РОССИЙСКАЯ АКАДЕМИЯ НАУК

ВСЕРОССИЙСКИЙ ИНСТИТУТ НАУЧНОЙ И ТЕХНИЧЕСКОЙ ИНФОРМАЦИИ
(ВИНИТИ)

НАУЧНО • ТЕХНИЧЕСКАЯ ИНФОРМАЦИЯ

Серия 2. ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ И СИСТЕМЫ

ЕЖЕМЕСЯЧНЫЙ НАУЧНО-ТЕХНИЧЕСКИЙ СБОРНИК

Издается с 1961 г. **№ 5** **Москва 2014**

ИНФОРМАЦИОННЫЙ АНАЛИЗ

УДК 81’32

В.А. Яцко

Компьютерная лингвистика или лингвистическая информатика?

Предлагается термин "лингвистическая информатика" для обозначения предметной области, изучающей закономерности распределения текстовой информации, проблемы, принципы, методы и алгоритмы разработки лингвистического программного обеспечения и аппаратных средств. Рассматриваются основные термины и понятия предметной области, даётся классификация лингвистического программного обеспечения

Ключевые слова: *обработка единиц естественного языка, алгоритмы и программы, классификация, лингвистическая информатика.*

Характерная особенность нашего времени – повсеместное использование технологий обработки естественного языка, которые являются частью глобального процесса информатизации общества. Миллионы пользователей во всём мире, посылая запросы в информационно-поисковые системы, отдавая голосовые команды телефонам, выполняя автоматическое реферирование текстов, не подозревают, что это стало возможным в результате развития предметной области, в рамках которой проводятся исследования и разработки алгоритмов и программ обработки текстов на естественном языке.

Для обозначения этой предметной области в настоящее время используются разные термины. Наиболее распространен термин *компьютерная лингвистика* – калька с английского *computational linguistics*. Данный термин представляется нам не вполне удачным, поскольку использование наименования какой-либо дисциплины в качестве опорного слова в словосочетании ограничивает сферу применения данного термина представителями этой дисциплины. Мало кто слышал об "информационной биологии" или "компьютерной историографии", поскольку разработки, выполняемые в рамках этих предметных об-

ластей, предназначены для биологов и историков. Соответственно, и термины "компьютерная лингвистика" и "корпусная лингвистика" предполагают именно лингвистов в качестве контингента пользователей продуктов, создаваемых представителями этих дисциплин. Если во втором случае это так и есть, то в первом – структура словосочетания входит в противоречие с его денотатом, поскольку под компьютерной лингвистикой обычно понимается разработка любых лингвистических программ и систем, в том числе и, например, информационно-поисковых систем, контингент пользователей которых, разумеется, не ограничивается лингвистами. Ср. определение Р. Гришмана: "Компьютерная лингвистика занимается изучением компьютерных систем, предназначенных для анализа и генерации единиц естественного языка" [1, с. 4]. В зарубежной литературе также широко применяется термин "обработка естественного языка" (*natural language processing* – NLP), причём через него определяется понятие *computational linguistics*, см. определения в [2]. Данный термин, как мы полагаем, достаточно точно указывает на объект деятельности – единицы естественного языка, однако, в нём отсутствует упоминание предмета исследования. По сути дела, им обозначается область практической деятельности, а не научная дисциплина. Введение отдельного слова для обозначения научной составляющей, например, "теория обработки единиц естественного языка" делает термин слишком громоздким, а его сокращение (ТОЕЕЯ) не вполне соответствует звуковому строю русского языка.

Особо следует остановиться на термине "прикладная лингвистика", значение которого, сложившееся в отечественной науке, отличается от его интерпретации в англо-американской литературе, да в западной науке в целом. Вплоть до последнего времени под прикладной лингвистикой понималась методика обучения языкам: "До настоящего времени основная часть разработок в прикладной лингвистике была посвящена обучению и изучению языков, в первую очередь английского как иностранного или второго языка" [3, с. 4]. В настоящее время наметилась тенденция расширения предметной области прикладной лингвистики, в которую включаются также проблемы логопедии и перевода, ср. определение в *Oxford dictionary*: "отрасль языкознания, занимающаяся практическим применением исследований языка, например, в обучении языкам, переводе и логопедии" [4].

Такая интерпретация существенно отличается от понимания прикладной лингвистики в отечественной науке. Ю.В. Рождественский считает, что "задача прикладного языкознания состоит в том, чтобы ввести новые материалы речи, