аргументов.

Итак, нам необходимо задать функцию $Jus(A)$ для вычисления любого из аргументов в графе вывода. Будем считать, что для изначальных аргументов эта величина является определенной. На значение этой функции будут оказывать влияние два фактора – дерево вывода аргумента (т.е. степень обоснования аргументов, которые использовались в выводе данного аргумента) и конфликты с другими

аргументами. Для удобства рассмотрим эти два фактора отдельно: $Jus_{anc}(A)$ – унаследованная степень обоснования и $Jus_{con}(A)$ – на сколько конфликт подрывает обоснование аргумента. Пусть $Anc = \{Anc_i\}$, $i \in 1..n$ – множество аргументов Anc_i , учествовавших в выводе аргумента A , n – количество таких аргументов.

$$Jus_{anc}(A) = \min_{i \in 1..n} \{Jus(Anc_i)\} \quad (2.1)$$

Формулу (2.1) называют принципом слабой связи [Pollock, 2001]. Надо отметить, что это не единственный подход к вычислению степени обоснования, в ряде работ применяется байесовский подход (см. например [Naenni et al., 1999]).

Отметим, что из формулы (2.1) следует, что, если производить вычисление степеней обоснования рекурсивно, начиная от исходно заданных аргументов, то можно искать минимум, не из всех аргументов в базе, а только на предыдущем шаге. Таким образом, если у аргумента один предок, то его унаследованная степень обоснования будет равна степени обоснования его предка.

Если при вычислении Jus_{anc} ищутся наиболее слабые аргументы, то при определении того, насколько конфликт уменьшает обоснования, используется наиболее сильные аргументы. Пусть A_{confl} – множество аргументов, вступающих в конфликт с A , $n = |A_{confl}|$, тогда

$$Jus_{con}(A) = \begin{cases} \max_{i \in 1..n} \{Jus_{anc}(A_{confl_i})\}, & n > 0 \\ 0, & \text{в противном случае} \end{cases} \quad (2.2)$$

В формуле (2.2) используется Jus_{anc} для того, чтобы верно обрабатывать случаи, когда между аргументами есть конфликт типа опровержение. Итак, окончательно:

$$Jus(A) = \begin{cases} Jus_{anc}(A) - Jus_{con}(A), & Jus_{anc}(A) > Jus_{con}(A) \\ 0, & \text{в противном случае} \end{cases}$$

3. Алгоритмы

3.1. Алгоритм поиска конфликтов

Входные данные: Граф вывода, список подрывающих следствий **undercut_rules**

Выходные данные: Граф вывода с конфликтами

Шаг 1. Для каждого аргумента $Arg_1 = a_1/X_1$ проверить не существует ли $Arg_2 = a_2/X_2$, такого, что для a_1 и $\sim a_2$ существует унификатор U и

$X_1 * U \in X_2 * U$ или $X_2 * U \in X_1 * U$. Если такой унификатор существует, добавить конфликт между Arg_1 и Arg_2 .

Шаг 2. Для каждого подрывающего правила uc_rule вида $p \Rightarrow (q @ r)$ из **undercut_rules** выполнить:

Шаг 3. Взять левую часть правила - p . Для каждого аргумента $Arg = a/X$, проверить существует ли унификатор U_{left} для выражений a и p .

Шаг 4. Для тех аргументов Arg , которые унифицируемы с левой частью подрывающего правила выполнить следующие шаги:

Шаг 4.1. Применить унификатор U_{left} к q и r ($q_u = q * U_{left}$ и $r_u = r * U_{left}$).

Шаг 4.2. Составить массив

$X = [\{xq_1, xr_1\}, \{xq_2, xr_2\}, \dots, \{xq_n, xr_n\}]$ состоящий из пар аргументов, соединенных пересматриваемой. Для каждого элемента этого массива выполнить:

Шаг 4.2.1 Проверить существует ли унификатор U_{right} для q_u и xq_i . Если найденный унификатор так же является унификатором для r_u и xr_i , добавить новый конфликт в граф вывода.

3.2. Алгоритм вычисления степеней обоснования.

Процедура **recalculate_justification_degree()**;

Входные данные: Граф вывода.

Выходные данные: Граф вывода

Шаг 1. Составить очередь $Qlist$ вычисления степеней обоснования из аргументов входящих в состав графа вывода, начиная от изначально заданных аргументов.

Шаг 2. Пока очередь $Qlist$ не пуста извлечь из нее очередной элемент q и выполнить процедуру $Jus(q)$.

Процедура **Jus(argument q, [argument ex])**;

Входные данные: Аргумент q , необязательный параметр – аргумент ex , который не учитывается при определении степени обоснования аргумента q , граф вывода.

Выходные данные: Граф вывода

Шаг 1. Если q – изначально заданный аргумент, то положить $Jus_{anc}(q)$ равным изначально заданной степени доверия для этого аргумента. В противном случае из всех Anc_i являющихся непосредственным предком q (то есть q получается из Anc_i за один шаг) найти такой Anc_min , что его степень обоснования минимальна. Положить $Jus_{anc}(q) = Jus(Anc_min)$.

Шаг 2. Если аргумент q не имеет конфликтов, тогда положить $Jus_{con}(q) = 0$. В противном случае составить множество $\{A_i\}$ состоящее из аргументов имеющих конфликт с q . Если параметр ex задан, исключить его из множества $\{A_i\}$. Если к моменту вычисления $Jus(q)$ среди A_i нет аргументов с неопределенной степенью обоснования, то среди них найти тот аргумент A_max , у которого $Jus(A_max)$ максимален. Положить $Jus_{con}(q) = \max(Jus(A_max), Jus_{anc}(ex))$, выполнить $Jus(ex)$, перейти к шагу 4. В противном случае (т.е. среди $\{A_i\}$ есть аргументы с неопределенными степенями

обоснования) выполнить шаг 3.

Шаг 3. Для каждого A_i , для которого не определена степень доверия выполнить:

Шаг 3.1. Если A_i имеет конфликт с q , но аргумент q не имеет с ними конфликта (несимметричная форма конфликта) выполнить $Jus(A_i)$. В противном случае выполнить $Jus(A_i, q)$, т.е. рассчитать $Jus(A_i)$ без учета конфликта с q .

Шаг 4. Положить $Jus(q) = Jus_{anc}(q) - Jus_{con}(q)$, если $Jus_{anc}(q) > Jus_{con}(q)$. В противном случае $Jus(q) = 0$.

4. Пример работы системы аргументации

Имеется следующая задача: Предприниматель хочет вложить некоторую сумму в развивающийся бизнес и получить от этого доход. Есть компания CMP, которая занимается IT бизнесом. По статистике IT рынок находится на подъеме и 75% IT компаний показывают положительную динамику развития. Компания CMP занимается разработкой программного обеспечения под операционные системы семейства Linux. В России менее 10% пользователей Linux и следовательно спрос на продукцию будет не велик. Однако 80 процентов производства компании направлено на международный рынок программного обеспечения для веб серверов, которые в основном работают на Linux.

Запишем условия формально на языке ЛППП.

A1. IT(CMP) justification: 0,75

A2. под_Linux(CMP) justification: 1

A3. рынок_сбыта(для_серверов) justification: 0,8

A4. $(\forall x)(\text{инвестировать}(x) \rightarrow \text{доход})$

R1. $(\forall x)(\text{IT}(x)) \Rightarrow (\forall x) \text{рост_акций}(x)$
R2. $(\forall x)(\text{рост_акций}(x)) \Rightarrow \text{инвестировать}(x)$
R3. $(\forall x)(\text{под_Linux}(x)) \Rightarrow \text{плохой_спрос}(x)$
R4. $(\forall x)(\text{плохой_спрос}(x)) \Rightarrow \sim \text{инвестировать}(x)$
R5. $(\forall x) \text{рынок_сбыта}(\text{для_серверов}) \Rightarrow ((\text{под_Linux}(x)) @ \text{плохой_спрос}(x))$

На рисунке 2 приведен граф вывода для данной задачи.

Рассмотрим решение данной задачи поэтапно.

1) Из аргумента A1: IT(CMP) и правила вывода R1 получается пересматриваемый аргумент A6: $\text{рост_акций}(\text{CMP})$. $Jus(A6) = Jus(A1) = 0.75$.

2) Из аргумента A2: под_Linux(CMP) и правила вывода R3 получается пересматриваемый аргумент A8: $\text{плохой_спрос}(\text{CMP})$. $Jus(A8) = Jus(A2) = 1$.

3) Аргумент A8: $\text{плохой_спрос}(\text{CMP})$ с помощью правила R5 вступает в конфликт типа подрыв с аргументом A3: $\text{рынок_сбыта}(\text{для_серверов})$. Согласно формуле 1.3 $Jus(A3) = Jus_{anc}(A3) - Jus(A8) = 0.2$

4) Из аргумента A6 и правила R2 получается аргумент A7: $\text{Инвестировать}(\text{CMP})$. $Jus(A7) = Jus(A6) = 0.75$.

5) Из аргумента A8 и правила вывода R4 получается аргумент A9: $\sim \text{Инвестировать}(\text{CMP})$.

6) Аргументы A9 и A7 вступают в конфликт типа опровержение, степень обоснования аргумента A7 выше, и следовательно $Jus(A7) = 0.55$, а аргумент A9 становится полностью пораженным с $Jus(A9) = 0$.

7) Используя правила обратного вывода из интереса In5 получается интерес In11, из которого получается интерес In14.

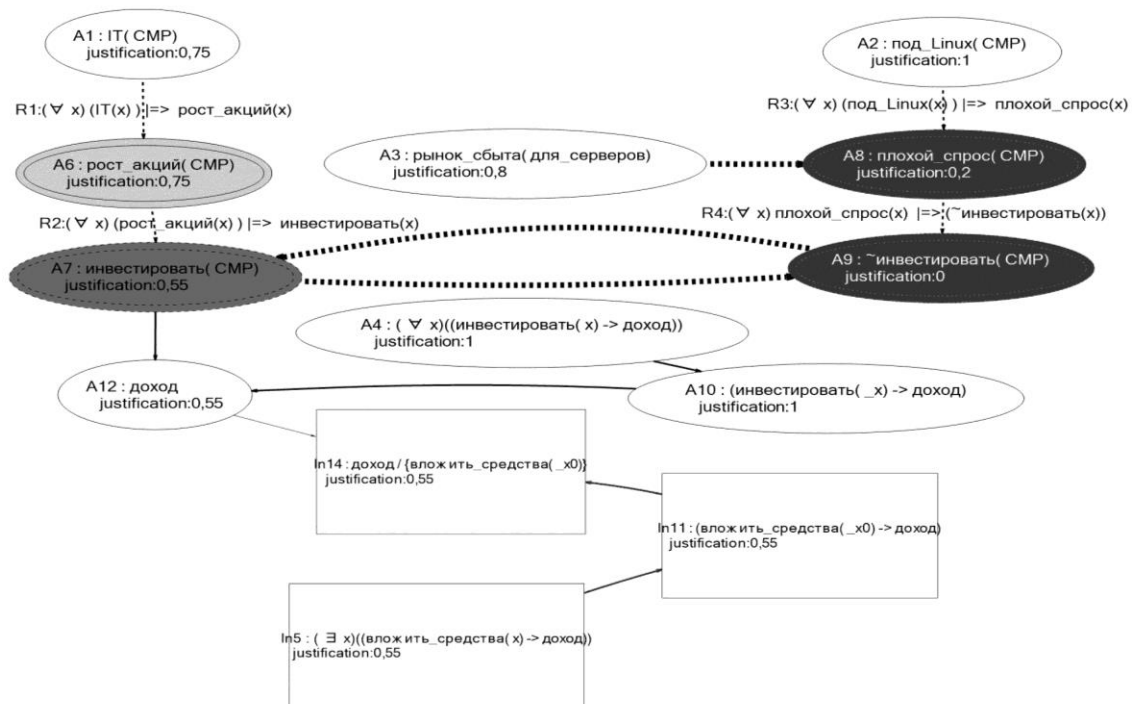


Рисунок 2 - Пример решения задачи со степенями обоснования

8) Из аргумента A10: Инвестировать(x) \rightarrow доход, аргумента A7:Инвестировать(CMP) применяя унификацию и правило Modus Ponens получаем аргумент A12. Согласно принципу слабой связи $Jus(A12) = 0.55$. Этот Аргумент подтверждает интерес In14.

9) Окончательно получаем, что интересы подтверждены со степенью обоснования 55%. То есть инвестиция в компанию CMP принесет доход с вероятностью 55%.

Заключение

Конфликты в базах знаний являются серьезной проблемой при построении любых интеллектуальных систем. Поиск, анализ и, возможно, разрешение этих конфликтов – важная и интересная задача. В данной статье был рассмотрен один из подходов к данной проблеме – использование аргументации, а именно теории пересматриваемых рассуждений. Были рассмотрены теоретические основы пересматриваемых рассуждений и дан обзор системы, основанной на них. Кроме того, применение механизма степеней обоснования позволило решить важную задачу – давать не только качественные оценки («поражен» или «не поражен»), но и количественные. Разработанная программная система справилась со многими задачами, не решаемыми с точки зрения классической логики. Она была протестирована и отлажена на многих тестовых задачах из статьи Врейсвика «Interpolation of Benchmark Problems in Defeasible Reasoning» [Vreeswijk, 1995].

Библиографический список

- [Besnard, 2008] Besnard P., Hunter A. Elements of Argumentation. – Cambridge MA: MIT Press, 2008. – 298 p.
- [Bondarenko et al., 1997] Bondarenko A., Dung P.M., Kowalski R.A., Toni F. An Abstract Argumentation-Theoretic Framework for Defeasible Reasoning // Artificial Intelligence. – 1997. – Vol. 93, № 1-2. – P. 63-101..
- [Lin et al., 1989] Lin F., Shoham Y. Argument Systems. A Uniform Basis for Nonmonotonic Reasoning // Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning. – San Mateo CA: Morgan Kaufmann Publishers Inc, 1989. – P. 245-355.
- [Vreeswijk, 1989] Vreeswijk G.A.W. Abstract Argumentation Systems // Artificial Intelligence. – 1997. – Vol. 90. – P. 225-279.
- [Pollock, 1992] Pollock J.L. How to Reason Defeasibly // Artificial Intelligence. – 1992. – Vol. 57. – P. 1-42.
- [Вагин и др., 2008] Вагин В.Н., Головина Е.Ю., Загорянская А.А., Фомина М.В. Достоверный и правдоподобный вывод в интеллектуальных системах / Под ред. В.Н. Вагина, Д.А. Поспелова. 2-е издание дополненное и исправленное. – М.: Физматлит, 2008. – 712 с.
- [Pollock, 2001] Pollock J.L. Defeasible reasoning with variable degrees of justification, // Artificial Intelligence. Artificial Intelligence. – 2001. – Vol. 133. – P. 233-282.
- [Haenni et al., 1999] Haenni R., Kohlas J., Lehmann N. Probabilistic Argumentation Systems, Handbook of Defeasible Reasoning and Uncertainty Management Systems // Algorithms for Uncertainty and Defeasible Reasoning. – 1999. – Vol. 5. – P. 221–287.
- [Vreeswijk, 1995] Vreeswijk G.A.W. Interpolation of Benchmark Problems in Defeasible Reasoning // Proceedings of the 2nd World Conference on the Fundamentals of Artificial Intelligence (WOCFAI 95). – Paris, France, 1995. – P. 453-468.

METHODS AND ALGORITHMS FOR CALCULATING JUSTIFICATION DEGREES IN ARGUMENTATION SYSTEMS

Morosin O.L. *

**Moscow Power Engineering Institute*

omorsik@gmail.com

This paper provides a brief overview of approaches to the formalization of argumentation systems. Opportunities of application of justification degrees are also observed. Justification degrees allow us to solve various argumentation problems that need numerical estimation of an answer. In contrast to classical logic, defeasible reasoning allows us to draw conclusions on the contradictory and incomplete sets of propositions.

Introduction

Most practical problems are rather poor formalized, and may contain inconsistent, uncertain and implausible information. The methods of the classical logics could not be applied for solving such problems. It is proposed to use the methods of argumentation to cope with uncertain and inconsistent data.

Main Part

There are several formalizations of the argumentation theory. In this paper, we will consider the development of defeasible argumentation systems that use the first order logic as an underlying language and based on the theory of defeasible reasoning proposed by John Pollock. All conclusions are not considered as reliable and may be revised at a later stage of reasoning when new knowledge (or even new conclusions from existing knowledge) appears.

The concept of the conflict is the main point in theory of defeasible reasoning. Moreover, the possibility of conflicts is the reason why defeasible reasoning is considered. The paper contains two types of conflict: rebutting and undercut. Defeasible reasoning allows us to find and analyze such conflicts.

Opportunities of application of justification degrees are also observed. Justification degrees allow us to solve various argumentation problems that need numerical estimation of an answer.

Conclusion

Argumentation system based on defeasible reasoning could be used in many intelligent systems that use the first order logic as a method of knowledge representation. It was shown that it is a powerful tool for dealing with inconsistent data. Nevertheless, the implementation of different mechanisms of calculating justification degrees, including methods of fuzzy logic, is a topic of a future work.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:658.51

ОНТОЛОГИИ ЖИЗНЕННОГО ЦИКЛА СЛОЖНОЙ ТЕХНИЧЕСКОЙ СИСТЕМЫ

Тарасов В.Б.^{*}, Федотова А.В.^{*}, Черепанов Н.В.^{*}

^{*} *ФГБОУ ВПО «Московский государственный технический университет им. Н.Э.Баумана»
г. Москва, Россия*

Vbulbov@yahoo.com

afedotova@acm.org

^{**} *ФГУП «НПО им. С.А. Лавочкина»
г. Химки, Московская область, Россия*

nv137@yandex.ru

Рассмотрены проблемы создания системы управления жизненным циклом сложной технической системы. Предложен онтологический подход к анализу и моделированию жизненных циклов. Построена система наглядных и абстрактных моделей ЖЦ. Особое внимание уделено проблемам построения гранулярных онтологий и развитию спиральных моделей жизненного цикла

Ключевые слова: представление знаний; управление знаниями; метамодель; онтология; метаонтология; грануляция; жизненный цикл.

Введение

Настоящая работа посвящена проблемам анализа и моделирования жизненного цикла сложных технических систем (СТС), включающего стадии проектирования, производства и эксплуатации [Валькман, 1998; Колчин, 2002; Stark, 2011; Tarassov, 1994].

Предложен новый подход к представлению и интеграции знаний о жизненном цикле, основанный на онтологическом моделировании и методах грануляции информации.

1. Подсистема управления жизненным циклом продукции в структуре интегрированной системы управления предприятием

Согласно традиционным представлениям об управлении предприятием, процессы управления ограничиваются его внутренней средой. Внутреннее управление охватывает такие проблемы как управление ресурсами предприятия, финансовое управление, управление персоналом, управление сбытом и пр. Однако в современную эпоху сетевой экономики в сферу внимания руководства компании и ответственности ее менеджеров начинает входить все, что как-либо влияет на производительность и

конкурентоспособность – как внутри предприятия, так и за его пределами. Появление «электронно прозрачного» рынка, развитие электронного бизнеса означает выведение процессов управления за пределы предприятия, их перенос в виртуальную среду, основанную на веб-технологиях. Возникают задачи управления компонентами внешней микросреды предприятия, организации тесного взаимодействия компании с ее клиентами, поставщиками и партнерами.

Все это привело к появлению систем планирования и управления ресурсами предприятия второго поколения ERP-II (Enterprise Resource Planning-II), для которых объединение ресурсов не ограничивается рамками монолитного предприятия и включает обмен ресурсами по корпоративной сети. Здесь ключевое место занимают системы управления отношениями: с заказчиками CRM (Customer Relationship Management) и поставщиками SRM (Supplier Relationship Management). Вместе с системами управления жизненным циклом продукции PLM (Product Lifecycle Modeling) и системами управления цепочками поставок SCM (Supply Chain Management) они образуют «джентльменский набор» инструментальных средств ERP-II [Тарасов, 2006; SAP ERP, 2008] (рисунок 1).

В данной работе основное внимание уделяется онтологическим вопросам управления жизненным

циклом сложных технических систем (СТС). Здесь под СТС понимаются искусственные системы, состоящие из большого числа неоднородных элементов и подсистем, которые обладают значительным разнообразием внутренних связей и разветвленной структурой. Для них типичными являются множественность возможных состояний, а также градации работоспособных состояний (отказ отдельных элементов снижает эффективность функционирования СТС в целом, но необязательно приводит к скорой и неизбежной аварийной ситуации). Яркими примерами СТС служат современные корабли, самолеты, космические аппараты, орбитальные обсерватории и т.п. Обычно подобные системы входят в состав еще более сложных человеко-технических комплексов.

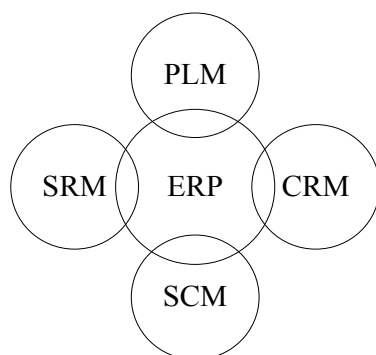


Рисунок 1 – Основные компоненты интегрированной системы управления предприятием ERP-II.

Управление жизненным циклом СТС означает интеграцию данных, процессов, структур, персонала ряда предприятий в рамках единой корпоративной сети [Тарасов, 1998; Spur, 1995; Stark, 2011]. Его ключевыми аспектами являются:

- безопасный и управляемый доступ к информации о создаваемой системе, а также ее совместное использование в спецификации СТС;
- обеспечение согласованности информации о СТС на протяжении всего ЖЦ;
- поддержка процессов создания, управления, распространения, разделения и использования информации о СТС.

С одной стороны, одной из главных подсистем системы управления жизненным циклом продукции является система управления данными о продукции PDM (Product Data Management). Сегодня PDM-система в достаточно полном объеме реализует функции управления составом СТС, структурой всех ее составных частей, деталей, узлов и агрегатов. Кроме того, сюда должны входить и структурированные информационные объекты технологической подготовки производства, состав которых отражает все необходимые данные для организации работ по производству самого изделия, структура оснастки, инструментального парка, операций и переходов, технологических приемов.

Задачи управления данными об СТС и управления работами по созданию СТС являются взаимосвязанными и должны базироваться на

единых информационных ресурсах. При этом важное место занимает построение системы управления проектом, основная задача которой состоит в координации действий исполнителей проекта, обеспечении согласованного планирования и контроля работ, предоставлении необходимой информации о ходе выполнения проекта.

Интеграцию данных на различных этапах ЖЦ удобно осуществлять на основе семантического подхода, опирающегося на онтологическое описание проблемной области в терминах теории множеств и алгебраических систем.

С другой стороны, важнейшей подзадачей задачи управления жизненным циклом СТС является управление знаниями, циркулирующими на его этапах и стадиях. Управление знаниями на протяжении жизненного цикла – это инновационная стратегия работы предприятия, направленная на обмен знаниями как ресурсами, коллективно используемыми на разных этапах ЖЦ, получение новых и обновление существующих знаний, позволяющих сотрудникам лучше понимать друг друга, эффективно взаимодействовать и совместно решать сложные задачи. Оно обеспечивает единый, интегрированный подход к созданию, сбору, организации, распределению, использованию информационных ресурсов предприятия и доступу к ним. Эти ресурсы включают структурированные базы данных, документы, неявные знания сотрудников [Попов, 2001; Кудрявцев, 2010].

1.1. Жизненный цикл системы

Понятие «жизненный цикл» и модели жизненных циклов (ЖЦ) для различных классов естественных и искусственных систем являются одними из важнейших объектов современной теории систем как междисциплинарной научной области [Волкова, 2004; Спицнагель, 2000]. В то же время различные по содержанию и структуре жизненные циклы изучаются в конкретных научных областях. Так в современном маркетинге одной из базовых концепций является представление о жизненном цикле товара – периоде времени, в течение которого товар обращается на рынке, начиная с момента его выхода на рынок и заканчивая уходом с рынка. Соответственно, выделяют такие этапы ЖЦ товара как его выведение на рынок, этап роста, этап зрелости, этап спада и ухода с рынка [Райзберг, 2007].

Любой цикл характеризуется завершенностью и повторяемостью взаимосвязанных этапов на определенном промежутке времени; его ключевыми характеристиками являются длительности: как отдельных этапов, так и цикла в целом [Субетто, 1989; Спицнагель, 2000]. Весь жизненный цикл СТС определяется интервалом времени от начала ее создания до конца эксплуатации; при этом за начало жизненного цикла традиционно принято считать формирование потребности в СТС (зарождение идеи или облика СТС), а за конец – снятие СТС с эксплуатации (утилизацию). По сути, ЖЦ – это

множество состояний СТС, связанных между собой допустимыми переходами.

Концепция жизненного цикла представляет собой основной вариант реализации системного подхода к сложным техническим объектам, направленный на отображение изменений состояний этих объектов в некоторый период времени. Она связана с интеграцией процессов проектирования, производства и эксплуатации СТС в рамках единой метамодели и предполагает выделение ряда стадий (этапов) ЖЦ и изучение взаимосвязей между ними.

В конце XX-го – начале XXI-го века понятие ЖЦ расширилось; в него также стали включать стадию рекуперации (Recycling), которая лежит в основе концепции «обращения ЖЦ» [Kimura, 1995]. В статье [Тарасов, 1998] введена трехмерная система жизненных циклов «ЖЦ продукта – ЖЦ процесса – ЖЦ предприятия» и предложены варианты ее грануляции.

Структура и продолжительность ЖЦ СТС влияют на затраты по ее созданию и эксплуатации. Это влияние является достаточно сложным и неоднозначным: так исключение из ЖЦ какого-либо этапа отнюдь не всегда означает уменьшение общих затрат, и напротив, удлинение и более тщательная проработка отдельных этапов (например, этапа технического обслуживания и ремонта) может дать существенную экономию и привести к снижению общих затрат ЖЦ.

В конце XX-го–начале XXI-го века возникла концепция инженерии жизненного цикла (Lifecycle Engineering) [Alting, 2006; Spur, 1996], которая предполагает широкое использование современных информационных и коммуникационных технологий в моделировании и интеграции его этапов. Главными аспектами инженерии ЖЦ выступают инженерия знаний и управление знаниями о ЖЦ, в частности, на основе наглядного представления его структуры, оптимизация временных соотношений между этапами и стадиями ЖЦ СТС (например, сокращение сроков проектирования и увеличение периода эксплуатации СТС), учет и управление неопределенностями, возникающими на разных этапах ЖЦ [Tarassov, 1996].

Ниже будет изложен онтологический подход к управлению знаниями в русле интеллектуализации систем управления жизненным циклом класса PLM.

1.2. Онтологический подход к анализу и моделированию жизненных циклов

В информатике под онтологией понимается наглядное и формализованное описание структуры некоторой проблемной области (темы). Подобное описание всегда опирается на концептуализацию этой области, которая обычно задается в виде исходных понятий, отношений между ними и ограничений.

Следуя Т.Груберу, онтологию определяют как

спецификацию разделяемой разными людьми концептуализации [Gruber, 1993], а по Н.Гуарино она представляет собой логическую теорию, которая состоит из словаря терминов, образующих таксономию, их определений и атрибутов, а также связанных с ними аксиом и правил вывода [Guarino, 1995]. Иными словами, онтологию понимают и как формальный взгляд на семантику, и как словарь, используемый логической теорией [Гаврилова, 2003]. По сути, онтологии отражают соглашения о единых способах построения и использования концептуальных моделей. Они выступают как удобный метод представления и повторного использования знаний, средство управления знаниями, способ обучения [Кудрявцев, 2010].

Следует отметить, что в рамках проекта семантической интерпретации информационных ресурсов Интернет (Semantic Web) в 2000-е годы был предложен стандарт описания метаданных о документе Resource Description Framework (среда описания ресурсов), в основе которого лежит представление информации с помощью троек «субъект-предикат-объект». В связи с этим нельзя не вспомнить, что еще в 1970-е годы известный белорусский ученый В.В.Мартынов предложил близкую по форме ядерную цепочку «субъект-действие-объект» (см. [Мартынов, 2001]) для своего языка представления знаний, названного им «универсальный семантический код».

Построение единственной понятной и согласованной онтологии, как правило, оказывается невозможным, что приводит к формированию иерархической системы онтологий. В ней на нижнем уровне представлены предметная онтология, онтология задач и онтология приложений; выше них находится онтология верхнего уровня, Общая схема взаимосвязей между онтологиями (см.[Тарасов,2012]) показана на рисунке 2.

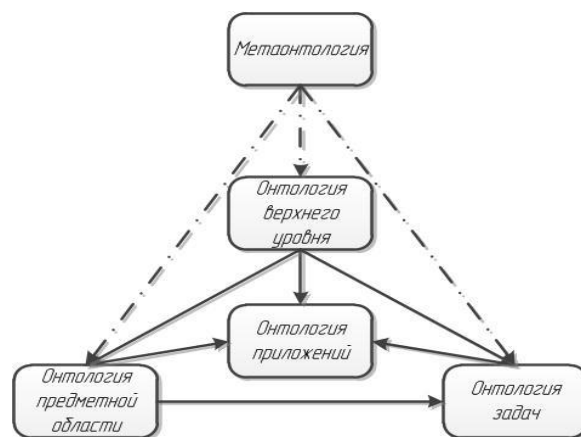


Рисунок 2 – Иерархическая система онтологий

В данной работе главное внимание уделяется выбору метаонтологии и построению онтологии верхнего уровня. Здесь термин «метаонтология» (т.е. онтология над онтологиями) понимается как основа представления, слияния и интеграции онтологий. Метаонтология обеспечивает как

точную, математическую спецификацию онтологий, так и формальный анализ их свойств. С ее помощью устанавливают соответствие между характером располагаемой информации (по сути, уровнем неопределенности) и выбираемым языком описания СТС. Отсюда видно, что выбор подходящей метаонтологии во многом определяет свойства нижележащих онтологий.

В [Тарасов, 2012] введено понятие «гранулярной метаонтологии», что означает выбор в качестве онтологических примитивов не точечных, а протяженных объектов. Гранулярная метаонтология задается формально как алгебраическая система (по А.И.Мальцеву) тройкой $GMONT = \langle C_G, R_G, \Omega_G \rangle$, где C_G – базовое гранулярное множество, понимаемое как основа онтологической грануляции, R_G – множество отношений на C_G , а Ω_G – множество операций на C_G и/или R_G .

В качестве базиса онтологической грануляции могут выступать фактор-множества, вложенные множества, приближенные множества, нечеткие множества, мультимножества и пр.

Согласно Дж.Сова [Sowa, 1995], онтологии верхнего уровня описывают наиболее общие, парадигматические концептуализации систем, используемые в разных предметных областях, формируя базовую систему понятий для онтологий нижнего уровня.

В настоящей разделе в качестве онтологий верхнего уровня, тесно связанной с различными временными онтологиями [Кандрашина, 1989; Еремеев, 2003; Euzenat, 2005], рассматривается гранулярная онтология жизненного цикла системы.

Гранулярные метаонтологии, опирающиеся на понятия гранулы, уровня, иерархии, отношений между уровнями, предполагают рассмотрение базовых понятий нижележащей онтологии на

различных уровнях абстрактности. В случае ЖЦ СТС такими базовыми понятиями являются этапы и стадии жизненного цикла.

Жизненный цикл охватывает ряд стадий, которые представляют собой наиболее крупные его периоды, а эти стадии, в свою очередь, разбиваются на отдельные этапы. Этапы ЖЦ обычно связывают с контрольными точками (или интервалами) изменений СТС на протяжении ее жизни.

Грануляция ЖЦ предполагает рассмотрение таких вопросов как определение общих принципов и критериев грануляции; построение вариантов интерпретации и классификации гранул; развитие подходов и методов грануляции; разработка формальных моделей гранул; формирование отображений между разными уровнями грануляции; выделение количественные характеристик как гранул, так и самого процесса грануляции.

Следует отметить, что не существует одного оптимального уровня грануляции ЖЦ на отдельные этапы: размеры гранул являются проблемно-ориентированными и зависят от контекста исследования. Одни стадии ЖЦ могут быть представлены более детально, а другие – менее подробно в зависимости от целей моделирования [Валькман, 1998]. Будем рассматривать модели ЖЦ с разной степени абстрактности: а) более абстрактные круговые модели, связывающие между собой стадии ЖЦ; б) более конкретные модели взаимосвязей между этапами, а также между этапами и стадиями.

Удобным и наглядным средством визуализации онтологий являются ментальные карты (см. [Гаврилова, 2008]). На рисунке 3 приведен пример ментальной карты для онтологии ЖЦ СТС.

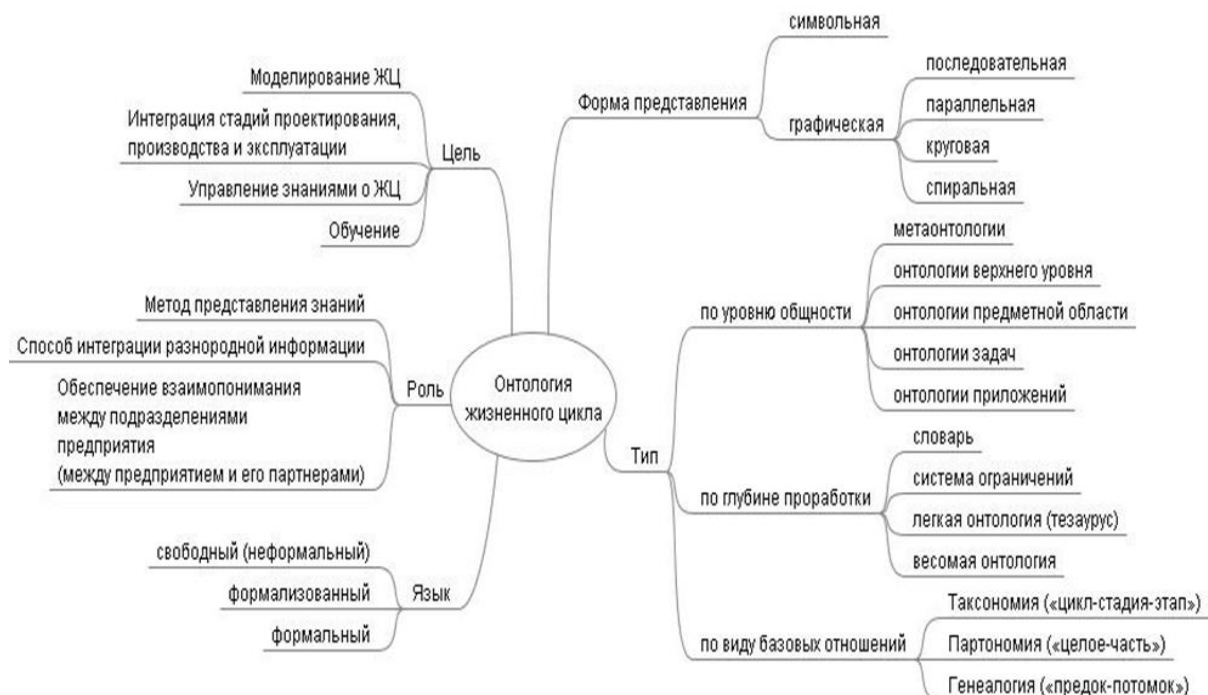


Рисунок 3 – Онтология жизненного цикла: представление с помощью ментальной карты

Введем формальное представление класса онтологий жизненного цикла ONT_{LC} сложной технической системы в виде модифицированной алгебраической системы, т.е. четверки

$$ONT_{LC} = \langle C_{LC}, R_{LC}, \Omega_{LC}, T_{LC} \rangle, \quad (1)$$

где C_{LC} – множество понятий, связанных с ЖЦ, R_{LC} – множество отношений между этими понятиями, Ω_{LC} – множество операций над понятиями и/или отношениями, T_{LC} – множество временных параметров, используемых на протяжении ЖЦ.

Поскольку базовыми понятиями ЖЦ являются его этапы и стадии, в качестве системного ядра (1) можно взять тройку

$$ONT_S = \langle S, R_s, \Omega_s \rangle, \quad (2)$$

где S – множество стадий (этапов) ЖЦ, R_s – множество связей между этими стадиями (этапами), Ω_s – множество операций, применяемых на стадиях (этапах) ЖЦ. В частном случае, имеем

$$ONT_{S1} = \langle S, < \rangle, \quad (2^*)$$

где $<$ есть отношение строгого линейного порядка, т.е. нереклексивное, асимметричное, транзитивное и полное отношение.

1.3. Краткий обзор концепций времени применительно к моделированию жизненного цикла

Время и изменение являются двумя ключевыми понятиями при описании жизненного цикла. Кроме собственно времени, на различных этапах ЖЦ применяются такие важнейшие показатели как производительность и трудоемкость. В частности, производительность предприятия измеряется количеством продукции, произведенной им в единицу времени. Трудоемкость, как величина, обратная производительности, характеризуется затратами рабочего времени на производство единицы продукции. Поэтому анализ различных концепций и моделей времени, построение и применение формальных методов представления времени, временных (темпоральных) показателей, утверждений и зависимостей, используемых на протяжении жизненного цикла, выступают как необходимые условия управления ЖЦ СТС.

Вначале остановимся на четырех классических концепциях времени: субстанциональной и реляционной, статической и динамической [Анисов, 2001; Молчанов, 1990].

В рамках субстанциональной концепции время трактуется как особого рода субстанция, наряду с пространством, веществом и полем. Эта субстанция существует по своим собственным законам и не зависит от пространства, вещества и полей, оказывая со своей стороны существенное влияние на их бытие и движение. Время в этой концепции, с одной стороны, определяет длительность всех происходящих явлений, а с другой – упорядочивает

события, разделяя их на прошлые, настоящие и будущие.

Напротив, в реляционной концепции Лейбница время раскрывается как система отношений между событиями. Эта система априори не существует, а определяется свойствами взаимодействующих тел.

Согласно статической концепции времени, ассоциируемой с Ньютоном, события прошлого, настоящего и будущего существуют реально и в известном смысле одновременно; их разделение условно и зависит от выбора начальной точки отсчета и направления. Все моменты времени как точки временной шкалы являются равноправными и могут рассматриваться как совокупность данных; никакой момент времени не имеет каких-либо отличий или преимуществ по отношению к другим моментам. Однако время течет, т.е. его можно представить как «поток времени».

В русле динамической концепции времени как меры изменения, восходящей к Гераклиту и Аристотелю, реально существуют только события настоящего времени, т.е. особо выделяется настоящий момент времени «сейчас», отделяющий прошлое от будущего. Тогда события прошлого считаются уже не существующими, а события будущего – еще не наступившими. При этом время непрерывно течет: настоящее уходит в прошлое, а будущее становится настоящим. Эту концепцию времени подчас называют «воспринимаемым временем», поскольку оно отражает субъективное восприятие изменений, происходящих в нашем окружении. Здесь за основу берется отношение «предшествования-следования» или отношение «раньше-позже».

С легкой руки МакТаггарта модели времени, соответствующие статической и динамической концепциям, стали называть А-моделями и В-моделями соответственно.

Двумя наиболее известными метафорами времени являются «стрела времени» и «колесо времени». Эти метафоры характеризуют две геометрические интерпретации времени – линейное время и циклическое время. Наряду с трактовкой времени с помощью прямой линии существует также его представление как последовательности точек и концепция исторического времени, имеющего предысторию и последующую историю [Любинская, 2002]. Развитие компьютерных моделей стало предпосылкой появления новой концепции времени как ресурса.

В целом, мы разделяем позицию Д.А.Поспелова, который утверждает, что процесс разработки множества альтернативных моделей времени и соответствующих временных логик развивается аналогично появлению неклассических геометрий, таких как геометрии Лобачевского или Римана (см. [Кандрашина, 1989]). В этой связи мы не согласны с А.М.Анисовым, который считает, что примирить идеи цикла и порядка невозможно

[Анисов, 2001]. Линейная модель времени нередко может рассматриваться как фрагмент или локальная модель для общей циклической модели. Еще более ярким примером является сочетание идей цикла и порядка в спиральных моделях времени.

2. Наглядные представления жизненного цикла системы

2.1. Круговые модели жизненного цикла

Вначале будем представлять отдельные стадии ЖЦ в рамках теоретико-множественной модели как гранулы, полученные путем разбиения. Введем естественные обозначения стадий ЖЦ СТС: $П$ – проектирование; $Пр$ – производство; $Э$ – эксплуатации, $Р$ – рекуперация (Recycling). Тогда имеем, например,

$$ЖЦ_1 = П \cup Пр \cup Э, \quad П \cap Пр = \emptyset, \quad Пр \cap Э = \emptyset, \quad Э \cap П = \emptyset \quad (3)$$

или

$$ЖЦ_2 = Пр \cup Э \cup Р, \quad Пр \cap Э = \emptyset, \quad Э \cap Р = \emptyset, \quad Р \cap Пр = \emptyset \quad (4)$$

Структура ЖЦ₂ в (4) отражает «экологический императив» современного производства и тесно связана с концепцией «обращения ЖЦ» от стадии утилизации СТС к стадии ее создания [Kimura, 1996]. Первый вариант разбиения ЖЦ СТС в виде структуры ЖЦ₁ (3) можно изобразить в виде секторов круга (рисунок 4).

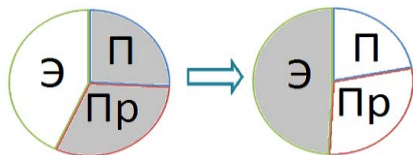


Рисунок 4 – Круговое представление жизненного цикла системы: иллюстрация идеи сокращения сроков проектирования и производства и продления периода эксплуатации системы

Одним из основных ресурсов управления жизненным циклом является время. Отличительной особенностью ЖЦ является гетерохронность, т.е. неравномерность, связанная с различием временных требований и критериев на разных стадиях. Так на стадии проектирования стремятся сократить сроки проектирования создаваемой системы, для чего применяют, например, стратегию совмещенного проектирования (Concurrent Design), но при этом также организуют проектирование, направленное на облегчение технического обслуживания (Design for Maintenance). Напротив, на стадии эксплуатации обычным критерием является увеличение периода эксплуатации системы, в том числе за счет большей детализации этапов данной стадии благодаря улучшению системы технического обслуживания и ремонта ТОиР.

Следует отметить, что представление ЖЦ как разбиения является весьма упрощенным и не

отражает имеющихся взаимосвязей, условий кооперации между стадиями, которые частично перекрываются, причем в этой области перекрытия реализуются важнейшие функции. Например, на пересечении $Э$ и $П$ формируется техническое задание на разрабатываемую систему, на стыке $П$ и $Пр$ разрабатываются технологии производства, а такие процессы как ТОиР предполагают взаимодействие специалистов из сфер эксплуатации и производства. Учет этих особенностей приводит к построению модели ЖЦ с нечеткими границами между его стадиями, т.е. грануляции ЖЦ на основе покрытия (рисунок 5). Здесь:

$$ЖЦ_i = П \cup Пр \cup Э, \quad П \cap Пр \neq \emptyset, \quad Пр \cap Э \neq \emptyset, \quad Э \cap П \neq \emptyset \quad (5)$$

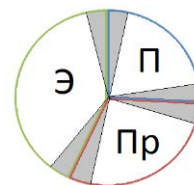


Рисунок 5 – Круговое представление ЖЦ на основе покрытия: наличие совместных работ на всех стадиях

2.2. Последовательные, инкрементальные и последовательно-параллельные модели

Классическая последовательная (водопадная) модель (рисунок 6) на основе ГОСТ 34.601-90 широко используется в проектно-конструкторской и производственной деятельности. В ней этапы идут один за другим: каждый следующий этап начинается только после окончания предыдущего этапа.

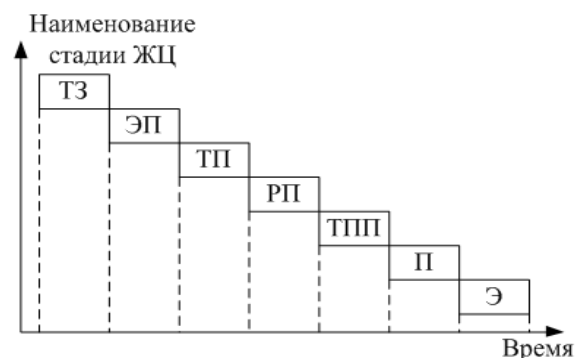


Рисунок 6 – Последовательная модель ЖЦ СТС (ТЗ – техническое задание; ЭП – эскизное проектирование; ТП – техническое проектирование; РП – рабочее проектирование; ТПП – технологическая подготовка производства; П – производство; Э – эксплуатация)

Главными преимуществами этой модели является простота и легкость ее понимания и использования, поскольку отсутствуют итерации и параллельно выполняемые задачи; она годится для небольших проектов и хорошо структурированного окружения. В то же время, данная модель плохо работает со сложными и долгосрочными проектами, а также при быстро меняющихся требованиях; она не поддерживает процедуры пересмотра для большого числа решений.

Инкрементальная модель также подразумевает линейную последовательность стадий ЖЦ, но предусматривает несколько инкрементов (версий) с запланированным улучшением продукции (рисунок 7). Достоинства и недостатки этой стратегии такие же, как и у классической (последовательной). Но в отличие от классической модели здесь заказчик может раньше увидеть результаты. Уже по результатам разработки и внедрения первой версии он может незначительно изменить требования к разработке, отказаться от нее или предложить создание более совершенного продукта с заключением нового договора.

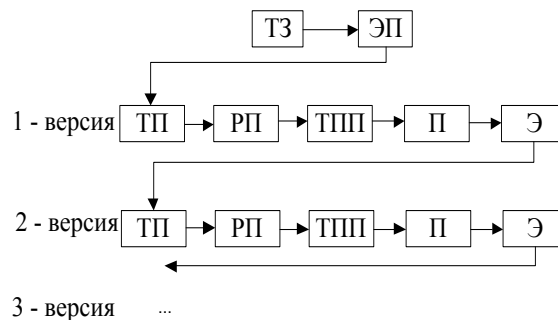


Рисунок 7 – Инкрементальная модель ЖЦ СТС

Последовательная и инкрементальная модели ЖЦ могут порой трактоваться как локальные фрагменты круговой модели

Таблица. Базовые временные отношения между этапами и стадиями жизненного цикла

Обозначения	Отношения и их инверсии	Иллюстрации	Примеры
r_1	Этап a выполняется раньше (предшествует) этапу b		Этап рабочего проектирования предшествует техническому обслуживанию
r_2	Этап b выполняется позже (следует за) этапом a		Этап технического обслуживания следует за рабочим проектированием
r_3	Этап a непосредственно предшествует (приближает к) этапу b		Этап эскизного проектирования непосредственно предшествует техническому проектированию
r_4	Этап b непосредственно следует за этапом a		Этап технического проектирования непосредственно следует за эскизным проектированием
r_5	Этап a частично пересекается с этапом b		Этап рабочего проектирования частично пересекается с этапом технологической подготовки производства
r_6	Этап b частично пересекается с этапом a		Этап технологической подготовки производства частично пересекается с этапом рабочего проектирования
r_7	Этап a лежит внутри стадии s		Этап технического обслуживания лежит внутри стадии эксплуатации
r_8	Стадия s содержит этап a		Стадия эксплуатации содержит этап технического обслуживания
r_9	Этап a лежит внутри стадии s так, что их начальные точки совпадают		Этап подготовки технического задания лежит внутри стадий разработки, причем их начальные точки совпадают
r_{10}	Стадия s содержит этап a так, что их начальные точки совпадают		Стадия разработки содержит этап подготовки технического задания, причем их начальные точки совпадают
r_{11}	Этап a лежит внутри стадии s так, что их конечные точки совпадают		Этап снятия СТС с эксплуатации лежит внутри стадии эксплуатации, причем их конечные точки совпадают
r_{12}	Стадия s содержит этап a так, что их конечные точки совпадают		Стадия эксплуатации содержит этап снятия с эксплуатации, причем их конечные точки совпадают
r_{13}	Этап a совпадает с этапом b		Этап детального проектирования совпадает с техническим и рабочим проектированием

Уменьшение сроков проектирования и производства в рамках ЖЦ СТС может быть достигнуто путем частичного запараллеливания отдельных этапов проектирования и производства (в русле совмещенной разработки) (рис.8).

В целом, моделирование связей между этапами и стадиями ЖЦ может опираться на базовые отношения между временными интервалами, лежащие в основе логики Аллена [Allen, 1983] (см. таблицу) и их расширения [Плесневич, 2003].

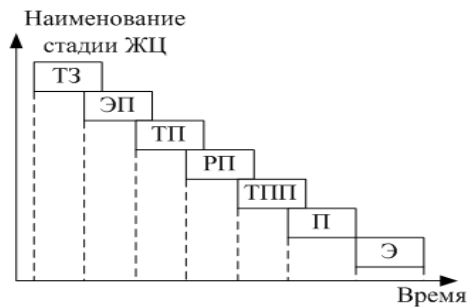


Рисунок 8 – Параллельно-последовательная модель ЖЦ СТС

В темпоральной логике Аллена в качестве временных примитивов используются интервалы. Здесь временной интервал a есть упорядоченная пара (a^-, a^+) , такая, что $a^- < a^+$, где a^- и a^+ рассматриваются как моменты времени (например, на вещественной оси). Структура времени может быть любой (например, линейной или ветвящейся) в зависимости от конкретной ситуации (см. [Еремеев, 2003; Кандрашина, 1989]).

Интервальной интерпретацией называется функция, отображающая временной интервал на числовую ось. Множество базисных интервальных отношений временной логики Аллена определяется через отношения между концами интервалов (см. таблицу).

Взаимное усиление достоинств и компенсация недостатков линейной и круговой моделей в отдельности достигается путем их интеграции и построения синтетической, спиральной модели ЖЦ.

Линейная модель времени выражает такие свойства времени как течение, направленность, необратимость, тогда как круговая модель делает акцент на итеративность и ритмичности процессов на протяжении ЖЦ. Итеративный характер работ адекватно отражается с помощью спиральной модели. При этом обеспечивается возможность перехода на следующий этап ЖЦ, не дожидаясь окончания предыдущего этапа. Более того, недостающую работу можно выполнить на следующей итерации.

Наиболее известной спиральной моделью ЖЦ считается модель Б.Бозма [Boehm, 1986], которая предназначена для отображения жизненного цикла программного обеспечения (ЖЦ ПО) и предусматривает анализ рисков (рисунок 9).



Рисунок 9 – Спиральная модель ЖЦ Бозма

Отметим, что исторически более ранней и существенно более детальной была спиральная модель ЖЦ Л.А.Кашубы и В.Б.Тарасова. Она изложена в кандидатской диссертации последнего, защищенной в 1982г., но подробные публикации об этой модели появились лишь в 1990-х годах [Кашуба, 1998; Tarassov, 1994].

2.3. Спиральные представления жизненного цикла системы: формальные модели и их содержательный анализ

Рассмотрим формальную запись спиральной модели ЖЦ СТС. Спираль – это кривая, которая огибает некоторую центральную точку или ось, постепенно приближаясь или удаляясь от неё, в зависимости от направления обхода кривой [Савелов, 2010]. Спирали обычно описывают в полярных координатах.

Пусть $M=(x,y)$ – произвольная точка плоскости, заданная своими декартовыми координатами. Поставим теперь в соответствие этой точке M два других числа – ее полярные координаты, а именно, число r , равное длине отрезка OM : $r = l(OM)$, и число φ , равное величине угла в радианах между положительной полуосью абсцисс и отрезком OM , причем угол отсчитывается в направлении против часовой стрелки (рисунок 10). В результате имеем $M = (r, \varphi)$, где число r называется радиусом точки M , а число φ – ее углом.

Спираль Архимеда (рисунок 10) есть плоская кривая, которую описывает точка, движущаяся равномерно-поступательно от центра O по равномерно вращающемуся радиусу. В полярных координатах уравнение Архимедовой спирали, положенной в основу модели ЖЦ [Tarassov, 1994] имеет вид

$$r = a\varphi + b \quad (6)$$

где a – параметр вращения спирали и b – характеристика расстояния между витками. Здесь возможна следующая интерпретация параметров спирали: угол φ – период времени, коэффициент a – производительность работ, коэффициент b – уровень грануляции, а r – состояние системы.

Главным недостатком спирали Архимеда (6) является равномерное расстояние между витками, что не позволяет отобразить такие явления как ускорение ЖЦ (уменьшение времени разработки) на его начальных этапах и замедление ЖЦ на стадии эксплуатации (продление сроков эксплуатации).

Более адекватным представлением ЖЦ видится логарифмическая спираль (рисунок 11), которая задается следующим уравнением в полярных координатах

$$r = \exp(a\varphi). \quad (7)$$

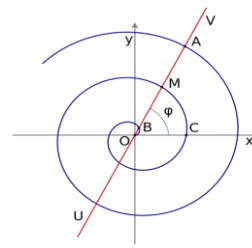


Рисунок 10 – Спираль Архимеда

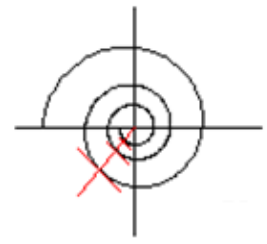
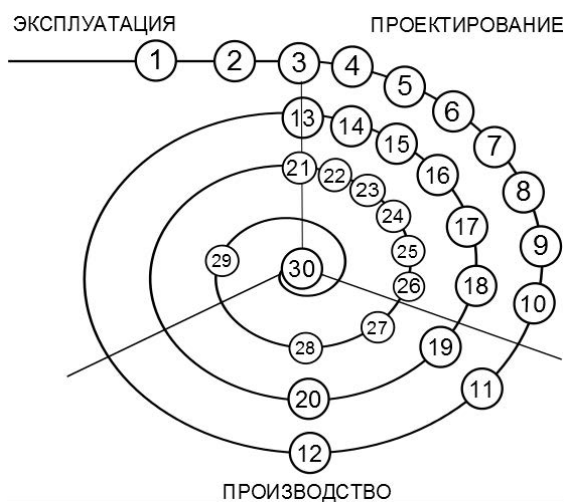


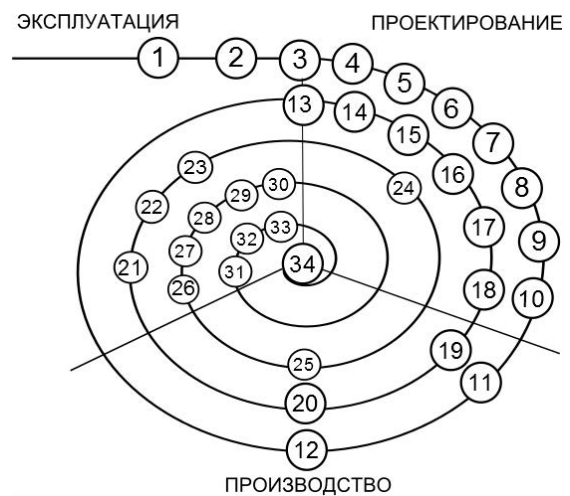
Рисунок 11 – Логарифмическая спираль

У логарифмической спирали (7) расстояние между витками изменяется, что обеспечивает возможность отображения эффектов нелинейности времени при различных вариантах развертывания ЖЦ (рисунок 11).



а

1. Формирование потребности в СТС. 2. Определение числа потенциальных потребителей, объема и стоимости производства. 3. Разработка технического задания (ТЗ). 4-7. Стадия проектирования: 4. Концептуальное проектирование (разработка и защита технического предложения). 5. Эскизное проектирование. 6. Техническое проектирование. 7. Рабочее проектирование. 8-10. Стадия технологической подготовки производства: 8. Разработка комплекта технологической документации. 9. Разработка конструкции технологического оснащения. 10. Разработка организационно-технической документации. 11. Изготовление технологического оснащения. 12. Изготовление опытного образца СТС. 13. Испытание опытного образца СТС. 14, 22. Корректировка технического проекта СТС. 15, 23. Корректировка рабочего проекта СТС. 16, 24. Корректировка документации по технологии изготовления СТС. 17, 25. Корректировка документации по оснастке. 18, 26. Корректировка организационно-технической документации. 19. Изготовление средств дооснащения. 20. Изготовление серийного образца СТС. 21. Испытание серийного образца СТС. 27. Модернизация дооснащения. 28. Изготовление партии заказанной СТС. 29. Использование СТС по назначению. 30. Утилизация СТС.



б

1. Формирование потребности в СТС. 2. Определение числа потенциальных потребителей, объема и стоимости производства. 3. Разработка технического задания (ТЗ). 4-7. Стадия проектирования: 4. Концептуальное проектирование (разработка и защита технического предложения). 5. Эскизное проектирование. 6. Техническое проектирование. 7. Рабочее проектирование. 8-10. Стадия технологической подготовки производства: 8. Разработка комплекта технологической документации. 9. Разработка конструкции технологического оснащения. 10. Разработка организационно-технической документации. 11. Изготовление технологического оснащения. 12. Изготовление опытного образца СТС. 13. Испытание опытного образца СТС. 14. Корректировка технического проекта СТС. 15. Корректировка рабочего проекта СТС. 16. Корректировка документации по технологии изготовления СТС. 17. Корректировка документации по оснастке. 18. Корректировка организационно-технической документации. 19. Изготовление средств дооснащения. 20. Серийное производство. 21. Транспортировка и хранение СТС. 22. Ввод в эксплуатацию. 23. Использование СТС в соответствии со служебным назначением. 24. Корректировка проекта в требуемом объеме. 25. Модернизация СТС. 26. Осмотры, диагностика СТС. 27. Гарантийное техническое обслуживание СТС. 28. Текущий ремонт СТС. 29. Применение СТС в соответствии со служебным назначением. 30. Периодическое ТО. 31. Капитальный ремонт. 32. Послегарантийное ТО. 33. Применение СТС в соответствии со служебным назначением. 34. Утилизация СТС.

Рисунок 12 – Спиральные модели жизненного цикла СТС с акцентом на задачу уменьшения сроков проектирования (а) и задачу обеспечения заданных сроков (увеличения сроков) эксплуатации (б)

В зависимости от целей моделирования степень детализации этапов на отдельных стадиях ЖЦ может различаться. Например, когда решается задача сокращения сроков проектирования, число этапов проектирования в спиральной модели может быть существенно больше, чем число этапов производства и эксплуатации. Напротив, если главное внимание уделяется обеспечению требуемого периода эксплуатации СТС, то в спиральной модели наиболее детально рассматривают этапы технического обслуживания и ремонта.

На рисунках 12 и 13 представлены модели ЖЦ СТС для этих двух случаев, где каждый виток спирали соответствует итерации жизненного цикла.

Заключение

В настоящей работе рассмотрены как наглядные, так и абстрактные модели онтологий жизненного цикла сложной технической системы. Показано, что одним из важнейших аспектов обеспечения эффективности современных систем управления жизненным циклом СТС является управление знаниями, относящимися к ЖЦ, на базе онтологий. Следует отметить, что в отличие от проблематики жизненного цикла онтологий, по которой имеется немало публикаций в зарубежной и отечественной литературе, вопросы построения онтологий ЖЦ как на верхнем, так и на нижнем уровне, все еще остаются недостаточно изученными.

Дальнейшее направление наших работ связано с развитием методик грануляции ЖЦ и разработкой моделей гранулярных онтологий, а также с построением конструктивных интерпретаций параметров спиральных моделей в интересах формализации, расчета и оптимизации ключевых характеристик ЖЦ СТС.

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проекты №11-07-00738-а и №14-07-00846).

Библиографический список

- [Анисов, 2001] Анисов А.М. Свойства времени/ А.М.Анисов // Логические исследования. Вып.8. – М.: Наука, 2001. – С.5-24.
- [Валькман, 1998] Валькман, Ю.Р. Интеллектуальные технологии исследовательского проектирования: формальные системы и семиотические модели. – Киев: Port-Royal, 1998.
- [Волкова, 2004] Волкова, В.Н. Системный анализ и принятие решений: словарь-справочник / В.Н.Волкова, В.Н.Козлов. – М. Высшая школа, 2004.
- [Гаврилова, 2008] Гаврилова, Т.А. Онтологический подход к управлению знаниями при разработке корпоративных информационных систем/ Т.А.Гаврилова // Новости искусственного интеллекта, 2003, №2, с.24-30.
- [Гаврилова, 2008] Гаврилова, Т.А. Визуальные методы работы со знаниями: попытка обзора / Т.А. Гаврилова, Н.А. Гулякина // Искусственный интеллект и принятие решений, 2008, № 1, с.15-21.
- [Еремеев, 2003] Еремеев, А.П. Методы представления временных зависимостей в интеллектуальных системах поддержки принятия решений / А.П.Еремеев, В.В.Троицкий // Известия РАН: Теория и системы управления, 2003, №5, с.75-88.
- [Кашуба, 1998] Кашуба, Л.А. Параллельное проектирование средствами CAE/CAD/CAM в жизненном цикле изделий

машиностроения/ Л.А. Кашуба // Программные продукты и системы, 1998, №3, с.24-30.

[Кандрашина, 1989] Кандрашина, Е.Ю. Представление знаний о времени и пространстве в интеллектуальных системах/ Е.Ю.Кандрашина, Л.В.Литвинцева, Д.А.Поспелов. – М.: Наука, 1989.

[Колчин, 2002] Колчин, А.Ф. Управление жизненным циклом продукции / А.Ф. Стрекалов, М.В. Овсянников, С.В. Сумароков. – М.: Анахарсис, 2002.

[Кудрявцев, 2010] Кудрявцев, Д.В. Системы управления знаниями и применение онтологий. – СПб: Изд-во Политехнического университета, 2010.

[Любинская, 2002] Любинская, Л.Н. Проблема времени в контексте междисциплинарных исследований/ Л.Н.Любинская, С.В.Лепилин. – М.: Прогресс-Традиция, 2002.

[Мартынов, 2001] Мартынов, В.В. Основы семантического кодирования. Опыт представления и преобразования знаний. – Мн.: Европейский гуманитарный университет, 2001.

[Молчанов, 1990] Молчанов, Ю.Б. Проблема времени в современной науке. – М.: Наука, 1990.

[Плесневич, 2003] Плесневич, Г.С. Дедукция в некоторых расширениях интервальной логики Аллена/ Г.С.Плесневич// Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов II-го международного научно-практического семинара (Коломна, 15-17 мая 2003г.) – М.: Физматлит, 2003. – С.83-92.

[Попов, 2001] Попов, Э.В. Корпоративные системы управления знаниями/ Э.В.Попов// Новости искусственного интеллекта, 2001, №1, с.14-25.

[Райзберг, 2007] Райзберг, Б.А. Жизненный цикл товара. Современный экономический словарь. 5-е изд., перераб. и доп. / Б.А.Райзберг, Л.Ш.Лозовский, Е.Б. Стародубцева. – М.: ИНФРА-М, 2007.

[Савелов, 2010] Савелов, А.А. Плоские кривые: систематика, свойства, применения. Справочное руководство. Изд.3. – М.: Либроком, 2010.

[Спицнадель, 2000] Спицнадель, В.Н. Основы системного анализа: Учебное пособие. – СПб.: Изд. дом «Бизнес-пресса», 2000.

[Субетто, 1989] Субетто, А.И. Проблема цикличности развития. – Л.: ВИИ им. А.Ф. Можайского, 1989.

[Тарасов, 1998] Тарасов, В.Б. Концепция метаКИП: от компьютерно-интегрированного производства к Internet/Intranet-сетям предприятий/ В.Б.Тарасов// Программные продукты и системы, 1998, №3, с.19-22.

[Тарасов, 2006] Тарасов, В.Б. Развитие компьютерных средств управления предприятиями: системы четвертого поколения/ В.Б.Тарасов, Д.В.Черемисов// Реинжиниринг бизнес-процессов на основе современных информационных технологий. Системы управления знаниями. Сборник докладов 9-й Российской научно-практической конференции (Москва, 26-27 апреля 2006 г.). – М.: МЭСИ, 2006. – С.399-403.

[Тарасов, 2011] Тарасов, В.Б. Грануляция информации, нестандартные и гибридные нечеткие множества/ В.Б.Тарасов // Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник трудов VI-й Международной научно-практической конференции (ИММВ-2011, Коломна, 16-19 мая 2011 г.). – М.: Физматлит, 2011. – Т.1. – С.35-49.

[Тарасов, 2012] Тарасов, В.Б. Гранулярные, нечеткие и лингвистические онтологии для обеспечения взаимопонимания между когнитивными агентами/В.Б.Тарасов, А.П.Калуцкая, М.Н.Святкина//Открытые семантические технологии проектирования интеллектуальных систем. Материалы II-й международной научно-технической конференции (Минск, БГУИР, 16-18 февраля 2012 г.). – Минск: БГУИР, 2012. – С. 267-278.

[Allen, 1983] Allen, J.F. Maintaining Knowledge about Temporal Intervals/ J.F.Allen// Communications of the ACM, 1983, vol.26, p. 832-843.

[Alting, 2006] Alting, L. Life Cycle Engineering and Management / L.Alting, M.Z.Hauschild, H.Wenzel // Sustainability in Manufacturing: Recovery of Resources in Product and Material Cycles. – Berlin: Springer Verlag, 2006. – P. 31-67.

[Boehm, 1986] Boehm, B. A Spiral Model of Software Development and Enhancement/ B.Boehm// ACM SIGSOFT Software Engineering Notes, 1986, vol.11, №4, p.14-24.

[Bralla, 1996] Bralla, J.G. Design for Excellence. – New York: McGraw-Hill, 1996.

[Euzenat, 2005] Euzenat J. Time Granularity// J.Euzenat, A.Montanari // Handbook of Temporal Reasoning in Artificial Intelligence/ Ed. by M.Fisher, D.Gabbay, L.Vila. – Amsterdam: Elsevier, 2005. – P.59-118.

[Gruber, 1993] Gruber, T.R. A Translation Approach to Portable Ontologies/ T.R.Gruber// Knowledge Acquisition, 1993, vol.5, №2, P.199-220.

[Guarino, 1995] Guarino, N. Formal Ontology, Conceptual Analysis and Knowledge Representation/ N.Guarino// International Journal of Human-Computer Studies, 1995, vol.43, №5-6, p.625-640.

[Kimura, 1996] Kimura, F. Product Life Cycle Modeling for Inverse Manufacturing/ F.Kimura, H.Suzuki // Proceedings of IFIP WG 5.3 International Conference on Life Cycle Modeling for Innovative Products and Processes (PROLAMAT'95, November 29-December 1, 1995)/ Ed. by F.L. Krause, H. Hansen. – Berlin: Springer Verlag, 1996. – P.81-89.

[SAP ERP, 2008] SAP ERP. Построение эффективной системы управления: Пер. с англ. – М.: Альпина бизнес букс, 2008.

[Sowa, 1995] Sowa, J.F. Top-Level Ontological Categories/ J.F.Sowa// International Journal of Human-Computer Studies, 1995, vol.43, №5-6, p.669-685.

[Spur, 1996] Spur, G. Life Cycle Modeling as a Management Challenge/ G.Spur// Proceedings of IFIP WG 5.3 International Conference on Life Cycle Modeling for Innovative Products and Processes (PROLAMAT'95, November 29-December 1, 1995)/ Ed. by F.L. Krause, H. Hansen. – Berlin: Springer Verlag, 1996. – P.3-13.

[Stark, 2011] Stark, J. Product Lifecycle Management: 21st Century Paradigm for Product Realization, 2nd ed. – London: Springer-Verlag, 2011.

[Tarassov, 1996] Tarassov, V.B. Some Theoretical Issues of Lifecycle Engineering/ V.B.Tarassov// Proceedings of IFIP WG 5.3 International Conference on Life Cycle Modeling for Innovative Products and Processes (PROLAMAT'95, November 29-December 1, 1995)/ Ed.by F.L.Krause, H.Hansen. – Berlin: Springer Verlag, 1996. – P.90-92.

[Tarassov, 1994] Tarassov, V.B. Concurrent Engineering and AI Methodologies: Opening New Frontiers/ V.B.Tarassov, L.A. Kashuba, N.A.Cherepanov // Proceedings of the IFIP International Conference on Feature Modeling and Recognition in Advanced CAD/CAM Systems. – Valenciennes, 1994. – Vol.2. – P. 869-888.

[Zadeh, 1997] Zadeh L.A. Toward a Theory of Fuzzy Information Granulation and its Centrality in Human Reasoning and Fuzzy Logic/ L.A.Zadeh// Fuzzy Sets and Systems, 1997, vol.90, p. 111-127.

LIFECYCLE ONTOLOGIES FOR COMPLEX TECHNICAL SYSTEMS

Tarassov V.B. *, Fedotova A.M. *,
Cherepanov N.V. **

**Bauman Moscow State Technical University,
Moscow, Russia*

Vbulbov@yahoo.com
afedotova@acm.org

*** Lavochkin Corporation, Khimki,
Moscow Region, Russia*
nv137@yandex.ru

Some problems of Product Lifecycle Management for a Complex Technical System are faced. An ontological approach to lifecycle analysis and modeling is proposed. A system of both visual and abstract lifecycle models is built. A special attention is paid to the problems of constructing granular ontologies and spiral lifecycle models

Keywords: knowledge representation, knowledge management, meta-model, ontology, meta-ontology, granulation, lifecycle

Introduction

The paper is devoted to the problems of lifecycle analysis and modeling for complex technical systems. The lifecycle includes the stages of design, production and use. A new approach to representing and integrating lifecycle knowledge based on ontological modeling and information granulation techniques is suggested.

Main Part

Some ontological problems of lifecycle management for complex technical systems are considered. By a complex technical system we mean artificial system that includes a great number of heterogeneous elements and subsystems possessing a considerable variety of internal links and sophisticated structure. Such systems have a plurality of possible working states and graduated faulty states (here the fault of some elements decreases the efficiency of the whole system but not necessarily brings about complete systems's breakdown). Typical examples of complex technical systems are modern ships, aircrafts, spaceships, orbital observatories. As a rule such complex systems are an intrinsic part of even more sophisticated socio-technical systems.

Product Lifecycle Management is the process of managing the entire lifecycle of a product from its conception, through design and manufacture, to service, disposal and dismantling. It integrates data, processes, personnel and organizations to provide a product information backbone for modern computer integrated (in particular, virtual and extended) enterprises. Three core aspects of PLM are:

Universal, secure, managed access and use of product definition information;

Maintaining the integrity of product-related information throughout the life of the product or plant;

Managing and maintaining various processes used to create, manage, disseminate, share and apply product-related information.

The concept of lifecycle represents a basic implementation of systemic approach to complex technical objects that consists in visualizing their state changes for a temporal interval. By the end of XXth century-the beginning of XXIst century the notion of lifecycle has become wider. Now it also encompasses the stage of recycling that underlies the idea of lifecycle conversion.

Product Lifecycle Engineering consists in using modern information/ communication infrastructure to design, simulate and integrate various stages of product lifecycle. Its main goal is to construct an overall vision of the product lifecycle by using modern modelling and simulation techniques and tools in the framework of virtual engineering. Nowadays lifecycle engineering is seen as a basis for enterprise integration; it encompasses such aspects as lifecycle simultaneous engineering, lifecycle knowledge (meta-knowledge) engineering and lifecycle uncertainty/vagueness management. Lifecycle knowledge engineering includes both visual representation of lifecycle structure and resource (in particular, time) optimization between lifecycle stages and phases, for instance, the decrease of design time and the increase of exploitation time.

Our ontological approach to lifecycle knowledge engineering supposes the construction of both visual and formal models of lifecycle ontologies. Here formal models are based on Maltsev's concept of algebraic system, whereas visual representations encompass linear, circular and spiral models.

In this paper the main attention is paid to lifecycle ontology viewed as higher order ontology together with granular meta-ontology.

The term meta-ontology means «ontology over ontologies». Meta-ontology provides us with both appropriate mathematical specification of ontology and necessary tools for representing and merging various ontologies. Here granular meta-ontology is of primary concern; it is based on such concepts as granule, level, hierarchy, relations between levels.

Today granule may be seen as a basic object of mereology – the theory of parthood relations: both «part to whole» and «part to part within a whole» relations. Generally, granule can be viewed as a subset of a set. A more restricted definition was suggested by L.A.Zadeh: granule is a collection of objects which are drawn together by indistinguishability, similarity, proximity or functionality. Granules are investigated at a particular level that specifies the degree of abstraction or precision. Different levels focus on different, though related, types of granules. Levels are connected through an order relation. A family of ordered levels forms a hierarchy. Such a hierarchy represents different levels of granularity; by selecting these levels we can obtain different layers of knowledge.

Any granular meta-ontology supposes the consideration of basic notions of the lower ontologies on various abstraction levels. In case of system's lifecycle the basic notions are lifecycle stages and phases. Here lifecycle stages are usually divided into lifecycle phases, where each phase corresponds to a specific system's state. So the stage is viewed as a coarse-grained lifecycle object, whereas the phase is a fine-grained object.

Generally, lifecycle granulation supposes the consideration of such problems as: 1) definition of basic granulation principles and criteria; 2) classification and interpretation of lifecycle granules; 3) analysis of lifecycle granulation approaches and techniques; 4) development of formal granular lifecycle models; 5) construction of mappings between various granularity levels; 6) specification of quantitative parameters of both lifecycle granules and granulation process itself.

It is worth stressing that it does not exist an optimal granulation level; granule sizes are problem-oriented and depend on investigation context. Some lifecycle phases can be considered in a more detailed way and other – less thoroughly, with taking into account modeling objectives. We also envisage lifecycle representations with various abstraction degrees: a) rather simple circular representation based on either partition or covering; b) more sophisticated spiral representations showing interrelations between lifecycle phases, as well as between its phases and stages.

Besides, lifecycle ontology seen as top-level ontology is representable by a mind map. Here such ontology characteristics as its goal, role, language, basic relations and representation form are of special concern.

Basic time theories are mentioned in the context of

lifecycle modeling: substantial and relational, static and dynamic, pointwise and interval time. Our approach is based on relational time model and interval time primitives. We take well-known Allen's temporal logic relations to model the links between lifecycle phases (or lifecycle stages and phases). These are basically two types of relations: consequence and overlapping relations.

Two time metaphors – «time wheel» and «time arrow» – bring about lifecycle circular and consequent models respectively. On the one hand, consequent linear models express such time properties as course, ordering facility, irreversibility. On the other hand, circular time models make emphasis on alternations, reiterations, rhythms, self-sustaining processes. In our paper, we try to reconcile these opposite models by constructing and analyzing spiral lifecycle models.

Let us recall that in polar coordinates each point on a plane is determined by a distance from a fixed point r , $r \geq 0$ and an angle $\varphi \in [-\pi, +\pi]$ from a fixed direction: $M = (r, \varphi)$. A spiral is a curve that winds around a fixed center point at a continuously increasing or decreasing distance from the point. In our paper we consider two spirals – Archimedean spiral and logarithmic spiral. The first one is the locus of points corresponding to the locations over time of a point moving away from a fixed point with a constant speed along a line which rotates with constant angular velocity. It is given by the equation $r = a + b\varphi$, where changing the parameter a will turn the spiral, while b controls the distance between successive turnings. In the context of lifecycle we interpret these spiral parameters in the following way: φ is the time interval, a is the productivity index, b is the level and r is system's state.

The Archimedean spiral has the property that any ray from the origin intersects successive turnings of the spiral in points with a constant separation distance. Hence, such lifecycle features as time acceleration on early phases of lifecycle (for instance, decrease of design time) or time deceleration on later phases (increase of exploitation period) cannot be taken into account by using the Archimedean spiral.

In contrast to this, in a logarithmic spiral these distances, as well as the distances of the intersection points measured from the origin, form a geometric progression.

In the paper we give two spiral representations of system's lifecycle: the first one is oriented to the design for X concept and the second one helps to ensure a required exploitation period by a more detailed consideration of maintenance procedures.

Conclusion

A very important way of enabling the effectiveness of modern PLM systems consists in implementing lifecycle knowledge management on the basis of ontological approach. To differ from the rather popular topic of ontology lifecycle the problems of ontological consideration of complex system's lifecycle are still not sufficiently studied.

Our future work will be related to the analysis of lifecycle granulation techniques and further development of basic indices for granular ontologies.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 001.53

АНАЛИЗ ДИНАМИКИ ИЗМЕНЕНИЙ НАУЧНОЙ ОБЛАСТИ МЕТОДАМИ ОНТОЛОГИЧЕСКОГО ИНЖИНИРИНГА

Лещева И.А.^{*}, Лещев Д.В.^{**}

^{} Санкт-Петербургский Государственный Университет,
г. Санкт-Петербург, Россия
leshcheva@gsom.pu.ru*

*^{**} Центр Перспективных Исследований Санкт-Петербургского Государственного
Политехнического Университета
г. Санкт-Петербург, Россия
dimvovich@gmail.com*

Многие современные предметные области подвержены быстрым изменениям. Не смотря на то, что онтологии являются статическим срезом представлений о предметной области, серия онтологий может быть использована для анализа динамики предметной области. В работе описан общий алгоритм построения серии онтологий, отражающих изменения во времени, для последующего анализа. Алгоритм апробирован на предметной области знаний об углеродных нанокластерах, отраженных в тезисах докладов конференции IWFAC.

Ключевые слова: онтологии; серия онтологий.

Введение

Развитие предметных областей приводит к тому, что онтологии, отражающие знания о них, также изменяются. В настоящее время существует ряд работ, посвященных проблеме изменения онтологий во времени. Их авторы сосредоточены на том, чтобы при внесении изменений/дополнений онтология оставалась непротиворечивой. В таких работах само изменение онтологии является проблемой, которую надо решить, например, с помощью специального инструментария [Загорулько, 2007; Farquhar, 1997; Karp, 1999].

С другой стороны, развивается направление, посвященное разработке временных (темпоральных) онтологий, например, [De Beule, 2004]. Основная идея — сделать онтологии пригодными для моделирования динамики изменения предметных областей.

В данной работе именно изменения онтологии во времени (динамика) являются предметом анализа, но без прямого построения темпоральных функций. С помощью подобного изучения динамики онтологий можно отслеживать развитие знаний в любой предметной области на базе статического онтологического подхода.

1. Процесс построения онтологий

1.1. Общий алгоритм

Общий алгоритм, позволяющий построить онтологию для будущего анализа их динамики можно разбить на три шага.

Шаг 1. Сначала строится базовая онтология, покрывающая все основные концепты предметной области [Gavrilova, 2007]. Если необходимо, процесс построения базовой онтологии может быть итеративной.

Шаг 2. Полученная на предыдущем шаге онтология расширяется и уточняется экспертами с учетом изменений знаний во времени. В результате нескольких итераций получается всеобъемлющая онтология, охватывающая все более-менее существенные концепты предметной области. Данная онтология может служить классификатором или рубрикатором для всех возможных тем в данной области знаний.

Шаг 3. Строятся проекции общей онтологии на некоторые заранее определенные моменты времени. По факту, все устаревшие или еще не появившиеся концепты-листья и даже целые ветви исключаются. Также на данном этапе возможно сопоставление каждому концепту некоторого количественного

показателя, например, число публикаций по данному вопросу. Именно этот показатель и становится в будущем основой анализа темпоральных изменений.

1.2. Применение алгоритма

В качестве примера предметной области для апробации алгоритма была взята конференция «Фуллерены и атомные кластеры» ("Fullerenes and Atomic Clusters", IWFAC), которая в 2011 году была преобразована в конференцию «Продвинутые углеродные наноструктуры» (International Conference "Advanced Carbon Nanostructures" (ACN'2011)) [ACN, 2013] в связи с сокращением числа выступлений, связанных с неуглеродными кластерами. Конференции являются удобным средством для отражения концептов определенной небольшой предметной области: они меньше подвержены отбору в отличие от статей в научных журналах и позволяют легко регулировать глубину строящейся онтологии.

Онтологии строились на материале сборников тезисов конференции (биеннале с 1993 по 2009, всего 9 конференций, 300-400 докладов на каждой конференции). Применительно к данной задаче алгоритм упростился:

Шаг 1. Была построена всеобъемлющая онтология, покрывающая тематику докладов самой «богатой» по темам конференции IWFAC'07. Верхние уровни ветви «Углеродные материалы» приведены на рисунке 1. Реальная онтология весьма велика, и помещать ее целиком в статье не имеет смысла. Самым верхним уровнем концептов стали «Углеродные материалы» и «Неуглеродные материалы»

Шаг 2. Полученная на предыдущем шаге онтология была расширена экспертом за счет потенциально-возможных областей исследования, не освещенных в докладах конференции 2007 года. Таким образом полученная на 2 шаге онтология послужила рубрикаторм для тем докладов.

Шаг 3. Доклады каждого года были распределены по темам в соответствии с полученным на 2 шаге рубрикаторм. Для всех узлов каждой из 9 полученных онтологий, помимо названий концептов, были рассчитаны веса, показывающие, сколько докладов было сделано по соответствующей теме.

Всего было выделено 8 уровней иерархии. Второй уровень иерархии определялся наличием трансляционной симметрии материала: кристаллы, кластеры (молекулы), полимеры и аморфные вещества. Следующее деление зависело от развитости концепта второго уровня. Например, кластеры были разделены на фуллерены, нанотрубки, наноалмазы, онионы и остальные. Четвертый уровень — это уровень модификации

чисто углеродной основы: немодифицированные, производные, комплексы, эндоэдральные и допированные вещества. Следующий уровень определял вид модификации. И последний уровень иерархии содержал собственно темы тезисов. Если тезисы могли быть отнесены к нескольким разным первичным концептам, то доклад становился листом более общего родительского концепта. Таким образом, у каждого концепта появился вес, равный количеству докладов, отнесенных к данному концепту.

При классификации каждого доклада учитывалась только ключевая идея тезисов. Построенная онтология достаточно субъективна, и тезисы можно классифицировать и другими, отличными от предложенного образцами. Это отличает данный способ построения онтологии/таксономии от машинных способов классификации, которые используют вхождения всех употребляемых слов формально. Например, в большинстве докладов во вводной части говорится о фуллеренах, как о символе новых форм углерода, хотя сама работа может быть посвящена углеродным нанотрубкам, полимерам или вообще неуглеродным материалам.

2. Анализ динамики предметной области

Построенная последовательность онтологий была проанализирована с точки зрения изменений во времени.

Типичный сценарий развития темы состоит из 3 основных стадий: рождения, роста и угасания. Для примера на рисунке 2 приведена диаграмма числа докладов по теме «Органические производные фуллеренов» и ее подтемам за 10 лет. Для удобства визуализации использована логарифмическая цветовая шкала. Из рисунка, например, видно, что максимум исследований галогенсодержащих органических производных фуллеренов приходится на 2007 г.

Если рассматривать две связанные темы «Галогенсодержащие органические производные фуллеренов» и «Галогениды фуллеренов», то видно, что рост первой темы сопровождается уменьшением числа докладов по второй. Можно предположить, что исследователи заканчивают работу над исчерпавшей себя темой и переходят в близкую. Подобный феномен наблюдается во многих парах.

На рисунке 1 можно также заметить, что концепты зрелых областей исследования содержат 3 дочерние ветви: синтез, свойства и применения. В молодых областях такого разделения нельзя провести. Согласно этому выводу, к зрелым областям можно отнести только две темы: немодифицированные фуллерены и немодифицированные нанотрубки.

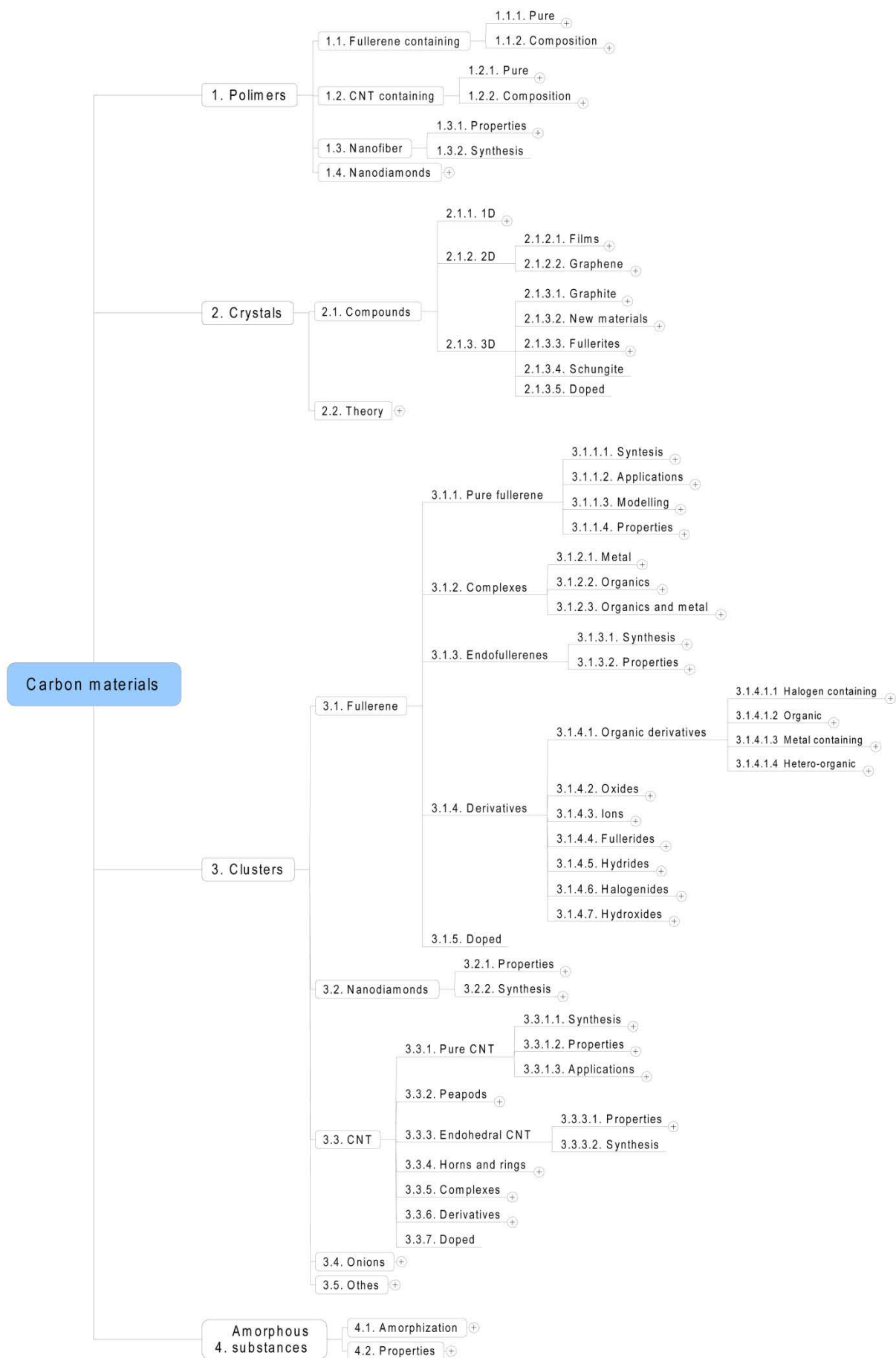


Рисунок 1 — Верхние уровни ветви «Углеродные материалы» общей онтологии

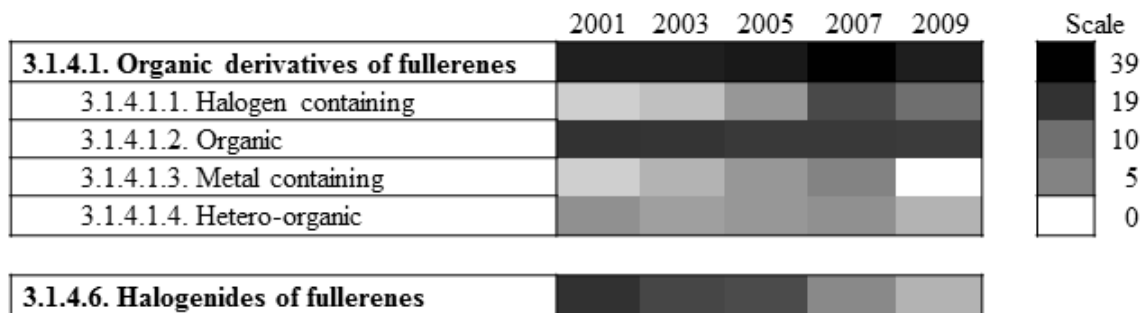


Рисунок 2.— Эволюция популярности двух тем

Заключение

В работе предложен подход использования онтологий для анализа динамики предметных областей, формализованный в виде общего алгоритма. Алгоритм апробирован на предметной области, охватывающей темы докладов одной из конференций, посвященной углеродным нанокластерам.

Были отмечены следующие закономерности:

- Типичный сценарий развития темы состоит из 3 основных стадий: рождения, роста и угасания.
- «Молодые» темы являются листьями онтологии, в то время как развитые области содержат разветвленную структуру.
- Некоторые темы связаны между собой: потеря интереса к одной из областей соответствует росту исследований в другой. Можно предположить, что развитие второй темы сопровождается оттоком в нее исследователей из области.

Таким образом, в результате анализа изменения онтологий во времени можно проследить эволюцию отдельных тем, выявить тренды развития, оценить перспективность того или иного направления для исследований или инвестиций и даже предсказать, что будет происходить в ближайшем будущем.

Библиографический список

- [Загоруйко, 2007] Загоруйко Ю.А., Боровикова О.И. Технология построения онтологий для порталов научных знаний/ Вестник Новосибирского государственного университета. Серия: Информационные технологии, 2007, Т.5, № 5, сс.42-52.
- [ACN, 2013] 11th International Conference Advanced Carbon NanoStructures Home Page <http://www.ioffe.ru/ACNS/index.html> (дата обращения: 30.11.13).
- [De Beule, 2004] De Beule, J. Creating Temporal Categories for an Ontology of Time. In Rineke Verbrugge and Niels Taatgen and Lambert Schomaker, editors, BNAIC-04 [Электронный ресурс] / De Beule, J. Creating. — P. 107-114. — Режим доступа: <http://arti.vub.ac.be/~joachim/Bnaic04.pdf>
- [Farquhar, 1997] Farquhar A., Fikes R., Rice J. The Ontolingua Server: a tool for collaborative ontology construction/ International Journal of Human-Computer Studies, 1997, V. 46, I.6, pp. 707-727.
- [Gavrilova, 2007] Gavrilova, T., Ontological Engineering for Practical Knowledge Work, Lecture Notes in Artificial Intelligence, 2007, 4693, pp. 1154-1162.
- [Karp, 1999] Karp P., Chaudhri V., Paley S. A Collaborative Environment for Authoring Large Knowledge Bases/ Journal of Intelligent Information Systems. 1999, V.13, I.3, pp.155-194.

ANALYSIS OF TRENDS IN SCIENTIFIC DOMAINS BY ONTOLOGICAL ENGINEERING

Leshcheva I.A. *, Leshchev D.V. **

* Graduate School of Management, St. Petersburg University, St.Petersburg, Russia

leshcheva@gsom.pu.ru

**Center for Advanced Studies, St. Petersburg State Polytechnic University, St. Petersburg, Russia

dimvovich@gmail.com

In this work a general algorithm for constructing a scientific domain ontologies to analyze the scientific domain dynamics is proposed. The algorithm is tested on an analysis of conference abstracts for 20 years.

Introduction

The subject of this work is applying ontology engineering approach to analyzing of trends in scientific domains. By constructing series of static ontologies at different moments of time the dynamics of knowledge in a subject area can be traced.

Creating of ontologies series

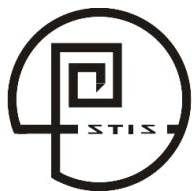
A three steps algorithm is proposed. On the first step, a basic ontology covering all main concepts of the domain is constructed. On the second step, the obtained ontology is expanded by experts. These steps can be iterative. On the third step, a projection of the received ontology on a particular point of time is built.

Domain trends analysis

The described algorithm was used to analyze dynamic changes in carbon nanoclusters domain using abstracts of International Workshop "Fullerenes and Atomic Clusters" (IWFAC) [ACN, 2013].

Conclusion

The proposed approach can be used to define points of domains growth and stagnation, relationship between subdomains and there maturity.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8.032.26

ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ В ДИАГНОСТИКЕ ИШЕМИЧЕСКИХ АТАК

Головко В.А.^{*}, Войцехович Г.Ю.^{*}, Апанель Е.Н.^{**}, Мастыкин А.С.^{**}

^{} Брестский государственный технический университет,
г. Брест, Республика Беларусь*

gva@bstu.by

napster@tut.by

*^{**} Научно-практический центр неврологии и нейрохирургии
г. Минск, Республика Беларусь*

apanel@rambler.ru

masmas2014@yandex.by

Приведено описание нейросетевой системы для прогнозной диагностики транзиторных ишемических атак (ТИА). Предлагаемый подход основан на интегрировании нелинейного метода главных компонент (NPCA) и многослойного персептрона (MLP). При совмещении двух различных по функциональному назначению нейронных сетей (NPCA и MLP) можно получить эффективное прогнозирование возможного возникновения транзиторной ишемической атаки.

Ключевые слова: нейронные сети; транзиторные ишемические атаки; многослойный персептрон; система диагностики.

Введение

В настоящее время все больше возрастает тенденция проектирования систем искусственного интеллекта на основе нейронных сетей, искусственных иммунных систем, эволюционного программирования и других, биологически инспирированных подходов. Использование методов искусственного интеллекта широко применяется и в медицине. Каждый год в журналах появляется более 500 академических публикаций по применению искусственных нейронных сетей в медицинских приложениях [Paulo, 2006]. Согласно публикуемой литературе, искусственные нейронные сети являются эффективным средством для автоматической диагностики заболевания и помогают врачу принять адекватное решение. Медицинская диагностика является сложной задачей, которая зависит от квалификации доктора и поэтому во многих случаях очень субъективна.

Сложность также состоит в том, что заболевание, как правило, определяется большим числом факторов и симптомов. К сожалению, медицинские работники очень часто не обладают достаточным опытом в диагностике многих

заболеваний. Недостаточная квалификация медицинских работников увеличивает смертность среди пациентов. Так, например, диагностика транзиторных ишемических атак (ТИА) врачами первой помощи была правильной лишь в 30% случаев. В то время как автоматическая система поддержки принятия решений в медицинской диагностике позволяет повысить качество диагностирования и избавить врача от рутинной работы.

Для рассматриваемого класса заболеваний большое значение имеет оперативное обнаружение, позволяющее предотвратить катастрофическое развитие ситуации и помогающее доктору поставить адекватный диагноз. Задачи оперативного обнаружения транзиторных ишемических атак являются трудноформализуемыми, для них невозможно или очень сложно формализовать процесс принятия решений. Основным подходом в медицинских учреждениях для диагностики таких заболеваний является нейропсихологическое тестирование, которое зависит от квалификации врача и очень часто приводит к неадекватному диагнозу.

Поэтому эффективным средством для

автоматической диагностики таких заболеваний являются нейросетевые методы для обработки биомедицинских данных. Методы, основанные на искусственных нейронных сетях, являются подходящим средством для решения трудноформализуемых задач и, как будет показано ниже, позволяют повысить качество диагностирования и, соответственно, помочь врачу поставить адекватный диагноз.

Существуют различные методы диагностики ТИА: нейропсихологическое тестирование (neuropsychological testing), статистические подходы, методы искусственного интеллекта [Easton, 2009]. Основным подходом в медицинских учреждениях для первичной диагностики транзиторных ишемических атак (ТИА) является визуальная инспекция и нейропсихологическое тестирование, которые зависят от квалификации врача и очень часто приводят к неадекватному диагнозу. Так, например, диагностика транзиторных ишемических атак (ТИА) врачами первой помощи является корректной лишь в 30% случаев [McNeill, 2008]. Параметрами для диагностики являются тестирование речи (внезапное затруднение понимания речи), соматической чувствительности, вегетативные нарушения (внезапное появление односторонней слабости мускулатуры лица или конечностей) и т.д. Основным недостатком такого подхода является зависимость от квалификации врача и низкая точность первичной диагностики. Статистические подходы требуют наличия больших баз данных, что нередко является большим препятствием на пути создания системы диагностики. К методам искусственного интеллекта относятся нейронные сети, генетический алгоритм, нечеткая логика или их комбинации. Эти технологии характеризуются высокой точностью и менее требовательны к размеру базы данных в сравнении со статистическими алгоритмами. По этой причине методы искусственного интеллекта могут быть подходящим инструментом для диагностики и классификации подтипов ТИА. В данной статье для диагностики ТИА предлагается нейросетевая модель. Нейросетевые методы используются для уменьшения времени диагностики и числа ошибочных диагнозов, а также для помощи доктору в принятии решения. Эффективность предлагаемой нейросетевой модели в диагностике транзиторных ишемических атак иллюстрируется результатами экспериментов.

1. Существующие подходы

В настоящее время существует множество подходов для предварительной диагностики ТИА.

Один из статистических подходов для диагностики транзиторных ишемических атак описан в [Dawson, 2009]. Авторы применили многомерную логистическую регрессию, используя ROC (receiver operating characteristic curves) анализ, для создания клинической системы диагностирования ТИА. Для этого была

использована база данных из West Glasgow Stroke Registry, которая содержала информацию приблизительно о 225 000 пациентов. После регрессионного анализа были выделены 9 переменных с соответствующими коэффициентами регрессии: возраст, головные боли, двоение в глазах, обмороки, приступы, речевые нарушения, односторонняя слабость в конечностях, изменения на лице, история ТИА. Для вычисления критерия ТИА значения всех коэффициенты регрессии суммируются. При использовании ROC кривых ТИА идентифицировалась, если значение критерия превышало 6.1. Такая система правильно идентифицировала 85% пациентов с цереброваскулярными отклонениями и 54% с нецереброваскулярными. Основным недостатком такого подхода является необходимость иметь большую базу данных.

Нейронные сети в задаче диагностики ишемических заболеваний описываются в [Barnes, 2006]. Предлагаемые модели были разработаны для быстрой диагностики ТИА в соответствии со следующими выходными значениями: нормальное состояние, ТИА в левой сонной артерии, правой сонной артерии и др. Исходными данными являлись анкеты пациентов, в 6 секциях которых содержалась следующая информация: потеря или изменение речи, потеря зрения, зрительные двоения, нечувствительность или дрожь, паралич или слабость, головокружения или потеря равновесия. Каждый входной образ состоял из цифр 0 или 1 из анкеты. Для каждой секции анкеты использовался многослойный персептрон. Так, например, сеть для зрительных двоений состояла из 4 входных и 5 выходных нейронов. Три выхода соответствовали случаю “нет события” и оставшиеся 2 – удар (stroke) или ТИА.

В [Barnes, 2006] описывается пример применения генетического алгоритма для отбора признаков-предикторов с наиболее выраженным дифференциально-диагностическим потенциалом и последующей классификацией с помощью многослойного персептрона для прогнозирования тромбоэмболических инсультов. Архитектура нейронной сети состоит из 20 входных нейронов, 10 скрытых и 10 выходных нейронов. В качестве входных данных были использованы следующие параметры: гипертония, диабет, миокардит, холестерин в крови и др. Выходные нейроны отображают различные категории заболеваний: ТИА, левая гемиплегия, афазия, правая гемиплегия, дисфазия и квадриплегия. Точность прогнозирования составляет 78.52% на обучающей выборке и 90.61% – на тестовом множестве.

Рассмотренные выше подходы отличаются друг от друга используемыми входными и выходными данными, а также базами данных пациентов. Поэтому трудно сравнивать эффективность различных подходов.

В следующих разделах рассматривается нейросетевая диагностическая система для

предварительной диагностики ТИА. Эффективность предлагаемой нейросетевой системы для диагностики транзиторных ишемических атак иллюстрируется результатами экспериментов.

2. Исходные данные для прогнозной диагностики

В исследование были приняты следующие диагностические параметры (признаки-предикторы): возраст; пол; место жительства; смена места жительства за послед. 10 лет; аускультация сердца; метеозависимость; образование; профессия; особенности ночного сна; артериальная гипертензия; боли в области сердца; нарушения сердечного ритма; хрон. гастрит, язв. болезнь желудка; остеохондроз шейный; курение (возраст); снижение остроты зрения (время появления); головные боли; головные боли (время появления); головокружение; конфликты по работе и с администрацией; бессонница; наследствен. по патологии сосудов мозга; наследственность по другим заболеваниям; диастолическое давление; границы сердца; изменения на ЭКГ; хронический бронхит; хронический гепатохолецистит; почечнокаменная болезнь; употребление алкоголя; курение (количество); работоспособность; раздражительность повышенная; снижение памяти (степень); снижение памяти (время появления); снижение остроты зрения (степень); зрительные нарушения; смена профессии за последние 10 лет.

Здесь еще необходимо учитывать оптимальное необходимое и достаточное количество диагностических признаков для того, чтобы их излишнее количество не было бы слишком обременительной нагрузкой на врача в процессе обследования пациента.

Дифференциация прогнозных диагнозов осуществляется по четырем классам выходного вектора: атеротромботический подтип ТИА, кардиоэмболический, гипертензивный, прочие.

3. Нейросетевая диагностическая система

Рассмотрим нейросетевую систему распознавания транзиторных ишемических атак. Данная система основывается на двух различных нейронных сетях. В качестве входного вектора выступают описанные выше 38 параметров, содержащие данные о пациенте. Выходными данными нейросетевой системы является 4-х размерный вектор, где 3 выхода соответствуют подтипам ТИА и один – состоянию без ТИА. Обработка входных данных осуществляется в 2 этапа. Первый этап соответствует выделению наиболее информативных признаков (feature selection). Один из наиболее важных вопросов, касающихся входных данных, является следующий: какие входные параметры действительно полезны и оказывают наибольшее влияние на результат диагностики? Часто, в аналогичных системах для

выделения наиболее информативных признаков используется генетический алгоритм. В данной работе для выделения наиболее значимой информации из исходных данных и уменьшения размерности входных данных предлагается использовать нейронную сеть нелинейного анализа главных компонент NPCA (Nonlinear Principal Component Analysis), с помощью которой входные 38-размерные векторы для каждого пациента преобразуются в 12-размерные выходные векторы. Они являются уже преобразованной, очищенной от «информационного шума», входной диагностической информацией для вычисления вероятностей по 4-м классам распознавания

На втором этапе обработки данных производится классификация ТИА. Сжатые на предыдущем шаге данные теперь уже содержат только необходимую информацию, взятую из исходного входного множества. По этой очищенной диагностической информации многослойным персептроном (MLP) и осуществляется вычисление вероятностей классов распознавания, т.е. прогнозных диагнозов, дифференцированных по подтипам, и класса ПРОЧИЕ. Выходной слой персептрона состоит из 4 нейронов: 3 для выдачи результатов вычислений по каждому подтипу ТИА, и один для состояния ПРОЧИЕ.

4. Результаты экспериментальных исследований

Рассмотрим отображение входного пространства образов на плоскость двух первых главных компонент. На рис. 1 графически представлена классификационная картина разделения клинических случаев по подтипам ТИА с применением нелинейной NPCA. Из рисунка видно, что данные, принадлежащие разным типам атак, распределены по разным областям.

Отображение входного пространства образов для нормального состояния и транзиторных ишемических атак в трехмерном пространстве показано на рис. 2.

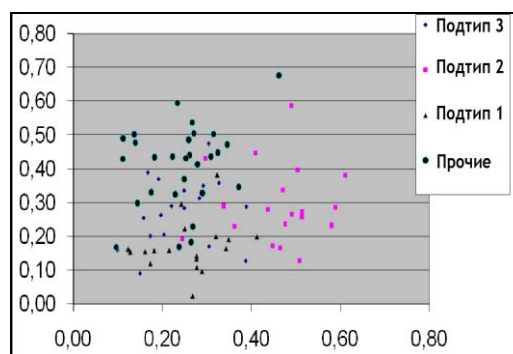


Рисунок 1 - Данные обработанные нелинейной NPCA

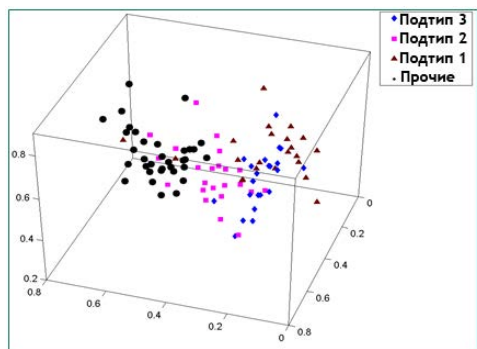


Рисунок 2 – Отображение входного пространства образов на 3 главные компоненты

На этих рисунках четко обозначена тенденция всех отдельных индивидуальных клинических случаев объединяться в кластеры по этиопатогенетическому принципу «ближайшего соседства».

Все пространство исходных образов разделено на 2 группы: обучающую и тестовую выборки. Точность распознавания составляет до 100% на обучающей выборке и 78% на тестовых образах (табл. 1).

Таблица 1. Точность распознавания

Число образов в обучающей выборке	Число образов в тестовой выборке	Точность распознавания на обучающей выборке	Точность распознавания на тестовой выборке
90	24	100%	78%

Заключение

В работе рассмотрен подход к построению системы классификации подтипов транзиторных ишемических атак, который базируется на нейросетевых технологиях. В основу положена идея представления входного пространства образов в виде главных компонент, для повышения качества обработки и скорости функционирования программы. Был предложен подход, основанный на интеграции нелинейной РНС и многослойного персептрона. В целом, результаты показали, что возможности и потенциал систем, основанных на связке рециркуляционной сети и многослойного персептрона, достаточно велики. Поэтому, продолжая работу в этом направлении, можно ожидать улучшения основных показателей функционирования систем такого типа.

Главное преимущество использования нейросетевого подхода заключается в возможности распознавать новые ТИА, неиспользуемые в обучающей выборке, а также скорость работы.

Наиболее трудные моменты, которые выявились в ходе решения поставленной задачи:

- нечеткость входных множеств, т.е. отсутствие выраженных границ между

рассматриваемыми классами атак;

- ограниченность исходных данных. С накоплением данных о новых пациентах, вероятно, процент правильного распознавания будет расти.

В любом случае цель разработанной системы диагностики – это предварительный анализ состояния пациента, базирующийся на легкодоступных данных, получение которых не требует дорогих и времязатратных исследований. Система позволяет определить наиболее вероятный вектор развития патологии в кратчайшие сроки.

Таким образом, полученные прогнозно-диагностические данные дают возможность ориентироваться в упорядоченной симптоматике пациента уже при первом обследовании, что значительно уменьшает временной интервал между началом обследования пациента и назначением целевых этиотропных лечебно-профилактических мероприятий.

Библиографический список

- [Paulo, 2006] Paulo J. Lisboa, Azzam Taktak. The use of artificial neural networks in decision support in cancer: A systematic review, Neural networks 19 (2006), p. 408-415.
- [Easton, 2009] J. Easton et al. Definition and evaluation of transient ischemic attack. Stroke (Journal of the American Heart associations), 2009.
- [McNeill, 2008] A. McNeill. How Accurate Are Primary Care Referral Letters For Presumed Acute Stroke? Scottish Medical Journal, V. 53, № 4, 2008. p. 11-12.
- [Dawson, 2009] J. Dawson et al. A recognition tool for transient ischemic attack, O J Med, 192, p. 43-49, 2009.
- [Barnes, 2006] Barnes R., Toole J., Nelson J., Howard V, Neural networks for ischemic stroke, Journal of stroke and cerebrovascular diseases, Vol. 15, No. 5, p. 223-227, 2006.
- [Barnes, 2006] D. Shanthi, Dr.G. Sahoo, Dr.N. Saravanan. Input feature selection using hybrid neuro-genetic approach in the diagnosis of stroke disease, IJCSNS International Journal of Computer Science and Network Security, Vol.8 No.12, December 2008, p.99-107.

NEURAL NETWORKS USAGE IN ISCHEMIC ATTACKS DIAGNOSTICS

Golovko V.A. *, Vaitsekhovich H.Y. *, Apanel E.N. **, Mastyskin A.S. **

*Brest State Technical University, Brest, Republic of Belarus

gva@bstu.by

napster@tut.by

** Scientific and Practical Center of Neurology and Neurosurgery, Minsk, Republic of Belarus

apanel@rambler.ru

masmas2014@yandex.by

In this work the neural network model for transient ischemic attacks (TIA) recognition is described. The proposed approach is based on integration of nonlinear principal component analysis (NPCA) neural network and multilayer perceptron (MLP). Combination of two different functional neural networks (NPCA and MLP) can be an effective way of transient ischemic attack diagnostics.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ОСОБЕННОСТИ ПОСТРОЕНИЯ ИНФОРМАЦИОННЫХ ПРОСТРАНСТВ ПРОМЫШЛЕННЫХ ПРЕДПРИЯТИЙ НА ОСНОВЕ СВЯЗАННЫХ ДАННЫХ

Галушка И.Н.^{*}, Щербак С.С.^{*}

^{}Кременчугский национальный университет имени Михаила Остроградского,
г. Кременчуг, Украина*

ilona.galushka@ya.ru

sergey.shcherbak@gmail.com

В работе рассмотрены вопросы построения информационных пространств, как интеграционного процесса объединения источников данных информационных систем промышленных предприятий для формирования единого интерфейса доступа на основе объектно-ориентированного подхода. Разработана формальная модель информационного пространства и предложены соответствующие отображения в качестве интеграционных компонентов существующих источников в информационное пространство

Ключевые слова: интеграция; связанные данные; паттерны реализации; информационное пространство

Введение

В современных условиях управление данными на промышленных предприятиях (ПП) характеризуется наличием большого количества распределенных гетерогенных источников данных (ИД) зачастую не связанных едиными механизмами управления. Причиной накопления информации в таких источниках является отсутствие реализации политики автоматизации ПП по единому принципу, что приводит к тому, что на одном ПП могут функционировать несколько локальных информационных систем (ИС). Интеграция подобных источников данных является ключевым фактором для их объединения в единое информационное пространство (ИП) связанных данных предприятия (англ. Linked Enterprise Data, LED), что существенно упрощает доступ к данным, которые необходимы для лиц, принимающих решения на ПП. В рамках такого подхода информационное пространство будем рассматривать как некий промежуточный интеграционный компонент с универсальным коммуникационным интерфейсом (рис.1).

В последнее время в качестве эффективного средства коммуникации с низкой стоимостью развертывания используется сеть Интернет и гипертекстовые технологии Всемирной Паутины (WWW). Документно-ориентированная природа WWW и отсутствие эффективных средств описания содержащихся в документах данных со временем привела к развитию новых направлений и

концепций, которые могут стать эффективной платформой для информационных систем промышленных предприятий.

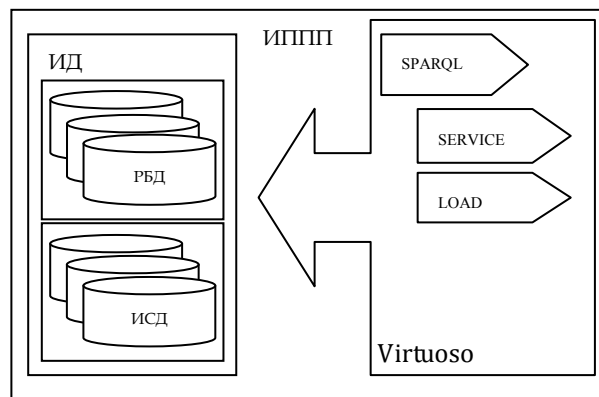


Рисунок 1 — информационное пространство на основе различных типов источников данных

Одной из таких является концепция связанных данных. В рамках этой концепции каждому объекту реального мира или объекту предметной области, автоматизация которой производится, ставится в соответствие его виртуальный аналог, а объектные связи выражаются как набор отношений между объектами, каждый из которых связан с другими с помощью универсального идентификатора ресурсов URI или его подкласса универсального локатора ресурсов URL. В рамках такого подхода к моделированию предметных областей данные представляются как открытая структура данных, модификация которой может быть осуществлена без

существенного изменения программных средств их обработки, и доступ к которой может быть при необходимости ограничен. Кроме того, концепция связанных данных обеспечивает эффективную платформу для создания распределенных информационных систем, основанную на технологиях искусственного интеллекта и Семантического Веба (англ. Semantic Web), причем ориентация этой концепции на распределенное хранение данных является на сегодня одним из лучших решений для организации как межсистемного, так и внутрисистемного взаимодействия распределенных информационных систем, что в условиях территориальной распределенности подразделений предприятий является одним из лучших решений.

При разработке ИС на основе связанных данных, в частности на основе RDF, некоторые трудности обусловлены тем, что связанные данные предназначены для распределенного хранения и описания данных и их схем, но не предоставляют возможности описания методов обработки данных (операционного аспекта), что в условиях применения на промышленных предприятиях требует создания новых моделей представления данных для учета операционной составляющей процессов.

Промышленные предприятия в условиях постоянного повышения скорости ведения бизнеса и недостаточного финансирования нуждаются в средствах, позволяющих повышать эффективность оперативного управления производственным предприятием на основе развитых технологий поддержки принятия решений, поэтому актуальным и целесообразным является создание информационного пространства производственного предприятия на основе связанных данных, которое могло бы повысить связность компонент ИС производственного предприятия при сохранении их территориальной распределенности и дало бы возможности модификации и добавления новой функциональности без перепрограммирования модулей обработки данных. Одной из таких модификаций является необходимость постоянного повышения эффективности процессов поиска распределенных между компонентами ИС данных в связи с постоянным ростом количества данных, хранящихся на производственных серверах, и требованием хранения разноаспектных характеристик некоторых видов данных, применяемых, например, при принятии решений уполномоченными лицами.

Информационное обеспечение производственных предприятий, построенное в условиях лоскутной автоматизации, строится на основе стихийной архитектуры и характеризуются использованием ИС контрольно-учетного характера, отсутствием единой формальной системы процесса выработки управленческих решений, отсутствием непосредственного доступа ЛПР к информационным ресурсам и т.д., что необходимо

обеспечить для эффективного решения задач как поддержки принятия решений, так и задач оперативного управления в целом.

Формальная модель информационного пространства промышленного предприятия

Для разработки модели информационного пространства уточним определения основных понятий, используемых в работе.

Определение 1. Связанные данные – это структурированные данные вида «поименованный граф, субъект, предикат, объект», каждый компонент которых для идентификации использует унифицированный идентификатор ресурсов URI (англ. Uniform Resource Identifier).

Замечание 1. Учитывая необходимость использования информации о местонахождении данных в работе, используется подкласс URI универсальный локатор ресурсов URL (англ. Uniform Resource Locator), обеспечивающий помимо идентификации ресурса представление информации о местонахождении ресурса.

Определение 2. Информационное пространство промышленного предприятия – это совокупность распределенных источников данных, соответствующих информационным системам предприятия, средств их сопровождения и использования, функционирующих на основе общих принципов.

Замечание 2. В связи с тем, что информационное пространство промышленного предприятия предложено создавать в соответствии с положениями концепции связанных данных, классы источников данных, используемых в работе, ограничим классом источников связанных данных, что с учетом развитых средств интеграции не сужает общность реализуемых в работе идей.

Сохраняя общность с концепцией связанных данных, определим информационное пространство промышленного предприятия согласно определению (2) как совокупность множеств источников связанных данных Src и присоединенных процедур Pr^U , которые реализуют функциональность информационного пространства ПП:

$$I = \langle Src, Pr^U \rangle, \quad (1)$$

Источники связанных данных Src представим как множество четырехкомпонентных структур – квадов (англ. Quad), которые состоят из поименованных графов (G), субъектов (S), предикатов (P) и объектов (O), причем каждый компонент квада с учетом распределенности их хранения должен быть представлен в виде URL:

$$Src = \langle G, S, P, O \rangle, \quad (2)$$

Квады в источниках связанных данных

группируются в контексты $G^c \subseteq G$ – множества квадов, сгруппированных по G . Каждому контексту g с учетом формулы (2) поставим в соответствие триплет $\langle S, P, O \rangle$:

$$\forall g \in G^c : g = \langle S, P, O \rangle, \quad (3)$$

где g – контекст источника связанных данных, G^c – множество всех контекстов, S – множество субъектов, P – множество предикатов, O – множество объектов.

Далее, если это не будет приводить к противоречию, «контекст» и «поименованный граф» будем использовать как синонимы.

С учетом необходимости хранения не только связанных данных в источниках, но и их схем, множество G^c разделим на два подмножества G^S и G^I , причем так, что $G^S \cap G^I = \emptyset$, тогда множество триплетов $\langle S, P, O \rangle$, принадлежащих G^S будут определять схему связанных данных, а G^I собственно сами связанные данные.

Далее опишем связанные данные в терминах, понятных пользователям ИС, а именно в терминах объектно-ориентированного проектирования.

Пусть $G_t^S \subset G^c$ – структурно-логическая схема t -го источника связанных данных (рис. 2.), тогда определим ее в терминах объектно-ориентированного проектирования (класс, подкласс, экземпляр класса, отношения между классами):

$$G_t^S = \langle C, R, F \rangle, \quad (4)$$

где G_t^S – схема связанных данных, C – множество классов, причем $C \subset G_t^S$, R – множество отношений между классами ($R \subset G_t^S$), F – множество свойств классов ($F \subset G_t^S$).

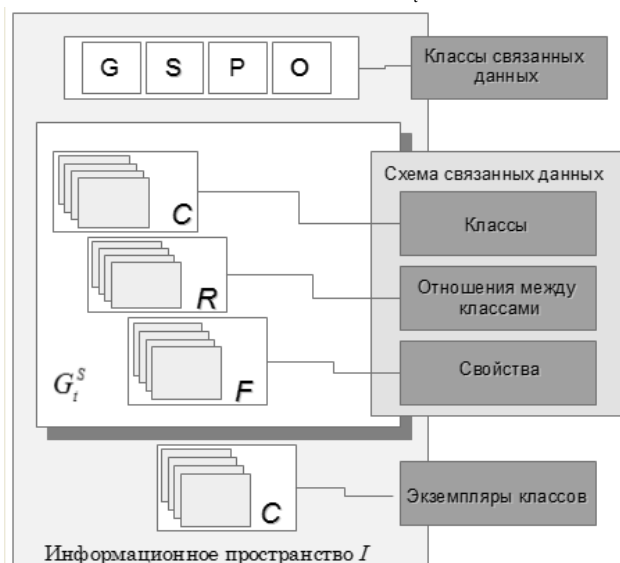


Рисунок 2 – Схема источника связанных данных

Множество свойств классов представим следующим образом:

$$F = \langle T, L, V \rangle, \quad (5)$$

где T – тип данных, L – язык представления, V – значение свойства.

Учитывая, что язык представления L необходим при определении только значений символического или строкового типов свойств экземпляров класса, для всех остальных установим нулевое значение L :

$$P = \langle T, \emptyset, V \rangle, \quad (6)$$

Для определения структуры классов значение свойств не имеют смысла, поэтому $V = \emptyset$, а выражение (6) будет иметь вид:

$$P = \langle T, \emptyset, \emptyset \rangle, \quad (7)$$

Определим множество доступных отношений между классами R для организации связей «быть подклассом» и «быть экземпляром класса» и др., формально определенных следующим образом:

$$R = \{isSubclass, instanceOf, isEquivalent\}, \quad (8)$$

где *isSubclass* – отношение «быть подклассом», *instanceOf* – отношение «быть экземпляром класса», *isEquivalent* – отношение «быть эквивалентным».

Тогда с учетом формулы (8) представим множество $R = R^S \cup R^I \cup R^E$ как объединение непересекающихся подмножеств отношений «быть подклассом» R^S , «быть экземпляром класса» R^I и «быть эквивалентным» R^E ($R^S \cap R^I \cap R^E = \emptyset$), логическая структура которых задана формулами (5) – (8).

Логическая структура R^S :

$$R^S = \{ \langle c'_1 \in C : c'_1 isSubclass c \rangle, \dots, \langle c'_v \in C : c'_v isSubclass c \rangle \}, \quad (9)$$

где c, c' – классы, C – множество классов, v – общее количество классов, связанных отношением «быть подклассом».

Логическая структура R^I :

$$R^I = \{ \langle c'_1 \in C : c'_1 instanceOf c \rangle, \dots, \langle c'_w \in C : c'_w instanceOf c \rangle \}, \quad (10)$$

где c, c' – классы, C – множество классов, w – общее количество классов, связанных отношением «быть экземпляром класса».

Логическая структура R^E :

$$R^E = \{ \langle c'_1 \in C : c'_1 isEquivalent c \rangle, \dots, \langle c'_z \in C : c'_z isEquivalent c \rangle \}, \quad (11)$$

где c, c' – классы, C – множество классов, z – общее количество классов, связанных отношением «быть эквивалентным».

С учетом вышесказанного, класс $c \in C$ схемы G_t^S определим как логическую

структуру с установленными отношениями Rс другими классами, свойствами F формулы (5) и функциональностью $Pr \subset Pr^U$ следующим образом:

$$\forall c \in C^{G_i} : c = \langle R, F, Pr \rangle, \quad (12)$$

Важно отметить, что для разделения классов и их экземпляров в работе используется не только отношение «быть экземпляром класса», но и установленные значения (V) свойств F класса.

Таким образом, формальную модель информационного пространства на основе связанных данных представим формулами (1) – (12).

Интеграционный процесс составляющих компонентов информационного пространства представим как отображение (δ_1) существующих данных и схем источников (SRC) на информационное пространство (I) предприятия и формально представим следующим образом:

$$\delta_1 : \forall s \in SRC \xrightarrow{sh^1} I, \quad (13)$$

где sh^1 – интеграционная схема информационного пространства.

С учетом положений модели, программное обеспечение для обработки связанных данных, будем рассматривать как расширение функциональных возможностей информационного пространства, реализующее конкретные необходимые для пользователя процедуры. В рамках такого подхода практическая реализация изложенных в работе положений подразумевает использование протокола SPARQL и промежуточного программного обеспечения OpenLink Virtuoso или любого другого, поддерживающего коммуникацию посредством протокола SPARQL.

Для реализации подключения к источнику данных и загрузки данных в информационное пространство можно использовать следующую процедуру языка Virtuoso/PL ПП OpenLink Virtuoso:

```
create procedure connectRS ()
{
  declare graph varchar;
  for ( sparql SELECT ?graph ?s ?p ?o
  WHERE
    {
      SERVICE
<http:// tniks.kdu.edu.ua/sparql>
      {
        graph ?graph {?s ?p ?o }
      }
    })
  do {
    sparql load ?graph into ?graph;
  }
};
```

где graph –именованный граф, s- субъект, p - предикат, o - объект.

Обращаем внимание, что вышеприведенный скрипт является реализацией обобщенного

алгоритма загрузки без учета специфических особенностей для каждого источника данных, загрузка которых осуществляется в информационное пространства.

Из вышесказанного следует, что при построении информационного пространства промышленного предприятия задача интеграции составляющих информационное пространство ресурсов требует проведение дополнительных исследований, которые будут проведены в дальнейшем.

Заключение

В работе были рассмотрены вопросы построения информационных пространств, как интеграционного процесса объединения источников данных информационных систем промышленных предприятий для формирования единого интерфейса доступа на основе объектно-ориентированного подхода.

Разработана формальная модель информационного пространства и предложены соответствующие отображения в качестве интеграционных компонентов существующих источников в информационное пространство

Библиографический список

- [Майер-Шенбергер В., 2014] Майер-Шенбергер В. Большие данные. Революция, которая изменит то, как мы живем, работаем и мыслим / Майер-Шенбергер В., Кукьер К. – М.: Манн, Иванов и Фербер, 2014. – 240 с.:ил.
- [DuCharme, 2011] DuCharmeB. Learning SPARQL / B.DuCharme. - O'ReillyMedia: 2011. - 258 с.
- [Powers, 2008] Powers S. Practical RDF / S. Powes. - O'Reilly Media: 2008. - 352 с.
- [Холзнер, 2004] Холзнер С. XML. Энциклопедия, 2-издание / СтивенХолзнер. - СПб.:Питер, 2004. - 1101 с.:ил.
- [Бек, 2008] Бек К. Шаблоны реализации корпоративных приложений.: Пер. с англ. - М.:ООО "И. Д. Вильямс", 2008. – 176 с.:ил.
- [Гудсон, 2013] Гудсон Дж. Практическое руководство по доступу к данным / Д. Гудсон, Р.Стюарт. – СПб.:БВХ-Петербург, 2013. – 304с.:ил.
- [Палкин, 2013] Палкин Н.Б. Бизнес-аналитика от данных к знаниям, 2-е издание / Палкин Н.Б., Орешков В.И. – СПб.: Питер, 2013. – 704 с.:ил.

PECULIARITIES OF LINKED DATA INFORMATIONAL SPACE DEVELOPMENT FOR INDUSTRIAL ENTERPRISES

Galushka I.N. *, Shcherbak S.S. *

*KremenchukMykhailoOstrohraskyiNational
University, Kremenchuk, Ukraine

ilona.galushka@ya.ru

sergey.shcherbak@gmail.com

Issues on development of informational spaces are observed as an integration process of uniting information system data sources of industrial enterprises for generation of unique access interface based on object-oriented design. Informational space model is developed with corresponding mappings (as integration components) of existing resources into the informational space



УДК 004.822:514

МЕТОД ЦЕЛЕНАПРАВЛЕННОГО ПОИСКА СВЯЗАННЫХ ДАННЫХ В УСЛОВИЯХ СТРУКТУРНОЙ НЕОПРЕДЕЛЕННОСТИ

Тертышный В.А. *, Шаповал И.С. *, Щербак С.С. *

**Кременчугский национальный университет имени Михаила Остроградского,
г. Кременчуг, Украина*

vladislafus@gmail.com

sergey.shcherbak@gmail.com

В работе рассмотрены вопросы организации поиска связанных данных в условиях отсутствия структурно-логических схем распределенных источников. Разработан метод поиска на основе взаимных отображений объектов и связанных данных, и предложены рекомендации по созданию пользовательских интерфейсов поиска.

Ключевые слова: поиск; связанные данные; паттерны реализации; информационное пространство

Введение

Распределенность информационных ресурсов (ИР), содержимое которых создано на основе концепции связанных данных (англ. Linked Data) приводит к необходимости переосмысления понятия информационного поиска как такового. В рамках концепции связанных данных рассмотрим ИР (IR), как составляющие некоторого информационного пространства I. В рамках такого подхода будем рассматривать ИР как совокупность объектов, объединенных единым контекстом и принадлежащих некоторой предметной области, и связанных между собой отношениями «быть подклассом» и другими, характерными для объектно-ориентированного подхода.

Информационное пространство (ИП) обеспечивает распределенное хранение объектов - экземпляров классов, описанных в ИР с наборами действий в виде присоединенных к объектам процедур, применяемых для организации поиска. Таким образом, целью поиска связанных данных в информационном пространстве будет нахождение наиболее релевантного запросу объекта в виде набора характеристик объекта или объектов и их состояний. Так как хранилища, соответствующие источникам связанных данных, из которых состоят информационные пространства, позволяют для обеспечения синтаксической гибкости представлять одни и те же синтаксические конструкции по-разному, то пользователь перед взаимодействием с источником информационного пространства должен указать, на основе какого отображения

интерпретировать связанные данные в терминах объектно-ориентированного подхода, поэтому определим два отображения (1), (2).

Выражение δ_1 определяет, что контексты G^c отображаются на классы C , предикаты P на свойства F , объекты O – на значения свойств V , а субъекты S – на экземпляры класса C' , которые связаны с C' отношением «быть экземпляром класса» R^i :

$$\delta_1 : G^c \rightarrow C, P \rightarrow F, O \rightarrow V, S \rightarrow C'^i C, \quad (1)$$

Выражение δ_2 определяет, что контексты G^c отображаются на экземпляры класса C' , который связан с C отношением «быть экземпляром класса» R^i , предикаты P – на свойства F , а объекты O – на значения свойств V :

$$\delta_2 : G^c \rightarrow C'^i C, P \rightarrow F, O \rightarrow V, \quad (2)$$

С учетом выражений δ_1, δ_2 и необходимостью при поиске наиболее релевантных объектов анализа не только составляющих структурно-логических схем классов, но и состояний их экземпляров, релевантность запроса экземпляра класса будем определять на основе метрики подобия. Таким образом, наиболее релевантным запросу экземпляром класса будет экземпляр класса, степень сходства которого согласно выбранной метрике будет наибольшей.

Метод целенаправленного поиска связанных данных

Для разработки метода целенаправленного поиска связанных данных уточним определения используемых в работе понятий.

Определение 1. Запрос представляет собой набор характеристик и их состояний, описывающий в терминах концепции связанных данных искомого в информационном пространстве объекта.

Определение 2. Наиболее релевантный запросу класс – это класс, у которого число совпадающих свойств класса и запроса наибольшее.

Замечание 1. Осознавая необходимость учитывать не только составляющие схемы класса, но и состояния его экземпляров, введем также понятие наиболее релевантного запросу экземпляра класса, соответствующего определению 2, но с дополнением в виде «метрики сходства». Таким образом, наиболее релевантным запросу экземпляром класса будет являться экземпляр класса, мера сходства которого согласно выбранной метрике является наибольшей.

Определение 3. Наиболее релевантная иерархия классов – это часть иерархии классов информационного пространства, базовый класс которой наиболее релевантен запросу.

Определение 4. Базовый класс – это класс источника связанных данных информационного пространства, с которым другие классы находятся в отношении «быть подклассом» (англ. «isSubclass»).

Для распознавания состояния объекта будем использовать следующую последовательность действий, а именно распознавание класса объекта, соответствующего запросу, на основе источников связанных данных ИП в соответствии с правилами интерпретации, представленных в виде присоединенных процедур, каждая из которых может использоваться экземпляром класса в соответствии с его структурно-логической схемой.

Замечание 3.2. В дальнейшем, с целью упрощения математической реализации предлагаемого метода задачу поиска решений будем рассматривать как задачу поиска на основе одного запроса, что ни в коей мере не сужает общности суждений.

Учитывая вышесказанное, последовательность метода поиска решений определим как совокупность действий, представленных следующими этапами и направленными на нахождение экземпляра класса информационного пространства, соответствующего состоянию объекта информационного пространства информирования процедуры визуализации состояния этого объекта:

Этап 1. Формирование запроса к ИП через пользовательский интерфейс, анализ его на противоречивость и построение соответствующего запроса экземпляра класса, поиск которого будет осуществляться.

Этап 2. Выбор метрики сходства, на основе которой будет осуществляться поиск.

Этап 3. Передача запроса к источнику СД ИП, а именно к их экземплярам базового класса, по подклассам которого будет проводиться дальнейший поиск.

Этап 4. Целенаправленный поиск наиболее релевантного запросу класса объектов для сокращения пространства поиска путем установления соответствия между схемами классов запроса и ИП.

Этап 5. Определение наиболее релевантных объектов по состоянию их характеристик на основе выбранной метрики сходства путем анализа ограничений назначения свойств класса, соответствующих объектов с целью выявления шаблонов визуализации в виде присоединенных процедур, формирующих ответ на запрос в виде визуализации в определенной форме содержимого объекта и синтез при необходимости нового шаблона визуализации.

Этап 6. Агрегация результатов поиска.

Этап 7. Оценка и формирование результатов поиска, упорядочивание на основе их рейтинга и представление в виде перечня визуализированных объектов по запросу пользователю.

Особенностью подхода, изложенного в методе, является то, что среда WWW, используемая в работе в качестве коммуникационной среды, позволяет параллельно осуществлять запросы к различным источникам связанных данных ИП. За счет этого одновременно осуществляется поиск в различных источниках. Кроме того, за счет анализа схем классов объектов и запроса, осуществляется целенаправленный поиск объектов и соответствующих им шаблонов визуализации путем сокращения пространства поиска благодаря выбору наиболее релевантных данному запросу объектных иерархий источника связанных данных, по которому осуществляется поиск.

Согласно вышеизложенному, реализацию математического обеспечения метода осуществим путем добавления к ИП присоединенных процедур, соответствующих положениям этого метода для визуализации содержимого объектов, что позволит находить объекты и формировать или использовать существующие шаблоны визуализации объектов.

Математическое обеспечение метода целенаправленного поиска связанных данных

Пусть Pr^{ds} – множество присоединенных процедур модели специализированной поисковой системы M1 (СПС), которые реализуют операционную составляющую этой системы в виде подмножеств присоединенных процедур Int, Viz , реализующих интерпретацию и визуализацию

содержимого объектов, которые соответствуют запросу, представленному в виде экземпляра класса $E \subset G_i^S$, где G_i^S – это структурно-логическая схема связанных данных; A_1, \dots, A_k – источники ИП, хранящие данные об объектах, A_r – хранилище специализированной поисковой системы, причем $\{A_1, \dots, A_k\} \cup A_r = \text{Src}$, тогда определим присоединенную процедуру ρ^{cpn} , обеспечивающую взаимодействие пользователя специализированной поисковой системы с ИП как средство формирования управляющего воздействия в виде запроса к ИП через специализированную поисковую систему, выполнение соответствующей входному запросу q процедуры объекта СПС $\rho^{\text{act}} \in \text{Pr}^{\text{ds}}$ (реакция ИП на входное воздействие) и визуализация результатов выполнения запроса (ρ^{viz}):

Пусть $\rho^{\text{act}}, \rho^{\text{search}}, \rho^{\text{int}}, \rho^{\text{dec}} \in \text{Pr}^{\text{ds}}$ – процедуры обработки управляющих воздействий пользователя, поиска, добавления критериев интерпретации значений показателей объекта, формирования шаблонов визуализации пользователя соответственно, тогда ρ^{act} формально определим так:

Рассмотрим математическую реализацию ρ^{search} с учетом (3) – (4) как средства представления основных положений метода целенаправленного поиска объектов в ИП (рис. 1):

$$\rho^{\text{search}}(q) = [\rho^{\text{get}}(q) : \{E^{A_1}, \dots, E^{A_k}\}, \quad (5)$$

Процедура ρ^{get} формально определена через $\rho_{\Lambda}^{\text{rcpn}}$ – присоединенные процедуры взаимодействия с удаленными источниками $\Lambda_1, \dots, \Lambda_k$ – следующим образом:

E^{Ai} – экземпляры наиболее релевантного класса ПС (прецеденты), q – запрос, ρ_i^{ls} – i -я ниже определенная процедура поиска прецедентов источников СД на основе метрики, определенной в $\rho_{A_i}^{metric}$.

Процедура ρ^{rat} предназначена для определения среди источников A_1, \dots, A_k наиболее релевантного результата поиска в виде класса объектов и соответствующих экземпляров объектов на основе их оценки с помощью процедуры $\rho_{A_i}^{\text{rat}}$:

Визуализацию содержимого объектов определим с помощью ρ^{dec} :

ограничения, которые будут отслеживаться СПС и учитываться при поиске объектов. Для этого пользователь может вызвать процедуру ρ^{cpn} , определенную выражением (4) с установленным параметром $in = 2$. В этом случае запрос q к СПС будет представлен парой с минимальным (\min^{prm}) и максимальным (\max^{prm}) ограничением значения показателя, идентифицируемый порядковым номером в множестве показателей Crt^c , а присоединенная процедура добавления нового показателя и его ограничений представлена так:

$$\rho^{int}(\min^{prm}, \max^{prm}) = [Crt^c \cup [\min^{prm}, \max^{prm}]] \quad (11)$$

Особое место в интерактивном взаимодействии пользователя через СПС с ИП занимает процедура формирования запроса. Рассмотрим ее более подробно.

Так как объект – экземпляр класса объектов, тогда его структурно-логическая схема может быть задана следующим образом:

$$x = \langle \langle x \text{ instanceOf } X' \rangle, f_1 \dots f_k \in F, p^{dec} \rangle, \quad (12)$$

где x – экземпляр класса объектов X' , свойства $f_1 \dots f_k \in F$, p^{dec} – процедура визуализации содержания объектов.

Класс объектов X' , подкласс базового класса c^{base} ИСД ИП с установленным отношением $isSubclass$ и определенным множеством F , зададим с помощью следующего выражения:

$$X' = \langle \langle X' \text{ isSubclass } c^{base} \rangle, F, \emptyset \rangle, \quad (13)$$

Запрос к ИП, задаваемый пользователем через СПС определим как шаблон объектах класса X' в виде набора свойств F_c установленными значениями v_1, v_k следующим образом:

$$q = \langle \langle x \text{ instanceOf } X' \rangle, \langle t_1, l_1, v_1 \rangle, \dots, \langle t_k, l_k, v_k \rangle \in F, \emptyset \rangle, \quad (14)$$

где t_1, t_k – типы данных, l_1, l_k – языки представления, k – номер свойства.

Учитывая возможность построения противоречивых запросов к ИП, определим понятие противоречивости запроса и разработаем процедуру проверки запроса на противоречивость.

Определение 5. Под противоречивостью запроса будем понимать появление в запросе двух и более одинаковых свойств, как по структуре, так и по содержанию.

Для проверки запроса на противоречивость, согласно определению (5), определим следующее выражение, результат которого 1 означает наличие противоречия в запросе:

$$pq(q) = \begin{cases} 1 & | \exists f \in F^q \mid f = f' \end{cases}, \quad (15)$$

где $f, f' \in F$ – свойства, определенные в запросе q .

С учетом вышесказанного, математическое обеспечение метода поиска решений в информационном пространстве может быть определено с помощью формул (1) – (15).

В качестве рекомендаций по проектированию поисковых интерфейсов можно выделить следующие положения, а именно подобные интерфейсы должны:

- обеспечивать своевременное информирование пользователя о текущем состоянии системы, смене состояния в результате выполнения поисковых запросов;

- формировать команды по принципу модели "объект - действие" вместо модели «действие-объект»;

- поддерживать монотонность интерфейса и интерфейсы на различных устройствах и платформах, включая режим сенсорного ввода.

Заключение

В работе рассмотрены вопросы организации целенаправленного поиска связанных данных в условиях отсутствия структурно-логических схем распределенных источников. Разработан метод поиска на основе взаимных отображений объектов и связанных данных, и предложены рекомендации по созданию пользовательских интерфейсов поиска.

Библиографический список

- [Майер-Шенбергер В., 2014] Майер-Шенбергер В. Большие данные. Революция, которая изменит то, как мы живем, работаем и мыслим / Майер-Шенбергер В., Кукьер К. – М.: Манн, Иванов и Фербер, 2014. – 240 с.:ил.
- [Гудсон, 2013] Гудсон Дж. Практическое руководство по доступу к данным / Д. Гудсон, Р.Стюарт. – СПб.:БВХ-Петербург, 2013. – 304с.:ил.

METHOD OF LINKED DATA SEARCH IN CONDITIONS OF STRUCTURAL UNCERTAINTY

Tertishniy V.A., Shapoval I.S., Shcherbak S.S.*

*Kremenchuk Mykhailo Ostrohraskyi National University, Kremenchuk, Ukraine
vladislafus@gmail.com
sergey.shcherbak@gmail.com

In this paper linked data search issues have been examined in conditions of structural and logical schema absence for distributed resources. A search method has been developed, which is based on mutual mappings of objects and lined data. Recommendations concerning search user interface development have been proposed.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ПРАКТИЧЕСКИЕ АСПЕКТЫ ТЕСТИРОВАНИЯ РАСПРЕДЕЛЕННЫХ ПРИЛОЖЕНИЙ НА ОСНОВЕ СВЯЗАННЫХ ДАННЫХ

Солошич С.Н.*

** Кременчугский национальный университет имени Михаила Остроградского,
г. Кременчуг, Украина*

soloshich@gmail.com

В данной работе рассмотрены практические аспекты тестирования распределенных приложений, использующих триплесторы в качестве хранилищ связанных данных и предложен новый подход к тестированию подобных хранилищ, который позволяет выявлять проблемы, связанные с внутренним состоянием хранилища.

Ключевые слова: тестирование; распределенные приложения; связанные данные; SPARQL.

Введение

Целесообразность проведения тестирования удаленных открытых хранилищ связанных данных (англ. Triplestore) обусловлена необходимостью проверки работоспособности разрабатываемого пользовательского программного обеспечения (ПО) с этими хранилищами. Эта необходимость обусловлена тем, что хранилища от различных разработчиков программного обеспечения поддерживают разное подмножество запросов SPARQL (англ. SPARQL Protocol and RDF Query Language) и обладают различной функциональностью для обработки связанных данных. Уровень поддерживаемой функциональности хранилища различается от источника к источнику и определяет уровень поддерживаемой функциональности клиентского ПО, что зачастую ограничивает клиентское ПО в функциональности, делая недоступными функции доступа к связанным данным, например, по изменению этих данных, а именно функции SPARQL Update не доступны. Также важными проблемами в этом вопросе являются выявления противоречивости, полноты и избыточности связанных данных.

Тестирование хранилищ связанных данных также имеет важное значение для гарантирования высокого качества программного обеспечения, так как незамеченные ошибки могут привести к неисправимому искажению данных. Проблему тестирования хранилищ связанных данных в целом можно разбить на проблемы создания тестов, подготовку тестируемых данных и истолкование

результатов испытаний. Среди трех проблем, проблема генерирования тестов напрямую влияет на эффективность тестирования. Традиционно, тестирование хранилищ связанных данных основано на проверке его способности выполнять набор предопределенных функций. Достижение элементарного (базового) уровня качества, стратегии тестирования методом черного ящика (англ. Black box) хранилищ связанных данных, является важным заданием, но использование метода белого ящика (англ. White box) необходимо для более тщательного ее тестирования. Несмотря на важность данной задачи, семантика операторов языка запросов к связанным данным SPARQL, встроенных в хранилища связанных данных, редко рассматривается в обычных методах тестирования «белого ящика». Мы предлагаем дополнить методы тестирования «белого ящика» с помощью использования семантики SPARQL. Данный подход заключается в преобразовании встроенных операторов SPARQL в процедуры в некоторых языках программирования общего назначения (далее – ЯПН) и тем самым генерации тестовых случаев с использованием традиционных методов тестирования «белого ящика». Дополнительные тестовые случаи, которые не охватываются в традиционном тестировании методом белого ящика, создаются для повышения эффективности тестирования пользовательского программного обеспечения хранилищ связанных данных.

Увеличение, как самой сложности хранилищ связанных данных, так и ожиданий относительно ее надежности способствовало возрастанию требований к методам тестирования программного обеспечения. Эффективное и действенное

тестирование программного обеспечения имеет решающее значение в цикле разработки программного обеспечения и технического обслуживания.

Тестирование хранилищ связанных данных имеет большое значение как на стадии разработки, так и в процессе работы, поскольку незамеченные ошибки в этих приложениях могут привести к неправильной модификации или случайному удалению важных данных. После того как данные ошибочно изменены, ошибка может распространяться и привести к увеличению разрушения данных, если она не отслежена. Так как не все транзакции могут быть развернуты, восстановления данных из резервных копий базы данных не сможет решить проблему. Хорошо известные правила функциональной зависимости в состоянии обеспечить целостность хранилищ связанных данных, но понесенные расходы, как правило, слишком велики для нетривиальных коммерческих хранилищ связанных данных [1].

Для тестирования хранилищ связанных данных должны быть разработаны как статические, так и динамические испытания. Статические мероприятия, такие как осмотр и проверка, позволяют определить соответствие прикладной программы и дизайна хранилищ связанных данных всем функциональным требованиям к данным и отсутствие конфликта между этими требованиями [2]. Кроме того, приложения должны ограничивать несанкционированный доступ [3], позволять одновременный доступ, и поддерживать восстановление данных после аппаратных или программных сбоев. Хотя различные исследования были проведены для определения статических методов тестирования проектирования хранилищ связанных данных, относительно мало внимания было уделено непосредственно динамическому тестированию хранилищ связанных данных. Предложен подход тестирования, который преобразует встроенные запросы SPARQL хранилищ связанных данных для процедур на языке программирования общего назначения. Целью преобразования является включение в семантику SPARQL таких запросов, которые позволят сгенерировать больше тестов и выявить недостатки, относящиеся к изменениям внутренних состояний распределенной базы данных. Подход позволяет сгенерировать тесты, которые функционируют в семантике GPL и SPARQL запросов, которые созданы с использованием традиционных методов тестирования «белого ящика» и, следовательно, помогает выявить больше ошибок, содержащихся в распределенных базах данных. В частности, он обнаруживает неисправности, которые произошли в некоторых специфических внутренних состояниях хранилищ связанных данных.

Хранилища связанных данных в целом можно разделить на две категории. Первая категория состоит из приложений, которые являются

исключительно встроенными в SPARQL Update (далее – SPARUL). В этих приложениях запросы и транзакции указываются через SPARUL, а пользовательские интерфейсы определены с использованием элементов на языке запросов хранилища связанных данных. Другая категория включает в себя приложения, которые построены в SPARUL и ЯПН, таких как C++, C# и Java [4]. Запросы в SPARUL, мощном декларативном языке запросов, встраиваются в программы приложений, написанных на любом ЯПН, который называется “принимающим языком” [4]. В этих приложениях запросы и изменения данных записываются как встроенные операторы SPARQL. Другие функции, такие как взаимодействие с пользователями и отправка результатов в графическом интерфейсе пользователя, записываются посредством ЯПН. Задачей является тестирование этой категории приложений баз данных.

Тестирование приложений хранилищ связанных данных, отличается от тестирования структурированных программ. Исходные данные хранилищ связанных данных включают в себя как данные, вводимые пользователем, так и экземпляры объектов хранилищ связанных данных. Как указано выше в дополнение к проверке исхода с ожидаемым результатом, программисты или тестеры должны также проверить, является ли хранилище связанных данных согласованным и удовлетворяет ли исходным условиям в тестировании, которые предъявляются к хранилищам связанных данных. Проблему тестирования подобных хранилищ можно разделить на три подзадачи, как показано на рисунке 1. Это проблемы генерирования тестов, подготовки данных к тестам, выполнение тестов и проверка результата.

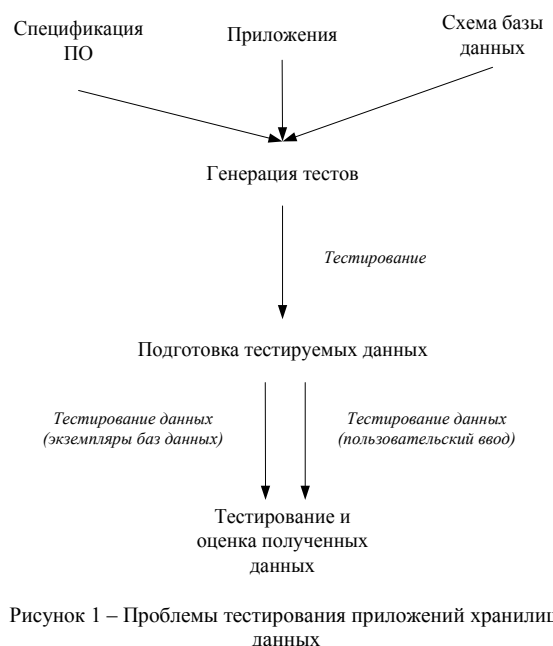


Рисунок 1 – Проблемы тестирования приложений хранилищ данных

Среди трех подзадач, проблема генерирования тестов нуждается в решении больше, чем остальные, поскольку эффективность тестирования

хранилища связанных данных непосредственно связана с созданием тестов. Как уже говорилось во введении, незамеченные недостатки в хранилищах связанных данных может привести к серьезному повреждению важных результирующих данных. Данная работа ориентируется на проблемы генерирования тестов. Традиционно методы тестирования «черного ящика» используются для создания функциональных тестов. Тестирование методами «белого ящика» требуется для достижения более высокого уровня качества и надежности программного обеспечения. В работе [5], Robbert и Maryanski предложил концептуальную модель подхода к генерации тестов, которые можно применить к хранилищам связанных данных.

1. Тестирование на основе методов черного ящика

В настоящее время большинство приложений хранилищ данных проверяются с использованием методов тестирования черного ящика, например, эквивалентное разделение, анализ граничных значений и причинно-следственные графические методы. В рамках этих методов тестовые сценарии получаются без посылания на построение прикладных программ. В соответствии с характеристиками программного обеспечения, тесты проводятся для проверки работоспособности независимо друг от друга. Это обычно используется для проверки соответствия характеристик программного обеспечения к требованиям приложения.

Одно из преимуществ методов тестирования черного ящика является то, что тестовые варианты могут быть получены независимо от программ на более ранней стадии цикла разработки программного обеспечения. Программисты могут проводить тесты во время разработки приложения. Таким образом, итоговое приложение, как правило, удовлетворяет большую часть сгенерированных тестов. Относительно небольшие усилия требуются для тестирования и отладки приложений. Еще одним преимуществом является то, что затраты на генерацию тестовых случаев с использованием методов «черный ящик» довольно низкие по сравнению с расходами, связанными с методами белого ящика. Таким образом, стоимость разработки может быть снижена. Кроме того, многие методы тестирования «черного ящика» уже хорошо апробированы. Тестеры могут выбрать наиболее подходящий метод для тестирования их приложений хранилищ данных. С другой стороны, без изучения программы приложения, невозможно знать, какая часть приложения проходит это тестирование. Практически, функциональные спецификации, как правило, указаны на естественном языке. Как отмечено в [2], использование естественного языка с целью определения ошибок в функциональной спецификации, таких как: дублирование,

противоречивость и неполнота. Кроме того, большинство методов тестирования «черного ящика» нечувствительны к некоторым видам неисправностей. Общие виды неисправностей еще могут остаться незамеченными, даже если все тесты, определенные методом тестирования успешны. Типичным примером является своего рода неисправности, что происходят только в некоторых специфических последовательностях выполнения функций. Таким образом, необходимо искать методы для применения за пределами тестирования «черного ящика».

2. Традиционный подход к тестированию методами белого ящика

Подходы к тестированию методами «белого ящика» использующие покрытие операторов, условий, путей, функций, значений параметров могут быть использованы для тестирования некоторой части или применены ко всему хранилищу связанных данных для достижения более полного тестирования этого хранилища. Методы тестирования белого ящика позволяют изучить код подробно и убедиться, что, по крайней мере, в определенной степени тестовое покрытие, такое как выполнение каждого оператора было достигнуто. Тем не менее, традиционные методы тестирования «белого ящика» имеют свои ограничения при тестировании хранилища связанных данных. Самое главное, они не полностью учитывают семантику запросов SPARQL, встроенных в прикладные программы распределенных приложений. Запросы SPARQL рассматриваются как «черные ящики». Как правило, на основе тестирования «белого ящика» генерируется только несколько тестов для тестирования встроенных SPARQL запросов. Следует отметить, что преднамеренно не генерируется никаких тестов, включающих семантику запроса SPARQL, которые отражают изменение внутреннего состояния хранилища связанных данных. Следовательно, можно утверждать, что традиционное тестирование методом «белого ящика» может пропустить тип неисправностей, связанных с внутренними изменениями в хранилище.

Чтобы удовлетворить условия этих методов для приложений хранилищ связанных данных, предлагается механизм, который позволяет сгенерировать дополнительные тесты из встроенных запросов SPARQL. Механизм, который показан на рисунке 2, преобразовывает операторы SPARQL хранилища связанных данных в операторы ЯПН. Затем стандартные методы «белого ящика» применяются к трансформированным запросам и другим операторам, записанным на «принимающем языке». Также задачей является включение семантики оператора SPARQL в генерацию теста. Другими словами, выполняется проверка правильности семантика запроса SPARQL в

сочетании с запросом языка программирования общего назначения.

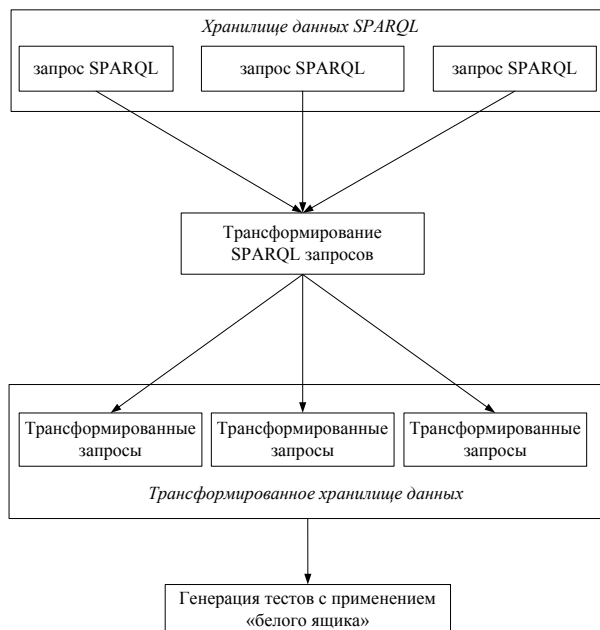


Рисунок 2 – Трансформация SPARQL запросов

Так же следует учитывать отсутствие возможности тестирования SPARQL операторов по отдельности, как «черные ящики», потому что функциональная спецификация не определена для запросов SPARQL. Кроме того, SPARQL не может быть достаточно эффективным, чтобы выполнить все функции хранилища связанных данных и операторы ЯПН, возможно, придется использовать для дополнения запросов SPARQL при выполнении некоторых функций.

Таким образом, проблему тестирования хранилищ связанных данных распределенных приложений декомпозируем на проблемы генерирования тестов, подготовки данных исследования, выполнения тестов и итогового контроля, что согласуется с положениями функционального тестирования, которое используется для генерации тестов для тестирования приложений баз данных, в то время как запросы SPARQL, встроенные в прикладные программы конкретно не рассматриваются в тестировании методом «белого ящика».

Проведенное исследование подходов к генерации тестов для хранилищ связанных данных выявило, что все они имеют слабые места, но объединение разных методов позволяет их устранить.

Заключение

В данной работе рассмотрены практические аспекты тестирования хранилищ связанных данных и предложен новый подход к проведению тестирования, который преобразует запросы SPARQL в процедуры на языке программирования общего назначения и применяет традиционные методы «белого ящика» для процедур и

принимающих запросов, используемых для генерации тестов. Полученные в результате тесты помогут выявить недостатки, связанные с внутренним состоянием хранилища связанных данных.

В дальнейшем будет проведено больше эмпирических исследований для повышения эффективности предложенного подхода к тестированию.

Библиографический список

- [Гергор, 2007] Гергор, Х. В. Бобби. Шаблоны интеграции корпоративных приложений / Х. Гергор, В. Бобби. // Пер. с англ. – М.: ООО «И.Д. Вильямс», 2007. – 672 с.
- [Громюк, 2011] Громюк Б.П. Аналіз та оцінка скриптів SQL / Б.П. Громюк // Восточно-Европейский журнал передовых технологий, 2011, 6/2 (54), С. 22-25.
- [Малишевський, 2009] Малишевський О.Г. Пріоритизація тестів як метод швидкого виявлення серйозних помилок / О.Г. Малишевський // Комп'ютерні технології, 2009, Вип. 93, С. 100-101.
- [Фаулер, 2007] Фаулер, М. Архитектура корпоративных программных приложения / М. Фаулер // Пер. с англ. – М.: Издательский дом «Вильямс», 2007. – 544 с.
- [DuCarme, 2011] DuCarme, B. Learning SPARQL / B. DuCarme // O'Reilly, 2011. – 256с.
- [Фаулер, 2007] Фаулер, М. Архитектура корпоративных программных приложения / М. Фаулер // Пер. с англ. – М.: Издательский дом «Вильямс», 2007. – 544 с.
- [DuCarme, 2011] DuCarme, B. Learning SPARQL / B. DuCarme // O'Reilly, 2011. – 256с.
- [Harinath, 2010] Harinath, S. Microsoft SPARQL Server Analysis Services 2008 with MDX / S. Harinath, R. Zare. – «Sams», 2010. – 1072 с.
- [Kaner, 2001] Kaner, C. Testing computer software / C. Kaner, J. Falk, H. Nguyen. – М: DiaSoft, 2001. – 538 с.
- [Robbert, 1991] Robbert M.A. Automated Test Plan Generator for Database Application Systems, Proceedings of the 1991 ACM SIGSMALL/ M.A. Robbert, F.J. Maryanski // PC Symposium on Small Systems, ACM Press, 1991, p.100-106.
- [Майер-Шенбергер В., 2014] Майер-Шенбергер В. Большие данные. Революция, которая изменит то, как мы живем, работаем и мыслим / Майер-Шенбергер В., Кукьер К. – М.: Манн, Иванов и Фербер, 2014. – 240 с.:ил.
- [Гудсон, 2013] Гудсон Дж. Практическое руководство по доступу к данным / Д. Гудсон, Р.Стюарт. – СПб.:БВХ-Петербург, 2013. – 304с.:ил.
- [Палкин, 2013] Палкин Н.Б. Бизнес-аналитика от данных к знаниям, 2-е издание / Палкин Н.Б., Орешков В.И. – СПб.: Питер, 2013. – 704 с.:ил.

PRACTICAL ASPECTS TESTING DISTRIBUTED APPLICATIONS BASED ON THE RELATED DATA

Soloshych S.N.*

*Kremenchuk Mykhailo Ostrohraskyi National
University, Kremenchuk, Ukraine

soloshich@gmail.com

This thesis overviews the practical aspects of distributed applications testing based on the linked data, a new approach to the testing of distributed databases is proposed, that can identify problems associated with the internal state of a distributed database.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.827

ПРОГРАММНЫЙ ИНСТРУМЕНТАРИЙ ДЛЯ РАЗРЕШЕНИЯ МОРФОЛОГИЧЕСКОЙ МНОГОЗНАЧНОСТИ В ТАТАРСКОМ ЯЗЫКЕ

Сулейманов Д.Ш. *, Гильмуллин Р.А. *, Гатауллин Р.Р. *

**Казанский (приволжский) федеральный университет,
Научно-исследовательский институт «Прикладная семиотика» АН РТ,
г. Казань, Республика Татарстан*

dvd.slt@gmail.com

rinatgilmullin@gmail.com

ramil.gata@gmail.com

В работе описывается программный инструментарий, предоставляющий широкие возможности для создания, редактирования, а также тестирования Базы контекстных правил автоматического разрешения морфологической многозначности в татарском языке.

Ключевые слова: морфема; омоформа; разрешение морфологической многозначности; База контекстных правил.

Введение

Разрешение многозначности в тексте является одной из наиболее сложных и важных задач компьютерной лингвистики. Данная проблема с особой остротой обнаруживается при попытке создания автоматических систем обработки текстов, т.е. полностью автоматизировать такие процессы как поиск информации, перевод текста с одного языка на другой, разметка текстов в электронном корпусе языка, извлечение знаний, контент-анализ и др.

Научные исследования по разрешению многозначности, в частности, лексической многозначности (word sense disambiguation, WSD), исследованию которой для татарского языка наиболее активно занимаются авторы статьи, находятся в поле зрения прикладной и компьютерной лингвистики достаточно давно и имеют многолетнюю историю. Несмотря на то, что с течением времени количество предложенных решений и их эффективность неуклонно растут, а сравнительно-эффективные показатели точности достигли определённого, более-менее приемлемого уровня (т.е. достигается ситуация, когда автоматическое разрешение многозначности дает результат, который уже может быть использован в практических задачах, и ручная обработка уже не является предпочтительной в силу ее трудоемкости и дороговизны), полного решения проблема пока не

получила, поскольку на пути успешного решения стоит множество задач, напрямую связанных с языковыми особенностями человеческой речи [Зализняк, 2004].

Одной из подзадач WSD является разрешение грамматической омонимии (морфологической многозначности). Исследование этой проблемы весьма важно для задач корпусной лингвистики и компьютерной лексикографии, в особенности при создании электронного корпуса татарского языка, в котором одной из наиболее важных задач является разрешение морфологической многозначности в размеченной коллекции текстов.

Для решения данной задачи необходимо разработать принципы классификации и оформить татарском языке и построить классификацию омоформ на основе разработанных принципов, далее – исследовать контекстные ограничения в различных типах омоформ и разработать правила разрешения многозначности на основе обнаруженных контекстных ограничений.

В данной работе приводится описание программного инструментария (ПИ), реализованного для решения вышеуказанных задач.

1. Методы разрешения омонимии

В настоящее время в задачах автоматического разрешения омонимии используются следующие методы: контекстный, статистический и гибридный.

Метод контекстного разрешения грамматической омонимии сводится к разработке для каждого функционального типа омонимии группы правил, задающих синтаксический контекст разрешения омонима, и построение управляющей структуры группы, определяющей порядок применения правил. В работе «Разрешение функциональной омонимии в русском языке на основе контекстных правил» [Невзорова и др., 2005] подробно описаны основные достоинства и недостатки данного метода, приведены конкретные структуры обобщенных правил для разрешения функциональной омонимии некоторых типов. Подход, основанный на правилах, является чрезвычайно трудоемким, требует проведения тщательной лингвистической экспертизы каждого типа омонимии.

Несмотря на указанные недостатки, приходится констатировать, что в настоящее время для татарского языка наиболее предпочтительным является метод основанный на контекстных правилах. Во-первых, в настоящее время не имеется достаточно объемного электронного корпуса татарского языка, позволяющего полноценно задействовать статистические и гибридные модели, во-вторых, регулярность грамматики и строгая подчиненность правилам практически на всех языковых уровнях [Сулейманов Д.Ш., 1994], позволяют рассчитывать на обнаружение и описание четких контекстных ограничений.

1.1. Обобщенный контекстный метод

Обобщенный метод контекстного разрешения функциональной омонимии для татарского языка включает несколько этапов:

1) построение полной классификации типов функциональных омонимов;

2) выделение минимального множества разрешающих контекстов для каждого типа. Минимальность множества означает, что для каждого типа функционального омонима следует оценить сложность распознавания каждой части речи, принадлежащей данному типу. Затем необходимо построить множество разрешающих контекстов (МРК), имеющих минимальную сложность распознавания. В алгоритмической записи данное требование выражается следующим правилом: если для функционального омонима X , имеющего тип $T1$ или $T2$, применено правило из МРК, то тип омонима X определяется примененным правилом, иначе приписывается альтернативный тип;

3) построение управляющей структуры обобщенного правила, обеспечивающего максимальную точность распознавания.

1.1.1. Пример разрешения функциональной омонимии для татарского языка

Рассмотрим разрешение функциональной омонимии типа $(V+Refl)/(N+3PossSg+Acc)$, где $(V+Refl)$ – глагол с аффиксом возвратно-

страдательного залога, $(N+3PossSg+Acc)$ – существительное с аффиксами принадлежности 3 лица единственного числа, на примере омоформы: *асылын*.

Варианты аффиксальной структуры омоформы:

(1) $ac(V)+Bl(Refl)+In(Refl)$ (вешайся) (в предложениях «Муенга асылын»)

(2) $асыл(N)+CЫ(3PossSg)+нЫ(Acc)$ «истинность, сущность, суть (чего-л., кого-л.)» («Гомернең асылын аңлагыз» «Поймите суть жизни»)

Потенциальные модели, главные компоненты и семантика словосочетаний:

(1) в качестве зависимого компонента не встречается;

(2) $N+Acc \rightarrow V$ (главный компонент – глагол, семантика прямого объекта).

Данный тип аффиксальной омонимии разрешается следующим правилом:

если в правом контексте находится глагол, возможна потенциальная модель словосочетания $N+Acc \rightarrow V$

Соответственно, если реализуется данная модель словосочетания, омонимия разрешается по 2-му варианту морфемной структуры, т.е. $N+3PossSg+Acc$: $асыл(N)+CЫ(3PossSg)+нЫ(Acc)$.

В формализованном виде обобщенное правило выглядит следующим образом:

if $(X_1 \cap Acc V^)$ then $X = N + 3PossSg + Acc$
else $X = V + Refl$*

1.2. Ключевые понятия, архитектура контекстного правила

Для решения поставленных задач разработана соответствующая архитектура программного инструментария, включающая следующие базовые объекты и понятия:

- Омоформа (или функциональный омоним) – слова, совпадающие в своем звучании лишь в отдельных формах (той же части речи или разных частей речи);
- База типов омоформ (или База контекстных правил) – иерархически упорядоченный список типов омоформ; для каждого типа определено множество разрешающих контекстов. На основе этих правил происходит разрешение многозначности для отдельно взятого типа омоформ;
- Обобщенное правило разрешения (ОПР) – правило, на основе контекстной информации определяющее актуальный вариант структуры омоформы. Для каждого типа функционального омонима следует оценить сложность распознавания каждой части речи, принадлежащей данному типу

[Сулейманов Д.Ш., 1994];

- Множество разрешающих контекстов (МПК) – совокупность минимальных разрешающих контекстов, достаточных для распознавания функционального омонима как определенного варианта структуры омоформы;

- Управляющая структура обобщенного правила обеспечивает контроль порядка применения правил (рисунок 1);

- Минимальный разрешающий контекст – неделимое в данном контексте простое условие, имеющее минимальную сложность распознавания.

Пример: существование в левом контексте слова с набора морфем вида «V+Refl» в пределах трех слов; возвращает «истинно» или «ложно»;

- Текст – совокупность предложений; обладает рядом свойств, которые могут быть полезным и при расширении контекстного метода;

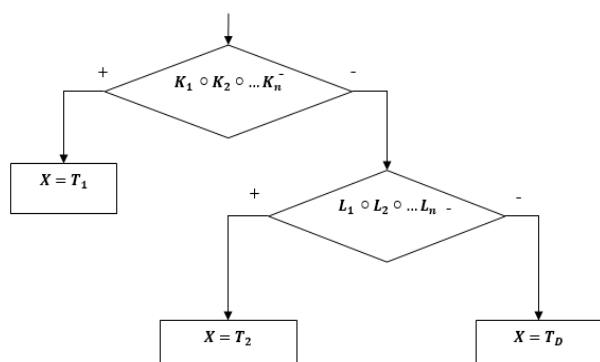


Рисунок 1 – Управляющая структура обобщенного правила $T_1/T_2/T_d$

Здесь, T_1 – морфологический тип, соответствующий распознаванию МПК $K_1 \circ K_2 \circ \dots \circ K_n$,

T_2 – морфологический, тип соответствующий распознаванию МПК $L_1 \circ L_2 \circ \dots \circ L_n$,

T_d – морфологический тип по умолчанию; K_i и L_i – минимальные разрешающие контексты для соответствующих типов, \circ – булевы операции (AND, OR ит.д.).

Процесс распознавания омонимии происходит следующим образом:

- 1) У анализируемого слова определяется тип функциональной омонимии, и в соответствии с этим типом из Базы контекстных правил находится обобщенное правило разрешения;

- 2) Управляющая структура задает порядок применения правил;

- 3) При применении каждого правила, проверяется каждый минимальный контекст разрешения этого правила;

- 4) Если при проверке правила получили подтверждение о его истинности, то функциональная омонимия распознается в

соответствии с этим вариантом структуры омоформы;

- 5) Иначе, если есть другое правило, осуществляется переход к следующему правилу и выполняется то же самое;

- 6) И если нет другого правила, то в качестве структуры выбирается тип по умолчанию;

- 7) Если нет такого типа, то многозначность помечается как неразрешенная.

Минимальными разрешающими контекстами могут служить:

- Наличие в левом (в правом, или с обеих сторон) контексте слова определенной формы (или конкретного ожидаемого слова) на определенном расстоянии от проверяемого слова. Проверяется в пределах предложения;

- Наличие определенных характеристик предложения. В программе нужно обеспечить расширяемость этих характеристик (элемент синтаксического анализа);

- Наличие определенных знаков пунктуации на определенном расстоянии слева (или справа);

- Наличие определенной лексики в предложении.

2. Программный инструментальный для создания и тестирования контекстных правил

2.1. Общие требования к программному комплексу и функционалу

Программный комплекс обеспечивает автоматизацию процесса создания Базы контекстных правил и является «дружественным», т.е. удобным для конечного пользователя.

Программный комплекс содержит следующий функционал:

- создание, редактирование и удаление типов омоформ;
- создание, редактирование и удаление правил и минимальных разрешающих контекстов;
- тестирование: нахождение слов, подходящих выбранному типу омоформ; отображение контекста и морфологической структуры слова.

2.2. Входные и выходные данные

В процессе работы конечный пользователь (в частности, филолог) имеет возможность создавать, редактировать и тестировать Базу контекстных правил. В качестве выходных данных формируется база контекстных правил, которая впоследствии может быть интегрирована в другие программы и использована в целях разрешения морфологической многозначности.

База контекстных правил сохраняется в файл с помощью стандартных методов языка программирования Java.

Для тестирования, в качестве входных данных, программа принимает текст с уже проведенным морфологическим разбором. Текст устроен таким образом, что каждой словоформе отводится две строки: 1) сама словоформа, 2) морфологический разбор данной словоформы (варианты разбора разделяются символом «;»).

2.3. Рабочие области программного инструментария

Программный комплекс содержит два основных модуля: а) модуль для создания, редактирования и тестирования Базы контекстных правил иб) модуль для разрешения морфологической многозначности на базе этих контекстных правил.

2.4. Модуль для создания, редактирования Базы контекстных правил

Основное окно (рисунок 2) состоит из рабочих областей. Каждая область отображает определенную информацию о состоянии базы.

Область №1

В этой области отображается выборка слов из загруженного текста с неоднозначным морфологическим разбором. При выборе правила, каждое слово подсвечивается зеленым цветом, если слово по типу подходит выбранному правилу, иначе – красным цветом.

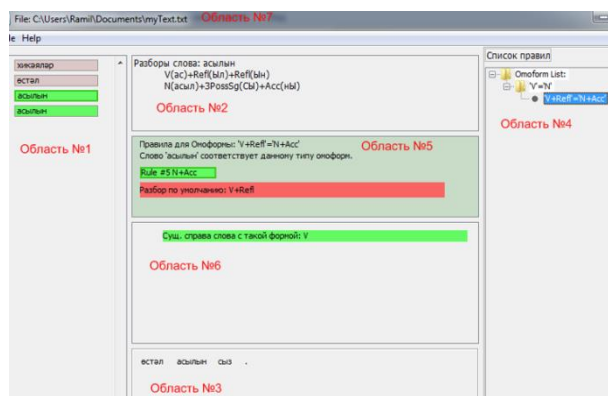


Рисунок 2 – Рабочие области программного инструментария

Область №2

В этой области отображаются разборы выбранного слова (из Области №1 или №5)

Область №3

В этой области отображается предложение, в которое входит выбранное слово. Подводя к слову указатель мыши, можно узнать ее морфологический разбор в Области №2.

Область №4

В этой области отображается текущая База правил. Для того, чтобы загрузить сохраненную базу, необходимо нажать в верхнем меню *Файл*-

>Загрузить Базу омоформ и в появившемся окне выбрать нужный файл с базой правил. Также с помощью *Файл->Загрузить Базу по умолчанию* можно загрузить базу по умолчанию с одним единственным правилом. Для сохранения необходимо открыть *Файл->Сохранить Базу омоформ* и в появившемся окне выбрать папку для сохранения.

Для добавления нового правила, нужно сначала выбрать к какому типу оно будет относиться. Правила привязаны к типам. У каждого типа может быть подтип. База правил достаточно гибкая. После того как определен тип правила, выбирается родительский тип (если это подтип), или корневой элемент (Список омоформ), далее необходимо нажать правую кнопку мыши и в контекстном меню выбрать *Добавить новый тип омоформ*. Затем появится окно с несколькими полями: первый тип разбора, второй тип разбора, тип разбора по умолчанию и ID, которые необходимо заполнить. Для редактирования или удаления правила, необходимо также выбрать нужный тип омоформ и выполнить нужное действие.

Область №5

После выбора типа омоформы, в этой области отображается информация о правилах. Здесь же их можно добавлять, изменять и удалять, нажав на правую кнопку мыши и выбрав соответствующий функционал в контекстном меню (рисунок 3).

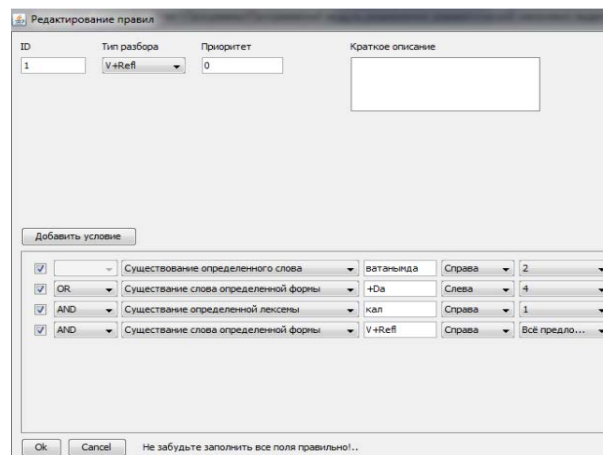


Рисунок 3 – Окно создания и редактирования правила

Кроме правил, в конце списка может располагаться разбор по умолчанию. Выбрав нужный тип омоформ из Области №4, и выбрав нужное слово из Области №1, можем проверить соответствие правила этому слову:

- если область окрашивается в красный цвет (и появляется надпись «Слово не соответствует данному типу»), то это значит, что неоднозначность разбора слова не соответствует данному типу. Иначе неоднозначность соответствует данному типу, т.е. она может быть разрешена с помощью этого правила;

- если одно из правил окрашивается в зеленый цвет, то это означает, что неоднозначность разбора

будет разрешена по данному правилу (также, если нажать на данное правило, то в Области №6 тоже загорятся выполненные атомарные условия)

- если, ни одно из правил не будет окрашено в зеленый цвет, т.е. если ни одно из правил не будет выполнено, то неоднозначность разбора будет разрешена разбором по умолчанию, если разбор по умолчанию отсутствует, то неоднозначность останется неразрешенной.

Область №6

В этой области отображаются атомарные условия правила, выбранного в Области №5. В настоящее время в программе реализованы две разновидности: 1) Проверка существования определенного слова, 2) Проверка существования определенной формы слова

Каждому типу можно добавить дополнительные условия: 1) Контекст проверки (слева, справа или с обеих сторон), 2) Расстояние проверки.

Атомарные условия определяются логическими операциями (AND, OR, AND NOT, OR NOT).

Область №7

Здесь отображается адрес выбранного текста.

2.5. Модуль разрешения морфологической многозначности с использованием Базы контекстных правил

Модуль разрешения морфологической многозначности представляет собой программное приложение, использующее ту же модель, что и модуль для создания, редактирования и тестирования Базы контекстных правил. В отличие от предыдущего модуля, данный модуль предназначен, в первую очередь, для разрешения многозначности в тексте, используя Базу контекстных правил. При отсутствии определенного правила имеется возможность пометить текст вручную. Выходной файл текста сохраняется в том же виде (с той же структурой), что и входной файл.

Модуль состоит из следующих основных разделов (рисунок 4):

1. Область №1 – Основной раздел текста,
2. Область №2 – Раздел Базы контекстных правил,
3. Область №3 – Раздел разбора слова и редактирования форм,
4. Область №4 – Консоль,
5. Область №5 – Титул окна с адресом файла текста,
6. Область №6 – Кнопки базовых действий.

Область №1 – Основной раздел текста

Здесь отображается текст из загруженного файла. Как и в предыдущем модуле структура файла такова, что каждой словоформе соответствуют две строки: 1) сама словоформа, например *өстэл* (стол), 2) Варианты морфологического разбора для данной

словоформы: *V(өстэ (добавь))+REFL(HI):V(өстэл)–добавься/N(өстэл) – стол;*

Загружая такого рода текст, для каждого слова создается объект. Слова собираются в предложения, предложения в абзацы. Таким образом, массив слов отображается в этой области. В отличие от предыдущего модуля здесь присутствуют все слова, независимо от количества разборов и их типов. Это объясняется тем, что, в первую очередь, с текстом будет работать человек, которому для разрешения неоднозначности необходим весь контекст. Для снятия неразрешенных модулем неоднозначностей, пользователю предоставляется возможность проверять правильность и самой Базы контекстных правил. Для удобства работы «слова»разделяются на несколько типов (состояний) и отличаются цветовым оформлением (каждому типу соответствует свой цвет фона).

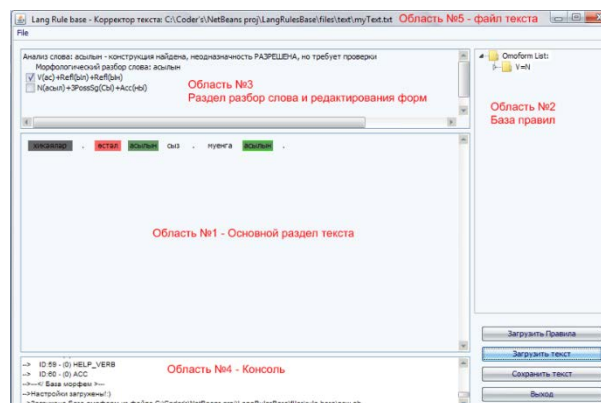


Рисунок 4 – Окно модуля разрешения морфологической многозначности.

Фоновый цвет слов с однозначным разбором совпадает с фоном самого текста. Слова с неразрешенными неоднозначностями обозначаются красным цветом. Разрешенные слова, но не проверенные человеком, светло-зеленым; проверенные (такого же цвета будут и слова, разрешенные вручную) - темно-зеленым. Слова, для которых нет правил, выделяются серым цветом. При нажатии на слово из этой области в Области №3 отображаются формы разбора, а также информация о правиле, по которой его можно разрешить.

Область №2 – Раздел База контекстных правил

В этом разделе отображается список правил, загруженных с файла База контекстных правил, разработанная с помощью предыдущего программного модуля. Однако, в данном случае, правила нельзя редактировать, так как здесь они представлены только в качестве информационного раздела.

Область №3 – Раздел разбора слова и редактирования форм

Как показано выше, в этом разделе отображается форма разбора выбранного слова, а также информация о правиле, по которой его можно разрешить. Кроме того, здесь же представлена и кнопка «Сохранить» (при нажатии на правую

кнопку мыши в этой области), которая сохраняет в качестве правильной ту форму, которая отмечена галочкой (слева в этой же области).

Область №4 – Консоль

Этот раздел носит вспомогательный характер, отражает информацию об истории и стабильности работы, такого типа, как: «Настройки загружены!», или: «Загружена База оморформ из файла C:\Coder's\NetBeans proj\LangRulesBase\files\rulebase\new.ob»

Область №5 – Титул окна с адресом файла текста

Здесь отображается название загруженного файла текста.

Область №6 – Кнопки базовых действий

Базовые действия, такие как «Загрузить правила», «Загрузить текст», функционируют также, как и в предыдущем модуле.

Заключение

В данной работе описан программный инструмент, предоставляющий широкие возможности для создания, редактирования, а также тестирования Базы контекстных правил автоматического разрешения морфологической многозначности. Выходной файл Базы правил в последующем может быть применен также в других программных приложениях для промежуточного разрешения морфологической многозначности в самых разных лингвистических задачах.

Программный инструмент также фиксирует проанализированные слова, их типы и результаты разрешения. В итоге собирается статистика, способная помочь в дальнейших разработках и в исправлениях разного рода ошибок в самих контекстных правилах.

В дальнейшем нами планируется расширение возможностей описанного программного инструментария как за счет добавления новых правил для выявленных и описанных неоднозначностей, так и за счет добавления правил разрешения новых типов неоднозначностей. Потенциал повышения эффективности представленного в данной статье программного инструментария видится также в совмещении различных подходов к разрешению многозначности, когда наряду с контекстными методами используются вероятностные модели и статистические методы.

Библиографический список

[Закиев и др., 1993] Татарская грамматика. Т. II. Морфология / Ред. М.З.Закиев, Ф.А.Ганиев, К.З.Зиннатуллина. – Казань: Татарское кн. изд-во, 1993. – 397 с.

[Сулейманов, 1994] Сулейманов Д.Ш. Регулярность морфологии татарского языка и типы нарушений в языке / Д.Ш.Сулейманов // Когнитивная и компьютерная лингвистика /

Науч. ред. Р.Г.Бухараев, В.Д.Соловьев, Д.Ш.Сулейманов. – Казань: КГУ, 1994. – С.77-106.

[Зализняк, 2004] Анна А. Зализняк. ФЕНОМЕН МНОГОЗНАЧНОСТИ И СПОСОБЫ ЕГО ОПИСАНИЯ. Вопросы языкознания. — М., 2004. — № 2. — С. 20-45.

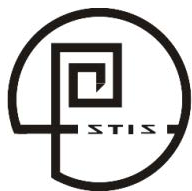
[Невзорова, 2005] Разрешение функциональной омонимии в русском языке на основе контекстных правил // Труды межд. конф. Диалог'2005. – М.: Наука, 2005. С. 198-202.

SOFTWARE TOOL FOR MORPHOLOGICAL DISAMBIGUATION IN THE TATAR LANGUAGE

Suleymanov D.Sh. *, Gilmullin R.A. *, Gataullin R.R. *

**Kazan Federal University,
Research Institute of Applied Semiotics of
Tatarstan Academy of Sciences, Kazan, Russia
dvd.slt@gmail.com
rinatgilmullin@gmail.com
ramil.gata@gmail.com*

The article describes a software tool for creating, editing, and testing base contextual rules for the automatically resolve morphological ambiguity in the Tatar language.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.912

ВОЗМОЖНОСТЬ АВТОМАТИЧЕСКОГО ОПРЕДЕЛЕНИЯ АДРЕСАТА НА ОСНОВЕ СЕМАНТИКО-СИНТАКСИЧЕСКИХ ОСОБЕННОСТЕЙ ТЕКСТА

Глазкова А.В.

*Тюменский государственный университет,
г. Тюмень, Россия*

anna_glazkova@yahoo.com

Решение проблем организации эффективного доступа к информации становится все более актуальным в связи с широким использованием интернет-технологий. Вследствие этого много внимания привлекает к себе задача автоматической классификации текстовых документов. В работе обсуждается возможность автоматической классификации текстов в зависимости от того, какой аудитории они адресованы.

Ключевые слова: классификация текстов; информационные ресурсы; информационный поиск; искусственный интеллект.

Введение

Объем информации в мире постоянно растет, и в связи с этим одним из ключевых направлений современной компьютерной науки является разработка методов систематизации имеющихся данных.

Необходимость интенсивного развития средств информационного поиска – в том числе для работы с поисковыми системами и электронными сообщениями – включает в себя также потребность в улучшении методов и алгоритмов автоматической классификации текстов, что является необходимым условием улучшения результативности обработки текстовой информации.

1. Краткий обзор текущего состояния технологий автоматической обработки текстов

В середине XX века в рамках дисциплины «искусственный интеллект» выделилось направление, связанное с обработкой естественного языка (Natural Language Processing). Трудность задач данного направления заключается в том, что естественный язык представляет собой сложную и многоуровневую систему, постоянно изменяющуюся в процессе жизнедеятельности человека. Многообразие естественных языков, их лексического, морфологического и синтаксического составов, также порождает сложность в создании и реализации методов обработки текстов на естественном языке.

В основе решения задач обработки текстов на естественном языке лежит использование моделей документов, отражающих словообразовательные, грамматические и смысловые особенности текстов [Рафанов, 2010]. Создание этих моделей опирается соответственно на методы морфемного, синтаксического и семантического анализа, методы статистического анализа и морфологические характеристики лексем.

Результаты автоматизации морфологического анализа текста в настоящее время применяются в текстовых редакторах, поисковых системах, модулях проверки правописания. Входными данными морфологического анализа является текстовый документ, выходом – список предложений, каждое из которых представляет собой список слов. Каждое слово, в свою очередь, – список лексем, то есть групп словоформ, соответствующих одной нормальной (словарной) форме слова и одной части речи, а также стоящих в одном числе. На данном этапе помимо грамматической информации для каждого слова запоминается, как правило, его смещение от начала текста и следующий за словом разделитель [Осипов, 2013].

Задачи анализа синтаксического и семантического анализа в полной мере не решены, несмотря на богатую теорию в области возможных путей их автоматизации. Широкое применение находят лишь методы, основанные на статистических характеристиках анализируемого текста (количественной оценке частоты встречаемости тех или иных слов и

словосочетаний). Основной задачей синтаксического анализа является установление различных зависимостей между лексемами, выявленными на этапе морфологического анализа, – в частности, выделение именных групп. Понятие именной группы – словосочетания, в которое входит корневое существительное – используется в задачах классификации текстов, а также при сравнении поисковых образов документа и запроса. Помимо установления зависимостей, синтаксический анализ обнаруживает однородные члены предложения. Таким образом, на этом этапе между лексемами могут быть установлены два вида связей: управление и однородность. Входными данными синтаксического анализа являются, как правило, предложения на выходе морфологического анализа, выходными – предложения в виде списков вариантов, каждый из которых может представлять собой список деревьев зависимостей [Осипов, 2013].

При этом системы, реализующие методы синтаксического анализа документов, не предусматривают в полной мере возможности настройки процесса обработки текста и средств пополнения правил грамматики используемого естественного языка. Область семантического анализа документов нуждается в разработке методов, аналогичных по своей сути методам, применяемым человеком при анализе информации, то есть в создании компьютерных систем, основанных на базах знаний и процедурах логического вывода и принятия решений, которые могли бы эффективно использоваться для задач автоматической обработки текстов [Большакова, 2011] [Мокроусов, 2010]. Разработка семантических моделей необходима для решения различных задач информационного поиска, в том числе и задачи классификации документов.

К основным методам классификации текста на основе обучающей выборки относятся [Маннинг, 2011]:

- вероятностные методы (байесовские – например, наивный байесовский метод на основе мультиномиальной модели или модели Бернулли, методы, оценивающие распределения вероятностей – например, метод максимальной энтропии);
- нечисленные методы (деревья и правила принятия решений);
- методы на основе векторной модели (классификация Роккио и метод k ближайших соседей);
- геометрические методы (метод опорных векторов и его модификации) и другие.

2. Задача классификации текстовых документов

2.1. Постановка задачи классификации

Для представления задачи классификации в общем виде рассмотрим некоторое конечное

множество категорий C и конечное множество документов D :

$$C = \{c_1, c_2, \dots, c_{|C|}\}. \quad (1)$$

$$D = \{d_1, d_2, \dots, d_{|D|}\}. \quad (2)$$

Целевая функция Φ , которая для каждой пары <документ, категория> определяет, соответствуют ли они друг другу, неизвестна:

$$\Phi: D \times C \rightarrow \{0, 1\}. \quad (3)$$

Необходимо найти классификатор Φ' , то есть функцию, максимально близкую к функции Φ [Sebastiani, 2002].

Таким образом, целью классификации текстовых документов является разделение документов из имеющегося множества по классам. Для этого используется обучающая выборка документов и алгоритм обучения, при помощи которого можно получить классификатор, или функцию классификации, отображающую документы в классы. В процессе выполнения классификации необходимо обеспечить достижение высокой точности не только на обучающей выборке, но и на новых данных.

2.2. Применение систем классификации текстов

Системы автоматической классификации текстов используются для решения многих прикладных задач информационного поиска, требующих сопоставления различных типов документов одной или нескольким выделенным категориям на основании содержания анализируемых текстов. К таким задачам относятся:

- поиск в электронных библиотеках и сети Интернет;
- фильтрация почтового спама;
- распределение текстов (например, новостных) по тематическим категориям;
- составление интернет-каталогов;
- подбор контекстной рекламы;
- автоматическое аннотирование и реферирование текстов;
- снятие неоднозначности при автоматическом переводе текстов;
- ограничение области поиска в поисковых системах;
- определение кодировки и языка текста;
- идентификация автора текста;
- жанровая классификация текстов и другие.

3. Возможность определения адресата текста

Одной из задач классификации документов является идентификация автора текста. Развитие вычислительной техники в настоящее время дает возможность не только для реализации таких

традиционных функций, как определение средней длины слов в тексте, подсчет числа слов в предложении, предложений в абзаце и так далее, но и для вычисления более сложных показателей – как на основе статистических данных, так и на основе семантико-синтаксических моделей лексико-грамматических структур документов.

В различных работах предпринимались попытки на основе анализа естественно-языкового текста сделать выводы о его языке, эпохе написания, литературном направлении – в случае, если текст художественный, типе (проза или поэзия), формате (роман, повесть, очерк, эссе), жанре и, наконец, авторе [Орлов, 2012]. В данной работе рассматривается возможность классификации текстов на естественном языке в зависимости от того, кому он адресован, – в частности, в зависимости от возраста аудитории. Решение этой задачи может быть использовано для усовершенствования механизмов информационного поиска, в частности поиска в сети Интернет и библиотеках электронных документов.

Процесс идентификации потенциального адресата текста подразумевает обращение к некому набору «эталонов» – базе знаний, отражающей характерных черты текстов, предназначенных для той или иной категории читателей. Для текста с неизвестной категорией будет требоваться определить его наиболее вероятный класс, то есть соотнести предложенный текст с одним из известных классов или с несколькими из них.

Часть признаков, лежащих в основе классификации, может быть получена, исходя из лексического (словарного) состава текста – подобной характеристикой может быть, например, отношение количества терминов к общему количеству слов. Остальные признаки должны определяться в зависимости от уровня синтаксической и семантической сложности документа. К синтаксическим особенностям, влияющим на классификацию, можно отнести:

- количество сложносочиненных и сложноподчиненных предложений;
- длина предложений;
- наличие обособлений, причастных и деепричастных оборотов;
- длина слов.

Также следует оценить сложность текста, подлежащего классификации, с помощью математических методов, основанных на понятии информационной энтропии [Кутузов, 2010]. Одной из классических работ, посвященных этой проблеме, является статья А. Н. Колмогорова, где описаны комбинаторный, вероятностный и алгоритмический подход к определению понятия «количество информации». На их основе выведена взаимосвязь между сложностью текста и его энтропией [Колмогоров, 1965]. С понятием сложности текста связано вычисление индекса удобочитаемости, определение степени

неоднозначности. Таким образом, для решения задачи потребуется выявление не только лингвистических, но и логических отношений на языковых объектах, то есть семантических связей между ними.

Заключение

В настоящей статье рассмотрена возможность реализации классификации текстов на основе категорий их адресатов. Основанием для классификации могут послужить как данные о лексическом составе текста, так и информация о его грамматической и семантической сложности.

Совершенствование алгоритмов и методов автоматической обработки текстовых документов является актуальной задачей в связи с широким использованием средств обработки информации.

Библиографический список

- [Sebastiani, 2002] Sebastiani, F. Machine Learning in Automated Text Categorization / F. Sebastiani // ACM Computing Surveys. - 2002. - Vol. 34, No. 1.
- [Jurafsky, 2002] Jurafsky, D. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition / D. Jurafsky, J. H. Martin. // New Jersey: Pearson, 2009. – p. 988.
- [Yang, 1999] Yang Y. A re-examination of text categorization methods / Y. Yang, X. Liu // School of Computer Science Carnegie Mellon University Pittsburgh, PA 15213-3702, USA, 1999. – p. 8.
- [Большакова, 2011] Большакова, Е.И. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика: учеб. пособие/ Е. И. Большакова, Э. С. Клышинский, Д. В. Ландэ, А. А. Носков, О.В. Пескова, Е.В. Ягунова // М.: МИЭМ, 2011. – 272 с.
- [Колмогоров, 1965] Колмогоров А.Н., Три подхода к определению понятия «количество информации» / А. Н. Колмогоров // Проблемы передачи информации, 1965, 1:1, С. 3-11.
- [Кутузов, 2010] Кутузов, А. Б. Методики определения сложности текста в рамках переводческого анализа / А. Б. Кутузов // Вестник Нижегородского государственного лингвистического университета им. Н.А. Добролюбова. Вып.4. Лингвистика и межкультурная коммуникация, 2009.
- [Маннинг, 2011] Маннинг К. Введение в информационный поиск. : Пер. с англ. / К. Маннинг, П. Рагхаван, Х. Шютце // М.: ООО «И.Д. Вильямс», 2011. – 528 с.
- [Мокроусов, 2010] Мокроусов, М.Н. Разработка и исследование методов и системы семантического анализа естественно-языковых текстов: автореф. дис. ... канд. техн. наук: / М. Н. Мокроусов // Ижевск, 2010. – 24 с.
- [Орлов, 2012] Орлов, Ю. Н. Методы статистического анализа литературных текстов / Ю. Н. Орлов, К. П. Осминин// М.: Книжный дом «ЛИБРОКОМ», 2012. – 312 с.
- [Осипов, 2013] Осипов, Г. С. Лекции по искусственному интеллекту. Изд. 2-е, испр. и доп. / Г. С. Осипов // М.: Книжный дом «ЛИБРОКОМ», 2013. – 272 с.
- [Рафанов, 2010] Рафанов, С. В. К проблеме классификации текстов в машинном переводе / С. В. Рафанов // Вестник Московского государственного областного университета. Серия: Лингвистика, 2010, № 3, С. 36-42.
- [Семенов, 2002] Семенов, А.Л. Математика текстов / А. Л. Семенов // М.: МЦНМО, 2002. – 16 с.
- [Электронный ресурс] Автоматическая обработка текстов [Электронный ресурс]. – Режим доступа: <http://www.aot.ru/index.html>.
- [Электронный ресурс] Лившиц, Ю. Классификация текстов. Алгоритмы для Интернета / yury.name/internet [Электронный ресурс]. – Режим доступа: <http://yury.name/internet/>.
- [Ягунова, 2012] Ягунова, Е. В. Вариативность структуры нарратива и разнообразие стратегий понимания / Е. В. Ягунова // Человек и язык, 2012, С. 65-83.

THE CAPABILITY OF AUTOMATIC TEXT ADDRESSEE RECOGNITION BASED ON SYNTACTIC AND SEMANTIC FEATURES

Glazkova A.V.

Tyumen State University, Tyumen, Russia

anna_glazkova@yahoo.com

The article deals with the task of natural language text processing. The paper discusses the capability of automatic text categorization based on characteristics of the text addressee.

Introduction

Currently, it is necessary to handle a large amount of information. Therefore, one of the important directions of modern computer science is to develop methods to systematize the available information.

Consequently, there emerged a need to improve the methods and algorithms of information retrieval, including automatic text categorization.

Document categorization methods are used to solve a variety of word processing tasks: filtering and spam recognition, automatic annotation and summarization, news distribution, and other categories.

Main Part

Natural language processing is a branch of science related both to artificial intelligence and linguistics that became a separate discipline in the middle-of the XX century. Natural language is a complex and evolving system. Diversity of natural languages, their lexical, morphological and syntactic structures also creates difficulty in developing and implementing methods for processing natural language texts. The basis of solving natural language processing is methods of morphemic, syntactic and semantic analysis, statistical analysis and morphological characteristics of lexical items. Current document analysis systems and methods of analysis don't give the full opportunity to set up text processing tools and renew grammar rules. Therefore, it's necessary to continue improving of information retrieval systems; this also applies to systems of text categorization.

Text categorization is the task of assigning a Boolean value to each pair $\langle \text{document}, \text{category} \rangle$, where D is a domain of documents and C is a set of predefined categories. More formally, the task is to approximate the unknown target function Φ' called the classifier by means of a function Φ : $\Phi: D \times C \rightarrow \{0,1\}$.

Document categorization is a problem in library science, information science and computer science. The main difficulty of text categorization is complexity and ambiguity of natural language. Automatic document classification techniques include Naive Bayes classifier, k-nearest neighbour algorithms, method of selected points, decision trees, decision rules and others.

Classification techniques have been applied to:

- spam filtering, a process which tries to discern E-mail spam messages from legitimate emails;
- email routing, sending an email sent to a general address to a specific address or mailbox depending on topic;
- language identification, automatically determining the language of a text;
- genre classification, automatically determining the genre of a text;
- readability assessment, automatically determining the degree of readability of a text, either to find suitable materials for different age groups or reader types or as part of a larger text simplification system.

This paper deals with the classification of natural language text, depending on the addressee of the text-in particular, depending on the age of the audience. The solution to this problem can be used to improve information retrieval mechanisms. The process of identifying a potential addressee of the text means an appeal to a set of "standards" – the knowledgebase that reflects the characteristic features of texts intended for a particular category of readers. It is required to assign the text with an unknown category to the most probable class. Some features for text categorization may be derived from the dictionary of the text (the ratio of the number of terms to the total number of words, etc.). Other features should be determined depending on the syntactic and semantic complexity of the document. Syntactic features affecting the classification include:

- complexity of sentences;
- average length of words;
- length of sentences;
- presence of participial constructions and adverb phrases.

Complexity of the text can also be evaluated using mathematical methods based on the concept of information entropy. One of the classic works devoted to this problem is the article of Kolmogorov A. N., which describes a combinatorial, probabilistic and algorithmic approach to the definition of "quantities of information". In this paper is derived the relation between the complexity of the text and its entropy. The notion of complexity associated the text readability index calculation and determination of the degree of ambiguity.

Conclusion

In this paper we consider the capability of realization of the text classification based on categories of recipients. The basis for classification may be data about the lexical composition of text and information about its grammatical and semantic complexity.

Development of algorithms and methods for automatic processing of text documents is an important task of artificial intelligence due to the wide use of information processing facilities.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8 + 004.9

СОЗДАНИЕ ХРАНИЛИЩА НАУЧНЫХ ПУБЛИКАЦИЙ С ПРИМЕНЕНИЕМ МЕТОДОВ КОРПУСНОЙ ЛИНГВИСТИКИ

Елохов Е.С., Югов А.С., Ланин В.В.

*Национальный исследовательский университет «Высшая школа экономики» – Пермь,
г. Пермь, Российская Федерация*

eugene.yelokhov@gmail.com, yugovas@live.ru, lanin@perm.ru

В работе рассматривается подход к созданию удобного хранилища публикаций, основанный на методах поиска и обработке неструктурированных текстов (научных публикаций) с применением программного комплекса GATE.

Ключевые слова: корпус документов; выделение ключевых понятий; аннотирование текста; GATE.

Введение

Число доступных научных публикаций растёт с каждым днем. Это объясняется прежде всего тем, что Интернет сделал доступными большое число работ, растёт число и самих электронных библиотек (РИНЦ, Springer, ACM и т.д.).

В связи с этим все более и более возрастает сложность задачи поиска статей по интересующей исследователей тематике. Для увеличения скорости поиска информация должна быть систематизирована. Распространенной технологией упорядочивания информации является рубрикация документов, то есть описание содержания документа через элементы некоторого замкнутого списка тем – рубрикатора [Петровский, Глазкова, 2007]. В больших информационных коллекциях (от библиотеки статей факультета, которая становится достаточно объемной через несколько лет работы факультета, до крупных хранилищ, электронных библиотек), имеет смысл говорить только об автоматической рубрикации.

Примерами систем рубрикации электронных документов являются Softinform Search Server (<http://www.softinform.ru>) и модуль рубрикации текстов в системе asknet (<http://asknet.ru/>). В большинстве из них применяются статистические методы, методы сравнения с образцом и методы поиска по ключевым словам. Однако статистические подходы малоэффективны по временным показателям, а метод ключевых слов зачастую выдает неполные или, напротив, недостоверные результаты. Становится очевидным, что решение задачи рубрикации невозможно без использования метазнаний.

Особенно актуальными становятся задачи семантического поиска по корпусу документов и автоматической семантической разметки документов для дальнейшего использования этой разметки при поиске.

1. Описание задачи

В рамках проекта по созданию системы поиска и аналитической обработки публикаций по методам и средствам моделирования информационных систем возникла задача выбора инструментальных средств обработки неструктурированных текстов. В качестве требований к таким системам были сформулированы следующие требования: расширяемость, поддержка русского и английского языков, возможность работы с тезаурусами и другими онтологическими ресурсами. После анализа существующих систем выбор пал на программный комплекс GATE (General Architecture for Text Engineering, сайт проекта <http://gate.ac.uk>).

При анализе публикаций с использованием возможностей GATE планируется решить следующие задачи:

- организация аннотированного хранилища ресурсов (статей);
- реализация механизма автоматического выделения ключевых слов текста;
- реализация механизма автоматического структурного анализа публикаций в произвольных форматах (выделение основных структурных элементов статьи: аннотации, введения и т.д.);
- выявление разнообразных связей между публикациями.

Очевидно, что все перечисленные задачи не покрываются функциональными возможностями GATE, поэтому планируется реализовать их путем разработки собственных решений с использованием GATE API.

2. Инструментарий GATE

GATE — это пакет инструментов Java для разработки и развертывания компонентов программного обеспечения, предназначенных для обработки естественного языка. Данная система с открытым исходным кодом подходит для любых операций обработки текстов различных размеров.

Необходимо отметить, что в лингвистических ресурсах, которые используются при работе с GATE, содержится три типа данных: документы, корпуса и аннотации. Аннотации представлены в виде графов, которые моделируются как Java-наборы. Имеется возможность работать с различными форматами документов (XML, RTF, HTML, SGML, текстовый файл). В каждом случае анализируемый текст конвертируется в единую унифицированную модель аннотации. Формат аннотации является видоизмененным TIPSSTER-форматом [Grishman, 1997], который, в свою очередь, совместим с Atlas-форматом [Bird and Liberman, 1999] и использует механизм «отдельно хранимой разметки». Документы, корпус документов и аннотации хранятся в базах данных, визуализируются с помощью инструментов среды разработки. Эти типы данных доступны на уровне кода в фреймворке.

Кроме того, GATE располагает интеллектуальной системой под названием ANNIE (A Nearly-New Information Extraction System), которая включает в себя набор ресурсов, обеспечивающих токенизацию, POS-тэггинг, разбивку на предложения, извлечение именованных сущностей и анализ кореферентности.

Таким образом, GATE предоставляет набор инструментов для первичной обработки текстов на естественном языке, которые позволяют структурировать текст и подготовить его к дальнейшему более сложному анализу.

3. Реализация ИС для хранения и обработки научной литературы

3.1. Создание хранилища документов

Первым этапом обработки корпуса является создание единого хранилища документов и наполнение этого хранилища. На этом этапе необходимо обеспечить удобное и эффективное ведение (хранение и добавление) исходных документов.

Для применения GATE в дальнейшем на этом этапе требуется создать отдельный каталог для хранения рубрицированных заранее экспертами документов.

3.2. Лингвистическая разметка документов корпуса

Лингвистическая разметка — одно из основных понятий корпусной лингвистики. Разметка даёт возможность идентифицировать тексты по различным параметрам, позволяя осуществлять осмысленный поиск по корпусу. Разметка текста заключается в приписывании текстам и их компонентам дополнительной информации (метаданных).

Метаданные должны удовлетворять ряду требований, «семи максима» Д. Лича [Leech, 1993] (Leech's seven maxims of annotation). Разметка должна быть независима от текста: должна быть возможность убрать разметку и просмотреть текст без неё и, наоборот, вычленив только разметку. Принципы разметки и их разработчики должны быть известны конечному пользователю. Пользователь должен быть поставлен в известность о том, что разметка не является безошибочной, а представляет собой лишь потенциально полезный инструмент. В основу разметки должны быть положены общепринятые и, по возможности, теоретически нейтральные лингвистические принципы. И, наконец, ни одна разметка не может априорно считаться стандартом.

Реализацию базовой разметки предоставляет готовый функционал комплекса GATE: токенизация и выделение абзацев, предложений, слов на ее основе; морфо-синтаксический анализ (определение части речи).

Более сложную разметку (библиографические данные, сведения об авторе и т.п.) можно реализовать при помощи доработки инструментов GATE.

3.3. Выделения ключевых понятий в документе

Набор ключевых слов формирует представление о работе в первом приближении и является важной характеристикой, показывающей, стоит ли данную конкретную работу рассматривать более детально или можно сразу сделать вывод о ее нерелевантности. Поэтому важно, чтобы каждый научный текст был охарактеризован таким образом.

Выделение ключевых понятий не входит в базовый набор функционала программной системы GATE, поэтому планируется реализовать дополнительный модуль с использованием GATE API.

На данный момент для поиска ключевых слов используются закономерности, открытые Ципфом в чистом виде (TF-IDF), либо алгоритмы LSI (latent semantic indexing). В рамках текущего исследования остановимся на первом варианте, как на наиболее простом для реализации исследовательского прототипа.

Чаще всего в качестве ключевых слов выступают устойчивые словосочетания [Белоногов и др., 2002],

поэтому расширим выбранный метод выделения ключевых слов на основе частотных характеристик при помощи добавления алгоритма выделения словосочетаний, в основе которого лежит последовательное вычисление частотных характеристик ($L = 2, 3, \dots, L_{max}$) и фильтрация повторяющихся L -грамм по критерию устойчивости [Гусев и Саломатина, 2004].

3.4. Применение GATE для автоматического аннотирования текстов

Как правило, процессы анализа текста состоят из одних и тех же этапов. Ключевым моментом является постановка задачи извлечения, причем аннотаторам (в роли которых выступают люди) следует решить эту задачу подобным образом. Кроме того, необходимо создать высококачественный образец, который задаст направление разработки и измерения результатов автоматизированного анализа. Довольно распространено использование двойных или тройных аннотаций, когда несколько человек занимаются задачей аннотирования независимо друг от друга, после чего измеряется уровень сходства (так называемое «меж-аннотаторское сходство») с целью оценки и контроля качества предоставленных данных.

Рассмотрим основные этапы процесса анализа текста, а также инструменты GATE, которые используются на каждом этапе.

На первом этапе необходимо собрать множество текстов, на основе которых будут созданы аннотации. Примерами таких текстов могут являться научные статьи, истории болезней, технические задания, отчеты по клиническим испытаниям, электронные письма, сообщения из социальных сетей, парламентские акты и т.д.

Далее составляется структурированное описание интересующих пользователя тем текста. Например, описание может быть представлено в виде корпоративной телефонной книги или списка названий препаратов. Таким образом, будет сформирована «черновая» онтология.

После этого нужно точно поставить задачу аннотирования и удостовериться в правильности этой постановки. Рекомендуется использовать инструмент GATE Teamware (или GATE Developer для небольших проектов) для того, чтобы задать «золотой стандарт» набора аннотаций корпуса, относящихся к данной онтологии.

Кроме того, необходимо разработать прототип цепочки процессов анализа текста. Рекомендуется использовать инструмент GATE Developer для построения цепочки процессов с целью автоматизированного аннотирования и оценки результата работы, сопоставляя результат с образцовым набором аннотаций.

Следующим шагом является развертка и удостоверение в работоспособности системы

анализа. Рекомендуется применить цепочку процессов анализа текста к анализируемому корпусу с использованием GATE Cloud или внедрить ее в систему, используя GATE Embedded.

Важным этапом является наполнение сервера индексного поиска информации. Рекомендуется использовать GATE Mimir для хранения аннотаций, относящихся к онтологии в мультипарадигматическом сервере индексного поиска информации.

В заключении необходимо предоставить результаты конечным пользователям одним из перечисленных ниже способов: выгрузить данные для анализа в статистические пакеты, базы данных и т.д.; создать интерфейс пользователя, специфический для этой предметной области, с целью эффективного использования GATE Mimir, или заняться интеграцией в существующую фронтальную систему через веб API со стилем построения REST.

Очень часто используется итеративная разработка (некоторые этапы или группы этапов повторяются) и интеграционное тестирование в рамках agile-методов [Bizer, Heath, and Berners-Lee, 2009].

При использовании предложенного подхода возможен поиск по корпусу документов с использованием аннотаций или онтологий. Кроме того, имеет место процесс поддержки системы и предусмотрена вероятность изменения информационных потребностей пользователя. В каждом из случаев используется справочник или полуавтоматическое аннотирование, а также автоматизированные измерения и регрессионное тестирование (для проверки стабильности существующих результатов анализа или для структурирования проведения анализов в будущем).

Заключение

В данной работе был описан подход к реализации информационной системы, реализующей функции хранилища научных публикаций. Кроме непосредственно функции хранения документов ИС реализует функции формирования метаданных, служащих для повышения скорости и точности поиска как информации по документам, так и самих документов. В качестве метаинформации выступает набор ключевых понятий и аннотация.

В дальнейшем планируется реализация системы для автоматического построения активных ссылок на другие документы (если они тоже находятся в системе) и построение карты публикаций.

Работа выполнена при поддержке Научного фонда НИУ ВШЭ по программе софинансирования грантов РФФИ (проект № 13-09-0143).

Библиографический список

[Белоногов и др., 2002] Белоногов, Г.Г. и др. Автоматический концептуальный анализ текстов // НТИ, 2002. сер. 2. № 10. С. 26–32.

[Гусев и Саломатина, 2004] Гусев В.Д., Саломатина Н.В. Алгоритм выявления устойчивых словосочетаний с учетом их вариативности (морфологической и комбинаторной) // Труды международной конференции Диалог'2004. М.: Наука, 2004. С. 530 – 535.

[Петровский, Глазкова, 2007] Петровский М., Глазкова В. Алгоритмы машинного обучения для задачи анализа и рубрикации электронных документов // Вычислительные методы и программирование. — 2007. — № 8. — С. 57–69.

[Bird and Liberman, 1999] Bird S., Liberman. M. A Formal Framework for Linguistic Annotation. Technical Report MS-CIS-99-01, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA, 1999. <http://xxx.lanl.gov/abs/cs.CL/9903003>.

[Bizer, Heath, and Berners-Lee, 2009] Bizer, C., Heath, T., & Berners-Lee, T. (2009). Linked Data - The Story So Far. International Journal on Semantic Web and Information Systems (IJSWIS), 5(3), 1-22. doi:10.4018/jswis.2009081901

[Grishman, 1997] Grishman. R. TIPSTER Architecture Design Document Version 2.3. Technical report, DARPA, 1997. http://www.itl.nist.gov/div894/894.02/related_projects/tipster/.

[Leech, 1993] Leech, G. 1993. Corpus annotation schemes. Literary and Linguistic Computing, 8(4):275–281.

PUBLICATIONS REPOSITORY CREATION BASED ON CORPUS LINGUISTICS TECHNIQUES

Elokhov E.S., Yugov A.S., Lanin V.V.

*National Research University Higher School of
Economics*

**eugene.yelokhov@gmail.com, yugovas@live.ru,
lanin@perm.ru**

The purpose of this work is to analyze an approach to publications repository creation based on search methods & processing of unstructured data (scientific papers) using GATE, a Java suite of tools.

Introduction

A problem of complicated search for specific scientific papers arises impacting people that intend to use accumulated knowledge. In order to increase data processing speed the data is to be systematized. The common way to achieve it is to implement the process of categorization, so text content is to be grouped into categories. It is worth bearing in mind that an automated process of categorization is needed in large data collections that range from faculty publications to works in Russian Science Citation Index. At present there's a deficit in software tools of e-documents categorization. Most of them use statistical methods, pattern matching techniques and methods of searching by keywords. Nevertheless statistical methods are ineffective in terms of time, and the results of searching by keywords implementation are often incomplete or excessive. It can be clearly seen that the only cure for this problem is using metaknowledge.

Main Part

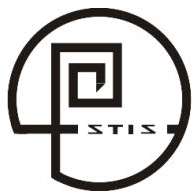
E-documents search and analytic processing system

creation comprises a number of tasks. The principal objectives to be reached are as follows. Firstly, publications warehouse should be created. Secondly, keywords automatic extraction technique is to be implemented. Moreover, publications structural elements extraction technique should be created. In addition, relations between publications must be detected.

In order to achieve this ultimate aim it was decided to use GATE (a Java suite of tools for all sorts of natural language processing tasks, including information extraction). Particularly semantic annotations are going to be used. Keywords extraction is going to be based on Zipf's law extended with collocations extraction. Furthermore, it will be possible to order publications by subject based on keywords extraction and annotations.

Conclusion

The main conclusion to be drawn is that the information system that performs a function of publications repository was described. It also performs functions of metadata collection. Metadata comprises a set of keywords and an annotation. It should be noted that metadata helps to increase a search speed and improve search accuracy. In the future in the context of the system it will be possible to automatically build active links to other documents and to establish a publication scheme.



УДК 004.8 + 004.4

О ПОДХОДЕ К ПОДБОРУ DSL НА ОСНОВЕ АНАЛИЗА КОРПУСА СПЕЦИАЛИЗИРОВАННЫХ ДОКУМЕНТОВ

Валеев М.Т., Елохов Е.С., Узунова Е.Н., Югов А.С.

*Национальный исследовательский университет «Высшая школа экономики»,
г. Пермь, Российская Федерация*

mt.vallev.1992@gmail.com, eugene.yelokhov@gmail.com, palgonuri@gmail.com, yugovas@live.ru

На сегодняшний день многие разработки ведутся с использованием различных DSL. При этом, чтобы использовать DSL, его надо либо создать, либо взять уже существующий. Создание нового языка требует, в большинстве случаев, определенных временных и финансовых затрат, что является экономически невыгодным. Выбор подходящего варианта из уже существующих DSL является трудоемким, т.к. перебор всевозможных DSL и определение того, подходит ли конкретный DSL для решения поставленной задачи выполняется человеком «вручную». Основная причина этой проблемы в том, что не существует хранилища DSL, и не существует инструмента подбора DSL для решения конкретной задачи. В данной статье предложен подход к реализации автоматического определения требований, которым должен удовлетворять DSL (требования формируются в виде онтологии) и автоматического подбора DSL, который в наибольшей мере подходит для решения конкретной задачи.

Ключевые слова: онтология; предметно-ориентированный язык; семантическая близость слов.

Введение

В настоящее время при разработке информационных систем широко используются технологии, основанные на применении метамоделирования и предметно-ориентированных языков (DSL, Domain Specific Languages) [Лядова, 2010]. При этом, DSL создается для решения какой-либо определенной задачи. Практически о любой возникающей задаче можно сказать, что ранее решалась похожая, и, возможно, она была решена. В нашем случае это означает, что необходимый для решения поставленной задачи DSL уже был реализован, либо был реализован некий DSL, но он не полностью удовлетворяет новым условиям. Получается, что можно либо найти уже готовый DSL, либо взять разработанный ранее и внести изменения, что, конечно, требует меньших затрат, чем разработка нового DSL.

Итак, для того, чтобы выбрать один из существующих DSL, необходимо:

1. Определить требования к целевому DSL.
2. Определить, насколько каждый из имеющихся DSL соответствует сформулированным требованиям.

Требования можно определить, проанализировав документы, относящиеся к данной предметной

области или к постановке и решению данной задачи. Требования предлагается представлять в виде онтологической модели, разработанной на основе анализа выбранных документов.

DSL подбирается на основе определения подобия онтологии, составленной пользователем и онтологией, построенной на основе DSL. Очевидно, для того чтобы DSL можно было сопоставить, все языки должны быть описаны в едином формате. В качестве такого формата возьмем представление, используемое в системе «MetaLanguage» [Sukhov and Lyadova, 2012].

В качестве входных данных выступают:

- Корпус документов, относящихся к определенной предметной области.
- Набор описаний DSL.

В качестве выходных данных должен быть получен список DSL, подходящих для решения поставленной задачи, упорядоченный по мере соответствия предъявленным требованиям.

В данной работе рассматриваются методы автоматизации определения требований к DSL и методы определения того, насколько каждый из DSL удовлетворяет предъявленным требованиям.

1. Существующие системы составления онтологий и трансформации моделей

На сегодняшний день имеются информационные системы, которые позволяют отдельно либо составлять онтологические модели документов на основе текстов, либо задавать соответствия онтологических моделей, тем самым преобразовывать одну модель в другую. Удалось найти два веб-ресурса, позволяющих автоматизировать разработку онтологий онтологий: OwlExporter и OntoGrid.

Основная идея OwlExporter состоит в том, чтобы проанализировать текст на естественном языке, составить краткую аннотацию, а затем распределить выделенные понятия по заранее подготовленной онтологии, т.е. «населить» онтологию [Witte и др., 2010]. Таким образом, OwlExporter не создает онтологию на основе текста, а только расширяет, дополняет уже существующую.

OntoGrid – инструментальная система для автоматизации построения онтологий предметных областей с использованием Grid-технологий и анализа текстов на естественном языке [Гусев и др., 2005].

Данная система оснащена двуязычным лингвистическим процессором для извлечения знаний из текстов на естественном языке. В качестве базы для морфологического анализа используется электронный словарь Д. Уорта [Worth и др., 1970]. Она содержит 3,2 млн. словоформ. Процесс индексации включает 200 правил. «Ключевая лексика» выявляется при помощи анализа распределения слов в тексте. Разработчиками предложен новый метод выявления сверхфразовых единств, образуемых сгущениями лексических единиц определенного типа. Построение семантических сетей текстовых документов осуществляется следующим образом: производится анализ текста при помощи соответствующего компонента системы анализа текста, в качестве формализма для представления смысла текста в ней используются семантические Q-сети [Загоруйко и др., 2004]. База лингвистических знаний системы анализа текста представляет собой набор элементарных и составных словосочетаний предметной области. Такую базу условно можно разделить на базу реализаций элементарных отношений (БРО) и набор критичных фрагментов (НКФ), по которым определяется, какие элементы онтологии затрагиваются в данном тексте. Далее происходит создание и развитие онтологии в GRID-сети. Для представления структуры онтологии используется известный стандарт OWL.

Кроме того, было найдено три информационные системы, выполняющих функции преобразования моделей [Сухов, 2012].

Язык трансформации ATL (ATLAS Transformation Language) является частью

архитектуры управления моделью ATLAS [Bezivin, 2005]. ATL – язык, позволяющий описывать трансформации любой исходной модели в указанную целевую модель.

GReAT (Graph REwriting And Transformation) – язык описания преобразований модели, базирующийся на подходе тройных трансформаций графа [14]. Трансформация, описанная на этом языке, представляет собой набор упорядоченных правил перезаписи графа, которые применяются к входной модели и в результате создают выходную модель.

VIATRA – основанный на правилах и паттернах, язык преобразования для управления графовыми моделями, который комбинирует в единую парадигму спецификации два подхода: математический формализм, основанный на правилах трансформации графа, для описания моделей и абстрактные конечные автоматы, предназначенные для описания потока управления [Kagypis и др., 1998].

Описанные выше программные системы либо помогают создавать онтологии, либо преобразуют модели (графы) к заданному виду, т.е. нет системы, которая объединяла бы в себе эти функции. Кроме того, не было найдено информационной системы, которая оценивала бы схожесть преобразованной модели с неким образцом.

2. Описание предлагаемого подхода

Процесс выбора подходящего для решения поставленной задачи DSL состоит из шести последовательных этапов, которые должны быть обязательно выполнены (рисунок 1).

Сначала обрабатывается корпус документов, относящихся к предметной области, с целью выделения ключевых понятий. На втором этапе проводится повторная обработка корпуса документов и задаются связи между концептами. Концепты и связи между ними формируют семантическую сеть. Третьим этапом является устранение синонимии среди концептов. На четвертом шаге сжатая семантическая сеть трансформируется в базовую онтологию при помощи алгоритма огрубления графов. После этого специалист уточняет онтологию (если в этом есть необходимость). И, наконец, из хранилища выбираются DSL, наиболее точно описывающие рассматриваемую предметную область.

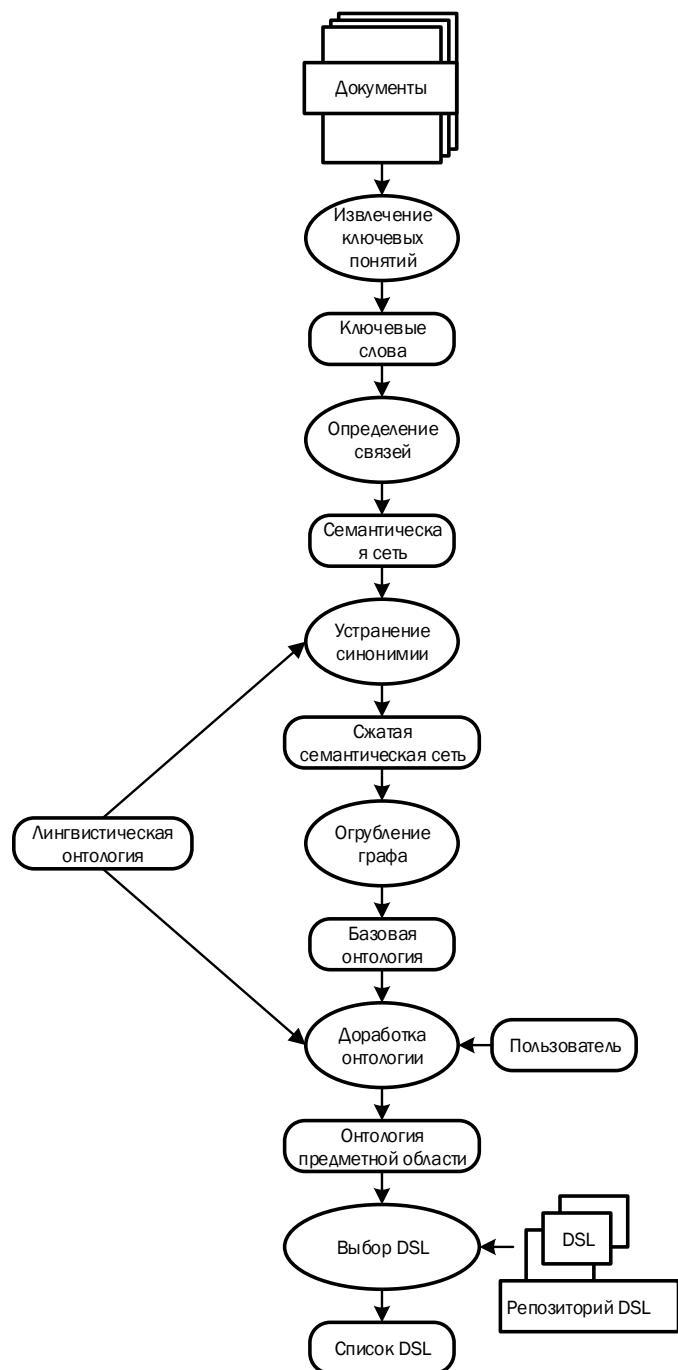


Рисунок 1 – Процесс подбора DSL

2.1. Извлечение ключевых понятий

Одним из наиболее распространенных способов структурирования информации являются онтологии.

Формально онтология определяется как

$$O = \langle X, R, F \rangle,$$

где X — конечное множество понятий предметной области, R — конечное множество отношений между понятиями, F — конечное множество функций интерпретации.

В рамках данной статьи рассмотрим задание множества понятий и множества отношений.

Будем считать, что основными терминами документа являются его ключевые слова, а именно

существительные. Исследования, связанные с алгоритмами выделения ключевых слов документа, активно проводятся в рамках решения задач поиска документов. Все они базируются на частотных законах, открытых лингвистом и филологом Джорджем Ципфом (George Kingsley Zipf). Первый из законов гласит, что произведение вероятности обнаружения слова в тексте на ранг частоты, есть число постоянное. Второй говорит, что частота и количество слов, входящих в текст с этой частотой, также имеют зависимость.

На данный момент для поиска ключевых слов используются закономерности, открытые Ципфом в чистом виде (TF-IDF), либо алгоритмы LSI (latent semantic indexing). В рамках текущего исследования остановимся на первом варианте, как на наиболее простом в реализации. Лингвистическая обработка будет проводиться программными средствами Aot.ru.

Рассмотрим построение онтологии на примере процесса сдачи экзамена. Предположим, с помощью частотного анализа были выделены следующие понятия: программирование, студент, преподаватель, учитель, дискретная математика (рисунок. 2).



Рисунок 2 – Ключевые понятия в процессе сдачи экзаменов

2.2. Определение связей

В результате проведения частотного анализа получим множество несвязанных между собой вершин. Теперь необходимо задать множество связей, т.е. достроить имеющийся несвязный граф до семантической сети.

Семантический граф представляет собой взвешенный граф, вершинами которого являются термины проанализированных документов. Наличие ребра между двумя вершинами означает тот факт, что термины семантически связаны между собой, вес ребра является численным значением семантической близости двух терминов, которые соединяет данное ребро [Гринева и др., 2009].

Величина близости онтологических концептов оценивается по ряду мер:

1. Сходство по Жаккару [Real и др., 1996]:

$$K_J = \frac{c}{a + b - c} \quad (1)$$

Коэффициент Жаккара — бинарная мера сходства, предложенная Полем Жаккаром в 1901 г., где a — количество упоминаний первого понятия, b — количество упоминаний второго понятия, c —

количество совместного упоминания понятий (понятия встречаются в одном контексте).

2. Взаимная информация [Mutual Information]:

$$MI = \sum_{u=\{0,1\}} \sum_{v=\{0,1\}} P(u,v) \log_2 \frac{P(u,v)}{P(u)P(v)} \approx \sum_{u=\{0,1\}} \sum_{v=\{0,1\}} \frac{(u,v)}{N} \log_2 \frac{(u,v)}{(u)(v)} N \quad (2)$$

где u, v – понятия, встречающиеся в документе; (u) – количество употребления понятия u , (v) – количество употреблений понятия (v) , – количество совместного употребления понятий (u, v) .

При вычислении сходства по частотам совместной встречаемости слов u и v в некотором контексте также использовалась как $MI(u, v)$, так и точечная взаимная информация PMI [9]:

$$PMI(u, v) = p\left(\frac{(u, v)}{p(u)p(v)}\right) \quad (3)$$

После подсчета оценок близости концептов различными способами производится усреднение этих оценок [Мисуно и др., 2005]. На основе средних оценок в k вершинам добавляются связи. В итоге получаем семантическую сеть (рисунок 3).

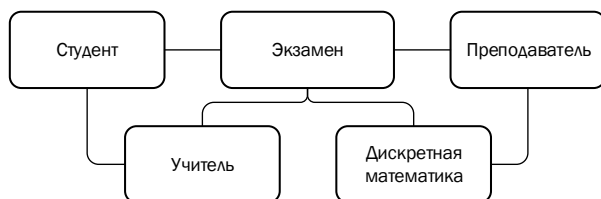


Рисунок 3 – Семантическая сеть понятий процесса сдачи экзамена

2.3. Устранение синонимии

Понятия в тексте могут обозначаться по-разному. В ходе частотного анализа названные иным образом понятия были выделены как отдельные концепты. Таким образом, объединить в одну вершину понятия, являющиеся синонимами.

Синонимия понятий определяется на основе лингвистической онтологии. Мы будем использовать WordNet, семантическую сеть, разработанную в Принстонском университете. Её словарь состоит из 4 сетей для основных частей речи – существительных, глаголов, прилагательных и наречий. Базовая словарная единица – синонимический ряд (синсет), объединяющий слова со схожим значением и, по сути, являющийся узлом сети. Синсеты связываются различными семантическими отношениями, такими, как: гипероним (завтрак → приём пищи), гипоним (приём пищи → обед), has-member (факультет → профессор), member-of (пилот → экипаж), мероним (стол → ножка), антоним (лидер → последователь). Используются различные алгоритмы, например, алгоритмы, учитывающие расстояние между концептуальными категориями слов, учитывающие иерархическую структуру онтологии WordNet.

В рассматриваемом примере понятия «преподаватель» и «учитель» являются синонимами (согласно лингвистической онтологии), поэтому заменяются одной вершиной «преподаватель» (рисунок 4). Вновь созданная вершина содержит связи обеих синонимичных вершин.

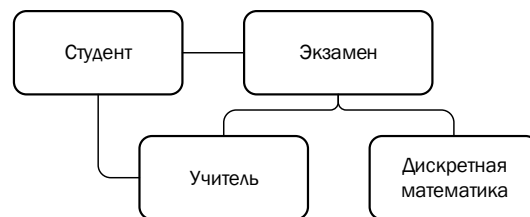


Рисунок 4 – Сжатая семантическая сеть понятий процесса сдачи экзамена

2.4. Огрубление графа

Далее необходимо привести построенную семантическую сеть к онтологической модели. В общей постановке эту задачу следует отнести к задаче огрубления графа [Karypis и др., 1998].

Классические методы решения задачи огрубления графа основаны на итерационном стягивании смежных узлов графа G_α в узлы графа $G_{\alpha+1}$, где $\alpha = 0, 1, 2, \dots$ – номер итерации, $G(0) = G(O)$. В результате этого процесса ребро между двумя вершинами графа G_α удаляется и создается мультиузел графа $G_{\alpha+1}$, объединяющий оба стягиваемых узла [Карпенко, 2010].

Стягиваемые вершины должны иметь общего «родителя». Когда вершины заменяются на одну, то значения, которые имели эти вершины, заменяются значением вершины-родителя из лингвистической онтологии.

В нашем примере «программирование» и «дискретная математика» были объединены в мультиузел «дисциплина» (рисунок 5).

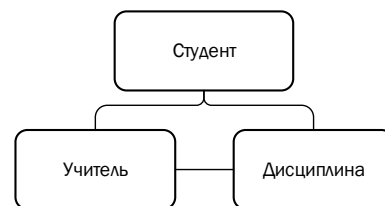


Рисунок 5 – Базовая онтология понятий процесса сдачи экзамена

2.5. Доработка онтологии

В итоге получаем базовую онтологию, представляющую критерии, по которым будет подбираться искомый DSL. Но у построенной модели имеется ряд недостатков:

1. Не отражена семантика связей.
2. Могут присутствовать концепты, которые не важны с точки зрения рассматриваемой задачи (но были выделены в ходе анализа).
3. Могут отсутствовать важные для решения задачи концепты.

Поэтому, пользователю предлагается отредактировать базовую онтологию. Он может задать семантику связей, добавить или удалить концепты по своему усмотрению. Очевидно, чем полнее онтологическая модель, тем точнее будет подобран DSL.

Например, специалист заменил концепт «дисциплина» на «экзамен», удалил связь между студентом и преподавателем, и добавил семантическую значимость связям (студент сдает экзамен, а преподаватель принимает экзамен). Результат действий показан на рисунке 6.

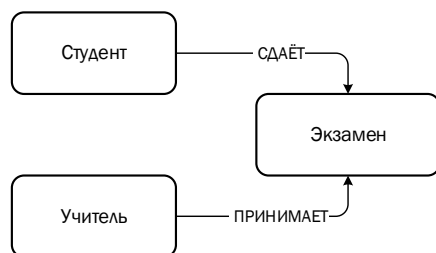


Рисунок 6 – Онтология предметной области «процесс сдачи экзамена»

2.6. Оценка соответствия DSL и созданной онтологии

Сопоставление онтологий сводится к вычислению или выявлению связей или соответствий между понятиями разных онтологий или модулей онтологий, используя различные методы (лексические, структурные и т.д.). Результатом сопоставления является множество соответствий между семантически связанными сущностями.

Мерой соответствия DSL составленной онтологии является показатель изоморфизма графов (насколько графы изоморфны по отношению друг к другу), но с учетом того, что важно рассматривать соответствие не только вершин графов и связей между ними, но и их семантики.

Два графа $(V1; E1; g1)$ и $(V2; E2; g2)$ изоморфны, если существует взаимно однозначное соответствие:

$$f1 : V1 \rightarrow V2 \text{ и } f2 : E1 \rightarrow E2 \quad (4)$$

таким образом, для каждой вершины

$$\begin{cases} a \in E1 \\ g1(a) = x - y \end{cases} \quad (5)$$

тогда и только тогда

$$g2[f2(a)] = f1(x) - f1(y). \quad (6)$$

Далеко не всегда возможно встретить полную изоморфность двух графов. В том случае, если графы не изоморфны, а только похожи, необходимо проверить существует ли соответствие:

$$f: V1 \rightarrow V2, \quad (7)$$

которое представляет соответствие вершин.

Следует отметить, что в нашем случае графы, вероятнее всего, будут неизоморфны. Очевидно, чем

больше «показатель изоморфности» (например, таким показателем может стать число вершин, для которых не удалось установить соответствие, отсутствие или присутствие «лишних» связей и т.п.), тем точнее конкретный DSL описывает предметную область.

Заключение

В рамках данной статьи рассмотрена проблема подбора DSL для решения определенной задачи.

В дальнейшем планируется увеличить количество методов, при использовании которых устанавливаются связи в онтологической модели, для увеличения точности средневзвешенной оценки близости понятий. Кроме того, планируется рассмотреть возможность сравнения DSL не только на одном уровне, но и на разных (добавить возможность сравнивать иерархические структуры).

Работа выполнена при поддержке Научного фонда НИУ ВШЭ по программе софинансирования грантов РФФИ (проект № 13-09-0143).

Библиографический список

- [Гринева и др., 2009] Гринева М., Гринев М., Лизоркин Д. Анализ текстовых документов для извлечения тематически сгруппированных ключевых терминов. [Электронный ресурс]. URL: http://citforum.ru/database/articles/kw_extraction/2.shtml#3.3.
- [Гусев и др., 2005] Гусев В., Завертайлов А., Загоруйко Н. и др. Система «Онтогид» для построения онтологий. [Электронный ресурс]. URL: <http://www.dialog-21.ru/Archive/2005/Zagoruiko%20Gusev%20Zavertailov/Zagoruyko%20NG.htm>.
- [Загоруйко и др., 2004] Загоруйко Н., Налётов А., Соколова А. и др. Формирование базы лексических функций и других отношений для онтологии предметной области. [Электронный ресурс]. URL: <http://www.dialog-21.ru/Archive/2004/Zagorujko.htm>. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.
- [Карпенко, 2010] Карпенко А. Оценка релевантности документов онтологической базы знаний. [Электронный ресурс]. Режим доступа: <http://technomag.edu.ru/doc/157379.html>.
- [Лядова, 2010] Лядова Л.Н. Многоуровневые модели и языки DSL как основа создания интеллектуальных CASE-систем. [Электронный ресурс]. URL: http://www.hse.ru/data/2010/03/30/1217475675/Lyadova_LN_2.pdf.
- [Мисуну и др., 2005] Мисуну И., Рачковский Д., Слипченко С. Векторные и распределенные представления, отражающие меру семантической связи слов. [Электронный ресурс]. URL: http://www.immsp.kiev.ua/publications/articles/2005/2005_3/Misuno_03_2005.pdf.
- [Сухов, 2012] Сухов А. О. Методы трансформации визуальных моделей. [Электронный ресурс]. Режим доступа: <http://www.hse.ru/pubs/share/direct/document/68390345>.
- [Bezivin, 2005] Bezivin J. An Introduction to the ATLAS Model Management Architecture. [Online]. Available: <http://www.ie.inf.uc3m.es/grupo/docencia/reglada/ASDM/Bezivin05b.pdf>.
- [Mutual Information] Mutual Information [Online]. Available: <http://cats.lse.ac.uk/homepages/liam/st418/mutual-information.pdf>.
- [Karypis и др., 1998] Karypis G., Kumar V. Multilevel k-way Partitioning Scheme for Irregular Graphs // Journal of Parallel and Distributed Computing. 1998. Pp. 96-129.
- [Real и др., 1996] Real R., Vargas J., The Probabilistic Basis of Jaccard's Index of Similarity [Online]. Available: <http://sysbio.oxfordjournals.org/content/45/3/380.full.pdf>.
- [Sukhov and Lyadova, 2012] Sukhov A.O., Lyadova L.N. MetaLanguage: a Tool for Creating Visual Domain-Specific Modeling Languages // Proceedings of the 6th Spring/Summer Young

Researchers' Colloquium on Software Engineering, SYRCoSE 2012, Пермь: Институт системного программирования Российской академии наук, 2012. Pp. 42-53.

[Witte и др., 2010] Witte R., Khamis N., and Rilling J. Flexible Ontology Population from Text: The OwlExporter // Dept. of Comp. Science and Software Eng. Concordia University, Montreal, Canada. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2010/pdf/932_Paper.pdf.

[Worth и др., 1970] Worth D., Kozak A., Johnson D. Russian Derivational Dictionary / New York, NY: American Elsevier Publishing Company Inc, 1970.

AN APPROACH TO THE SELECTION OF DSL BASED ON CORPUS OF DOMAIN-SPECIFIC DOCUMENTS

Valeev M.T., Elokhov E.S., Uzunova E.N.,
Yugov A.S.

*National Research University Higher School of
Economics, Perm, Russia*

mt.vallev.1992@gmail.com

eugene.yelokhov@gmail.com

palgonuri@gmail.com

yugovas@live.ru

Today many problems that are dedicated to a particular problem domain can be solved using DSL. Thus to use DSL it must be created or it can be selected from existing ones. Creating a completely new DSL in most cases requires high financial and time costs. Selecting an appropriate existing DSL is an intensive task because such actions like walking through every DSL and deciding if current DSL can handle the problem are done manually. This problem appears because there are no DSL repository and no tools for matching suitable DSL with specific task. This paper observes an approach for implementing an automated detection of requirements for DSL (ontology-based structure) and automated DSL matching for specific task.

Introduction

Nowadays metamodeling and DSL-based technologies (DSL – Domain Specific Language) are widely used in information system developing. DSL is created for solving some specific problem. Almost every arising problem is similar to the one that was solved before. In this case it means that a suitable DSL was already implemented or an implemented DSL does not fully meet the requirements. Therefore, you can either find a ready-to-use DSL or complete and configure a DSL implemented earlier. This requires less costs rather than developing a completely new DSL.

This paper shows generating process of requirements ontology based on domain-specific documents and how a particular DSL meets given requirements.

Main Part

The suggested approach of the DSL selection process consists of six stages that can be described as a series of sequential operations which should be implemented.

Firstly, a corpus of documents is processed. As a result, the key words (concepts related to specific domain) are retrieved. Secondly, when re-viewing the document, the relations between concepts are built. These concepts and relations form a semantic network. The next step is to eliminate synonymy (to merge nodes containing synonymic concepts). In order to achieve this, a linguistic ontology is used. After that, it is necessary to transform –contracted” semantic network into ontology model, using the graph coarsening algorithm with implementing linguistic ontologies. The next step is to qualify the ontology model by a specialist. This step includes concepts editing and relations marking semantically.

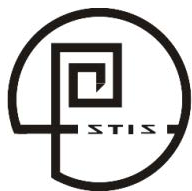
When the ontology is complete, i.e. it meets user requirements, DSLs are taken from the repository, and the measures of DSLs correspondence to ontology requirements are calculated.

Conclusion

In this paper a problem of matching a suitable DSL for specific task was observed.

The requirements for DSL are based on domain documents analysis. Requirements are formed as ontological model which is generated in two steps: defining concepts using frequency analysis of terms found and defining relations based on average weighted score obtained using Jaccard index and mutual information index.

The second step of DSL matching is comparison of DSL's that was implemented earlier with ontology based on domain documents analysis. The core of this comparison is the method of determining graphs' isomorphism and semantic match is controlled by linguistic ontology.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8 + 004.9

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОГО СЕРВИСА АНАЛИЗА ПРЕДЛОЖЕНИЙ НА РЫНКЕ НЕДВИЖИМОСТИ

Анисимова Т.В.^{*}, Нестеров Р.А.^{**}, Печенежский А.Б.^{*}

^{} Пермский государственный национальный исследовательский университет,
г. Пермь, Пермский край*

**hvestya@gmail.com
nextzucker@gmail.com**

*^{**} Национальный исследовательский университет «Высшая школа экономики»,
г. Пермь, Пермский край*

ranesterovhse@gmail.com

В статье описывается реализация интеллектуального сервиса агрегации предложений на рынке недвижимости. При анализе объявлений используется подход на основе онтологий. Набор онтологий, описывающий структуру конкретных сайтов, может быть расширен, что делает возможным использование сервиса для широкого набора источников.

Ключевые слова: интеллектуальный сервис, недвижимость, онтология.

Введение

Специфика работы специалиста по работе с недвижимостью заключается в постоянном анализе информационных потоков, поэтому для успешной деятельности им необходимы средства интеллектуального анализа и мониторинга предложений на рынке. Большая часть такой информации является слабоструктурированной и при традиционном подходе работа с ней занимает значительную часть времени. Источниками информации для риэлтора являются тематические Интернет-ресурсы, бумажные издания и специализированные БД.

Задача агрегации информации из разных источников и ее структуризация является чрезвычайно актуальной. Кроме того, требуются решения задачи устранения дублирования информации и поиска противоречий. Слабая структурированность информации и гетерогенный характер её источников диктуют применение средств и методов искусственного интеллекта для решения данной задачи (например, text mining, технологий Semantic Web и мультиагентные технологии).

Предлагаемым решением описанной задачи является разработка интеллектуального сервиса, собирающего информацию о предложениях на

рынке недвижимости из различных источников в единую базу данных.

1. Классификация агрегаторов предложений на рынке недвижимости

Интернет-ресурсы и сервисы, аккумулирующие существующие предложения на рынке недвижимости, принято называть агрегаторами. Основными характеристиками данных сервисов, влияющими на востребованность пользователями, являются полнота базы объектов, актуальность данных, достоверности информации, возможности поиска и фильтрации и цена доступа [Агрегаторы, 2013].

Существующие на данный момент ресурсы можно классифицировать по двум признакам: территориальному охвату базы объектов и способу организации работы с контентом. В первой классификации выделяют два класса: глобальные, созданные на платформе известного портала («Яндекс.Недвижимость»), и локальные, относящиеся к региональным проектам по недвижимости. Вторая классификация предполагает деление на описанные ниже классы.

Доска объявлений появилась одной из первых. Обычно это бесплатная, тематически организованная база данных. На профессиональном языке это так называемая «грязная» база, т.е.

неорганизованные, практически не регламентированные системы.

Следующим важным агрегатором информации являются *электронные версии печатных изданий частных объявлений*. Одним из главных преимуществ, по оценкам экспертов, позволивших этим ресурсам занять ведущее место в своих рынках, является совмещение концепции газеты бесплатных объявлений с электронной версией.

Мультилистинговые системы являются наиболее популярным и востребованным видом ресурсов среди профессиональных риэлторов. В России на сегодня нет глобального портала, который объединял бы информацию обо всех предложениях на рынке недвижимости.

Информационные порталы по недвижимости или *специализированные сайты* на сегодняшний день – наиболее распространенные агрегаторы информации о недвижимости в сети Интернет.

Социальные сети также относят к агрегаторам информации. Тем не менее, сейчас наблюдается сближение сайтов-агрегаторов и соцсетей с точки зрения общих черт и применяемых сервисов.

Мета-агрегаторы – класс систем, которые объединяют предложения с нескольких ресурсов. Данные сервисы имеют дополнительные функции, например, интеллектуальная фильтрация объявлений только от собственников, а не от посредников.

2. Архитектура сервиса

Для решения автоматического пополнения базы данных объектов недвижимости в рамках проекта по созданию системы автоматизации агентства недвижимости реализован *интеллектуальный сервис*. Работа сервиса заключается в извлечении информации об объектах недвижимости из неструктурированных объявлений, получаемых из различных источников. Решение базируется на онтологическом подходе. Общая архитектура разработанного сервиса представлена на рис. 1.

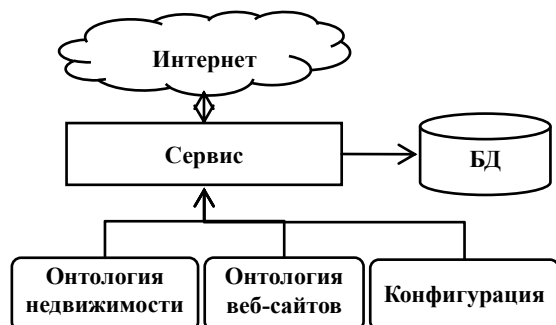


Рисунок 1 – Архитектура сервиса

3. Схема работы сервиса

Общая схема работы сервиса представлена на рис. 2.

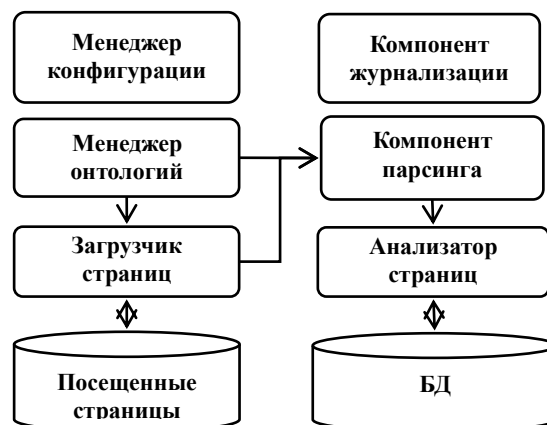


Рисунок 2 – Схема работы сервиса

Компонент журнализации выполняет запись информации о работе сервиса. Полученная с помощью данного компонента информация используется для мониторинга работы сервиса и его отладки.

Менеджер конфигурации предоставляет доступ к переданным сервису настройкам и производит динамическую настройку работы сервиса при необходимости.

Менеджер онтологий реализует операции по управлению онтологическими ресурсами.

Компонент загрузки страниц создает локальную копию исходной страницы, а также производит её предварительную обработку. Информация о пройденных страницах заносится в специальную базу данных, что позволяет оптимизировать работу сервиса за счёт исключения повторного прохода по одинаковым ссылкам в текущую сессию поиска. На базе онтологии сайтов о недвижимости данным компонентом осуществляется извлечение информативной части страницы. Таким образом, на вход *компонента парсинга страниц* фактически передаётся предобработанный текст объявления о продаже недвижимости, из которого за счёт использования онтологии объектов недвижимости извлекаются знания, которые приводятся к некоторому стандартному виду (например, происходит приведение к одинаковым единицам измерения площади и др.).

Компонент анализа страниц осуществляет логический вывод по онтологии объектов недвижимости, основываясь на полученных знаниях, а также проверяет некоторые дополнительные эвристики, после чего формирует сведения об объекте недвижимости, который заносится в соответствующую базу данных.

4. Онтология сайтов о недвижимости

Онтология сайтов о недвижимости хранит специфичные для конкретных сайтов настройки.

Среди интересующих нас параметров выделяют:

- Описание позиции на странице, где вероятнее всего находится интересующая

информация, а также описание, позволяющее получить заголовок этой информации.

- Описание позиции на странице, где могут находиться полезные ссылки.
- Описание фильтров, позволяющих определить «мусорные» для сервиса ссылки.
- Настройки механизма «перелистывания» страниц (описан ниже).

5. Онтология объектов недвижимости и регулярные выражения

Онтология объектов недвижимости содержит некоторые общие понятия предметной области и связи между ними.

В процессе парсинга страниц выполняется попытка «привязать» конкретные понятия, основываясь на знаниях, имеющихся в онтологии. К каждому конкретному понятию в онтологии приписаны определённые регулярные выражения. Выделяются регулярные выражения двух типов: *общие* и *настроенные под конкретный сайт*. Регулярные выражения второго типа могут использоваться для привязки только на конкретных сайтах и в общем случае неверны (такие регулярные выражения позволяют хорошо разбирать используемые на сайте специфические формулировки). Регулярные выражения общего типа построены таким образом, чтобы срабатывать в общих случаях. При привязке конкретных понятий сначала выполняется попытка привязки по второму типу и, в случае неудачи, – по первому.

В структуре регулярных выражений можно выделить два типа элементов: те, что говорят о нахождении совпадения, и те, что свидетельствуют об ошибочной привязке понятия. Например, привязывается понятие «телефон» (то есть к объекту недвижимости проведена телефонная линия), однако в объявлениях часто пишут в конце телефон того, кто подал объявления. Элементы второго типа как раз и выявляют признаки, позволяющие определить, что речь идёт вовсе не об интересующем понятии.

Кроме того, в процессе извлечения знаний об объектах устанавливаются определённые их показатели («Площадь квартиры», «Этаж» и т.п.). Общая структура регулярных выражений в целом аналогична описанной выше, однако дополнительно выделяются логические части, позволяющие, например, выполнить перевод показателей в некоторую единую систему (если была указана цена в тысячах рублей за сотку, то сервис приведёт эту характеристику к унифицированной единице – рубли за квадратный метр и т.п.).

6. Механизм «перелистывания» страниц

В ходе изучения структуры сайтов, содержащих объявления об объектах недвижимости, было выяснено, что зачастую они содержат списки, где

представлены только ссылки на сами объявления. Поскольку объявлений на сайте достаточно много, они могут быть расположены на нескольких страницах, переход по которым обычно осуществляется кнопками навигации.

Разработанный механизм «перелистывания» страниц осуществляет последовательный переход по страницам со списками. Необходимые для этого настройки индивидуальны для каждого сайта и хранятся в онтологии сайтов о недвижимости.

Особо стоит отметить, что некоторую сложность представляет переход по ссылкам, где информация подгружается с помощью javascript. На исследуемых сайтах данная проблема была решена посредством использования особых классов.

7. Настройки сервиса и список загрузок

Настройки сервиса содержат параметры, отвечающие за работу сервиса. Среди параметров можно выделить:

- Путь к списку загрузок, где указаны адреса, которые будет сканировать сервис.
- Путь в файловой системе, куда будут сохранены выгруженные страницы.
- Период, через который сервис возобновит свою работу (остановка сервиса может быть связана с тем, что он пройдёт по всем указанным в списке закладки адресам).
- «Глубина сканирования сайтов» – длина пути, на которую переходит сервис по ссылкам.
- Флаг, указывающий, разрешать ли сервису переход на сторонние сайты по ссылкам при поиске «в глубину».

8. Используемые программные и инструментальные средства

Сервис был разработан в среде Microsoft Visual Studio 2010 на языке программирования C#. Онтология разрабатывалась в редакторе онтологий Protégé. Использовались также библиотеки HtmlAgilityPack (для осуществления парсинга html-страниц) и OwlDotNetApi (для чтения онтологии из файла).

9. Тесты и оценки

Сервис демонстрирует достаточно высокие показатели точности. Адекватно распознаются примерно 97% объявлений. В 93% случаев точно распознается атрибутивный состав объявления. Точность распознавания может быть повышена за счёт настройки онтологии на формат представления объявления на конкретном сайте. Кроме того, в компонент логирования встроены средства анализа, объясняющие причину неудачного сопоставления и предоставляющие рекомендации по настройке онтологии.

Заключение

На данный момент реализован действующий прототип сервиса интеллектуального анализа и агрегации предложений на рынке недвижимости, архитектура которого и особенности реализации описаны выше. Одна из ключевых особенностей сервиса заключается в возможности его настройки на анализ новых ресурсов без изменения программного кода – настройка осуществляется через редактирование онтологии. В рамках проекта на основе информации, хранящейся в системе, планируется также реализовать экспертную систему по подбору объектов недвижимости и их оценке.

Библиографический список

[Segaran, 2009] Segaran T., Evans C., Taylor J. Programming the Semantic Web, O'Reilly Media, 2009.

[Яндекс, 2013] Что такое Яндекс.Недвижимость <http://help.yandex.ru/realty/>.

[Агрегаторы, 2013] Недвижимость online: агрегаторы <http://media-office.ru/?go=2082914&pass=f79e9c77f077cfd060a615834c3c2d1>

INTELLIGENT SERVICE FOR ANALYSIS OF REAL ESTATE MARKET OFFERS

Anisimova T.V. *, Nesterov R.A. **,
Pechenezhskiy A.B. *

**Perm State National Research University,
Perm, Russia*

**hvestya@gmail.com
nextzucker@gmail.com**

***National Research University «Higher School of
Economics», Perm, Russia*

ranesterovhse@gmail.com

This article contains the implementation description of a real estate market offers aggregator service. Advertisement analysis is made with the aid of ontologies. A set of ontologies to describe specific websites can be extended, so the aggregator can be used for many diverse resources.

Introduction

Real estate agents constantly analyze different information flows, so real estate market offers intellectual analysis and monitoring services are required for their efficient work. Most of this information is semistructured and its conventional processing is time-consuming. Real estate information resources are topical Internet resources, papers and special databases.

Information aggregation and structuring tasks are increasingly timely. Apart from that, it is necessary to address information duplication and contradiction search tasks. Semistructured information and its heterogeneous resources implies application of artificial intelligence means: text mining, Semantic Web technologies and multi-agent technologies.

Our solution is to develop intelligent service to accumulate real estate market offers information from different resources in a single database.

Main Part

First of all, we have categorized Internet resources aggregating information on real estate market offers: bulletin boards, electronic versions of free advertisements newspapers, electronic versions of free advertisements newspapers, multilisting systems, real estate information portals, social networks, meta-aggregators.

As for the service, to address automatic real estate items database in the context of a real estate agency automation system project a smart service was implemented. It extracts real estate items information from unstructured advertisements placed on different resources. The solution is based upon an ontological approach using a general real estate ontology, ontology of websites and settings for service configuring.

The service was developed using Microsoft Visual Studio 2010 and C# programming language. Ontology was developed in Protégé ontology editor. Also, HtmlAgilityPack (for html-pages parsing) and OwlDotNetApi (for reading ontologies from a file) libraries were used.

The service demonstrates rather high accuracy rates. Approximately 97 per cent of all advertisements are recognized in an adequate way. In 93 per cent of the time advertisement attributes are recognized precisely. Recognition accuracy can be improved with the aid of adjusting ontology to the specific resource advertisement representation. Besides, logging component include analytical tools to find a reason for a fail correlation and to recommend on required ontology settings.

Conclusion

In this paper we described the architecture and peculiar implementation properties of the real estate market offers aggregator service. At the moment the pilot system of the service is implemented. One of the core features of this service is that it can be adjusted to new resources analysis without changing program code; configuration is only about ontology editing. Also, in the context of this project and on the basis of the information kept in the system, we intend to develop an expert system on real estate items selection and estimating.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.522

ПОТРЕБНОСТЬ В РЕЧЕВЫХ ФУНКЦИЯХ ИНТЕЛЛЕКТУАЛЬНЫХ САПР

Абрамов Г.А., Боргест Н.М., Коровин М.Д.

*Самарский государственный аэрокосмический университет имени академика С.П. Королева
(национальный исследовательский университет), г. Самара, Россия*

borgest@yandex.ru

maks.korovin@gmail.com

belg.ru@mail.ru

В работе рассмотрены основные принципы построения голосовых интерфейсов автоматизированных систем, вопросы распознавания и синтеза речи. С целью увеличения производительности труда и снижения временных затрат на обучение сложных программных приложений обосновывается необходимость применения голосовых технологий на примере разрабатываемого интеллектуального помощника проектанта в предметной области «самолетостроение».

Ключевые слова: голосовой интерфейс; распознавание; синтез; помощник проектанта; голосовые подсказки; голосовое управление.

Введение

Современные темпы развития пользовательских приложений приводят к усложнению программных интерфейсов. Наиболее ярко эта тенденция проявляется в «тяжелых» САД-системах, на освоение которых требуются значительные временные затраты. Одним из подходов решения указанной проблемы является применение справочных систем, построенных на основе речевых технологий. Внедрение в пользовательские инженерные инструменты «голосовых помощников» способствует наилучшему восприятию передаваемой информации и уменьшает время, необходимое на подготовку пользователя и на принятие решений. Реализация голосовых технологий в рабочем инструменте способствует увеличению производительности системы человек-машина.

1. Голосовой интерфейс

Пользовательский интерфейс представляет собой совокупность средств и методов, при помощи которых пользователь взаимодействует с различными машинами, устройствами и аппаратурой [Фисун, 2004].

Голосовой интерфейс – это разновидность пользовательского интерфейса, в которой обмен информацией проходит с помощью речи [Фролов, 2003]. Такой вид взаимодействия очень

привлекателен в различных областях, для которых характерна высокая интенсивность обмена данными между пользователем и информационной системой (ИС). Наиболее полно воспринимается та информация, которая передается одновременно несколькими методами, например визуальным и звуковым. Такой подход помогает обеспечить малые сроки освоения пользования системой и повысить эргономичность интерфейсов ИС в широком спектре предметных областей.

2. Краткая история развития речевых технологий

Работы по созданию систем автоматического распознавания речи ведутся с конца 50-х годов.

В 60-х годах появляются технологии, которые позволяют использовать имеющиеся голосовые метки для выполнения конкретных задач.

В 80-е годы компании Bell Laboratories и IBM приступили к исследованиям в направлении N-грамм (непрерывных последовательностей из N элементов заданного текста или речи), и спикерозависимых (зависимых от свойств звукового входного сигнала) тренируемых систем.

В 90-е годы начинает развиваться технология нейронных сетей. Данный подход увеличивает степень точности распознаваемой речи и ведет к ускорению обработки информации [Petsche, 1998].

В 2006 году Microsoft выпускает операционную систему с полноценной поддержкой функции распознавания речи Windows Vista.

В 2011 году была разработана программа Siri – самообучаемый персональный помощник с голосовым интерфейсом.

Активное внедрение элементов голосового управления в популярные пользовательские приложения позволяет надеяться на скорую готовность речевых технологий к применению в более сложных предметных областях, например, в тяжелых CAD.

3. Распознавание речи

Для успешной реализации голосового интерфейса в определенной предметной области, необходимо принимать во внимание наличие специфической лексики. Уровень развития вычислительной техники и речевых технологий в настоящий момент не позволяют пока реализовать диалог на естественном языке между ИС и пользователем. Это обусловлено такими факторами, как:

- ограниченность словаря ИС;
- особенности дикции оператора и различные скорости произношения;
- недостаточное качество работы аппаратных интерфейсов ввода речевой информации;
- наличие посторонних шумов, влияющих на точность получения команд;
- отсутствие на сегодняшний день технологий семантического анализа текста, готовых к применению в пользовательских приложениях.

Для устранения ограничений, связанных с возможностью некорректной выдачи и восприятия информации, были разработаны специальные языки, например, упрощенные технические языки, главной целью которых является обеспечение однозначности передаваемой информации [Степура, 2013]. Примером такого языка является Simplified Technical English (STE) - упрощенный технический английский язык (<http://www.simplifiedenglish.net/>) или УТР – упрощенный технический русский (<http://www.technorus.ru/>). Переход к упрощенному языку обеспечивает снижение погрешности машинного восприятия текста на 20-25%. STE был разработан при участии компании Boeing для сокращения номенклатуры и содержимого выпускаемой документации, обеспечения полного и однозначного понимания информации.

При использовании упрощенных языков, положительный эффект достигается за счет:

- использования слова только в одном определенном смысле;
- использования простых предложений;
- запрещения в простом предложении постановки подлежащего в конец предложения;
- запрещения изменения инвертированного порядка частей составного сказуемого;

- запрещения постановки прямого дополнения перед сказуемым;
- запрещения использования причастных оборотов в качестве синтаксических оборотов.

Использование упрощенного технического языка при работе с голосовым интерфейсом ИС позволяет повысить качество работы за счет уменьшения количества ошибок распознавания.

На сегодняшний день существуют два принципиально отличающихся метода распознавания звукового потока. Это метод распознавания голосовых меток и метод распознавания лексических элементов.

Первый подход предполагает распознавание фрагментов речи по заранее записанному образцу. Этот подход широко используется в системах, предназначенных для исполнения заранее записанных речевых команд.

Второй подход реализуется выделением из потока речи лексических элементов — фонем и аллофонов (реализация фонемы, её вариант, обусловленный конкретным фонетическим окружением), которые затем объединяются в слоги и морфемы. Применение этого подхода в интерактивных системах осложняется его высокой ресурсоемкостью.

Вне зависимости от реализуемого подхода, распознавание состава речи происходит в три этапа:

- выделение лексических элементов речи;
- выделение слогов и морфем;
- выделение слов, предложений и сообщений.

После завершения распознавания речи, система обрабатывает полученные команды и, в случае необходимости, синтезирует голосовые комментарии.

4. Синтез речи

Системы синтеза речи подразделяется на 4 основных типа [Феррейра, 2012] :

- параметрический синтез – синтез, где речевой сигнал представляется набором небольшого числа непрерывно изменяющихся параметров. При данном подходе, выходной сигнал, в зависимости от количества параметров, может быть очень высокого качества. Недостатком является невозможность применения параметрического синтеза в случаях произвольных, не подготовленных диалогов;
- конкатенативный или компиляционный (компилятивный) синтез – синтез, который сводится к составлению сообщения из предварительно записанного словаря исходных элементов. Размер элементов синтеза не меньше слова;
- синтез по правилам – синтез, реализуемый путём моделирования речевого тракта человека. В процессе синтеза речи фонемы выводятся последовательно через определённый временной интервал;
- предметно-ориентированный синтез –

синтез, который компилирует слова, записанные заранее, а также фразы для создания полных речевых сообщений. Этот тип синтеза используется в приложениях, где многообразие текстов системы будет ограничено определенной темой.

5. Применение речевых технологий в автоматизированных САПР

В силу того, что разрабатываемый помощник проектанта реализуется в узкоспециализированной предметной области, в частности, предварительное проектирование самолета, функциональный лексикон пользователя ограничен тезаурусом предметной области.

Тезаурус является одним из действенных инструментов описания отдельных предметных областей. Он состоит из контролируемого, но при этом изменяемого словаря терминов, между которыми устанавливаются смысловые связи. Возможность формализации семантики речевых команд через взаимосвязь лексических единиц с множеством структурированных понятий предметной области является отличительной особенностью тезаурусов от толковых словарей. В тезаурусе описание понятий предметной области осуществляется систематизировано и направлено на выявление отношений иерархического и неиерархического типов. Таким образом, можно сказать, что тезаурус обеспечивает удовлетворение одного из важных требований при использовании САПР в практике проектирования – наличие полной информации о проектируемом объекте. Тезаурус является важным информационным ресурсом при обработке и анализе данных речевой системой [Боргест, 2012].

В настоящий момент речевая подсистема в интеллектуальном помощнике проектанта исполняет роль интерактивного тьютора. Задачей речевой компоненты является озвучивание для

пользователя алгоритма работы с графическим интерфейсом помощника, а так же принятых в автоматическом режиме ключевых решений на этапе предварительного расчета проекта самолета. Это позволяет упростить контроль над системой и повысить удобство работы пользователя.

Перспективным направлением развития речевого интерфейса интеллектуального помощника проектанта видится реализация голосового управления на основе тезауруса предметной области и статистической базы данных голосовых команд, что позволило бы освободить пользователя от необходимости ручного ввода параметров задачи.

Исходя из этого, можно утверждать, что для автоматизированной системы поддержки принятия конструкторских решений целесообразно применить предметно-ориентированный синтез.

Учитывая сложность интерфейса автоматизированных приложений ведущие разработчики САПР ставят перед собой задачу разработки системы, способной общаться с пользователем на тривиальном языке, что позволило бы свести к минимуму время, необходимое для освоения интерфейса программы. Исследователи из Университета Гонконга предложили схему реализации общения между CAD системой и проектантом, представленную на рисунке 1 [Kou, 2010]. Авторы этого исследования предлагают создать базу знаний, специфических для предметной области слов и выражений, характерных для конечных пользователей. Такой подход позволит повысить точность распознавания, так как система анализирует характерные фразы, которым назначены конкретные действия. Успехи, полученные при применении системы, демонстрируют перспективность такого подхода в специализированных системах, предназначенных для работы в областях, где традиционно часто применяются специальные термины и жаргонизмы.



Рисунок 1 - Модель голосового управления CAD системой на тривиальном языке

Заключение

Усложнение графических интерфейсов современных CAD продуктов вследствие расширения их функциональных возможностей диктует необходимость выработки новых подходов к реализации взаимодействия между системой и пользователем. Одним из наиболее перспективных способов снизить время, потребное для освоения ИС, является внедрение речевых технологий, способных к интерактивной выдаче справочной информации, а в перспективе – к реализации диалога с пользователем на естественном языке.

Библиографический список

- [Боргест и др., 2012] Боргест Н.М., Чернов Р.В., Шустова Д.В. Разработка интерфейса интеллектуального помощника проектанта / Н.М. Боргест, Р.В. Чернов, Д.В. Шустова // материалы международной научно-технической конференции OSTIS-2012. БГУИР. 2012
- [Соловов, 2006] Соловов А.В. Электронное обучение: проблематика, дидактика, технология / А.В. Соловов // Самара: «Новая техника», 2006. – С. 464
- [Степура, 2013] Степура Л. В. Автоматическое реферирование текстовой информации на основе моделирования ситуативных связей между понятиями предметной области / Л.В. Степура // Информатика, 2013, № 39
- [Сулейманов, 2011] Сулейманов Д.Ш. Двухуровневый лингвистический процессор ответных текстов на естественном языке // Материалы международной научно-технической конференции OSTIS-2011, Минск, БГУИР, 2011
- [Феррейра, 2012] Феррейра Опасо Е.В. Интерактивный голосовой интерфейс / Е.В. Феррейра Опасо // <http://sibac.info/index.php/2009-07-01-10-21-16/3955-2012-10-02-04-01-00/> / актуально на 02.10.2012
- [Фисун и др., 2004] Фисун А.П., Гращенко П.А. Теоретические и практические основы человеко-компьютерного взаимодействия: базовые понятия человеко-компьютерных систем в информатике и информационной безопасности / А.П. Фисун, П.А. Гращенко // Орел, ОГУ, 2004. – С. 169.
- [Фролов и др., 2003] Фролов А.В., Фролов Г.В. Синтез и распознавание речи. Современные решения / А.В. Фролов, Г.В. Фролов // <http://www.frolov-lib.ru/books/hi/index.html/> / актуально на 15.10.2013
- [Kou, 2010] Kou X.Y., Xue S.K.. Knowledge-guided inference for voice-enabled CAD / X.Y. Kou, S.K.. Xue // Tan Computer-Aided Design Volume. – 2010 - P. 545-557.
- [Markowitz, 2000] Markowitz A.J. Using speech recognition / A.J. Markowitz // Prentice Hall PTR. - 2000.
- [Petsche, 1998] T. Petsche, A. Marcantonio, C. Darken, S. J. Hanson, G. M. Huhn, and I. Santoso, "An autoassociator for on-line motor monitoring," in Industrial Applications of Neural Networks, F. F. Soulie and P. Gallinari, Eds, Singapore: World Scientific, 1998, pp. 91–97.

THE NEED FOR VOICE CONTROL IN INTELLECTUAL CAD

Abramov G.A., Borgest N.M., Korovin M.D.

*Samara State aerospace University
(national research university),
Samara, Russian Federation*

borgest@yandex.ru

maks.korovin@gmail.com

belg.ru@mail.ru

The paper discusses the basic principles of voice interfaces implementation into high-tech automated systems. The principles of speech recognition and synthesis are described. The use of voice technology in complex user applications is given. Increase in productivity and time spent on learning to use the software reduction is described based on the example of the developed intelligent designer's assistant in the preliminary design of an aircraft domain.

Introduction

The current pace of development of custom applications dictates the increase of their UIs complexity. One of the solutions to this problem is to use a voice technology based assistant. Introduction of custom engineering tool "voice assistant" promotes the best perception of the transmitted information and reduces the time needed to educate the user.

Main Part

The implementation of voice technology into the intellectual designers assistant is described:

- Advantages and disadvantages of voice technology in CAD software is given.
- Trend for the increase of voice systems usage in software interfaces is observed.
- Speech recognition systems are discussed from the point of their applicability to intellectual designers assistant.
- Speech synthesis systems are examined for their applicability to intellectual designers assistant.

Conclusion

The increasing complexity of modern CAD software due to the advance in their functionality calls for a new approach to UI. One of the most promising technologies that can drastically decrease the amount of time, needed to educate the user is the speech-aided UI element implementation. It can serve as an interactive reference system and potentially might be able to support a machine-user dialog using trivial language.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 025.4.03

РЕАЛИЗАЦИЯ ОНТОЛОГИЧЕСКОЙ МУЛЬТИАГЕНТНОСТИ ПРЕДМЕТНОЙ ОБЛАСТИ СРЕДСТВАМИ РЕЛЯЦИОННОЙ СУБД НА ПРИМЕРЕ ЗАЧИСЛЕНИЯ АБИТУРИЕНТОВ В УНИВЕРСИТЕТЫ РОССИИ

Боргест Н.М., Лысаковский И.А.

*Самарский государственный аэрокосмический университет имени академика С.П. Королева
(национальный исследовательский университет), г. Самара, Россия*

borgest@yandex.ru

rxnexus@gmail.com

Статья посвящена актуальной для российских ВУЗов проблеме – набору нового контингента, в частности, процедуре отбора и зачисления на первый курс университета. На основе онтологического анализа предметной области предложен алгоритм решения поставленной задачи средствами реляционной СУБД в объективно существующей мультиагентной среде. Приведена схема реализации программы автоматизированного зачисления абитуриентов в университеты России.

Ключевые слова: мультиагентные технологии; абитуриент; университет; автоматизированное зачисление.

Введение

Использование информационных технологий в приемных кампаниях высших учебных заведений (ВУЗ) обуславливается потребностью повысить эффективность управления ВУЗа, сделать ВУЗ успешной организацией, реализующей важную социальную функцию по подготовке высококвалифицированных специалистов и научных кадров. В условиях складывающейся демографической ситуации, меняющейся структуры рынка труда, возникают задачи внедрения инновационных методов, моделей, алгоритмов и специального программного обеспечения открытой архитектуры для принятия решений. Целью внедрения новых методов управления документооборотом потока абитуриентов в работе приемных комиссий ВУЗов является сокращение ошибок и конфликтных ситуаций при зачислении в ВУЗы, а также минимизация времени сопровождения каждого абитуриента при минимальном числе персонала, задействованного в приемной кампании [Николаева, 2008], [Кириллова, 2010].

Реализация задач, связанных с обработкой больших объемов информации в слабоалгоритмизированных областях, все чаще осуществляется с применением мультиагентных технологий. При таком подходе решение получается в результате взаимодействия множества самостоятельных целенаправленных программных

модулей (агентов) [Городецкий и др. 1998], [Скобелев, 2002], [Боргест, 2012].

1. Онтология предметной области

Онтология рассматриваемой предметной области (ПрО) включает в себя такие важные ключевые сущности как *абитуриент* и *университет*. Абитуриент и университет представляют собой классы сущностей, которые в данной ПрО имеют множество экземпляров, исчисляемых сотнями (университеты) и сотнями тысяч (абитуриенты). Причем, под университетом рассматривается конкретная специальность, на которую желает поступить абитуриент.

У каждой из сущностей имеются свои *Потребности* и *Возможности*, которые формируют ПВ сеть. Специальности в университетах имеют ограниченное (плановое) количество мест в каждом конкретном ВУЗе и ограниченное (минимальное) значение проходных баллов Единого государственного экзамена (ЕГЭ) на каждую из специальностей. Абитуриент имеет результаты ЕГЭ и должен определиться с будущей специальностью (ВУЗом). Для этого ему требуется формализовать свои предпочтения. Минобрнауки России допускает возможность подачи документов в пять университетов и три специальности в каждом из них. Представленная задача в такой постановке является типично мультиагентной, решение которой предполагает применение соответствующих методов поиска

матчинга, т.е. соотнесение потребностей и возможностей каждой из рассматриваемых сущностей.

Однако задача может быть значительно упрощена, если к началу матчинга исходные данные по потребностям и возможностям рассматриваемых сущностей будут определены. Т.е. изменение данных в сцене в процессе выполнения матчинга у различных экземпляров сущностей не происходит. Это условие характерно для данной ПрО, т.к. приемная кампания проходит одновременно во всех ВУЗах страны. Существенным элементом полноты данных является формализация предпочтений абитуриента. Для этого авторами предложен организационно-правовой механизм заполнения документа в виде листка приоритетов абитуриента (ЛПА), где представлены все желаемые варианты для поступающего в ВУЗ.

Для реализации поставленной задачи использовалась СУБД MS Access, с помощью которой был создан ряд таблиц, запросов, соответствующих форм и отчетов. Скомпилированы программные модули – агенты, каждый из которых реализует свою функцию обработки данных. На рис. 1 приведена онтологическая IDEF5 схема, на которой показана трансформация изменения статуса сущности Абитуриент, когда каждый из них проходит «сито» отбора и соответствие потребностей и возможностей каждой из сущностей ВУЗ-Абитуриент, когда фактически определяется удовлетворение всех условий матчинга и осуществляется прием абитуриентов в университеты на выбранные специальности. На рисунке 1 показаны следующие функции, выполняемые в процессе матчинга:

A1 – проверка баллов абитуриентов по основным предметам ЕГЭ на ограничение по минимальному порогу;

A2 – суммирование баллов абитуриентов по основным предметам и дополнительному предмету;

A3 – сопоставление специальности в ЛПА со специальностью ВУЗа;

A4 – сравнение наличия сданных абитуриентом предметов ЕГЭ с предметами ЕГЭ, необходимыми для выбранной специальности;

A5.1 – выбор абитуриентов с 1-ым приоритетом для конкретной специальности;

A5.N - выбор абитуриентов для конкретной специальности с N-ым последующим приоритетом ($2 \leq N \leq 15$);

A6.P – сортировка абитуриентов по убыванию баллов в рамках рассматриваемого приоритета и суммирование их количества ($1 \leq P \leq 15$);

A7.1 – сравнение количества абитуриентов с количеством мест на исследуемой специальности по 1-ому приоритету. Нахождение разницы для «отсечения», в случае превышения количества абитуриентов числу мест в ВУЗе на данную специальность, или дальнейшего заполнения вакантных мест абитуриентами, не прошедшими в выбранные специальности с другими приоритетами;

A7.T – выбор абитуриентов для заполнения оставшихся свободных мест по специальности.

На рисунке 2 приведена схема связи таблиц и данных в разработанной программе.

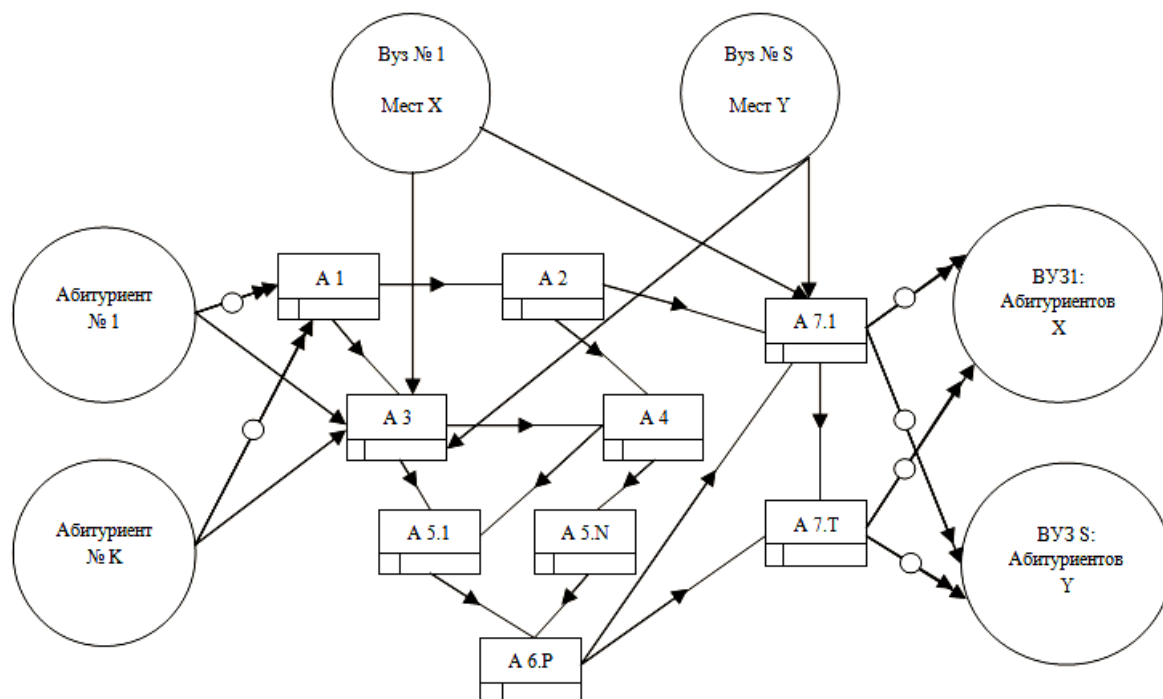


Рисунок 1 – IDEF5 схема ПрО ВУЗ-Абитуриент

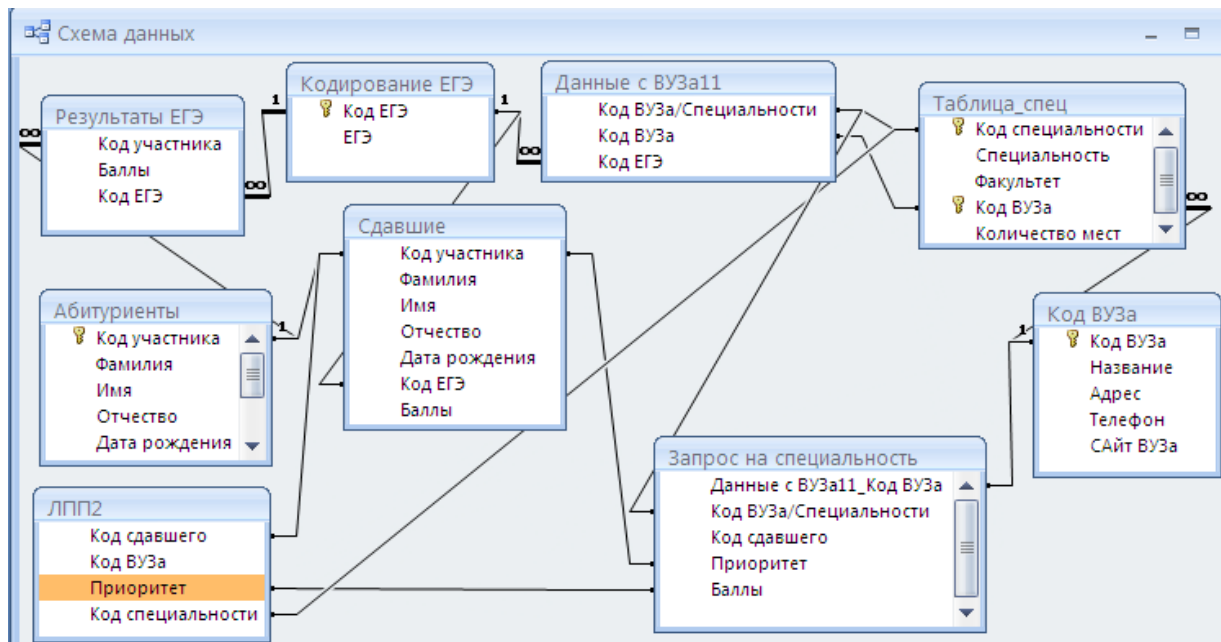


Рисунок 2 – Схема связи таблиц в базе данных

Таблицы в схеме данных:

Абитуриенты - множество абитуриентов, где Код участника – является уникальным номером абитуриента, заполняется автоматически нарастающим итогом при регистрации нового участника;

Результаты ЕГЭ – содержат баллы, полученные абитуриентом по сданным им предметам;

Сдавшие - множество абитуриентов, набравших баллы выше минимального порога;

Код ВУЗа – данные о ВУЗе;

Таблица_спец - множество специальностей всех ВУЗов с количеством плановых мест для приема;

Данные с ВУЗа11 – связь потребных предметов ЕГЭ для поступающих на конкретную специальность;

ЛПП2 - приоритеты абитуриентов;

Кодирование ЕГЭ – множество предметов, используемых для сдачи ЕГЭ.

Запрос на специальность – сопоставление данных по абитуриентам и ВУЗам.

2. Мультиагентность

Характерными особенностями программных агентов являются [Городецкий и др., 1998]:

- коллегальность, т.е. способность к коллективному целенаправленному поведению в интересах решения общей задачи;
- автономность, т.е. способность самостоятельно решать локальные задачи;
- активность, т.е. способность к активным действиям ради достижения общих и локальных целей;

- информационная и двигательная мобильность, т.е. способность активно перемещаться и целенаправленно искать и находить информацию, энергию и объекты, необходимые для кооперативного решения общей задачи;

- адаптивность, т.е. способность автоматически приспосабливаться к неопределенным условиям в динамической среде.

Эти возможности кардинально отличают мультиагентные системы от «жестко» организованных систем.

Большинство современных программных систем характеризуются отсутствием средств идентификации новых потребностей и возможностей, позволяющих оперативно принимать эффективные решения. Типичными примерами событий, вызывающих необходимость заново идентифицировать потребности и возможности, являются появление новых данных или изменение уже принятых. Чем чаще случаются новые события, тем ниже эффективность существующих систем, не способных самостоятельно принимать решение и автоматически перестраиваться под изменения в среде [Скобелев, 2002].

Система распределения абитуриентов по ВУЗам с учетом их желания и их результатов ЕГЭ, основанная на применении мультиагентных технологий, является эффективным решением проблемы зачисления абитуриентов. Данные абитуриентов и ВУЗов накапливаются и обрабатываются в единой базе данных. В ней создаются связи ВУЗ – специальности – абитуриенты. Когда все данные занесены в базу – запускается процесс матчинга. После его окончания, когда становится известен результат, абитуриенты по СМС могут узнать, что они зачислены в тот или иной ВУЗ на выбранную ими

специальность, и уже фактическим студентам необходимо принести в приемные комиссии подлинники документов.

Система максимально открыта для ВУЗов и абитуриентов. Доступ в систему происходит через Интернет с использованием общих сетевых служб. Все абитуриенты могут вносить данные о себе одновременно и независимо друг от друга. При этом в программе происходит параллельная обработка информации по ВУЗам и по абитуриентам. Каждый абитуриент может максимально использовать свои шансы на поступление, анализируя все данные ВУЗов в одной системе и оценивая свои шансы и сверяя их со своими предпочтениями.

3. Автоматизация зачисления

Специалисты ВУЗов совместно с Минобрнауки заносят в базу данных информацию об имеющихся в ВУЗах специальностях, предметах ЕГЭ и количестве мест для приема по каждой специальности. Персональные данные по выпускникам школ, включая полученные баллы по результатам сдачи ЕГЭ, имеются в базе данных Минобрнауки. Выпускники, желающие поступить в ВУЗ, после сдачи ЕГЭ заполняют Лист приоритета абитуриента (ЛПА). Примерная форма ЛПА приведена на рисунке 3. В соответствии с Приказом Министерства образования и науки Российской Федерации «Об утверждении Порядка приема граждан в образовательные учреждения» выбор абитуриента ограничен пятью ВУЗами и тремя специальностями в каждом ВУЗе. Абитуриент ориентируется на качество ВУЗа и ожидаемую

вероятность поступления в него, проставляя приоритеты по выбору специальности от 1 до 15.

Данные каждого абитуриента, добавляемые в систему, обрабатываются агентами по установленным критериям единого конкурсного приема в ВУЗы России.

Список всех подавших заявления на каждую специальность сортируется по сумме баллов ЕГЭ. При этом проверяется количество баллов, полученных по основным предметам (русский язык и математика) на минимальный порог. Абитуриенты, набравшие по какому либо из этих предметов меньше установленного количества баллов, в конкурсе не участвуют.

Количество абитуриентов на каждую специальность ограничивается числом мест, выделенных ВУЗу.


При анализе данных по заполненным приоритетам абитуриентов после осуществления матчинга можно определить конкурс в каждом ВУЗе на каждую специальность. Это позволит определить наиболее востребованные специальности и специальности ВУЗов не пользующиеся спросом, т.е. те, на которые существует недобор абитуриентов.


Система формирует отчет, показывающий распределение абитуриентов по специальностям ВУЗов с учетом максимально набранных баллов и увеличения приоритетов.

Блок-схема работы алгоритма приведена на рисунке 4.

Лист приоритета абитуриентов

Код участника:

Фамилия: 

Имя: 

Отчество:

Дата рождения:

Код	Название	Код спе	Специальность	Приоритет
103	Самарский государственный архите	010900	Прикладные математика и	1
106	Самарский Государственный Эконо	022000	Экология и природо-пользс	2
105	Самарский Государственный Униве	030300	Психология	3
102	Поволжский государственный униве	031300	Журналистика (Журналисти	4
101	Самарский государственный аэроко	035000	Издательское дело	5
101	Самарский государственный аэроко	080100	Экономика	6
102	Поволжский государственный униве	080500	Бизнес-информатика (Элек	7
101	Самарский государственный аэроко	160700	Двигатели летательных апп	8
102	Поволжский государственный униве	200700	Фотоника и оптоинформати	9
*				

Рисунок 3 – Лист приоритета абитуриента

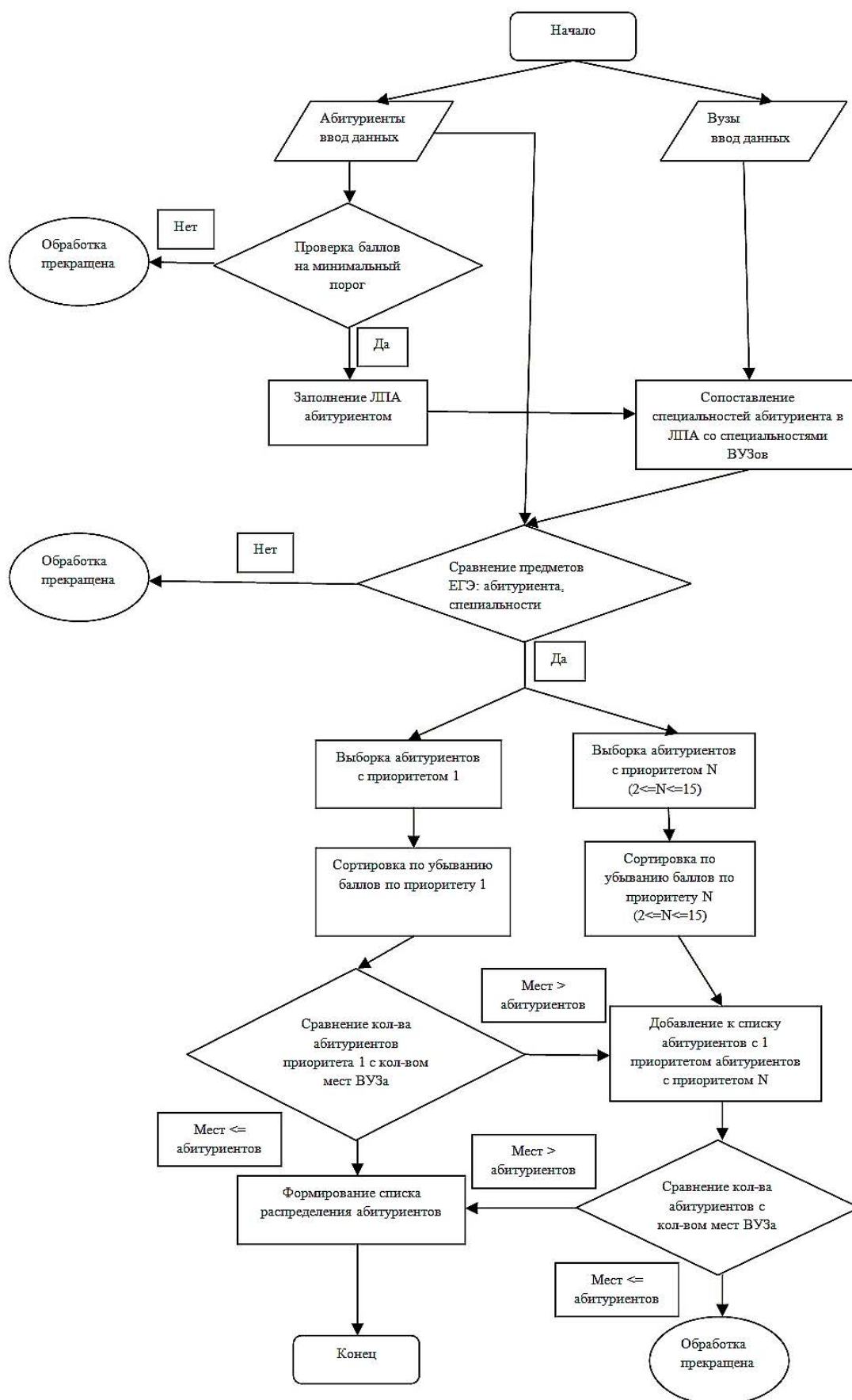


Рисунок 4 – Блок схема алгоритма работы программы зачисления абитуриентов в ВУЗы

ВУЗ	Специальность	Факультет	Баллы	Абитуриент	Количество мест
101 Самарский государственный аэрокосмический университет					
<i>Инженерно-технологический факультет (№4)</i>					
150400	Металлургия				10 мест
Приоритет 1					8 чел.
			270	(163) Антипов Василий Сергеевич, 08.06.1994	
			267	(333) Беркаев Иван Сергеевич, 25.11.1994	
			252	(1518) Бурсов Виталий Сергеевич, 22.02.1998	
			246	(1162) Зубарев Валерий Сергеевич, 03.03.1997	
			219	(146) Лыжанов Василий Сергеевич, 22.05.1994	
			216	(316) Лужьянов Иван Сергеевич, 08.11.1994	
			201	(1501) Барабаш Виталий Сергеевич, 05.02.1998	
			189	(248) Пугин Иван Сергеевич, 01.09.1994	
Приоритет 2					2 чел.
			240	(1514) Бурик Виталий Сергеевич, 18.02.1998	
			216	(145) Сибряев Василий Сергеевич, 21.05.1994	
<i>Итого абитуриентов по специальности</i>					10 чел.

Рисунок 5 – Фрагмент списка абитуриентов, прошедших по конкурсу на одну из специальностей в один из ВУЗов России (пример)

4. Заключение

На основе онтологического анализа Про разработана система автоматизированного зачисления абитуриентов в ВУЗы России на основе ЕГЭ. Тестирование системы на модельном примере показало ее работоспособность.

В результате автоматизированного зачисления каждый ВУЗ сможет получить свой список абитуриентов, прошедших по конкурсу на выбранные специальности (рисунок 5).

Библиографический список

- [Боргест и др., 2012] Боргест, Н.М., Кристалович, И.В., Куликов, П.С.. Автоматизация зачисления в ВУЗ: мультиагентный подход /Искусственный интеллект. Интеллектуальные системы. ИИ-2012: материалы Международной научно-технической конференции - Донецк: ИПШ «Наука і освіта», 2012, - 312 с. - С.292.
- [Городецкий и др. 1998] Городецкий, В.И., Грушинский, М.С., Хабалов, А.В. Многоагентные системы (обзор)// Новости искусственного интеллекта. №2, 1998 - с.64-116.
- [Кириллова, 2010] Кириллова С. Компьютерная программа вместо приемной комиссии. Газета «Первое сентября». №22, 2010.
- [Николаева, 2008] Николаева Н. Как повысить шансы на поступление? Автоматизированная система зачисления абитуриентов. <http://www.afportal.ru/teacher/instruction/odds>.
- [Скобелев, 2002] Скобелев П.О. Открытые мультиагентные системы для оперативной обработки информации в процессах принятия решений// Автоматизация. №6, 2002 - с.45-61.

REALIZATION ONTOLOGICAL MULTIAGENT OF SUBJECT AREA USING RDBMS BY EXAMPLE ADMISSION IN RUSSIAN UNIVERSITIES

Borgest N.M., Lysakovkiy I.A.

Samara State Aerospace University named after academician Korolev S.P. (National Research University), Samara, Russia

borgest@yandex.ru

rxnexus@gmail.com

The article is devoted to the problem of Russian universities - set a new contingent, in particular, the selection procedure and admission to the first year. A scheme of the program aided gaining admission.

Keywords: multi-agent technology; entrant; university; automated enrollment.

Use of information technology in foster campaigns driven by higher education institutions need to improve governance, to make a successful university organization carries out an important social function in preparing graduates and highly qualified scientific personnel.

Implementation of the tasks associated with processing large amounts of information in weak algorithmic areas is increasingly carried out with the use of multi-agent technologies. With this approach, the decision is obtained as a result of the interaction of multiple independent targeted software modules (agents).

Using this scheme for the implementation of this task may actually implement automatic enrollment of students in higher education institutions in accordance with their wishes and exam scores.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.02 : 004.9

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ВСПОМОГАТЕЛЬНЫХ АЛГОРИТМОВ ДЛЯ ПОСТРОЕНИЯ КЛАСТЕРНЫХ СТРУКТУР

Горюнов В.А., Жукевич А.И., Родченко В.Г.

*Гродненский государственный университет имени Янки Купалы,
г. Гродно, Республика Беларусь*

v.gorunov@grsu.by

san@grsu.by

rovar@mail.ru

При построении систем распознавания использование кластерных структур позволяет формализовать процедуру представления образов и эталонов классов в многомерном признаковом пространстве и в автоматическом режиме выполнить процедуры обучения и классификации. Процесс построения кластерных структур связан с использованием библиотеки программных процедур, реализующих вспомогательные алгоритмы кластеризации, вычисления метрик и межкластерных расстояний, критериев однородности.

Ключевые слова: интеллектуальная система, распознавание образов, кластерный анализ, алфавит классов, словарь признаков, классифицированная обучающая выборка.

Введение

В данной статье представлено описание и программная реализация вспомогательных алгоритмов для построения кластерных структур. Использование кластерных структур для формального представления образов классов при построении систем распознавания позволяет в автоматическом режиме на основе исходных данных классифицированной обучающей выборки (КОВ) реализовать процедуры обучения и окончательного принятия решений. [Родченко, 2006].

Формально весь процесс распознавания может быть реализован в результате выполнения следующей последовательности преобразований:

$$S \xrightarrow{F_1} C \xrightarrow{F_2} A \xrightarrow{F_3} T \xrightarrow{F_4} A^* \xrightarrow{F_5} E \xrightarrow{F_6} R$$

где S – алфавит классов; C – словарь наблюдаемых (измеряемых) характеристик; A – априорный словарь признаков (АСП); T – классифицированная обучающая выборка; A^* – уточненный словарь признаков для построения пространства решений; E – множество эталонов классов; R – множество решений; F_1 – алгоритм получения наблюдаемых характеристик; F_2 – алгоритм построения априорного словаря признаков; F_3 – алгоритм формирования классифицированной обучающей выборки; F_4 – алгоритм сепарирования признаков из априорного словаря по степени их

информативности для построения пространства решений; F_5 – алгоритм построения образов эталонов классов в пространстве решений; F_6 – алгоритм процедуры принятия решения [Родченко, 2012].

Алфавит классов S , словарь наблюдаемых характеристик C и априорный словарь признаков A вырабатываются совместно экспертами в соответствующей прикладной области и аналитиками. Классифицированная обучающая выборка T формируется на основе наблюдений состояний объектов с учетом определенных S , C и A на этапах F_1 и F_2 .

На основе данных КОВ и критериев однородности происходит сепарирование признаков из АСП и далее при отсутствии сдвигов среди признаков, выполняется построение кластерных структур объектов входящих в КОВ.

Признаки, значения которых по конкретному критерию оказались малоинформативными с точки зрения разделения образов классов исключаются из АСП. При этом выполняется исключение из КОВ строк соответствующих признакам, которые не влияют на процесс кластеризации. Однако можно отклонить предложение системы и приступить к формальному построению кластерных структур.

Классифицированная обучающая выборка фактически представляет собой прямоугольную матрицу числовых значений, которую можно

рассматривать и как один сплошной массив данных, не учитывая принадлежность отдельного объекта к конкретному классу. Программа позволяет в этом случае аттестовать АСП и произвести разбиение на классы, что достигается применением одного из алгоритмов кластеризации и таким образом получить рабочую КОВ. После чего результаты работы программы можно сохранить в файл, отредактировать, дополнить и т.п.

В случае, когда исходная КОВ – уже классифицирована, применяем алгоритмы аттестации АСП для сепарирования признаков по степени информативности и в результате получаем уточненный словарь признаков. Далее применяем алгоритмы кластеризации для аттестации КОВ на предмет пригодности при применении в алгоритме принятия решения. Для аттестации КОВ необходимо задать пороговое значение Q количества ошибочных включений векторов в состав другого класса. Если количество ошибок превышает Q , то предлагается возвратиться к самому началу алгоритма и сформировать новый вариант АСП.

В обоих случаях в финале приходим к реализации алгоритма F_5 , в результате которого получается E – множество эталонов классов. Принятие решения по распознаваемому классу происходит на основании исследования принадлежности каждого объекта этого класса к одному из классов множества E или же распознаваемый класс выделяется в отдельный джoker-класс.

1. Реализация процедуры построения эталонов классов в пространстве решений.

Для построения эталонов классов в пространстве решений формально необходимо построить кластерные структуры, разбив все объекты классифицированной обучающей выборки на группы, используя значения их признаков, рассматривая КОВ, как сплошной массив данных, ещё не разбитый предварительно на классы.

Если же КОВ представляет собой множество классов, состоящих из объектов признакового пространства, то для аттестации признаков предполагаем, что все признаки информативны – необходимо выполнить построение кластерных структур для объектов входящих в КОВ.

Аттестация признаков реализована посредством построения кластерных структур для заданной КОВ без учета отношения объектов к классу. Затем осуществляется проверка на предмет ошибочного включения объектов в классы исходной КОВ. Если количество ошибок не превышает заданного порогового значения, то признаки считаются пригодными для построения эталонов образов классов.

После выполнения этого этапа работа с КОВ и

АСП одинакова для обоих вариантов и на основе КОВ $T=\{X_1^*, X_2^*, \dots, X_k^*\}$ формируются эталоны классов $E=\{E_1, E_2, \dots, E_k\}$, где эталон i -го класса

$$E_i^T = (e_{1i}, \dots, e_{pi}) \text{ и } e_{ij} = \frac{1}{m_j} \sum_{j=1}^{m_j} x_{ij}^*, \text{ где } X_i^* (i=\overline{1, k})$$

– уточненные классы КОВ, после выполнения процедуры сепарирования априорного словаря признаков, p – количество признаков, попавших в уточненный словарь.

Сепарирование признаков априорного словаря производится на основе применением непараметрических критериев однородности. При отсутствии сдвига для очередного проверяемого признака по отношению ко всем остальным признакам, делается вывод о его пригодности для включения признака в уточненный словарь.

Для реализации процедуры обучения в автоматическом режиме разработана программа, которая реализована в виде многомодульного Windows-приложения. Модули выполнены по принципу плагинов, представленные в виде отдельных динамически подключаемых библиотек.

Существует несколько типов модулей которые реализуют алгоритмы вычисления метрики и межкластерных расстояний, которые в свою очередь используются в алгоритмах кластеризации, а так же модули проверки критериев однородности.

На данный момент реализованы алгоритмы кластеризации:

- FOREL [Загоруйко, 1999];
- FOREL-2 – модификация алгоритма FOREL, позволяющая строить заданное количество кластеров [Загоруйко, 1999];
- объектная кластеризация [Жукевич, 2010];
- равномерная кластеризация;
- равномерная кластеризация-2 – модификация алгоритма равномерной кластеризации, позволяющая строить заданное количество кластеров.

А так же модули с реализацией алгоритмов вычисления метрики (евклидова метрика, взвешенная евклидова метрика, метрика Махаланобиса, метрика Хэмминга); вычисления межкластерных расстояний (методы дальнего и ближнего соседа, метод центра тяжести, метод средних расстояний); критерии однородности (Е-критерий, критерий Хаги) [Кобзарь, 2006].

По мере написания новых алгоритмов кластеризации, признаков однородности, используемых метрик, методов расчета межкластерных расстояний, подключение их к программе происходит динамически.

Критерии аттестации АСП также реализованы в виде плагинов. Выбор нужных алгоритмов осуществляется посредством индексированных выпадающих списков (рисунок 1).

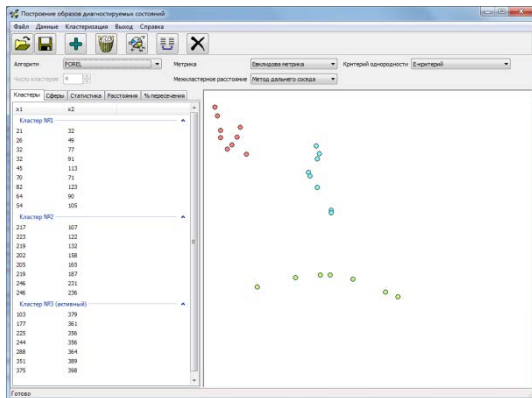


Рисунок 1 – Главное окно программы

Для случая размерности АСП равного двум (два признака) возможен интерактивный ввод данных с использованием манипулятора типа «мышь» или загрузка данных из файла в формате Microsoft Excel. В этом случае в правой рабочей области программы осуществляется наглядная визуализация результатов выполнения программы (рисунок 2).

Все имеющиеся алгоритмы кластеризации используют гипершары, из объединений которых строятся кластера. Поскольку визуализация предусмотрена только для случая двумерных векторов, то гипершары в этом случае представляют собой круги. Границы кругов изображены тонкими линиями. Контуры объединений кругов, принадлежащих одному кластеру, обведены более толстой линией. Области пересечения кластеров заштрихованы.

Каждому кластеру присваивается свой цвет, который генерируется автоматически. Границы кругов принадлежащих каждому кластеру, контуры кластера и точки, которые относятся к этому кластеру, рисуются одним и тем же цветом. Цвета подбираются автоматически так, чтобы они были как можно более хорошо различимы.

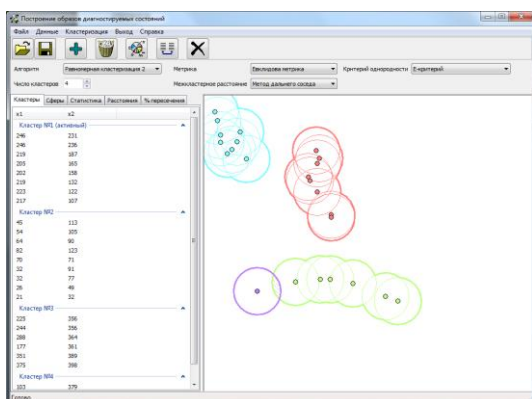


Рисунок 2 – Визуализация данных

На рисунке 2 представлены результаты кластеризации введенной КОВ для трех классов объектов. Для определения кластеров применили алгоритм равномерной кластеризации 2, с использованием евклидовой метрики

$$d_E(X_i, X_j) = \sqrt{\sum_{k=1}^p (x_i^{(k)} - x_j^{(k)})^2}, \text{ межкластерное}$$

расстояние вычисляется по методу дальнего соседа

$$\rho_{min} = \max_{X_i \in S_k, X_j \in S_m} d(X_i, X_j). \text{ Для проверки}$$

информативности признаков из АСП использовался непараметрический Е-критерий. На входе была загружена КОВ состоящая из 3-х классов, а на выходе получили 4-кластерную структуру, при этом два первых исходных класса образовали два отдельных кластера, а третий класс был разделен на два дополнительных класса. Результат работы алгоритмов построения кластерных структур существенно зависит от выбора используемой метрики.

Если объекты КОВ представляют собой векторы размерности более двух, то её загрузка в память, осуществляется только из файла в формате Microsoft Excel. При этом не проводится визуализация данных, но все характеристики построенных кластеров будут представлены на соответствующих закладках рабочего поля программы (рисунок 3).

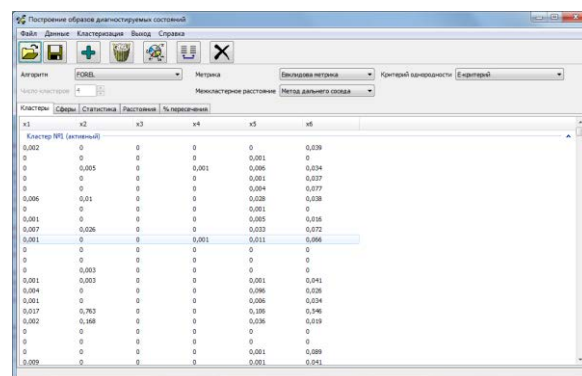


Рисунок 3 – Отображение данных

В остальном работа программы для векторов большей размерности ничем не отличается.

Результаты работы алгоритма построения кластеров и все их числовые характеристики можно сохранить в файл и в последующем, загрузив его, применить иной алгоритм кластеризации или иную метрику, алгоритм подсчета межкластерного расстояния, критерий однородности.

По внутренней структуре программный код реализации в полной мере использует архитектуру языка Object Pascal. В основе лежит обобщенная библиотека классов, которая должна быть одинакова и в главном модуле программы и в подключаемых библиотеках. Классы обобщенной библиотеки являются предками всех реализованных в программе классов, вызываемые методы которых в обобщенной библиотеке абстрактны и не имеют своей реализации, зато позволяют использовать принцип полиморфизма технологии объектно-ориентированного программирования.

2. Реализованные алгоритмы построения кластерных структур.

Опишем в общем виде реализованные в библиотеке алгоритмы построения кластерных структур.

2.1. Реализация алгоритма FOREL

Данный алгоритм кластеризации автоматически определяет количество формируемых кластеров.

В результате работы этого алгоритма получаются кластера сферической формы. Количество кластеров зависит от радиуса сфер. Чем меньше радиус, тем больше получается кластеров.

При реализации алгоритма создаётся список векторов, которые ещё не были добавлены в кластеры. Первоначально он равен переданному списку векторов для кластеризации. Создаётся другой список, который будет содержать точки, которые присутствовали в прошлой «плавающей» сфере.

Далее организован цикл, который останавливается, когда список неиспользованных векторов пуст. Внутри него вычисляется минимальная сфера, которая содержит все неиспользованные точки. Есть ещё внутренний цикл, который вычисляет точки внутри текущей внутренней сферы, находящейся внутри сферы минимального радиуса. Радиус внутренней сферы равен k_1 – радиус минимальной сферы, где k_1 задаётся свойством метрики и по умолчанию имеет значение 0,9. Центр внутренней сферы – центр тяжести внутренних точек, полученных на предыдущем шаге алгоритма. Внутренний цикл продолжается пока прошлый список внутренних точек сферы не будет равен списку точек, полученных на данном шаге. Когда внутренний цикл завершает работу, внутренние точки подсферы добавляются в новый кластер и сама эта подсфера становится новым кластером.

2.2. Реализация алгоритма FOREL 2

Алгоритм FOREL 2 является модификацией алгоритма FOREL. Он требует явного указания количества кластеров, которые необходимо сформировать. При реализации этот алгоритм несколько раз вызывает метод базового алгоритма FOREL.

Сначала коэффициент k_1 устанавливается в его значение по умолчанию, т.е. 0,9. При этом значении, как правило, достигается минимальное количество кластеров. При необходимости этот коэффициент можно устанавливать в значение более близкое к единице (но меньше единицы).

Далее организован цикл, который останавливается, когда число кластеров, полученных алгоритмом FOREL оказывается больше, чем необходимо или коэффициент k_1 окажется меньше заданного значения. На каждом шаге цикла коэффициент k_1 уменьшается на

заданный шаг. Минимальное значение на данный момент равно 0,01. Шаг также равен 0,01:

$$0,01 \leq k_1 \leq 0,9. \quad (1)$$

Результаты, полученные во время последнего вызова алгоритма FOREL, служат в качестве результата вычисления алгоритма FOREL 2.

2.3. Реализация алгоритма объектной кластеризации

Этот алгоритм создаёт один кластер для каждого множества точек, принадлежащих одному классу. Кластеры, получаемые при помощи этого алгоритма, представляют собой объединения гипершаров.

Для формального представления образа класса в многомерном признаковом пространстве строим соответствующий кластер. С этой целью создаём список векторов, которые ещё не были добавлены в кластеры. Первоначально он равен переданному списку векторов для кластеризации. Создаётся новый кластер – единственный для этого класса, который генерируется этим алгоритмом кластеризации, так как данный алгоритм не является разделяющим.

Процесс построения кластера начинается с поиска наиболее удаленного от других вектора класса и добавляется в кластер. Далее организуется главный цикл, который останавливается, когда список неиспользованных векторов становится пустым.

Ищется вектор, ближайший к выбранному. Этот вектор добавляется в кластер. Создаётся шар с центром в предыдущей добавленной точке. Радиус – минимум от половины предыдущего сохранённого расстояния и половины текущего расстояния. Кроме того, вычисляются и запоминаются значения координат вспомогательного вектора, который указывает на середину отрезка, соединяющего два очередных экземпляра класса, и добавляется шар с центром, расположенным между прошлой и текущей точкой. Его радиус равен половине расстояния между текущей и предыдущей точкой. Алгоритм останавливается, когда не осталось ни одного неиспользованного вектора. Кроме того добавляется последний шар радиуса, равного половине последнего сохранённого расстояния и с центром в последней добавленной точке.

2.4. Реализация алгоритма равномерной кластеризации

2.4.1. Подробное описание алгоритма

Для работы алгоритма требуется хотя бы две точки пространства R^n .

1. Находится точка, наиболее удалённая от всех остальных. Расстояние от неё до ближайшей точки обозначим M . Минимальное расстояние между точками обозначим m . Введём величину δ такую,

что $m \leq \delta \leq M$, например среднее между m и M . Все кластеры будут состоять из объединений гипершаров радиуса δ , далее просто «шаров».

2. Возьмём любые две точки из всех имеющихся, расстояние между которыми минимально.

3. Если расстояние между ними больше δ , то определим два новых кластера, в первый из них добавим шар с центром в первой точке, а во второй – шар с центром во второй точке. Эти две точки исключаются из рассмотрения. Переходим к пункту 6.

4. Если расстояние между двумя выбранными точками $\leq \delta$, то определим новый кластер и добавим в него два шара с центрами в этих точках. Эти точки исключаем из рассмотрения

5. Все точки, которые попали в новый кластер, но ещё не исключены из рассмотрения, исключаем из рассмотрения. Добавляем в кластер шары с центрами в этих точках. Повторяем этот шаг, пока в новом кластере будут содержаться нерассмотренные точки.

6. Если все точки исключены из рассмотрения, то прекращаем работу алгоритма.

7. Если осталась только одна нерассмотренная точка, то определяем новый кластер, содержащий шар с центром в этой точке, и прекращаем работу алгоритма.

8. Если осталось хотя бы две точки, то переходим к пункту 2.

Изменяя δ , можно изменять количество кластеров. Чем меньше δ , тем больше в результате получается кластеров.

Как вариант, δ может быть взято как среднее арифметическое расстояний между всеми точками. Или M может быть вычислено как среднее арифметическое расстояний между всеми точками, а δ – по первоначальной формуле. Те или иные подходы могут быть более или менее удобными при работе с различными исходными данными.

2.4.2. Реализация

В начале алгоритма находится минимальное расстояние между всеми имеющимися точками. Для этого перебираются все возможные пары векторов и как только находится меньшее расстояние, чем были найдены до этого, это расстояние записывается в переменную и т.д.

Для нахождения максимально удалённого вектора от остальных перебираются все векторы. От каждого из них находится минимальное расстояние до всех остальных. Далее выбираем тот вектор, для которого это вычисленное значение максимально.

Вычисляется расстояние соседства в зависимости от коэффициента $k1$ и максимальной и минимальной величины. Вычисляется радиус будущих шаров. Создаётся список векторов,

которые ещё не были добавлены в кластеры. Первоначально он равен переданному списку векторов для кластеризации. Организуется внешний цикл, который заканчивается тогда, когда список неиспользованных векторов становится пустым.

Создаётся новый кластер. Ищутся два ближайших вектора по аналогии с поиском минимального расстояния в начале алгоритма. Эти два вектора добавляются в кластер и удаляются из списка неиспользуемых векторов. В последний кластер добавляются шары стандартного размера с центрами в этих точках. Все точки из списка неиспользованных точек, которые не дальше расстояния соседства от точек, включенных в последний кластер, тоже добавляются в последний кластер. Аналогичным образом добавляются и шары.

Если таких двух точек не нашлось, то все оставшиеся точки добавляются в отдельные кластеры. Соответственно добавляется один шар стандартного размера в каждый из этих кластеров.

2.5. Реализация алгоритма равномерной кластеризации 2

Равномерная кластеризация 2 – это модификация алгоритма равномерной кластеризации, которая может строить заданное количество кластеров.

Этот алгоритм несколько раз вызывает метод базового алгоритма равномерной кластеризации в процессе выполнения основного бесконечного цикла.

После каждого вызова этого алгоритма сохраняется предыдущее количество полученных векторов. Предыдущее число кластеров и текущее число кластеров сравниваются и анализируются. Если текущее число кластеров меньше необходимого, а предыдущее – больше, тогда цикл прекращается и наоборот. Если число построенных кластеров равно требуемому, то цикл тоже прерывается. В остальных случаях, если число кластеров слишком большое или слишком малое, то коэффициент $k1$ алгоритма равномерной кластеризации корректируется в соответствующую сторону. Также цикл прекращается при достижении этим коэффициентом крайних значений.

Окончательные результаты получаются во время последнего вызова алгоритма равномерной кластеризации.

Заключение

При реализации систем распознавания для представления образов классов предлагается использовать кластерные структуры. Исходными данными для построения кластерных структур будут являться системы векторов в многомерном признаковом пространстве. Первоначальное описание этих векторов в математическом выражении представляет собой прямоугольную матрицу, с числом строк равным размерности

признакового пространства. Формальное же описание образов классов можно реализовать в виде кластерных структур, получаемых на основе применения различных алгоритмов кластеризации.

В статье рассмотрен ряд вспомогательных алгоритмов, на основе использования которых можно осуществлять представление формальных образов классов в виде кластерных структур и представлена программная реализация этих алгоритмов. Описаны основные принципы реализации на уровне алгоритмов и с точки зрения организации программного кода на основе применения технологии ООП. Программное приложение имеет многомодульную структуру и обеспечивает возможность динамического подключения новых реализаций алгоритмов кластеризации, вычисления метрик, критериев однородности для выполнения аттестации априорного словаря признаков.

Библиографический список

[Родченко, 2006] Родченко, В.Г. Об одном методе реализации процедуры обучения при построении системы распознавания образов / В.Г. Родченко. // Известия Гомельского государственного университета имени Ф.Скорины. - 2006. - №4. - С.73-76

[Родченко, 2012] Родченко, В.Г. Метод построения компьютерной системы диагностики на основе анализа данных обучающей выборки / В.Г. Родченко, Е.В. Олизарович, А.И. Жукевич // Искусственный интеллект. - 2012. - №4. - С.381-386

[Загоруйко, 1999] Загоруйко, Н.Г. Прикладная методы анализа данных и знаний. / Н.Г. Загоруйко – Новосибирск: Изд-во Института математики СО РАН, 1999 – 264 с..

[Жукевич, 2010] Жукевич, А.И. Об одном методе построения формальных образов классов при реализации систем распознавания / А.И.Жукевич, В.Г.Родченко // Известия Гомельского государственного университета имени Ф.Скорины. – Гомель, 2010. – №5(62). – С.79-83.

[Кобзарь, 2006] Кобзарь, А.И. Прикладная математическая статистика. Для инженеров и научных работников. / А.И. Кобзарь – М.:ФИЗМАТЛИТ, 2006. – 816с.

SOFTWARE IMPLEMENTATION OF SUBSIDIARY ALGORITHMS FOR CONSTRUCTING CLUSTER STRUCTURES

Harunou V.A., Zhukevich A.I.,
Rodchenko V.G.

*Yanka Kupala Grodno State University, Grodno,
Republic of Belarus*

v.gorunov@grsu.by

san@grsu.by

rovar@mail.ru

Cluster structures allow to formalize the procedure of presenting images and standards of classes in a multidimensional feature space when build recognition systems. Also it allows to automatically perform procedures of training and classification. The process of building cluster structures is associated with using of library of software procedures. The procedures implement the subsidiary clustering algorithms, algorithms to compute metrics and intercluster distance and homogeneity criteria.

Introduction

This article describes the software implementation of support algorithms for building of cluster structures. In an automatic mode the cluster structures allow to implement procedures for training and final decision-making based on an initial data of classified training sample. It is used to formal representation of images in the construction of class recognition systems.

Main Part

It is necessary to build cluster structures for construction of class standards in the solution space. It is necessary to divide all objects of classified training sample into groups using the values of their attributes.

If classified training set is a set of classes which consist of the feature space so we assume for certification that all features are informative and they need to build cluster structures for objects that included in the classified training set.

Certification features is implemented by constructing a cluster structures for a given classified training sample without a relationship between objects and class.

Then it is performed a check to find a mistaken inclusion of properties in the initial classes of classified training sample. If the number of errors does not exceed a threshold value, the signs are considered suitable for the construction of a master image classes.

There is a program to implement the training procedure in an automatically mode that is implemented as a modular Windows-application. Modules are executed as plugins that represented as separate dynamic link libraries.

There are several types of modules that implement algorithms of computation metrics and intercluster distances, clustering algorithms and checker modules of homogeneity criteria.

Conclusion

The article presents a set of auxiliary algorithms that can be used to implement of presentation of formal master image classes as a cluster structures. In addition, there is a software implementation of these algorithms. The article describes basic principles of implementation at the level of algorithms. The software application has a multimodule structure and provides the ability to dynamically connect new implementations of clustering algorithms, computing metrics and homogeneity criteria to perform certification of priori dictionary of features.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ЛОГИЧЕСКАЯ И ЛИНГВИСТИЧЕСКАЯ СЕМАНТИКА НА ОСНОВЕ ИДЕЙ Г.ФРЕГЕ И В.В. МАРТЫНОВА

Демиров В.В.

*Институт философии НАН Беларуси,
г. Минск, Республика Беларусь*

vitaly.demirov@gmail.com

В работе рассматриваются логико-лингвистические проблемы семантики на основе идей Г.Фреге и В.В.Мартынова. Рассмотрение скоррелировано на практические аспекты связанные с созданием универсальных грамматик допускающих компьютерную реализацию.

Ключевые слова: смысл; значение; формализация; семантический код.

Введение

В данной статье ставится задача выявить на основе идей Г.Фреге и В.В.Мартынова границы и преимущества логических и лингвистических методов для реализации универсальных смысловых исчислений.

Понятие смысла является сегодня краеугольным понятием, математически точное выражение которого должно привести к прорыву в области интеллектуализации информационных технологий (от данного факта во многом зависят технологии проектирования компиляторов). Последние сегодня как нельзя остро нуждаются в универсальной платформе вычисления значений, основанной на формальной семантике.

На пути к формализации семантики логика и лингвистика исходят из различных предпосылок и используют различные средства. Это расхождение берет начало уже с двух фундаментальных для этих наук понятий – «смысл» и «значение».

Становление идей логической и лингвистической семантики как основы формальных методов представления смысла

Лейбницу принадлежит мысль о том, что если понятие «человек» соотносить с числом 2, а понятие «разумный» – с числом 3, тогда «Homo sapiens» будет выражаться соотношением $2 \times 3 = 6$. Истинность высказываний, соответственно, будет связываться с делимостью нацело числовых характеристик содержащихся в них понятий. Несмотря на высоко оцененный Винером замысел в качестве верной

начальной методологии создания думающих машин, привязывать смысловые свойства элементов высказывания к числовым характеристикам нельзя ввиду изначальных отличий логического умножения и сложения от арифметических, а также неполноте и противоречивости формальных систем, пытающихся выразить понятие натурального числа и операций с ними, что было показано К.Геделем.

При этом, сказанное не исключает возможности изучения количественных закономерностей языка: употребительность слов и словосочетаний и их порядок в частотном списке, длина звука и буквосочетания, информационный вес слова, слога или морфемы. В частности, на формализации фоносемантических закономерностей работают экспертные системы, способные определить стилистический ракурс и эмоциональный «вес» текста, возможность воздействия на определенную аудиторию. Но все же, мы понимаем, что здесь речь не идет о вычислении «значения» и его смыслового представления на реализацию которого направлена формальная семантика.

Фундамент современной логической семантики был заложен в работах немецкого философа и математика Г.Фреге, впервые заговорившего об интерпретации формальных систем и процедуре приписывания высказыванию истинностной оценки, основанной на условиях истинности данного высказывания. Затем А. Тарский определил эти условия истинности как принадлежащие к некоторому метаязыку, описывающему структуру языка-объекта. На основе его идей Д.Дэвидсон построил семантику условий истинности для естественного языка. На этой же основе также строится теоретико-модельная семантика расширением которой является семантика возможных миров. Восходящее к работам Г.Лейбница понятие

возможного мира было уточнено Р.Карнапом и в качестве термина современной математической логики вошло через работы С.Крипке, который впервые формулировал понятие возможного мира не в абсолютном понимании, а по отношению к какому-либо другому миру. Исходя из этого, необходимость некоторого события в определенном мире связана с присутствием этого события во всех возможных мирах, достижимых из этого определенного мира, а возможность некоторого события в определенном мире связана с существованием хотя бы одного мира достижимого из этого определенного мира, где это некоторое событие существует. Данные идеи, составляющие фундамент модальной логики дали толчок разработанной А.Приором темпоральной логике тематизирующей феномен времени в структуре высказывания и являющейся достаточно удобным инструментарием для спецификации и верификации параллельных и распределенных систем.

Успехи лингвистической семантики в области формализации начались после отхода в «порождающей» грамматике Н.Хомского от сосюрювской догматической линии в понимании языка в качестве системы, единицы которой определяются собственными имманентными отношениями. «Порождающей» здесь не означало то, что Хомский сфокусировался на психофизиологических особенностях порождения и понимания речи. В данном слове подразумевалось смещение акцента от единиц структуры языка в сторону базовых генеративных правил, порождающих эти единицы. Данная, исключительно синтаксическая формулировка содержала существенные трудности в соотношении с естественным языком, в котором синтаксис всегда имеет свою семантику. Иначе говоря, в языковом коде говорящих всегда хранятся некоторые смысловые примитивы, представляющие собой значения синтаксических моделей. Хомский напротив считал, что эти модели «порождаются» базовыми правилами «фразовой составляющей», которые бессознательны и автоматизированы. Превращение системы Хомского в инструмент анализа семантики естественного языка начали Дж. Дж. Кац и Дж. А. Фодор в своей работе «Структура семантической теории». Для этой цели они выстраивают «деревья», состоящие из «семантических маркеров» и «отличителей». Особая роль в формальном представлении семантики естественного языка принадлежит Р.Монтегю, разработавшем теоретико-модельный подход. Название его книги «Английский как формальный язык» говорит о его убежденности в возможности формализации естественного языка. При этом, недостатком грамматики Р.Монтегю как и аппликативной грамматики С.К. Шаумяна, является то, что интерпретации подлежит только поверхностная семантика. Далее, в рамках создания языков семантического представления можно выделить модель концептуальной зависимости Р.Шенка и модель «смысл \longleftrightarrow текст» И.А. Мельчука, в основе которых лежат следующие примитивы: примитивные действия – у Шенка и лексические функции – у Мельчука. Имея эмпирический источник своей разработки, примитивы данных моделей не претендуют на непротиворечивость

и полноту. Говоря о ролевой грамматике, основанной на теории глубинных падежей Ч.Филлмора, о примитивах А.Вежицкой, необходимо отметить, что они представляют собой эмпирически выделенные фрагменты универсальной семантики без собственных средств ее преобразования.

В отличие от указанных выше инструментов формализации смысла, языки типа УСК (созданные В.В. Мартыновым) представляют собой дедуктивные системы, обладающие собственными средствами как представления, так и преобразования семантики.

Говоря о дедуктивной семантической системе, возникает отчетливое представление о смысле как феномене, обладающем антипсихологическим и объективным статусом. А эта традиция, как известно, берет свое начало у Г.Фреге.

Истинностная нейтральность и объективность смысла в семантической модели Г.Фреге

Известно, что Фреге первый подверг понятие смысла тщательному анализу. Но в данном случае важно понять, что побудило его ввести это понятие в свою логическую теорию и почему оно как бы осталось вне ее.

Существенно то, что Фреге оказывается от традиционного разделения имен на единичные и общие, используя вместо них имена предметов и имен функций. Предмет, согласно Фреге, – это то, что не есть функция. Понятие при этом является частным случаем функции как такого действия над объектами (аргументами), которое устанавливает им в соответствие один и только один объект (значение). Для логических функций указанное значение берется из области состоящей из двух «объектов» – истины и лжи.

С каждым собственным именем (словом, знаком, соединением знаков, выражением) обозначающим предмет связывает, то, что он называет **значением** имени, и то, что он называет **смыслом** имени. Значение имени есть тот предмет, который обозначается этим именем. Смысл имени собственного – это информация, заключенная в имени и определяющая способ, каким имя обозначает предмет. Фреге говорит о том, что собственное имя **выражает** свой смысл, **означает** свое значение.

Фреге, в отличие от Булевой алгебры, принципиально нацелен создать семантическое исчисление. Он говорит: «Теперь подходу Буля я противопоставляю цель моей системы. С самого начала я имел ввиду выражение некоторого содержания. ...Но содержание должно быть передано более точно нежели на словесном языке» [Фреге, 2000].

В соответствии с этим, Фреге считает необходимым ввести в свое «исчисление понятий» особый **знак утверждения**. Фреге пишет, что равенство “ $-2^2 = 4$ ” еще не содержит никакого утверждения, оно просто выражает некоторое мыслимое «положение дел», что обозначается в записи Фреге при помощи

горизонтальной черты стоящей впереди выражения. Чтобы показать, что речь идет о суждении, в котором уже нечто утверждается, Фреге добавляет к этой горизонтальной черте вертикальную “┐”, так что в “┐-2² = 4” утверждается, что квадрат двух есть четыре. Подобный оператор утверждения, безусловно, позволяет отличать смысловой, информационный контекст, сопровождающий логические утверждения, от самих утверждений. Это отличие имеет большое значение, т.к. бинарные логические цепочки составляют достаточно небольшой процент нашего мышления, обрабатывающего информацию по вероятностным принципам. Кроме этого, современные смысловые инструменты обработки информации кроме средств исчисления семантики используют и средства представления семантики. Фрегеанская форма исчисления понятий, можно сказать, явилась прототипом семантических сетей, которыми успешно пользуется В.В. Мартынов при реализации своих исчислений. Выражая типизированные для определенной предметной области отношения понятий в виде ориентированного графа, сети сами не являются результатом некоторого алгоритма или вычислительной процедуры. На основе анализа математических свойств отношений входящих в сеть, разрабатываются методы решения для сетей конкретного вида.

Известно, что понятие смысла в его исчислении понятий появляется как следствие размышления над тем, что такое равенство. В частности, он замечает, что выражения “Утренняя звезда есть Утренняя звезда” ($a = a$) и “Утренняя звезда есть Вечерняя звезда” ($a = b$) выражают один и тот же факт: то, что планета Венера тождественна планете Венере. Но различие их познавательной ценности очевидно: если первое не несет никакого нового знания, то второе расширяет наше знание, сообщая некоторый астрономический факт. При этом, необходимо отметить, что поскольку обозначение предмета некоторым знаком зависит от нашего произвола (конвенции), выражение “ $a = b$ ” теряет связь с существом дела (объективностью смысла) и становится относящимся только к нашему способу обозначения. Различение познавательной значимости выражений “ $a = a$ ” и “ $a = b$ ” становится возможным, если к каждому имени отнести не только тот предмет, который обозначается этим именем (значение имени), но и тот способ, каким имя обозначает предмет, – его смысл. Различие в способе, каким предмет дан нам в его обозначениях, есть различие в сведениях и информации о предмете, содержащихся в его именах. Причем, если выявление значения предложения подчинено принципу композициональности и представляет собой функцию значений входящих в него имен, то смысл предложения определяется не значением его элементов, но только их смыслом. Формальной зависимости между смыслом имени и смыслом предложения Фреге найти не удалось. Несмотря на это он предложил **метод лингвистической редукции** выявляющий отличие между пониманием смысла и знанием значения, соответственно между суждением (как выражением смысла) и утверждением как переходом от суждения к

истинностному значению. “Заклочки в скобки” суждения об истинности или ложности мысли, есть лингворедукционный инструмент позволяющий преодолеть референциальные ловушки языка вызванные тем, что признание истинности мысли, имеет в языке всецело латентный характер и осуществляется самой формой утвердительного предложения. Невозможно обнаружить в языке специального знака, который указал бы на отличие мыслимого положения дел от утверждения истинности. Исходя из чего, следует отметить, что лингворедукционные процедуры, реализованные средствами языка Фреге, являются одним из методов канонизации естественного языка о которой, как известно, говорил и В.В. Мартынов.

Помимо данной канонизации редукция указывает на отличие суждения у Фреге от формально-логического. Будучи нейтральным по отношению истинности и ложности, оно обретается не как результат дедуктивного вывода, но как результат особой процедуры «схватывания». В контексте рассуждений В.В. Мартынова о роли «внутреннего времени» в реализации ситуационно свободной и, соответственно, сложной сигнальной системы это обстоятельство приобретает для нас особый смысл.

Специфика «исчисления смысла» В.В. Мартынова

Белорусский ученый-лингвист и создатель универсального семантического кода (вернее его различных версий) – В.В. Мартынов, является достойным продолжателем идей и традиций формальной семантики. Мартынов стремится к построению дедуктивной системы абсолютных языковых универсалий, разумеется, не путем сравнения языков, но путем построения семантического классификатора глаголов, соотносящегося с индексом синтаксических моделей предложений, а он, в свою очередь, с классификатором ситуаций.

Противникам подобного глобального подхода к языку, утверждающим сложность его семантического среза и многозначность в соотношении синтаксиса и семантики, Мартынов говорит о том, что это отношение не многозначно, а **неопределеннозначно**. Но последним свойством, вопреки мнению многих, обладают не только абстрактные и метафорические сферы, но и область первичных пространственных представлений, лежащих в основании логики и математики. Пример Мартынова, описывающий движение в комнате с окном и дверью, в форме выражений (стоять в дверях или под окном; при этом понятия дверь и окно не определены), не имеющих прямой зрительной ассоциации, это наглядно доказывает. Тем не менее, эти неопределенности нивелируются благодаря «здравому смыслу», лежащему в основе нашей модели мира, фундирующей коллективный и индивидуальный опыт. Первым шагом преодоления этой неопределенности в естественном языке должно быть **снятие его эллиптичности** – пропуска уточняющих элементов высказывания, когда

оказывается, что глагол управляет двумя и более объектами. Принцип одно-объективности говорит, что для каждого из таких объектов необходимо восстановить опущенный глагол, который может управлять только одним объектом. Именно в естественном языке, канонизированном подобным образом, возможна экспликация глубинной семантики. По мнению Мартынова, до сих пор не существует полного перечня типов языковых эллипсисов. Внутрязыковая (горизонтальная) неопределенность называется им презумпцией, а сверхфразовая неполнота, вызванная необходимостью синхронизации с моделью мира, называется пресуппозицией. Снятие горизонтальной неполноты позволяет Мартынову осуществить однозначное соответствие со смысловыми примитивами, т.е. соответствие между множеством синтаксических позиций и предложно-падежных конструкций. Ядерная цепочка с двумя объектами $S-P-O_1-O_2$ является формальным представлением предложения: Студент читает книгу в библиотеке. Предикатное ядро (P) обозначает любую форму воздействия субъекта (S) на объект (O). Но глагол читать не может управлять местным падежом. Для управления словоформами в местном падеже служат глаголы локализации. Соответственно, необходимо восстановить это предложение: Студент читает книгу (находясь) в библиотеке. Возможны так же цепочки с двумя субъектами S_1-S_2-P-O . И лишь при соблюдении стандартного порядка слов возможна реконструкция **одно-однозначных отношений** между синтаксической структурой и смыслом высказывания.

Итак, классифицируя в более ранних вариантах своей системы глаголы («акции»), Мартынов работает в рамках трехчленной семантико-эпистемологической конструкции (CAO – субъект, акция, объект), для которой формулируются аксиомы представления и преобразования знаний.

Далее, выделяя в процессе разработки УСК-6 четыре первичны неопределяемые понятия – смысловые примитивы (X – субъект, интеллектуальный агент; Y – инструмент, орудие потенциально замещающее субъекта; Z – объект воздействия субъекта; W – результат) и определяя УСК-алгебру (на основе абстрактной алгебры Линденбаума-Лукасевица имеющей следующую структуру $A = \langle M, \rightarrow, \neg \rangle$; Лукасевич доказал функциональную полноту системы состоящей из импликации и отрицания) их взаимодействия, Мартынов строит таблицы, в которых инфинитивам исходных глаголов и их интерпретациям в настоящем моменте с помощью УСК-примитивов ставит в соответствие классификационные формулы. Для реализации компьютерной поддержки данного исчисления Мартынов осуществляет построение классификатора примитивов в виде ориентированного графа, для которого аксиоматическая система УСК позволяет построить семантическое пространство в сотни тысяч вершин. Исходя из этого анализ выделенного слова определяется его формулой и местом в одном из классов (подклассов) семантического классификатора, для чего примитивы должны быть распределены в семантическом поле по степени близости их значений.

Примечательным является то, что несмотря на выстроенную линию построения алгебры смысла на основе математизации лингвистических феноменов, сам источник их структуры он относит к специфической форме переживания времени: «Сигнальная система животных **ситуационно связана**, т.е. сигнал и соответствующее действие животного совпадают во времени. Человек оперирует тремя обобщенными временами: прошлым, настоящим и будущим. Его сигнальная система **ситуационно свободна**. Он может анализировать прошлое, наблюдать настоящее и планировать будущее, что нашло отражение в его чрезвычайно богатой знаковой системе» [Мартынов, 2001]. Исходя из этого становятся более понятными идеи Фреге о допредикативном «схватывании» смысла, относимом к очевидности некоего объективного опыта нежели истинности логической структуры.

Заключение

Таким образом, несмотря на то, что для логики значение – это экстралингвистический предмет, денотат, а для лингвистики – концептуальное содержание лексической единицы закрепленное в словаре, системы Г.Фреге и В.В. Мартынова обнаруживают глубокую связь с одной стороны в стремлении точными логическими методами выявлять содержание языка, а с другой стороны в понимании того, что эти методы не автоматизируют источник этого содержания. По крайней мере, для этого необходимы другие методы включающие и когнитивную семантику, и понимание временной структуры актов «схватывания смысла».

Библиографический список

- [Мартынов, 1977] Мартынов В.В. УСК как язык представления знаний в автоматизированных системах // Семантика естественных и искусственных языков в специализированных системах. М., 1979.
- [Мартынов, 2001] Мартынов В.В. Основы семантического кодирования. Опыт представления и преобразования знаний. Минск, 2001.
- [Мартынов, 1982] Мартынов В.В. Категории языка. Семиологический аспект. М., 1982.
- [Фреге, 2000] Фреге, Г. Логика и логическая семантика : сб. тр. / Г. Фреге ; пер. с нем. Б.В. Бирюкова ; под ред. З.А. Кузичевой. – М. : Аспект пресс, 2000.
- [Фреге, 1997] Фреге, Г. Смысл и значение / Г. Фреге ; пер. В.А. Куренного // Избр. работы. – М., 1997. – С. 25–49.

LOGICAL AND LINGUISTIC SEMANTICS BASED ON THE IDEAS OF Г.ФРЕГЕ AND V.V. MARTYNOV

Demirov V.V.

*Institute of philosophy of NAS of Belarus,
Minsk, Republic of Belarus*

vitaly.demirov@gmail.com

The paper discusses the logical-linguistic problems of semantics based on the ideas of G.Frege and V.V.Martynov. Consideration founded on the practical aspects associated with the creation of universal grammar allow the computer implementation.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

СЕМАНТИЧЕСКИЕ МОДЕЛИ ИНТЕЛЛЕКТУАЛЬНОЙ ОЦЕНКИ ЗНАНИЙ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

Кабенов Д.И. *, Разахова Б.Ш. *

** Евразийский национальный университет им Л.Н. Гумилева, г. Астана, Казахстан*

kabenov73@mail.ru

utalina@mail.ru

В данной работе предлагается основа для проектирования и разработки интеллектуальной системы контроля и оценки знаний на естественном языке. Использование традиционных тестов предполагает выбор ответов на базе двоичной логики. Предполагается использовать их должным образом только в случае строго формально задаваемых вопросов. Но приобретение знания включает в себя не только (и не столько) запоминание априорных подлинных фактов, но способность понимания общих явлений, тенденций. Для управления этим знанием открытые тестовые задания являются более эффективными.

Ключевые слова: интеллектуальная система, анализ текста на естественном языке, оценка знаний, системы тестирования

Введение

Одной из важнейших задач системы непрерывного образования является обеспечение соответствия качества образования к международным стандартам. В сфере образования идет поиск нового содержания и новых форм, таких, как электронное обучение, дистанционное обучение, кредитная технология, рейтинговый контроль знаний. Все перечисленные формы обучения основаны на преимущественном использовании методики тестирования. Кроме того, важнейшим направлением реализации задачи улучшения существующей системы образования, является повышение объективности оценивания знаний обучаемых путем обоснования и разработки эффективных методик оценки результатов тестирования. В связи с этим в работе предлагается основа для проектирования и разработки интеллектуальной системы контроля и оценки знаний на естественном языке.

Одним из главных недостатков тестирования как метода контроля и оценки знаний является так называемый эффект "угадывания", существенно снижающий доверие к результатам контроля и свидетельствующий об отсутствии полной объективности оценки. Классическими методами борьбы с данным негативным аспектом являются [Аванесов, 2005], [Павлова, 2003], [Немировский, 2006], [Ким, 2006]: увеличение количества вариантов ответов в тестовом вопросе; введение

поправки на угадывание, корректирующей набранное количество баллов в зависимости от количества вариантов ответов в вопросе; максимальная детализация инструкций как в отношении проведения, так и в отношении подсчета результатов тестирования.

Эффективность каждого из данных методов достаточно широко исследована и проанализирована в современных научных разработках. Наряду с положительными аспектами, каждый из способов имеет ряд особенностей и ограничений в зависимости от специфики предметной области, психологических особенностей тестируемых, качества тестового материала, времени тестового сеанса, цели проведения теста (обучение, промежуточный или итоговый контроль) и много другого [Слободин, 2002], [Павлова, 2003].

Основным преимуществом компьютерных тестов является возможность задать всем студентам в равных условиях вопросы и в соответствии со шкалой оценивания равных оценок. Это повышает объективность контроля знаний по сравнению с традиционными методами.

Таким образом, проблема повышения объективности оценки качества знаний обучающихся является актуальной, а поиск эффективных путей её решения представляет собой важную современную научную задачу. Для решения этой проблемы целесообразно использовать "открытые" (без вариантов ответа) тестовые задания

с обработкой результатов тестирования применением методов искусственного интеллекта. Где ответы на заданные вопросы вводятся в свободной текстовой форме, что, является самой естественной и наиболее сложной задачей при организации системы контроля и оценки знаний.

1. Связанные работы

Для многих стран, теория электронного обучения подчеркивают важность установленного процесса познания и персонализированное обучение [Hung Jui-long, 2012]. Контроль знаний является интеллектуальной проблемой, требующей высокого качества решения, которые помогут выйти на новый этап в методике преподавания, так как это может дать возможность реализовать идею индивидуального подхода к обучению в массовых масштабах. Автоматизированное тестирование знаний становится очень популярным в наши дни, во-первых, потому что это экономит рабочее время учителя, освобождает его от рутинной работы и позволяет обеспечить объективную оценку знаний, результаты которых не зависят от субъективного мнения разных учителей.

В работе [Park et al, 2010] представлена разработанная компьютеризированная система конструктивного тестирования, с множеством выборов ответов. Система сочетает в себе краткий ответ (КО) и формат множества выборов (МВ), предлагая тестируемому лицу ответить на тот же вопрос дважды, сначала в формате КО, а затем в формате МВ.

В работе [Алексеев и др., 2007] авторы разработали программный инструмент, который позволяет подготовить тестовые вопросы и проведение тестирования с использованием любого из предлагаемых ниже типов вопросов. Описание этого программного средства и интеллектуальные алгоритмы для оценки знаний представлены в предыдущей работе авторов [Алексеев и др., 2003].

В работах [Shahbazova, 2000], [Shahbazova, 2011] описаны методы реализации механизма контроля знаний обучаемых с помощью теории нечетких множеств в сочетании с нейронной сетевой технологией. В работах применяются некоторые серьезные улучшения в логике оценки знаний и методик интерпретации данных ответов обучаемых. Представленная архитектура является типичной конфигурации аппаратного и программного обеспечения в среде интрасети образовательных учреждений.

В рамках проекта OSTIS (Открытые семантические технологии проектирования интеллектуальных систем) было создано справочная система «Решатель геометрических задач». На основе онтологии предметной области Геометрия. Работа системы построено так, что бы, отвечать на заданные вопросы пользователя и выдавать ему ответы на заданные вопросы, а так же выполнять операции для решения геометрических задач. В

работе «Семантическая технология компонентного проектирования интеллектуальных решателей задач» авторы [Заливако и др., 2012] представили возможности справочной системы интеллектуального решателя геометрических задач.

Анализ указанных работ показывает, что обычные линейные испытания с простыми формами ответа на этот вопрос не совсем отвечают требованиям комплексного контроля и оценки знаний обучаемых. Прежде всего, речь идет о естественных и математических науках, особенностью которых является тесная взаимосвязь понятия, темы и разделы курса, в качестве основного критерия для обучения - умение решать задачи различного характера и уровней сложности. Поэтому требуется разработка адаптивных, нелинейных и интеллектуальные методы испытаний с более различными типами задач и формами ответа. В то же время, новые системы тестирования должны включать в себя все достижения инструментов контроля знаний предыдущих поколений.

2. Применение тестового контроля для оценки знаний

Последние годы в системе непрерывного образования для контроля и оценки знаний широко используются компьютерное тестирование. Наиболее распространенной формой представления знаний являются естественно-языковые тексты. Текстовая форма знаний естественна для человека, такие знания легко воспринимаются, порождаются, тиражируются и модифицируются. В результате извлечения знания приобретают явный вид и становятся пригодными для автоматизированной обработки, например, системами сопоставляющего анализа, выполняющими сопоставление результата извлечения с эталонной моделью предметной области для оценки знаний обучаемых.

Использования тестирования в реальной ситуации позволит повысить детальность и точность оценивания и, кроме того, может являться единственно возможной формой процесса проверки знаний в ряде случаев, таких как:

- оценка знаний в системах дистанционного образования с использованием сетевых технологий Интернет;

- проведение массовой аттестации, сертификации специалистов, обучаемых.

Одним из перспективных направлений оценки знаний это применение открытых форм тестирования. Использование подобных заданий имеет следующие преимущества перед «традиционными» заданиями закрытого типа:

- увеличивает точность и детальность оценивания;

- позволяют определять правильность ответа на поставленный вопрос с определенной точностью;

- исключает фактор случайного выбора.

При построении тестирующих систем с открытой формой ответа в виде текста на естественном языке (ЕЯ) используются: методы искусственного интеллекта, методы лингвистической обработки текстов на ЕЯ. Использование тестирования позволяет, экономит рабочее время учителя, освобождает его от рутинной работы и позволяет обеспечить объективную оценку знаний, результаты которых не зависят от субъективного мнения разных учителей.

Оценка знаний в традиционном понимании рассматривается как определение итогового контроля уровня подготовки ученика в определенной предметной области после изучения некоторых разделов и глав курса. Результатом оценки знаний, как правило, является определение некоторой величины шкалы оценивания. Последствия оценивания могут иметь различные результаты – от чисто морального эффекта, до определения критических выводов вплоть до принятия судьбоносных решений. В связи с этим к оцениванию нужно подходить как к процессу объективного измерения и результаты такого измерения обрабатывать стандартными математическими методами и сопровождать стандартными характеристиками точности. Весь процесс оценки знаний должен выполняться в рамках научно-обоснованной методологии.

Одним из методов контроля знаний обучающихся является тестирование открытого типа. В тестах открытого типа выделяют следующие возможные варианты ответов:

- задания дополнения – испытуемый должен сам сформулировать ответы с учетом предусмотренных в задании ограничений;

- свободного изложения – испытуемый должен самостоятельно сформулировать ответы, ибо никакие ограничения на них в задании не накладываются.

Известно, что в настоящее время задача компьютерной реализации открытой формы тестирования в общем случае (свободно-конструируемый развернутый ответ на естественном языке) не полностью решена. Причина заключается в сложности извлечения смысла из фраз слабо формализованного естественного языка, на котором формулируется ответ.

Поскольку существующие системы оценок не удовлетворяют ученых и практиков, идет поиск других систем оценивания. Так, [Амонашвили, 1984] изучал вопрос о словесной оценке знаний учащихся и предложил методические рекомендации по словесной характеристике знаний в начальной школе.

В странах Европы и Америки имеются довольно многочисленные попытки отойти от цифровой,

символьной системы. В Германии был эксперимент по введению диагностических листов, в которых давались словесные и цифровые оценки знаний учащихся, мотивов учения, развития мышления, показанных при изучении школьного предмета и отдельных его тем. Они заносились в специальные таблицы. В Англии, подобно этому, имеются так называемые «профили». Они составляют тест и результаты, сведенные в таблицу – матрицу.

[Беспалько, 1989] в своих исследованиях предлагает в рамках педагогической технологии свою систему объективного контроля и оценки знаний учащихся. Главное в ней составляет разработка диагностических целей обучения, описание уровней усвоения знаний и инструментарий подсчета баллов по 12-балльной системе отметок.

Таким образом, педагогика делает активные попытки решить проблему объективного контроля и оценки знаний, но при этом сталкивается с рядом сложностей, в том числе организационных и психологических. Дело в том, что учителями неохотно принимаются нововведения контроля результатов обучения, так как от них требуются усилия понять новые системы и потратить время на их освоение и применение.

3. Методы представление знаний на естественном языке

Одним из методов искусственного интеллекта для представления знаний на естественном языке является семантические сети. Семантическая сеть – информационная модель предметной области, имеющая вид ориентированного графа, вершины которого соответствуют объектам предметной области, а дуги (рёбра) задают отношения между ними. Объектами могут быть понятия, события, свойства, процессы. Таким образом, семантические сети являются одним из способов представления знаний на естественном языке. Последние годы в трудах многих ученых предлагаются в качестве формальной основы проектируемых интеллектуальных систем, абстрактных логико-семантических моделей интеллектуальных систем, использовать графодинамические модели специального вида – семантические модели представления и обработки знаний в основе которых лежат **семантические сети** [Кузнецова, 1986], [Лозовского, 1984], [Плесневича, 1982], [Скороходько, 1989], [Шенка, 1980], [Sowa, 2008].

В своей работе [Голенкова и др., 2012] предложил создание формальных средств описания семантики различных видов знаний, и формальных средств описания обработки знаний на семантическом уровне. На основе предложенных принципов можно построить систему интеллектуальной оценки знаний обучающихся на примере предметной области Геометрия раздел Планиметрия.

Для анализа и проверки ответов обучаемых, на вопросы теста открытой форме предлагается анализ текста на естественном языке. Описание знаний в предметной области Планиметрия было представлено в виде онтологии [Kabenov et al., 2012].

1. Предварительно лингвистическая обработка исходного текста (морфологический и синтаксический анализ предложения) необходима для отделения терминов (классы, подклассы, свойства и отношения).

2. Формальное понимание текста как результат построения онтологического графа.

В работе были построены [Razakhova, 2009] формализованные синтаксические правила, анализ и синтез алгоритмов словосочетания и предложения на казахском языке. Результаты исследования могут быть использованы при создании интеллектуальных систем человек - машина с возможностью взаимодействия на казахском языке.

Опишем вкратце предложенный метод проверки геометрического решения задачи на основе онтологий. Тексты геометрической задачи представляют собой набор связанных предложений. К ним относятся простые и сложные предложения, неполные предложения (с анафорой и многоточиями). Формальное понимание текста геометрических задач является их представлением в языке знаний в предметной области онтологии Планиметрия. Это представление должно быть подключено и расширено с заполнением значения слота для случаев с описанием ситуации, представленной текстом.

Рассмотрение всего процесса анализа геометрических задач в этой статье не представляется возможным. Поэтому, давайте рассмотрим структуру ситуации, которая должна быть получена в результате онтологии на основе лингвистического анализа для одной геометрической задачи.

Задача. Дан прямоугольный четырехугольник сторонами 8 см и 18 см. Найдите сторону квадрата площадь которого равен площади прямоугольного четырехугольника.

Ответ обучаемого(далее ученик):

Ученик на естественном языке(казахский язык) излагает ход решения данной задачи. По условию задачи площади прямоугольного четырехугольника и квадрата равны. В связи с этим найдем площадь прямоугольного четырехугольника. То есть умножив стороны четырехугольника найдем $8 \cdot 18 = 144$. Зная что все стороны квадрата равны, далее извлекая корень квадратный из 144 находим сторону квадрата. Сторона квадрата равен 12.

Для оценки знаний ученика в системе тестирования должно быть заложено эталонное знание. Эталон знаний предлагает эксперт (в данном случае учитель). На основе эталонного

знания система тестирования будет оценивать ученика.

Эталон знаний:

1. Обработка задачи (вопроса):

1.1 Морфологический анализ: из онтологии предметной области Планиметрия найти ключевые слова связанные с задачей. По нашему примеру – это прямоугольный четырехугольник, сторона, квадрат, площадь, равно, корень квадратный.

1.2 Синтаксический анализ: для определения, какое свойство относится, к какому классу. Загрузка формул из базы данных для решения задачи. Формировать список нужных переменных и присвоить значения отобранные из текста задачи при анализе задачи.

1.3 Определение количества и порядок уровней решения (учитель здесь может указать весомость хода решения задачи, которое будет учитываться в дальнейшем при оценке знаний ученика) задачи.

Организация и проведение тестирования будет происходить при сопровождении учителя, для дополнения и совершенствования базовой онтологии по конкретному вопросу (это ввод формул, дополнительных описаний и т.п.). Не обработанные вопросы, по которым не было составлено эталонное знание к тестированию не допускается. Такой подход дает возможность качественного подбора вопросов и их уровня сложности.

Система интеллектуального тестирования должна заранее решить задачи, предложенные к тестированию. Для этого используется язык разметки математических формул MathML. Ввод и хранение нужных формул для решения задачи будет происходить в языке разметки математических формул MathML. В интеллектуальной системе имеется поддержка редактирования, просмотра и решения математических выражений в языке разметки математических формул MathML. Также имеется конвертер с естественного языка в математический язык разметки математических формул MathML и обратно на естественный язык.

2. Обработка ответа ученика:

2.1 Морфологический и синтаксический анализ: для конвертирования ответа ученика данного на естественном языке на математический язык в языке разметки математических формул (MathML).

2.2. Провести анализ по уровню ходу решения задачи определенных при вводе вопроса и проверить их реализацию в ответе ученика и оценивать каждый уровень хода по весу указанный учителем. То есть задача, делится на подзадачи. Для того чтобы оценить знание ученика, а не ответ. Как часто у нас происходит при использовании закрытых тестов или же тестов с несколькими вариантами ответов.

Работа автоматизированной интеллектуальной системы тестирования построенный в таком ключе, на наш взгляд даст ожидаемый объективный результат оценки знаний обучающихся.

Заключение

В этой статье мы представили подход анализа текста основанного на онтологии для автоматической оценки ответов обучающихся на естественном языке (казахский язык). Использование интеллектуальных алгоритмов может также быстро менять систему оценки и контрольной схемы, что значительно улучшает качество и скорость тестирования. Наша концепция системы тестирования использует результаты интеллектуальной оценки уровня пользователя и предоставляет набор тестов, приспособленных к уровню подготовки обучаемого. Система контроля знаний распределяет вопросы по сложности, основанные на данных, полученных во время тестирования. Это дает возможность построения адаптивных тестов, которые не требуют коррекции на уровне пользователей.

Разрабатываемая система тестирования интеллектуальной оценки знаний обучающихся в будущем даст предпосылку дальнейшего применения результатов исследования для разработки автоматизированной он-лайн системы тестирования, с целью организации и проведения таких масштабных мероприятий как ЕНТ (единое национальное тестирование) выпускников школ и КТА (комплексное тестирование абитуриента) поступающих в ВУЗы.

Библиографический список

- [Слободин, 2002] Слободин А.В., Часовских В.П. Совершенствование оценки знаний методом тестирования [Электронный ресурс] / IX Всероссийская научно-методическая конференция "Телематика" 2002.
- [Аванесов, 2005] Аванесов В.С. Форма тестовых заданий [Текст] / В.С. Аванесов Учебное пособие, 2 изд. переработанное и расширенное. – М.: Центр тестирования, 2005. – 156 с.
- [Павлова, 2003] Павлова И.Н., Одинокоев А.В. О проблеме угадывания правильного ответа в тесте [Текст] // Международный конгресс конференций "Информационные технологии в образовании" [Немировский, 2006] Немировский В.Б. Реализация разновидностей открытой формы компьютерного тестирования знаний. 2006. - Т. 309. - № 7.
- [Ким, 2006] Ким В.С. Коррекция тестовых баллов на угадывание. Педагогические измерения, 2006. - №4. – С.47-55.
- [Hung Jui-long, 2012] Hung Jui-long.: Trends of E-learning Research from 2000 to 2008: Use of text mining and bibliometrics. British J. of Educational Technology, Vol. 43.1, (2012) 5–16
- [Park et al, 2010] Park, Jooyong.: Constructive multiple-choice testing system. British J. of Educational Technology. Special Issue: Learning objects in progress, Vol. 41.6, (2010) 1054–1064.
- [Алексеев и др., 2007] Алексеев А.Н., Волков Н.И., Кочевский А.Н.: Типы вопросов для автоматизированного контроля знаний студентов компьютерного моделирования и новых технологии. № 11.3, Рига, 2007. С. 35–42.
- [Алексеев и др., 2003] Алексеев А.Н., Волков Н.И., Кочевский А.Н.: Элементы нечеткой логики в системы автоматизированного контроля знаний. Открытое образование, № 4, М.: 2003. – С. 23–25.
- [Shahbazova, 2000] Shahbazova, Sh., Freisleben, B.A.: Network-Based Intellectual Information System for Learning and Testing. Proceedings of the Fourth International Conference on

Application of Fuzzy Systems and Soft Computing, Siegen, Germany (2000) 308-313.

[Shahbazova, 2011] Shahbazova, Sh.: Application of Fuzzy Sets for Control of Students Knowledge. Application Comput. Math., Vol.10.1, Special Issue (2011) 195-208.

[Заливако и др., 2012] Заливако С.С., Шункевич Д.В. Семантическая технология компонентного проектирования интеллектуальных решателей задач. Материалы международной конференции «Открытые семантические технологии проектирования интеллектуальных систем. Минск: 2012. С.297-314.

[Амонашвили, 1984] Амонашвили Ш.А. Воспитательная и Образовательная функции оценки учения школьников – М.: 1984

[Беспалько, 1989] Беспалько В.П. Слабые педагогической технологии. – М.: 1989.

[Кузнецов, 1986] Кузнецов, И.П. Семантические представления / И.П. Кузнецов. – М: Наука, 1986.

[Лозовский, 1984] Лозовский, В.С. Семантические сети / В. С. Лозовский // Представление знаний в человеко-машинных и робототехнических системах. – М.: ВИНТИ, 1984. – С. 84-121.

[Плесневич, 1982] Плесневич, Г.С. Представление знаний в ассоциативных сетях / Г. С. Плесневич // Изв. АН СССР. Техн. кибернет. - 1982. – N 5. - с.6-22.

[Скороходько, 1989] Скороходько, Э.Ф. Семантические сети и автоматическая обработка текста. / Э. Ф. Скороходько. – Киев: Наук. думка, 1983.

[Шенк, 1980] Шенк, Р. Обработка концептуальной информации / Р. Шенк. – Москва: Энергия, 1980.

[Sowa, 2008] Sowa, J. Conceptual Graphs/ John F. Sowa, F. van Harmelen, V. Lifschitz, B. Porter// eds., Handbook of Knowledge Representation, Elsevier, 2008, pp. 213-237.

[Голенков и др, 2012] Голенков В.В., Гулякина Н.А. Графодинамические модели параллельной обработки знаний: принципы построения, реализации и проектирования. Материалы международной конференции «Открытые семантические технологии проектирования интеллектуальных систем. Минск: 2012. С. 23-52.

[Kabenov et al., 2012] Kabenov D., Muratkhan R., Satybaldina D.Zh., Razahova B.Sh., Sharipbayev A.A. Intelligent system of knowledge control for e-learning/ Journal of International Scientific Publications: Educational Alternatives, Volume 10, Part 2, 2012. pp.250-260. <http://www.scientific-publications.net>

[Razakhova, 2009] Razakhova, B.: Automation of the analysis and synthesis of word-combinations and sentences of the Kazakh language. PhD thesis (in Kazakh language). L. Gumilyov Eurasian National University. Astana. 2009.

SEMANTIC MODELS OF KNOWLEDGE INTELLECTUAL ASSESSMENT IN NATURAL LANGUAGE

Kabenov D.I.* , Razahova B.Sh.

* L. Gumilyov Eurasian National University,
Astana, Kazakhstan

kabenov73@mail.ru
utalina@mail.ru

This paper proposes a framework for the design and development of intellectual system for knowledge monitoring and evaluation in natural language. Using conventional test involves choosing answers based on binary logic. They are intended for proper use only in the form of strictly formally asked questions. But acquiring knowledge includes not only (and not so much) memorizing of priori true facts, but the ability to understand general phenomena and trends. Open test tasks are more effective in managing knowledge.

Introduction

One of the major objectives of continuous education system is to ensure education quality conformance to international standards. New content and new forms such as e-learning, distance learning, credit technology, rating control of knowledge are searched in education. All of these learning forms are based on the preemptive use of testing technique. Moreover, the most important direction for the implementation of task for improving the existing education system is to increase the objectivity of knowledge evaluation trained by justification and development of effective methodologies for the test results assessment. In this regard, the paper proposes a framework for the design and development of intellectual system for monitoring and evaluation of knowledge in natural language.

Main Part

Proposed testing system will track the progress of the task solving by the student.

Etalon knowledge:

1. Task (question) processing:

1.1 Morphological analysis: finding and selecting keywords based on ontology (classes and properties).

1.2 Syntactic analysis: to determine which property belongs to which class. Loading formulas from the database to solve the problem. Create a list of relevant variables and assign values selected from the text of the task in the task analysis.

1.3 Determination of the number and order of the task solution levels (here the teacher can specify the level significance that will further be taken into account in assessment).

All this will occur accompanied by a teacher in order to supplement and improve the basic ontology (this is entering formula, additional descriptions etc.). Necessary formulas will be entered and stored in MathML format. The intellectual system supports editing, viewing and solving mathematical expressions in MathML form. There is also a converter of natural language into MathML and back.

2. Processing of student response:

2.1 Morphological analysis and syntactic analysis: to convert the answer given in natural language in the language of formulas (MathML).

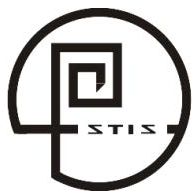
2.2. To make analysis on the task decision levels, determined in entering the question and to check their implementation in the student's answer and evaluate each level by significance, specified by teacher.

Work of automated testing system made in this way, in our opinion will give the expected objective result of the students' knowledge assessment.

Conclusion

In this article, we presented text analysis approach based on ontology for automatic evaluation of the students' responses in natural language (Kazakh). Using intellectual algorithms can also quickly change the evaluation system and control circuit, which significantly improves the quality and speed of testing. Our test system concept uses the results of user level intellectual assessment and provides a set of tests that are adapted to the level of student training. Knowledge control system distributes questions in complexity, based on data obtained during testing. This enables to make adaptive tests that do not require correction at the user level.

The developed system of testing of intellectual assessment of trainees knowledge in the future gives further premise of research results application for the development of an automated on-line testing system in order to organize and conduct large-scale events such as UNT (unified national testing) for school leavers and ECT (entrants' complex testing) for those entering High Schools.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

АДАПТИВНЫЙ ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЕЙ

Барлыбаев А.Б.

*Евразийский национальный университет имени Л.Н.Гумилева НИИ «Искусственный интеллект»,
г. Астана, Республика Казахстан*

frank-ab@mail.ru

В данной статье рассматривается работа разработанного нами адаптивного интерфейса. Он является одной из функций интеллектуального электронного университета. На его дизайн и структуру могут влиять пользовательские данные. Описывается процесс адаптации интерфейса на основе психологической информационной модели пользователя, в котором генерируются новые виды дизайна и коды для стиля интерфейса.

Ключевые слова: пользовательские данные, адаптивный интерфейс, интеллектуальные информационные системы, самообучение, фрактальная теория.

Введение

Интерфейс между человеком и компьютером является своеобразным коммуникационным каналом. Назначение этого канала – облегчить пользователю взаимодействие с некоторыми функциями программы, не обременяя лишними заботами и не всегда востребованными знаниями. Существует очевидная зависимость между уровнями внутренней сложности интерфейса и соответствующих знаний, необходимых человеку для его использования. Здесь первой проблемой является адаптация структуры информационных потоков и параметров интерфейса к нуждам индивидуального конечного пользователя. Второй проблемой – адаптация пользователя к компьютерной системе.

Адаптивный пользовательский интерфейс – автоматически изменяющийся интерфейс под профиль пользователя с целью сохранения или достижения оптимального состояния при изменении внешних условий с помощью своих аппаратных и программных средств.

Для адаптивного интерфейса можно использовать: цветовой тест Люшера [Luscher, 1971], вербальный тест Айзенка [Айзенк, 2003] и другие. Но мы не будем их рассматривать, потому что прохождение различных тестов занимает много времени и усилий у пользователей.

1. Цель

Нашей целью является генерация адаптивного

интерфейса на основе личных данных пользователя, где учитываются дата рождения, пол, национальность и другие данные для построения психологической информационной модели пользователя (далее - модель пользователя).

Обычно для определения модели пользователя используют различные тесты. Но из-за долгого прохождения тестов, легче будет выявить модель, узнав дату рождения и определить гороскоп пользователя. Складывая до получения однозначного числа, цифры дня, месяца и года рождения можно определить характер человека. На характер также влияет его пол, а менталитет человека сильно влияет его национальность. Правила подготовки статьи

При создании своей статьи используйте уже предопределённые стили данного документа, старайтесь избегать создания своих собственных стилей.

Редакционная коллегия оставляет за собой право осуществлять компьютерную верстку представленных статей, но не будет подвергать поступившие статьи литературному редактированию. В них будут сохраняться все особенности авторского стиля, в том числе и возможные не точности.

2. МЕТОДЫ ВЫЯВЛЕНИЯ ПСИХОЛОГИЧЕСКОЙ ИНФОРМАЦИОННОЙ МОДЕЛИ ПОЛЬЗОВАТЕЛЯ

Метод сложения даты рождения. Для каждого человека можно вычислить так называемое число

рождения (ЧР) следующим образом:

- 1) вычисление значения числа по формуле

$$N = N_1 + N_2 + N_3 + N_4 + N_5 + N_6 + N_7 + N_8$$

Где N_1, N_2 – цифры дня рождения; N_3, N_4 – цифры месяца рождения, а N_5, N_6, N_7, N_8 – цифры года рождения;

2) Если полученное число является однозначным (одноразрядным), то это число считается ЧР. В противном случае будем складывать цифры полученного неоднозначного числа, пока не получим однозначное число в промежутке от 1 до 9. Это означает, что для каждого человека имеется собственное ЧР, которое является носителем определённого потенциала [Jordan, 1984].

Предположим, что человек родился 05.10.1959. Это означает, что

$$N = 5 + 1 + 0 + 1 + 9 + 5 + 9 = 30, 30 = 3 + 0 = 3.$$

Значит, его ЧР равно 3.

Поскольку каждое ЧР соответствует определенной характеристике, то 3 соответствует следующей характеристике:

«Самовыражение - в этом ваша суть...» и т.д.

- 1 - Лидерство в любой сфере.
- 2 - Лучший из возможных помощников.
- 3 - Самовыражение - в этом ваша суть.
- 4 - Фундамент - прочный, логически выверенный, рассчитанный надолго.
- 5 - Вы гордитесь своей свободой.
- 6 - Умение приспособиться.
- 7 - Глубокие мыслители и интроверты.
- 8 - Процветание и власть.
- 9 - Редкая чувствительность.

По гороскопу. Описание характеристик по гороскопу можно легко найти в интернете. Поразительно, но астрология и время вашего рождения имеет некоторое влияние на ваш характер. Этот факт можно оспорить, но мы поддержим это для легкого и главного быстрого определения модели пользователя [Goodman, 1984].

Пол. В соответствии с матрицей пола, выпавшей на долю конкретного человека, он будет развиваться, осваивать социальный опыт, чтобы реализовать свою субъектную реальность. Воспитание при этом играет важную, но вторичную роль: природой дано, а воспитанием задано. Помимо специфических биофизиологических (строение тела, половые органы) и психофизиологических (функции полушарий мозга, интеллект, эмпатия, эмоции), матрица пола содержит ряд видовых психических свойств - мужских, маскулинных, и женских, феминных. Маскулинные и феминные качества - это, в

основном, врожденные психологические признаки, параллели половой принадлежности, входящие в характер. Маскулинность проявляется в таких чертах, как смелость, воля, напористость, стремление демонстрировать силу, авантюризм, жесткость и др. Феминности соответствует уступчивость, покорность, сострадание и т.п. [Udry, 1994].

Национальность. У каждой нации свой менталитет, традиции и обычаи. Все это влияет на формирование характера. Согласно современным исследованиям нейрофизиологов и психологов, интеллектуально-познавательные национально-психологические особенности являются следствием вовлечения в работу разных зон коры головного мозга, а также результатом разной взаимозависимости активности этих зон [Крысько, 2002].

3. Модели пользователей

Приведем краткий перечень основных критериев моделей пользователей, имеющих на сегодняшний день практическое применение [Денинг, 1984]. Условно разделены на 7 смысловых групп:

а) Демографические показатели: возраст; пол; антропометрические данные.

б) Индивидуально-психологические особенности: отношения к нововведениям; уровень субъективного контроля; способность самостоятельно принимать решения; заинтересованность в помощи.

в) Психомоторные качества: способность концентрироваться; подверженность ошибкам; способность переключать внимание; степень развитости самоконтроля.

г) Когнитивные особенности: характеристики когнитивного стиля; индуктивная/дедуктивная стратегия; функциональная асимметрия полушарий головного мозга; когнитивные возможности.

д) Подготовленность: образования пользователя; профессиональная компетентность; знание системных задач и интерфейса; экспертный уровень.

е) Мотивация: цели; потребности; задачи; ожидания.

ж) Характер системного взаимодействия: предпочтения; привычки; специфические ситуации; стресс – факторы.

Если каждому пользователю системы соответствует своя модель, отличная от других моделей пользователей, то такой подход является индивидуальным. При стереотипном подходе, наоборот, используется предполагаемая принадлежность пользователя к определенной модели (классу), количество которых строго ограничено. Обычно создают несколько моделей пользователей. И. Бомон выделяет 3 модели:

«модель новичка», «модель продвинутого пользователя» и «модель эксперта». При индивидуальном подходе заранее известно, какая модель будет сформирована для конкретного пользователя.

Пока ни один из известных интерфейсов не может считаться совершенным ни с позиции учета степени адаптации, ни с позиции искусственного интеллекта, ни с эргономической позиции (дизайн, «прозрачность», удобство, и т.п.), ни с профессиональной (модифицируемость, наличие специальных функций).

Системы с адаптивным интерфейсами могут вызвать у пользователя чувство потери контроля, возможны некоторые неточности в предсказании желаний и поведения пользователя, что может вызвать эффект «враждебности» со стороны пользователя. Поэтому конечный интерфейс можно перенастроить самому.

Модели пользователей.

Определим переменные X_i – число рождения, Y_j – знак зодиака (по гороскопу), W_k – пол, Z_l – национальность. Где $i = [1-9]$, $j = [1-12]$, $k = [1-2]$, $l = [1-195]$. $k = [1-2]$, цифра 1 – мужской пол, цифра 2 – женский пол. $l = [1-195]$, нумерация по списку государств членов ООН (United Nations).

$$X_i Y_j W_k Z_l = [X_1 Y_1 W_1 Z_1 \dots X_9 Y_{12} W_2 Z_{195}]$$

$X_i Y_j W_k Z_l$ – идентификатор. Возможное максимальное количество моделей пользователей 42120. Если количество методов вычисляющих информационную модель пользователей увеличится, то и возможное максимальное количество моделей пользователей увеличится по геометрической прогрессии. Увеличение по геометрической прогрессии вызовет проблему для программистов. Необходимо чтобы адаптивный интерфейс сам вычислял эти модели пользователей, при этом генерируя необходимый код для новых дизайнов. Ниже будет описано решение этой проблемы, с использованием фрактальной теории при обучении и генерации новых видов моделей пользователей. На рисунке 1 представлены уровни элементов адаптивного интерфейса.

Что можно настроить в интерфейсе?

1. Позиция меню;
2. Вид меню (список, таблица, иконки);
3. Палитра блоков в интерфейсе (меню, шапка, основной блок и т.д.).
4. Шрифт и размер текста;
5. Размер изображений;



Рисунок 1. Уровни элементов адаптивного интерфейса.

Идентификатор равен идентификационному вектору, который определяет параметры меняющие интерфейс, т.е.

$$X_i Y_j W_k Z_l = \overline{A_{ijkln}}$$

где n – количество меняющихся параметров интерфейса.

В нашем случае $n=5$. Но, это значение может меняться в зависимости от добавления новых элементов в интерфейс. Возможные значения переменных можно представить в следующем виде:

A_{ijkl1} = 'right-top' OR 'left-top' OR 'center-top' OR 'center-bottom'

A_{ijkl2} = 'list' OR 'table' OR 'icon'

A_{ijkl3} = 'red, green, blue' OR '...'

A_{ijkl4} = ('arial' OR '...') AND ('1' OR '2' OR '3' OR '...')

A_{ijkl5} = '1%' OR '...'

Количество параметров определяется интерфейсом. Для каждого параметра определяется возможные значения. После формирования модели пользователя, идентификатора, она сохраняется в базе данных. Далее происходит подстройка интерфейса системы под конкретную модель. Из модели пользователя передаются значения идентификационного вектора, например, в CSS (Cascading Style Sheets) файлы интерфейса. По специальным тэгам (id компонент) настраивается интерфейс. На самом деле в базе данных записывается не сама модель пользователя, а сгенерированные на ее основе значения параметров компонент (id) интерфейса. На рисунке 2 изображена блок-система адаптивного интерфейса.

Каждому значению этих параметров можно дать несколько различных значений, либо создать определенный класс, например, в CSS.

Пример. X_1 – овен, Y_1 – 1, W_1 – мужской, Z_1 – казах.

$X_1 Y_1 W_1 Z_1$ = center-top, icon, red, pink, coral, lucida, 14, 120%

Позиция меню align="center", valign="top". Вид меню icon. Палитра блоков в интерфейсе (меню, шапка, основной блок) red, pink, coral. Шрифт и размер текста lucida 14 nm. Размер изображений 120%.



Рисунок 2. Блок-система адаптивного інтерфейса.

Пользователь заполняет данные в профиле. Данные сохраняются в базе данных. Далее они передаются в специальный модуль, который запускается при сохранении данных и высчитывает модель пользователя. Высчитанная модель сохраняется в базе данных. При новом вхождении в систему интерфейс автоматически настраивается под модель пользователя. В итоге получается адаптивный интерфейс.

На рисунках 3-6 показаны варианты интерфейса пользователя. Мы изменяем данные пользователя, после интерфейс автоматически настраивается под информационную модель пользователя. Адаптивный интерфейс разработан на языках программирования Cache Object Scripts (mumps), Cache Server Pages, ZEN, CSS, HTML.



Рисунок 3. Адаптивный интерфейс пользователя.

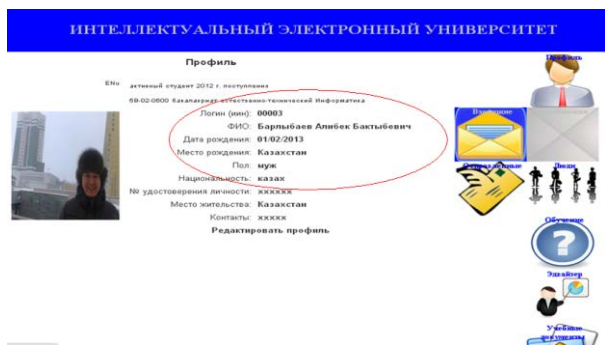


Рисунок 4. Адаптивный интерфейс пользователя.



Рисунок 5. Адаптивный интерфейс пользователя.

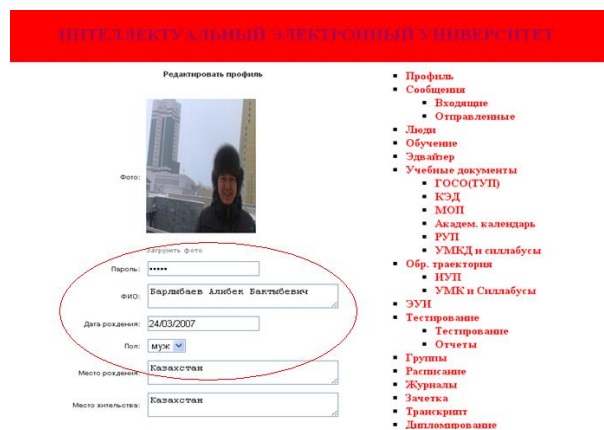


Рисунок 6. Адаптивный интерфейс пользователя.

4. Самообучение адаптивного интерфейса

Количество параметров (данных) вводимых в профиле влияет на максимальное количество информационных моделей. По моделям пользователей составляется конечный дизайн интерфейса. При этом появляется проблема увеличения количества информационных моделей пользователей по геометрической прогрессии. При использовании всего четырех параметров количество моделей пользователей может достигать 42120. А если параметров станет больше. Очевидно, эту проблему можно решить только самообучением адаптивного интерфейса.

Допустим, что каждый из параметров является измерением некоторого пространства. Если у нас 4 параметра, то мы должны использовать четырехмерное пространство. При увеличении количества параметров увеличится и количество измерений.

Очень сложно графически описать четырехмерное пространство, так как формат статей позволяет использовать максимум трехмерное пространство. Поэтому будем использовать только 3 параметра дату рождения, гороскоп и пол. Вводим несколько рандомных случаев моделей пользователей на трехмерное пространство, описано на рисунке 7. Эти случаи (экземпляры) объединяем в классы. К классам определяем шаблоны дизайна интерфейса (CSS интерфейса).

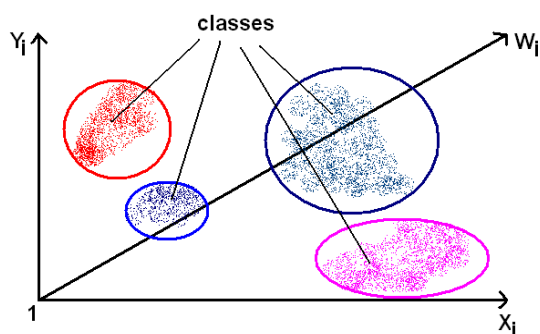


Рисунок 7. Случаи моделей пользователей в трехмерном пространстве.

Здесь ось X_i – число рождения, ось Y_j – знак зодиака (по гороскопу), ось W_k – пол.

При новом введении случая (экземпляра) в многомерное пространство вычисляется центр тяжести к классам. Исходя из этого, к какому классу он будет ближе, к такому классу будет принадлежать он. Тогда шаблон дизайна этого класса будет применяться новому экземпляру.

Согласно фрактальной теории, объект имеет бесконечное множество вложенности [Gouyet, 1996]. Если мы сможем описать один объект и разделить его на два, то ранее сделанное описание можно применить для нового объекта. В качестве объекта (фрактала) будем рассматривать классы со случаями. Исходя из этого, если классы переполняются случаями (допустим 20 случаев), то старые классы делятся на новые классы. Эти новые классы получают шаблон дизайна от старого класса. Нельзя забывать, что наш адаптивный интерфейс имеет функцию самонастройки. То есть шаблон дизайна может быть изменен пользователями. Шаблоны дизайна новых классов определяются средним или часто встречающимися параметрами случаев, которые входят в этот класс. В итоге количество классов увеличится в зависимости от их переполнения и добавления новых параметров. Таким образом, будет достигнуто главное свойство в интеллектуальных информационных системах – самообучение.

Заключение

Сформированные принципы генерации и автоматизации адаптивных интерфейсов в настоящей работе позволяют легко разрабатывать индивидуальные интерфейсы для различных приложений и облегчают понимание структуры интерфейса и его реализацию. Был предложен подход в решении проблемы самообучаемости в интеллектуальных информационных системах.

В дальнейшем мы попытаемся усовершенствовать наш адаптивный интерфейс до интеллектуального интерфейса и в уточнении ранее определенного характера пользователя будем использовать его аватар (фотографию) с помощью методов распознавания графических образов.

В результате можно получить онлайн автоматизированный справочный агент, который сможет самостоятельно вести диалог (устно и письменно) с пользователями и использоваться в различных интеллектуальных информационных системах.

Библиографический список

- [Luscher, 1971] Dr. Max Luscher, The Luscher Color Test, edited by Ian A. Scott, Pocket Books, New York, 1971
- [Айзенк, 2003] Айзенк Г. Новые IQ тесты. - М.: ЭКСМО, 2003. – 192с. -ISBN 5-04-006713-5
- [Jordan, 1984] Numerology: The Romance in Your Name. Dr. Juno Jordan, Jamie Grant Publisher: DeVorss & Company (January 1, 1984) ISBN-10: 0875162274
- [Goodman, 1984] Linda Goodman's Sun Signs. Linda Goodman. Published December 1st 1984 by Bantam. ISBN0553278827
- [Udry, 1994] Udry, J. Richard (November 1994). "The Nature of Gender". *Demography* 31 (4): 561–573. doi:10.2307/2061790. JSTOR2061790. PMID 7890091.
- [Крысько, 2002] Крысько В.Г. Этническая психология. - М.: Издательский центр "Академия", 2002. – 320с. ISBN 5-7695-0949-X
- [Денинг, 1984] Денинг В., Эсиг Г., Маас С. Диалоговая система «человек-ЭВМ». Адаптация к требованиям пользователя – М.: Мир 1984. – 110 с.
- [Gouyet, 1996] Gouyet, Jean-François (1996). *Physics and fractal structures*. Paris/New York: Masson Springer. ISBN 978-0-387-94153-0.

ADAPTIVE USER INTERFACES

Barlyabyev A.B. *

* *L.N.Gumilyov Eurasian National University, SRI
«Artificial Intelligence»,
Astana, Republic of Kazakhstan
frank-ab@mail.ru*

This paper examines the work of developed by us adaptive interface. It is one of the functions of intelligent electronic university. User data may affect to design and structure of the adaptive interface. Describes the process of adapting the interface based on the psychological informational model of user, in this process are generated new kinds of design and codes for interface style.

Introduction

The interface between humans and computers is a kind of communication channel. The purpose of this channel - without burdening unnecessary concerns and is not always popular knowledge to facilitate user interaction with some of the features of the program. There is a clear correlation between the levels of internal complexity of the interface and related knowledge, necessary for a person for using it. Here, the first problem is to adapt the structure of information flows and interface parameters to the needs of individual end-user. The second problem - the adaptation the user's to computer system.

An adaptive user interface - automatically changing the interface for the user's profile in order to preserve or achieve an optimal state with changing external

conditions with the help of their hardware and software tools.

For adaptive interface can be used: Luscher color test, a verbal test of Eysenck and others. But we will not discuss them, because the passage of various tests takes a lot of time and effort for users.

This paper examines the work of developed by us adaptive interface. It is one of the functions of intelligent electronic university. User data may affect to design and structure of the adaptive interface. Describes the process of adapting the interface based on the psychological informational model of user, in this process are generated new kinds of design and codes for interface style.

Main Part

Our purpose is the generation of adaptive interface based on the user's personal data, which takes into account the date of birth, gender, nationality, and other data to build a psychological informational model of user (hereinafter - the model of user).

Usually for determining the model of user used by various tests. But because of the long passage of tests, will be easier to identify the model, after learning date of birth and determining the user's horoscope. Adding to obtain a single digit, the digits of the day, month and year of birth can determine a person's character. Also the gender affects on the character, and person's nationality is greatly affects to his mentality.

If each user of system has its own model, which is different from other models of users, then such an approach is an individual. At stereotyped approach, by contrast, used the estimated user's belonging to a particular model (class), the amount of which is strictly limited. Usually create several models of users. I. Beaumont allocates 3 models: «model of beginner», «model of advanced user» and «model of expert». At individual approach is known beforehand which model will be generated for a particular user.

While none of the known interfaces can not be considered a perfect no with position accounting degree of adaptation, no with position of artificial intelligence, no with position of ergonomics (design, «transparency», convenience, etc.) no with professional (modifiability, availability of special functions).

System with adaptive interfaces can cause at the user a feeling of loss of control, may be some inaccuracies in the prediction of user behavior and his wishes, that can cause to the effect of «hostility» from the user's side. Therefore the final interface can be reconfigured by user.

Models of users.

Define the variable X_i - date of birthday, Y_j - sign of the zodiac (on a horoscope), W_k - gender, Z_l - nationality. Where $i = [1-9]$, $j = [1-12]$, $k = [1-2]$, $l = [1-195]$. $k = [1-2]$, digit 1 - male, digit 2 - female. $l = [1-195]$, the numbering of the list of members of the United Nations.

$$X_i Y_j W_k Z_l = [X_1 Y_1 W_1 Z_1 \dots X_9 Y_{12} W_2 Z_{195}]$$

$X_i Y_j W_k Z_l$ - identifier. Maximum possible number of the models of users is 42120. If the number of methods of calculating the information model of users will increase, then the possible maximum number of models of users will increase in geometric progression. Increase in a geometric progression will cause a problem for programmers. Necessary, that the adaptive interface computed these models of users by himself, thereby generating the necessary code for new designs. The following briefly describes the solution to this problem, using fractal theory in teaching and generating new types of the models of users. Figure 1 shows levels of elements of the adaptive interface.

What can be configured in the interface?

1. Position of menu;
2. View of menu (list, table, icon);
3. Palette of interface blocks (menus, header, main block, etc.);
4. Font and size of the text;
5. Size of the images;

Conclusion

Formed principles of generation and automation of adaptive interfaces in this paper make it easy to develop custom interfaces for different applications and facilitate the understanding of the structure of the interface and its implementation. Proposed an approach to solving the problem of self-learning in intelligent information systems.

In the future, we will try to improve our adaptive interface to intelligent interface and to clarify previously defined character of user we will use his avatar (photo) with the help of pattern recognition methods of graphic images.

As a result, we can get an online automated information agent, that can independently conduct a dialogue (oral and written) with users and used in a variety of intelligent information systems.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ВИРТУАЛЬНЫЙ ПОМОЩНИК АБИТУРИЕНТА ПРИ ВЫБОРЕ СПЕЦИАЛЬНОСТИ

Барлыбаев А.Б., Нургазина Г.Ш.

*Евразийский национальный университет имени Л.Н.Гумилева НИИ «Искусственный интеллект»,
г. Астана, Республика Казахстан*

frank-ab@mail.ru

Цель работы – автоматизация формирования ответа по запросу пользователя к базе знаний заданной предметной области. Предметная область – абитуриент выбирает специальность. База знаний предметной области представляется в виде семантической сети, состоящей из фактов и правил вывода. Требуется построить две семантические сети. Первая сеть строится экспертом, а вторая – абитуриентом, отвечая на вопросы, которые представлены на своих пользовательских интерфейсах. При этом функция решателя задач интеллектуальной системы сводится к выяснению вопроса «является ли семантическая сеть абитуриента частью семантической сети эксперта». Если да, то формируется ответ абитуриенту о подходящей ему специальности.

Ключевые слова: семантические сети, решатель задач, виртуальный помощник абитуриента, e-learning.

Введение

Семантическая сеть – это взвешенный ориентированный граф, в котором весами вершин являются понятия, события, объекты или субъекты, а весами дуг – отношения между вершинами. Любой фрагмент сети, например, одна вершина, две вершины и дуги между ними называют подсетью. Логический вывод (поиск решения) на семантической сети заключается в том, чтобы найти или сконструировать подсеть, удовлетворяющую некоторым условиям [Roussopoulos, 1976] [Quillian, 1968] [Sowa, 1987].

Отношения, представляемые дугами, в семантической сети могут быть различными. Примеры таких отношений приведены в таблице 1.

Типы отношений выбираются в зависимости от вида семантической сети (таблица 2) и решаемой задачи.

Таблица 1. Основные виды отношений в семантических сетях.

Тип	Описание
Является наследником (a-kind-of)	Задаёт иерархические связи между классами
Является экземпляром (is-a)	Определяет значение, описывает конкретный объект, понятие
Это есть	Может использоваться

(are)	вместо связи a-kind-of в отношениях подразумевающих равенство или эквивалентность
Быть частью (has-part)	Определяет структурные связи, описывает части или целые объекты
Функциональные или предикативные	Определяются обычно глаголами, отражают различные действия или отношения (учить, владеть и т.д.).
Количественные	Отображают количественные соотношения между вершинами (больше, меньше и т.д.)
Пространственные	Отображают пространственные отношения между вершинами (близко, далеко и т.д.)
Временные	Описывают временные связи между вершинами (скоро, долго, сейчас и т.д.)
Атрибутивные	Описывают свойства объектов, понятий
Логические	Описывают логические связи между вершинами (и, или, не)

Здесь глаголы могут быть разного падежа в зависимости от типов вершин и вида направленности отношения между ними, например: Субъект – учитель, субъект – ученик. «Учитель *учит* ученика», либо «ученик *учится* у учителя».

Таблица 2. Типы семантических сетей.

Тип	Описание
По типу знания	
Экстенциональные (терминальные)	Описывает конкретные отношения над конкретными значениями.
Интенциональные (нетерминальные)	Описывают имена понятий, событий, объектов и субъектов, а не их индивидуальные имена или значения, и отношения между ними.
По типу ограничений на вершины и дуги	
Простые	Вершины сети не обладают внутренней структурой (терминальные)
Иерархические	Вершины сети обладают внутренней структурой (нетерминальные).
Динамические (сценарии)	Вершины сети представляют некоторые действия или события
По количеству типов отношений	
Однородные	Обладают только одним типом отношений
Неоднородные	Количество типов отношений больше двух
По арности отношений	
Двуместные (Бинарные)	Все отношения в сети связывают ровно две вершины
Многоместные (n -арные, где $n > 2$ - число вершин)	В сети есть отношения, связывающие более двух объектов

Здесь *события* представляют действия, происходящие в реальном мире, и определяются указанием типа действия и ролей, которые играют объекты в этом действии.

Заметим, что в иерархической сети есть возможность разделять сеть на подсети и устанавливать отношения не только между вершинами, но и между подсетями (различные подсети, существующие в сети, могут быть упорядочены в виде дерева подсетей, вершины которого - подсети, а дуги - отношения видимости).

Важную роль в построении семантической сети играет предметная область, которая определяется следующим образом:

1. Определить абстрактные объекты и понятия предметной области, необходимые для решения поставленной задачи. Оформить их в виде вершин.

2. Задать свойства для выделенных вершин, оформив их в виде вершин, связанных с исходными вершинами атрибутивными отношениями.

3. Задать связи между этими вершинами, используя функциональные, пространственные, количественные, логические, временные, атрибутивные отношения, а также отношения типа «являться наследником» и «являться частью».

4. Добавить конкретные объекты и понятия, описывающие решаемую задачу. Оформить их в виде вершин, связанных с уже существующими отношениями типа «являться экземпляром», «есть».

5. Проверить правильность установленных отношений (вершины и само отношение при правильном построении образуют предложение, например «Двигатель является частью автомобиля»).

1. Проектирование виртуального помощника с помощью семантической сети

Задача 1. Построить сетевую модель представления знаний в предметной области «Абитуриент выбирает специальность». Выполняя вышеуказанные пункты 1-5, эту предметную область можно определить как:

1. Ключевые понятия данной предметной области – специальность, абитуриент. Абитуриент выбирает специальность для поступления. Специальность является частью группы специальностей. Группа специальностей часть уровней образования. При выборе специальности абитуриент использует характеристики имеющее на данный момент: образование и позицию (работу) на которой он хотел бы трудиться. Нам необходимо определить к какой части сектора труда относится позиция. В зависимости какое образование есть у абитуриента на данный момент, важную роль играет 4-тый предмет, сдаваемый при вступительных экзаменах (единое национальное тестирование). 4-тый предмет часть среднего образования. Важную роль играет также имеющийся у абитуриента уровень высшего образования и специальность. Исходя из этого, вершины графа будут следующими: «Абитуриент», «Специальность», «Образование», «Сектор труда», «Позиция», «4-тый предмет», «Уровень образования», «Уровень образования и специальность» и «Группа специальностей».

2. У этих объектов есть определенные свойства и атрибуты. Например, образование может

быть средним или высшим, уровень образования может быть бакалавриатом, магистратурой или докторантурой PhD. Поэтому добавим вершины «Среднее образование», «Высшее образование», «Бакалавриат», «Магистратура» и «Докторантура PhD».

3. Определим для имеющихся вершин отношения и их типы, используя таблицу 2. Полученный в результате граф изображен на рис. 1.

4. Добавим знание о конкретных фактах решаемой задачи. Пусть имеется два абитуриента: «Марк» и «Алекс», у первого есть только среднее образование, а у второго высшее образование. Марк при сдаче Единого Национального Тестирования выбрал 4-тым предметом физику. Алекс имеет высшее образование бакалавр экономики. Есть две группы специальностей «Социальные науки, экономика и бизнес» и «Естественные науки». У первой группы специальностей есть специальности «менеджмент» и «учет и аудит», у второй только «информатика». Каждая группа специальностей имеет весь набор уровней образования: бакалавриат, магистратура, докторантура PhD. Марк хочет работать в будущем программистом. А Алекс хочет быть топ-менеджером. Исходя из этого, добавим соответствующие вершины в граф и соединим их функциональными отношениями и отношениями типа «например или является экземпляром». Стоит учесть, что отношения «например» и «является экземпляром» эквивалентны. Полученный в результате граф изображен на рис. 2.

5. Осуществим проверку установленных связей. Например, возьмем вершину «Специальность» и пройдем по установленным связям. Получаем следующую информацию: специальность является частью группы специальности, примерами специальности могут служить менеджмент, учет и аудит, информатика.

Задача 2. Что должен делать абитуриент, чтобы выбрать нужную специальность. Для решения этой задачи нужно использовать Правила приема в высшие учебные заведения Республики Казахстан [Правительство РК, 2012].

Следуя правилам, абитуриент с средним образованием может поступать в высшие учебные заведения на специальности бакалавриата в зависимости от 4-го предмета, определяющего направление обучения (группу специальностей). А абитуриент, с высшим образованием, может поступать на все направления специальностей в бакалавриате. Для поступления на специальность магистратуры необходимо иметь образование бакалавриата по данному направлению обучения (группа специальности). Для поступления на специальность докторантуры PhD необходимо иметь образование магистратуры по данному направлению обучения (группа специальности).

Также абитуриенту необходимо определить

сектор труда и будущую профессию, по которой он в дальнейшем будет трудиться.

Чтобы выбрать правильную специальность, учитывая выше упомянутые факторы, абитуриент заполняет анкету-форму. Абитуриент отвечает на вопросы:

1. Вопрос «какое имеет образование?»
Варианты ответа: «среднее», «высшее».

1.1 Если выбрано «среднее», то вопрос «какой 4-тый предмет?»

1.2 Если выбрано «высшее», то вопрос «какой уровень образования и какая специальность?»

2. Вопрос «какой сектор труда?» и «какая позиция?»

На основе ответов строится семантическая сеть «анкета-форма». Далее полученные данные передаются в решатель задач.

Решатель задачи – процедура, которая выясняет, является ли построенная сеть «анкета-форма» частью «Абитуриент выбирает специальность».

Для получения ответа на какой-либо вопрос по этой задаче, необходимо найти соответствующий участок сети и, используя связи, получить результат.

Например, вопрос «Имея высшее образование бакалавр экономики, и имея желание работать топ менеджером, какую специальность должен выбрать Алекс?» Из запроса понятно, что необходимо найти следующие вершины: «Алекс», «Специальность», «Высшее образование», «Бакалавр экономики» и «Топ-менеджер». Часть семантической сети, находящаяся между этими вершинами, содержит ответ, а именно, высшее образование бакалавр экономики дает возможность поступать на все специальности бакалавриата, и на специальности магистратуры социальных наук и экономики и бизнеса. Соответственно, Алекс может поступить на бакалавриат специальностей информатики, учет и аудит, менеджмент, а также на магистратуру учет и аудит, менеджмент. Но желаемая позиция связана с одной специальностью, менеджментом. Больше информации в модели нет, поэтому делаем вывод – Алекс может поступить на бакалавриат или магистратуру специальности менеджмент.

Задача 3. Описание работы «виртуального помощника абитуриента при выборе специальности». На рис. 3 представлена блок-схема работы виртуального помощника.



Рис. 1. Интенсиональная (нетерминальная) семантическая сеть предметной области «Абитуриент выбирает специальность».

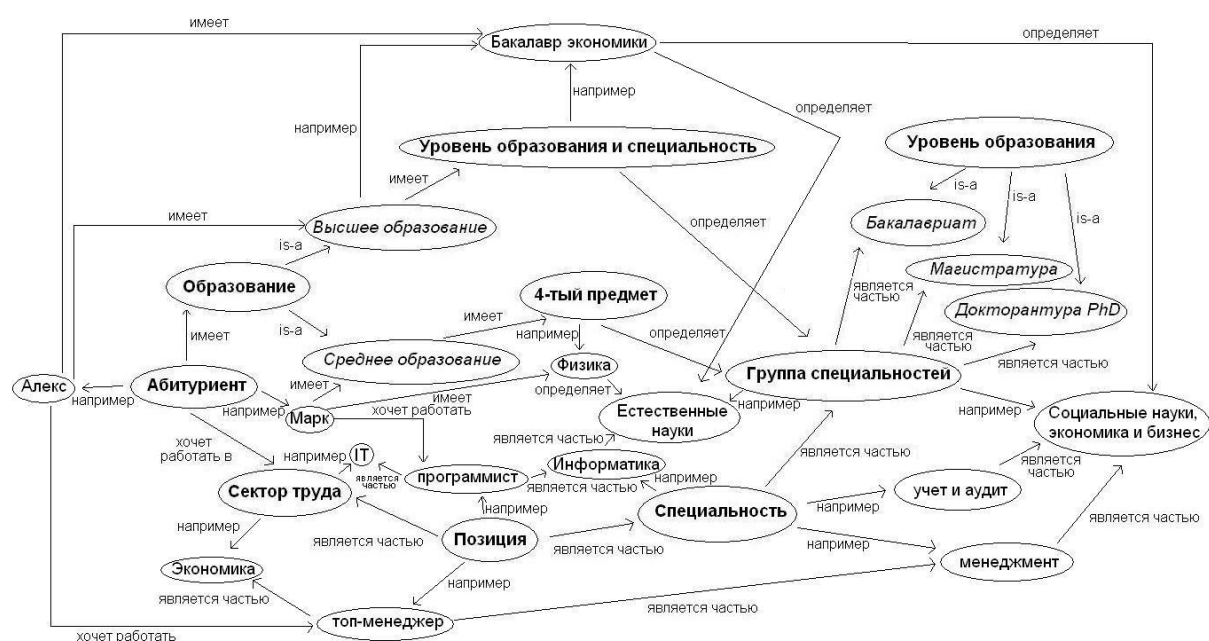


Рис. 2. Экстенсиональная (терминальная) семантическая сеть предметной области «Абитуриент выбирает специальность».

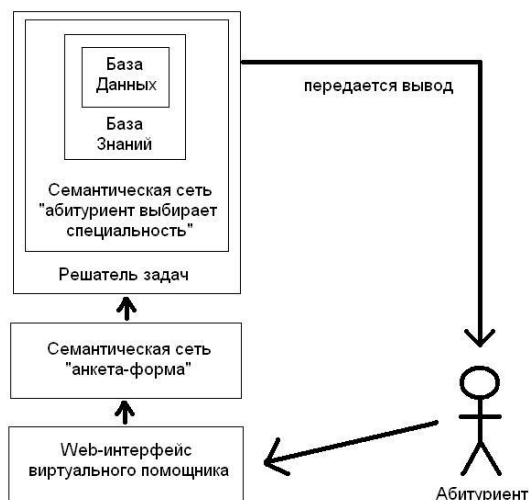


Рис. 3. Блок-схема виртуального помощника.

Абитуриент с помощью веб-интерфейса виртуального помощника строит свою семантическую сеть «анкета-форма». Построенная модель передается в решатель задач. Решатель задач состоит:

1. База данных, атомарный блок, в котором хранятся данные, факты.
2. База знаний, блок в котором хранятся правила обработки данных, фактов.
3. Семантическая сеть «абитуриент выбирает специальность», строится на основе базы данных и базы знаний.

Решатель задач сравнивает семантические сети «анкета-форма» и «абитуриент выбирает специальность». Если решатель задач выясняет, что построенная сеть «анкета-форма» является частью «абитуриент выбирает специальность», то запускается поиск решения по семантической сети. Решением будет промежуточные вершины, которые зависят от семантической сети «анкета-форма», но при этом существуют в семантической сети «абитуриент выбирает специальность». Подробнее описано в рис. 4.

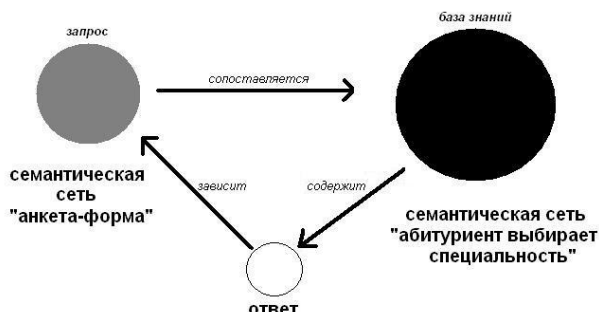


Рис. 4. Визуализация поиска решения в семантической сети.

Ниже будет построен алгоритм решателя задач, который сравнивает семантические сети «анкета-форма» и «абитуриент выбирает специальность».

Опишем следующие объекты, которые будут

использоваться при построении данного алгоритма в таблице 3.

Таблица 3. Описание объектов семантических сетей, используемых в решателе задач.

№	Обозначение объекта	Описание
1	->	Оператор связанно
2	∈	Оператор содержит
3	E	Образование
4	HE	Высшее образование
5	GS	Группа специальности
6	LEaS	Уровень образования и специальность
7	LE	Уровень образования
8	AGS	Список групп специальностей абитуриента
9	LM	Сектор труда
10	P	Позиция
11	ALM	Сектор труда абитуриента
12	AP	Позиция абитуриента
13	S	Специальность
14	AS	Список подходящих специальностей для абитуриента
15	FS	Четвертый предмет
16	AFS	Четвертый предмет абитуриента

НАЧАЛО АЛГОРИТМА «Решатель задач»

Исходные данные:

1. Семантическая сеть абитуриента анкета-форма, которая состоит из объектов (вершин): LEaS, ALM, AP, AFS;

2. Семантическая сеть эксперта абитуриент выбирает специальность, которая состоит из объектов (вершин): E, HE, GS, LE, LM, P, S, FS;

Результирующие данные:

1. Да: AS;

2. Нет: выполняется команда, выводящая сообщение «просим прощения, но у нас нет подходящей специальности для вас»; выполняется команда quit 0, закрывающая действие алгоритма;

Метод:

Сравнение семантической сети абитуриента с семантической сетью эксперта

Проверка на имеющиеся образование

Если высшее образование, То формируется AGS в зависимости от LEaS

Иначе формируется AGS в зависимости от AFS

Фильтрация AGS в зависимости от ALM и AP

$AGS \in GS \rightarrow S$

Если $S \rightarrow P = AP$ and $P \rightarrow LM = ALM$, То $AS = S$

Если они совпадают или $AS' = ""$, То ДА

Иначе НЕТ

КОНЕЦ АЛГОРИТМА «Решатель задач»

Этот алгоритм был реализован на языке

программирования *Cache Object Script (Mumps)* с использованием технологии *постреляционной СУБД Cache Intersystems* [Intersystems, 2014].

Заключение

Даны определения семантической сети. Спроектирована работа виртуального помощника абитуриента при выборе специальности. При проектировании были использованы методы нахождения информации в семантических сетях. Была спроектирована работа и структура решателя задач в семантических сетях. Построен алгоритм решателя задач, который сравнивает семантическую сеть эксперта с семантической сетью абитуриента. Написана программная реализация данного алгоритма. Проведены начальные экспериментальные наблюдения работы алгоритма. Данная проблема может быть расширена до решения задачи улучшения эффективности системы образования с учетом региональных трудовых ресурсов.

Библиографический список

[Roussopoulos, 1976] Roussopoulos N.D.A semantic network model of data bases. – TR No 104, Department of Computer Science, University of Toronto, 1976.

[Quillian, 1968] Quillian, M. R. (1968). Semantic memory. Semantic information processing, 227–270.

[Sowa, 1987] John F. Sowa (1987). "Semantic Networks". In Stuart C Shapiro. Encyclopedia of Artificial Intelligence. Retrieved 2008-04-29.

[Правительство РК, 2012] Типовые правила приема на обучение в организации образования, реализующие профессиональные учебные программы высшего образования. Утверждены постановлением Правительства Республики Казахстан от «19» января 2012 года № 111 (с изменениями от 19 апреля 2012 года № 487)

[Intersystems, 2014] <http://www.intersystems.com/cache/>

THE VIRTUAL ASSISTANT FOR APPLICANT IN CHOOSING OF THE SPECIALTY

Barlyabyev A.B. *, Nurgazinova G.Sh. *

* L.N.Gumilyov Eurasian National University, SRI
«Artificial Intelligence»,
Astana, Republic of Kazakhstan

frank-ab@mail.ru

This article's purpose is automation of answer formation to the user's request to the domain's knowledge base. The domain –the applicant chooses a specialty. Domain knowledge base is represented as a semantic network consisting of facts and rules of conclusion.

Introduction

The semantic network is the weighted oriented graph, in which vertex weights are concepts, events, objects or entities, and weights of arcs - the relation between the vertices. Any fragment of network, such as one vertex, two vertices and arcs between them is called subnet. The logical conclusion (decision search) on the

semantic network consists in finding or designing a subnet meeting some conditions.

Main Part

START OF THE ALGORITHM “problem solver”

Background:

The applicant's semantic network the profile-form which consists of objects (vertices): LEaS, ALM, AP, AFS;

The expert's semantic network the applicant chooses a specialty which consists of objects (vertices): E, HE, GS, LE, LM, P, S, FS;

The resulting data:

Yes: AS;

No: there are executing the command that displays the message “Sorry, but we do not have a suitable specialty for you”, quit 0 command is executed, this is stop operation of the algorithm;

Method:

Comparison of applicant's semantic network with a semantic network of the expert

Check on available education

If higher education, Then AGS is formed depending on LEaS

Else, AGS is formed depending on AFS

AGS filtration depending on ALM and AP

AGS ∈ GS → S

If S → P = AP and P → LM = ALM, then AS = S

If they are identical or AS' = “”, Then YES

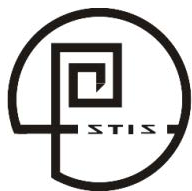
Else NO

END OF THE ALGORITHM “problem solver”

This algorithm has been implemented in the Cache Object Script (Mumps) programming language with using of technology of post-relational DBMS Cache Intersystems.

Conclusion

Definitions of the semantic network are given. The applicant's virtual assistant operation in choosing specialty is designed. At designing methods of information searching in semantic networks were used. It was projected operation and structure of the problem solver in semantic networks. The algorithm of a problem solver is constructed which compares the expert's semantic network with the semantic network of the applicant. Program realization of this algorithm is written. The initial experimental observations of the algorithm is conducted. This problem can be expanded up to solve the problems of the improvement the efficiency of the education system according to regional labor resources.



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.432.4

СОЗДАНИЕ ОНТОЛОГИЧЕСКОЙ МОДЕЛИ НАЛОГОВОЙ СИСТЕМЫ КАЗАХСТАНА

Шарипбаев А.А., Омарбекова А.С., Турмаганбетова Ш.К.

Евразийский национальный университет имени Л.Н.Гумилева,

г. Астана, Республика Казахстан

sharalt@mail.ru

omarbekova@mail.ru

sh_kurmangali@mail.ru

В рамках диссертационной работы определены основные понятия создаваемой онтологической модели, которая позволит организовать процесс реализации экспертной системы налоговой системы в Республике Казахстан. Таким образом, в рамках данной работы будет создана модель онтологии, обеспечивающая соответствующие условия работы системы и представляющая собой реализацию веб - приложения, которое использует технологии Semantic Web, и представляет собой интерфейс вывода машины экспертной системы в виде готового эксперта.

Ключевые слова: онтологическая модель, Semantic Web, налоговая система.

Введение

На базе Евразийского национального университета им. Л.Н. Гумилева проводятся исследования по созданию экспертной системы, которая на интеллектуальном уровне на основе базы знаний позволит самостоятельно выступить в роли эксперта.

В результате каждый гражданин в режиме онлайн сможет получить ответ в сфере налоговой системы Казахстана. Такой вид поддержки экономически более выгодно оказывать удаленно, в автоматическом режиме, то есть использовать формализованные знания эксперта. Знания, обеспечивающие техническую поддержку для различных предметных областей, имеют схожую структуру. Для их представления рационально использовать онтологии, и другие технологии Semantic Web.

1. Построение онтологии налога

Пользователь данной системы консультационной поддержки, как правило, не знает всех причинно-следственных связей используемой им системы, и имеет возможность только задавать вопросы эксперту. Вследствие этого при обращении к поисковой системе пользователь должен иметь возможность получить в ответ ресурсы релевантные смыслу запроса, то их поиск должен быть семантически

ориентированным. Для этого средства поиска соответствующей запросу информации предлагается организовать на основе онтологии, содержащей описания семантики ресурсов.

Задача экспертной системы поддержки – по неполной введенной информации, однозначно распознать вопрос и выдать пользователю полноценный ответ. Причем интеллектуальность системы зависит от того насколько быстро система сможет ответить на поставленный вопрос.

В области компьютерных наук и информатики, онтология формально представляет собой знания в виде набора понятий внутри домена, и отношения между парами понятий. Она может быть использован для моделирования и поддержки доменных рассуждения о понятиях.

Если рассматривать с позиции теории, онтология является формализация некоторой области знаний с помощью концептуальной схемы. Онтология обеспечивает общий словарь, который можно использовать для моделирования доменов, то есть тип объектов и/или концепций, которые существуют, и их свойств и отношений.

Онтологии структурных рамок для организации информации используются в области искусственного интеллекта, Semantic Web, инженерных систем, разработка программного обеспечения, биомедицинской информатики, библиотековедения и информационной

В результате исследований была построена онтологическая модель налоговой системы Казахстана (рисунок 1). В области искусственного интеллекта, экспертная система представляет собой компьютерную систему, которая эмулирует принятия решений способности человека-эксперта [1].

Экспертная система имеет уникальную структуру, таким образом, отличаясь от традиционного программирования. Она разделена на

Вследствие этих объективных обстоятельств весьма желательно, чтобы информация и знания были структурированы и описаны таким образом, чтобы получатель (пользователь) был способен понять и текст, и контекст (смысл) сообщения. В идеале, сообщение (знаниевая сущность) должна структурироваться таким образом, чтобы компьютер, а не только образованный человек был способен «понять» его. Под словом «понять» здесь имеется в виду, что компьютер будет способен обработать документ (знаниевую сущность) посредством использования известных ему правил с помощью некоторого логического языка, а также будет способен вывести новые факты и знания из данного документа (рисунок 2).

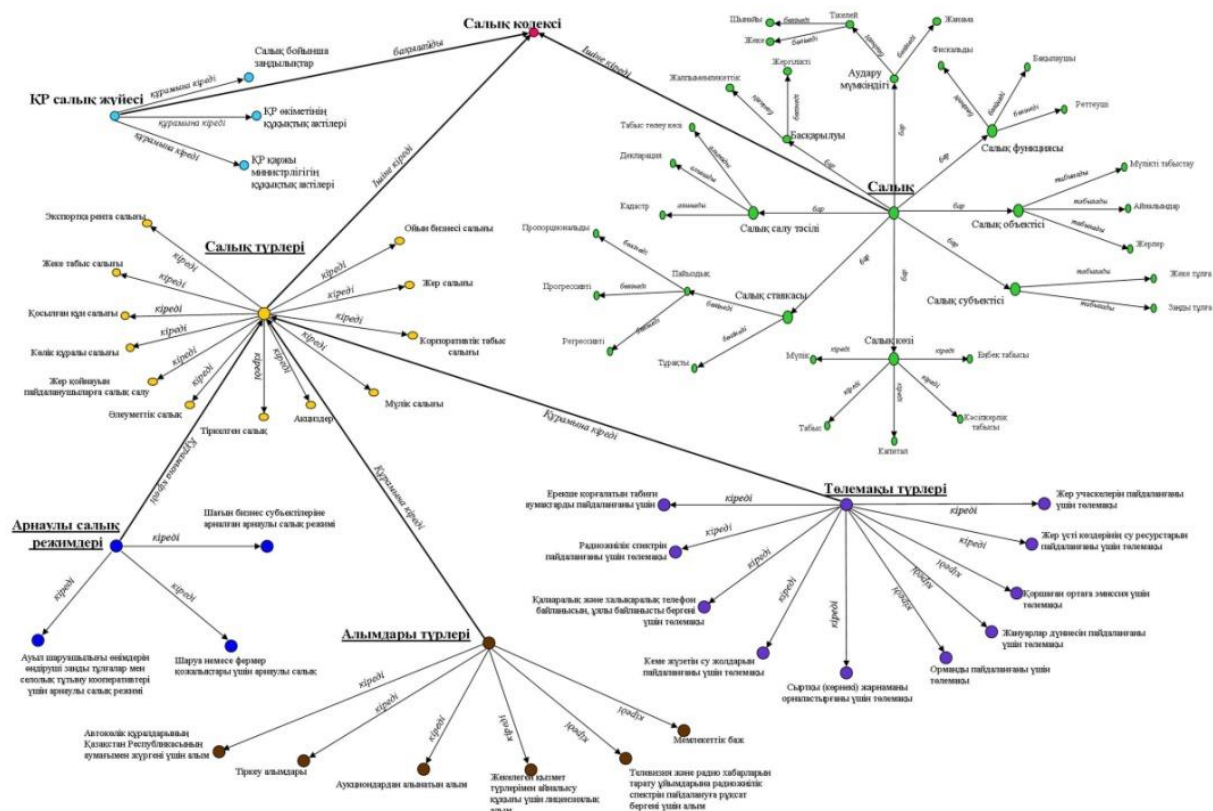


Рисунок 1 – Онтологическая модель налоговой системы Казахстана



Рисунок 2 – Схема функционирования системы показывает, как основные показатели взаимодействуют друг с другом в работе обработки

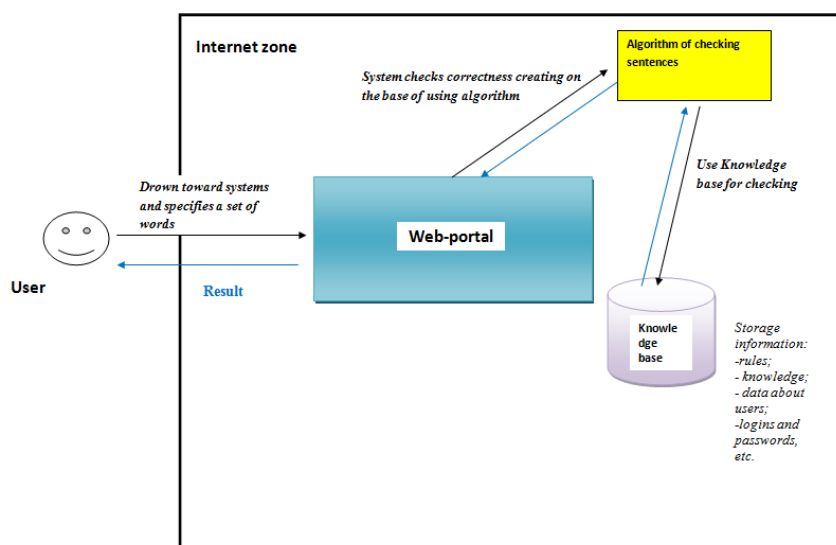


Рисунок 3 – Главные элементы системы, которые передают информацию друг другу до получения необходимого результата

Таким образом, система сможет выполнять следующие функции:

- формирование и фиксация общего разделяемого всеми экспертами знания о предметной области;
- явная концептуализация программного обеспечения, позволяющая описывать семантику данных;
- возможность переиспользования знаний;
- интеграция и возможность совместного использования разнородных данных и знаний в рамках одной системы;
- обеспечение лучшего понимания предметной области пользователями системы.

Система в ответ на конкретный вопрос, автоматически собирает ваш ответ и алгоритм строит на основе этого все необходимые правила,

связанные с задаваемым вопросом, с учетом конкретного пользователя (рисунок 3).

Система включает в себя базу знаний и три подсистемы: подсистема управления, интерфейс с пользователем, подсистема логического вывода. Все подсистемы представляют собой отдельные программные модули, работающие в составе общей исполняемой программы.

Система реализована многооконный интерфейс на основе меню. В ходе диалога, система отображает различные виды компонентов. Результаты ответа системы отображаются на экране. Вся система разработана таким образом, что пользователь может работать с ним без специальной подготовки.

Таким образом, мы определили основные элементы в созданной онтологической модели налоговой системы Республики Казахстан. Существуют внешние элементы, которые связаны с новой системой, далее отображены внутренние

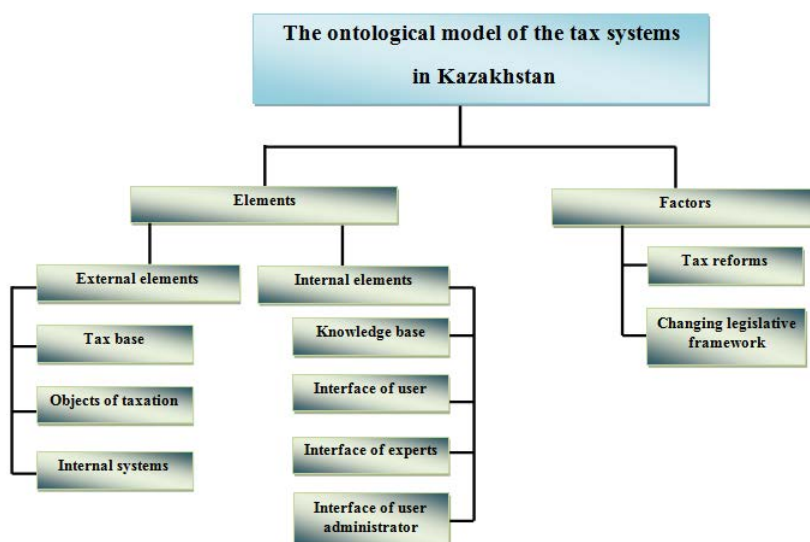


Рисунок 4 - Онтологическая модель налоговой системы в Республике

элементы системы. Также выделены основные факторы, которые влияют на изменение системы во временном пространстве (рис. 4).

Казахстан, которая состоит элементов (внутренних и внешних) и факторов влияющих на систему.

Заключение

В результате исследований мы создали модель онтологии налоговой системы Казахстана, которая позволит улучшить существующую налоговую систему в нашем государстве и реализовать экспертную систему и организовать связь граждан с правительством.

Библиографический список

- [Berners-Lee,2001] T., Hendler, J., Lassila, O.] The Semantic Web // Scientific American, May 17, 2001.
 [Stewart S.] Karlinsky and Daniel E. O'Leary. Tax-based expert systems: a first principles approach. Expert Systems in Finance D.E.1992] O'Leary and P.R. Watkins (Editors) 1992 Elsevier Science Publishers B.V.

CREATING AN ONTOLOGICAL MODEL FOR THE TAX SYSTEM IN KAZAKHSTAN

Sharipbaev A., Omarbekova A.,
Turmaganbetova S.

*L.N. Gumilyov Eurasian National University
Astana, Republic of Kazakhstan*

sharalt@mail.ru

omarbekova@mail.ru

sh_kurmangali@mail.ru

As result of research we have created ontology model of the tax system in Kazakhstan, which will improve the existing tax system in our state and to develop the expert systems that could link citizens with government.

As a result of introduction of the system the following tasks will be realized:

- automation of work process with citizens.
- transparency of the system
- lack of the human factor

АВТОРСКИЙ УКАЗАТЕЛЬ

Абрамов Г.А.	527	Елохов Е.С.	513, 517
Алсеитова А.Т.	411	Еремеев А.П.	461
Андрушеви А.	353	Ермоленко Т.В.	161
Анисимова Т.В.	523	Ефименко И.В.	29
Антипова О.В.	233	Жилякова Л.Ю.	239
Апанель Е.Н.	487	Житко В.А.	103
Барлыбаев А.Б.	553, 559	Жонин А.А.	161
Бердник В.Л.	439	Жукевич А.И.	537
Бойко И. М.	357	Заболеева-Зотова А.В.	431, 435, 439
Боргест Н.М.	155, 527, 531	Загорулько Г.Б.	117
Борисов А.Н.	97	Загорулько Ю.А.	117
Бурдо Г.Б.	195	Замятина Е.Б.	369, 375
Вакурина Т.Г.	221	Захарьев В.А.	453
Валеев М.Т.	517	Зуенко А.А.	251
Веремьев В.Л.	363	Иванов Р.А.	277
Вереник Н.Л.	211	Кабенов Д.И.	547
Виноградов Г.П.	227	Каешко А.И.	447
Вишняков В.А.	391	Канделинский С. Л.	357
Войцехович Г.Ю.	487	Катаев В.А.	427
Воробьева Е.В.	195	Кипаева Е.В.	431
Галушка И.Н.	491	Кириченко М.И.	431
Гатауллин Р.Р.	503	Клещев А.С.	131, 281, 285
Гатиатуллин А.Р.	173	Климов А.С.	443
Гецэвіч Ю.С.	333, 457	Климов В.С.	423
Гильмуллин Р.А.	503	Козлов О.А.	395
Гирель А.И.	211	Колб Д.Г.	447
Гладун А.	353	Комарцова Л.Г.	233
Глазкова А.В.	503	Коровин М.Д.	527
Глоба Л.С.	137, 297	Королев Ю.И.	461
Голенков В. В.	65	Корончик Д. Н.	79
Головко В.А.	487	Котеля В.В.	221
Гордей А.Н.	49	Кривдюк Н.М.	303
Горовая Д.О.	363	Кулинич А.А.	243
Горюнов В.А.	537	Курбатов С.С.	309, 315
Гракова Н.В.	83	Курбатский А.	353
Грибова В.В.	131	Лавренков Ю.Н.	233
Гулякина Н.А.	65	Ландэ Д.В.	199
Давыденко И.Т.	83	Ланин В.В.	381, 513
Данилин Н.С.	217	Лахин О.И.	221
Данко А.И.	423	Лещев Д.В.	483
Демиров В.В.	543	Лещева И.А.	483
Димитров Д.М.	217	Литвинович А.В.	309
Елисеева О.Е.	321	Лобанов Б.М.	103

Лобанов В.Н.	205	Снарский А.А.	199
Лобзин А.П.	309, 315	Соловьев С.Ю.	169
Лукашевич Н.В.	185	Солошенко А.Н.	435
Лысаковский И.А.	531	Солошич С.Н.	499
Лядова Л.Н.	375	Соснин П.И.	147
Мальковский М.Г.	169	Старжинский В.П.	403
Массель А.Г.	111, 277	Стефанюк В.Л.	191
Массель Л.В.	111	Сторож В.В.	415
Мастыкин А.С.	487	Стрижак А.Е.	297
Матюшин М.М.	221	Субхангулов Р.А.	385
Михайлов Ю.Ф.	395	Сулейманов Д.Ш.	173, 503
Михеев Р.А.	369	Сухов А.О.	375
Моросин О.Л.	465	Тарасов В.Б.	39, 471
Наместников А.М.	385	Татур М.М.	211
Невзорова О.А.	179	Терновой М.Ю.	297
Нестеров Р.А.	523	Тертышный В.А.	495
Ниязова Р.С.	411	Тимченко В.А.	291
Новикова А.О.	423	Турмаганбетова Ш.К.	565
Новогрудская Р.Л.	137	Узлов Д.Ю.	267
Нургазинова Г.Ш.	559	Узунова Е.Н.	517
Окрут Т.И.	333	Федотова А.В.	471
Омарбекова А.С.	565	Филатова Н.Н.	227
Орлова Ю.А.	431, 443	Фоменкова М.А.	443
Островский С.А.	297	Хайрова Н.Ф.	267
Пакладок Д.А.	457	Харламов А.А.	161
Палюх Б.В.	195	Хахалин Г.К.	309, 315
Петровский А.А.	453	Хорошевский В.Ф.	29
Петровский А.Б.	205	Черепанов Н.В.	471
Печенежский А.Б.	523	Чугунов А.П.	381
Плесневич Г.С.	143	Шалфеева Е.А.	285
Попова М.А.	297	Шаповал И.С.	495
Прокопчук Ю.А.	271	Шарипбаев А.А.	565
Разахова Б.Ш.	547	Шаронова Н.В.	267
Рогущина Ю.В.	339, 353	Шихнабиева Т.Ш.	399
Родченко В.Г.	537	Шункевич Д.В.	93
Розалиев В.Л.	435, 443	Шчурко М.Л.	457
Русецкий К.В.	83	Щербак С.С.	491, 495
Сабиров И.Х.	217	Югов А.С.	513
Сейткулов Е.Н.	211	Югов А.С.	517
Скобелев П.О.	221	Ягунова Е.В.	199
Скопинава А.М.	333	Ямшанов А.В.	303
Смагин С.В.	281	Янковская А.Е.	303
Сметанин Ю.М.	257, 263		

AUTHOR INDEX

Abramov G.A.	527	Kabenov D.I.	547
Alseitova A.A.	411	Kandelinsky S.L.	357
Andrushevich A.	353	Kataev V.A.	427
Anisimova T.V.	523	Kayeshko A.I.	447
Antipova O.V.	233	Khairova N.	267
Apanel E.N.	487	Khakhalin G.K.	309, 315
Barlyabyev A.B.	553, 559	Kharlamov A.A.	161
Berdnik V.L.	439	Khoroshevsky V.F.	29
Borgest N.M.	155, 527, 531	Kipaeva E.V.	431
Borisov A.N.	97	Kirichenko M.I.	431
Boyko I.M.	357	Kleshev A.S.	131, 281, 285
Burdo G.B.	195	Klimov A.S.	443
Cherepanov N.V.	471	Klimov V.S.	423
Chugunov A.P.	381	Kolb D.G.	447
Danilin N.S.	217	Komartsova L.G.	233
Danko A.I.	423	Korolev Y.I.	461
Davydenko I.T.	83	Koronchik D. N.	79
Demirov V.V.	543	Korovin M.D.	527
Dimitrov D.M.	217	Kotelya V.	221
Efimenko I.V.	29	Kozlov O.A.	395
Elokhov E.S.	513, 517	Krivdyuk N.M.	303
Eremeev A.P.	461	Kulinich A. A.	243
Fedotova A.M.	471	Kurbatov S.S.	309, 315
Filatov N.N.	227	Kurbatski A.	353
Fomenkova M.A.	443	Lakhin O.	221
Galushka I.N.	491	Lande D.V.	199
Gataullin R.R.	503	Lanin V.V.	381, 513
Gatiatullin A. R.	173	Lavrenkov Ju.N.	233
Gilmullin R.A.	503	Leshchev D.V.	483
Girel A.I.	211	Leshcheva I.A.	483
Gladun A.	353	Litvinovich A.V.	309
Glazkova A.V.	503	Lobanov B.M.	103
Globa L.S.	137, 297	Lobanov V.N.	205
Golenkov V.V.	65	Lobzin A.P.	309, 315
Golovko V.A.	487	Loukachevitch N.V.	185
Gorovaya D.O.	363	Lyadova L.N.	375
Grakova N.V.	83	Lysakovkiy I.A.	531
Gribova V.	131	Malkovsky M.G.	169
Guliakina N.A.	65	Massel A.G.	111, 277
Hardzei A.	49	Massel L.V.	111
Harunou V.A.	537	Mastykin A.S.	487
Hetsevich Yu.S.	333, 457	Matushin M.	221
Ivanov R.A.	277	Mikhailov J.F.	395

Mikheev R.A.	369	Soloshych S.N.	499
Morosin O.L.	465	Soloviev S.Y.	169
Namestnikov A.M.,	385	Sosnin P.I.	147
Nesterov R.A.	523	Starzhinsky V.P.	403
Nevzorova O.A.	179	Stefanuk V.L.	191
Niyazova R.S.	411	Storozh V.V.	415
Novikova A.O.	423	Stryzhak O.Y.	297
Novogrudskaya R.L.	137	Subkhangulov R.A.	385
Nurgazinova G.Sh.	559	Sukhov A.O.	375
Okrut T.I.	333	Suleymanov D. Sh.	173, 503
Omarbekova A.	565	Tarassov V. B.	39, 471
Orlova Yu.A.	431, 443	Tatur M.M.	211
Ostrovskiy S.A.,	297	Ternovoy M.Y.	297
Pakladok D.A.	457	Tertishniy V.A.	495
Palyukh B.V.,	195	Timchenko V.A.	291
Pechenezhskiy A.B.	523	Turmaganbetova S.	565
Petrovsky A.A.	453	Uzlov D.	267
Petrovsky A.B.	205	Uzunova E.N.	517
Plesniewicz G.S.	143	Vaitsekhovich H.Y.	487
Popova M.A.	297	Vakurina T.	221
Prokopchuk I.A.	271	Valeev M.T.	517
Razahova B.Sh.	547	Veremyev V.L.	363
Rodchenko V.G.	537	Verenik N.L.	211
Rogushina J.	339, 353	Vinogradov G.P.	227
Rozaliev V.L.	435, 443	Vishniakou U.A.	391
Rusetski K.V.	83	Vorobyeva E.V.	195
Sabirov I.Kh.	217	Yagunova E.V.	199
Seitkulov Y.N.	211	Yamshanov A.V.	303
Shalfeeva E.	285	Yankovskaya A.E.	303
Shapoval I.S.	495	Yeliseyeva O.E.	321
Sharipbaev A.	565	Yermolenko T.V.	161
Sharonova N.	267	Yugov A.S.	513, 517
Shcherbak S.S.	491, 495	Zaboleeva-Zotova A.V.	431, 435, 439
Shchurko M.L.	457	Zagorulko G.B.	117
Shicknabieva T.SH.	399	Zagorulko Yu.A.	117
Shunkevich D.V.	93	Zahariev V.A.	453
Skobelev P.	221	Zamyatina E.B.	369, 375
Skopinava A.M.	333	Zhilyakova L.Yu.	239
Smagin S.V.	281	Zhitko V.A.	103
Smetanin Yu. M.	257, 263	Zhonoin A.A.	161
Snarskii A.A.	199	Zhukevich A.I.	537
Soloshenko A.N.	435	Zuenko A.A	251

Научное издание

Открытые семантические технологии проектирования интеллектуальных систем

OSTIS-2014

Open Semantic Technologies for Intelligent Systems

МАТЕРИАЛЫ
IV МЕЖДУНАРОДНОЙ
НАУЧНО-ТЕХНИЧЕСКОЙ КОНФЕРЕНЦИИ
(Минск, 20–22 февраля 2014 года)

В авторской редакции
Ответственный за выпуск *В. В. Голенков*
Компьютерная верстка *Н. В. Гракова*

Подписано в печать 10.02.2014. Формат 60×84 1/8. Бумага офсетная. Гарнитура «Таймс».
Отпечатано на ризографе. Усл. печ. л. 67,43. Уч.-изд. л. 78,5. Тираж 140 экз. Заказ 30.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
ЛИ №02330/0494371 от 16.03.2009. ЛП №02330/0494175 от 03.04.2009.
220013, Минск, П. Бровки, 6



Проект OSTIS

(Open Semantic Technologies for Intelligent Systems)

Это открытый проект, направленный на создание массовой семантической технологии компонентного проектирования интеллектуальных систем различного назначения

Цели проекта OSTIS

- Создать массовую, комплексную и активно развивающуюся технологию проектирования интеллектуальных систем, включающую в себя теоретические и практические, программные и аппаратные аспекты
- Создать инфраструктуру, обеспечивающую сочетание научной и учебной, инженерной и коммерческой деятельности в области искусственного интеллекта

Особенности проекта OSTIS

- Является открытым комплексным проектом, состоит из большого числа частных проектов и предоставляет полный пакет документации по всем компонентам предлагаемой технологии (включая исходные тексты соответствующих программных средств)
- Ориентирован на широкий контингент разработчиков прикладных интеллектуальных систем (на массовое распространение предлагаемой технологии)
- Ориентирован на существенное сокращение сроков проектирования интеллектуальных систем

Возможными направлениями Вашего участия в развитии проекта OSTIS могут быть

- Разработка конкретных прикладных интеллектуальных систем в самых различных предметных областях. В каждой такой разработке Вы можете принять участие в качестве эксперта соответствующей предметной области, в качестве инженера знаний, в качестве разработчика операций обработки знаний и в качестве разработчика пользовательского интерфейса.
- Разработка различных ip-компонентов проектирования интеллектуальных систем:
 - ip-компонентов баз знаний
 - ip-компонентов машин обработки знаний
 - ip-компонентов пользовательских интерфейсов интеллектуальных систем
- Разработка различных фрагментов инструментальных средств проектирования интеллектуальных систем:
 - инструментальных средств проектирования баз знаний
 - инструментальных средств проектирования программ, ориентированных на обработку баз знаний
 - инструментальных средств проектирования машин обработки знаний (предметно независимых систем операций обработки знаний)
 - инструментальных средств проектирования пользовательских интерфейсов интеллектуальных систем
- Разработка интеллектуальных help-систем, предназначенных для информационного обслуживания и обучения разработчиков интеллектуальных систем:
 - интеллектуальных help-систем по проектированию баз знаний
 - интеллектуальных help-систем по проектированию программ, ориентированных на обработку баз знаний
 - интеллектуальных help-систем по проектированию машин обработки знаний
 - интеллектуальных help-систем по проектированию пользовательских интерфейсов интеллектуальных систем
- Комплексная разработка различных частных технологий проектирования различных классов прикладных интеллектуальных систем

Перечисленные направления Вашего участия в проекте OSTIS, а также и любые другие возможные направления Вы можете уточнить на основе документации проекта OSTIS (<http://ims.ostis.net/>).

Если Вы заинтересованы в наших продуктах или партнёрстве – свяжитесь с нами

Пишите нам по любым возникающим вопросам на следующие e-mail:

- golenkov.v.v@gmail.com,
- ostisconf@gmail.com



OSTIS-2015

**IV Международная научно-техническая конференция
«Открытые семантические технологии
проектирования интеллектуальных систем»
Open Semantic Technologies for Intelligent Systems**

19 – 21 февраля 2015 г. Минск. Республика Беларусь

И Н Ф О Р М А Ц И О Н Н О Е С О О Б Щ Е Н И Е

Приглашаем принять участие в V Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2015).

Конференция пройдет в период с **19 по 21 февраля 2015** года в Белорусском государственном университете информатики и радиоэлектроники, г. Минск, Республика Беларусь.

Рабочие языки конференции: русский, белорусский, английский.

ОСНОВНЫЕ ОРГАНИЗАТОРЫ КОНФЕРЕНЦИИ

- Российская ассоциация искусственного интеллекта (РАИИ)
- Белорусский государственный университет информатики и радиоэлектроники (БГУИР)
- Государственное учреждение «Администрация Парка высоких технологий» (Республика Беларусь)

НАПРАВЛЕНИЯ РАБОТЫ КОНФЕРЕНЦИИ:

- *Принципы, лежащие в основе семантического представления знаний, и их унификация. Типология знаний и особенности семантического представления различного вида знаний и метазнаний. Связи между знаниями и отношения, заданные на множестве знаний. Семантическая структура глобальной базы знаний, интегрирующей различные накапливаемые знания*
- *Языки программирования, ориентированные на параллельную обработку семантического представления баз знаний*
- *Модели решения задач, в основе которых лежит обработка знаний, осуществляемая непосредственно на уровне семантического представления обрабатываемых знаний. Семантические модели информационного поиска, интеграции знаний, анализа корректности и качества баз знаний, сборки информационного мусора, оптимизации баз знаний, дедуктивного и индуктивного вывода в базах знаний, правдоподобных рассуждений, распознавания образов, интеллектуального управления. Интеграция различных моделей решения задач*
- *Семантические модели восприятия информации о внешней среде и отображения этой информации в базу знаний*
- *Семантические модели мультимодальных пользовательских интерфейсов интеллектуальных систем, в основе которых лежит семантическое представление используемых ими знаний, и унификация этих моделей*
- *Семантические модели естественно-языковых пользовательских интерфейсов интеллектуальных систем. Структура семантического представления лингвистических баз знаний, описывающих естественные языки и обеспечивающих решение задач понимания естественно-языковых текстов и речевых сообщений, а также задач синтеза естественно-языковых текстов и речевых сообщений, семантически эквивалентных заданным фрагментам баз знаний*
- *Интегрированные комплексные логико-семантические модели интеллектуальных систем, основанные на семантическом представлении знаний, и их унификация*
- *Различные технические платформы и варианты реализации интерпретаторов унифицированных логико-семантических моделей интеллектуальных систем, основанных на семантическом представлении знаний*
- *Средства и методы, основанные на семантическом представлении знаний и ориентированные на проектирование различных типовых компонентов интеллектуальных систем (баз знаний, программ, решателей задач, интерфейсов)*
- *Средства и методы, основанные на семантическом представлении знаний и ориентированные на комплексное проектирование различных классов интеллектуальных систем (интеллектуальных справочных систем, интеллектуальных обучающих систем, интеллектуальных систем управления, интеллектуальных робототехнических систем, интеллектуальных систем поддержки проектирования и др.)*
- *Прикладные интеллектуальные системы, основанные на семантическом представлении используемых ими знаний*

ЦЕЛЬ И ФОРМАТ ПРОВЕДЕНИЯ КОНФЕРЕНЦИИ

Целью конференции является обсуждение проблем создания **открытой комплексной семантической технологии компонентного проектирования интеллектуальных систем**. Этим определяется и формат её проведения, предполагающий (1) пленарные доклады, (2) секционные заседания; (3) круглые столы, посвященные обсуждению различных вопросов создания указанной технологии; (4) выставочные презентации докладов.

Выставочная презентация докладов даёт возможность каждому докладчику продемонстрировать результаты своей разработки на выставке. Формат проведения конференции предполагает точное время начала каждого доклада и точное время его выставочной презентации.

Важнейшей задачей конференции является привлечение к её работе не только учёных и аспирантов, но и студенческой молодежи, интересующейся проблемами искусственного интеллекта, а также коммерческих организаций, готовых сотрудничать с научными коллективами, работающими над созданием современных технологий проектирования интеллектуальных систем.

КОНКУРС СТУДЕНЧЕСКИХ ПРОЕКТОВ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ И КОНКУРС ДОКЛАДОВ МОЛОДЫХ УЧЁНЫХ

В рамках конференции OSTIS проводится конкурс студенческих проектов интеллектуальных систем. Правила оформления заявок на участие в этом конкурсе **опубликованы на сайте конференции** (http://conf.ostis.net/index.php?title=Конкурс_студенческих_проектов).

Кроме того, проводится конкурс докладов молодых учёных. В конкурсе принимают участие доклады, включенные в Программу конференции и представленные на ней молодыми авторами, имеющими возраст до 30 лет и не имеющими ученых степеней.

УСЛОВИЯ УЧАСТИЯ В КОНФЕРЕНЦИИ

В конференции имеют право участвовать все те, кто интересуется проблемами искусственного интеллекта, а также коммерческие организации, готовые сотрудничать с научными коллективами, работающими над созданием современных технологий проектирования интеллектуальных систем.

Для участия в конференции OSTIS-2015 необходимо до 1 декабря 2014 года на электронную почту конференции ostisconf@gmail.com отправить:

- **статью** для публикации в Сборнике материалов конференции OSTIS-2015. Статья на конференцию должна быть оформлена в соответствии с шаблоном оформления статей;

- **заявку доклада** на конференцию OSTIS-2015. Каждое поле заявки обязательно для заполнения. Заполняя регистрационную форму, Вы подтверждаете согласие на обработку Оргкомитетом конференции персональных данных, публикацию статей и информации об авторах в печатном и электронном виде. В заявке доклада должна содержаться информация по каждому автору. К заявке доклада должны быть прикреплены **цветные фотографии** всех авторов статьи (это необходимо для публикации Программы конференции).

Отбор статей для публикации в Сборнике и участия в работе конференции осуществляется рецензентами из числа членов Программного комитета конференции.

Заявки и статьи, оформленные без соблюдения предъявляемых требований, не рассматриваются.

До 20 января 2015 года, авторам статей, включённых в Программу конференции, направляются приглашения для участия в конференции.

Участие в конференции не предполагает организационного взноса.

ПОРЯДОК ПРЕДСТАВЛЕНИЯ НАУЧНЫХ СТАТЕЙ

Статьи (только по перечисленным выше направлениям) представляются в готовом для публикации виде. Текст статьи должен быть логически законченным и содержать новые научные и практические результаты. От одного автора допускается не более двух статей.

Оргкомитет оставляет за собой право отказать в приеме статьи в случае, если статья не будет соответствовать требованиям оформления и тематике конференции, а также, если будет отсутствовать заявка доклада, соответствующая этой статье.

ПУБЛИКАЦИЯ МАТЕРИАЛОВ КОНФЕРЕНЦИИ

Оргкомитет конференции предполагает публикацию статей, отобранных Программным комитетом по результатам их рецензирования, в Сборнике материалов конференции и на официальном сайте конференции <http://conf.ostis.net>. Неимущественные права принадлежат авторам статей, поэтому публикация и распространение материалов статей на иных информационных ресурсах допускается только с согласия авторов статей.

ФОРМИРОВАНИЕ ПРОГРАММЫ КОНФЕРЕНЦИИ

Программа конференции формируется Программным комитетом по результатам рецензирования, представленных статей, а также на основании подтверждения автора(-ов) статьи о прибытии на конференцию.

КОНТАКТНЫЕ ДАННЫЕ ОРГАНИЗАТОРОВ КОНФЕРЕНЦИИ OSTIS

Вся необходимая информация по предстоящей и предыдущих конференциях OSTIS находится на сайте конференции <http://conf.ostis.net>.

Материалы для участия в конференции представляются в Оргкомитет конференции по электронной почте ostisconf@gmail.com.

Методическая и консультативная помощь участникам конференции осуществляется только через электронную почту конференции.

Конференция проходит в Республике Беларусь, г. Минск, ул. Платонова, 39 (5-ый учебный корпус Белорусского государственного университета информатики и радиоэлектроники).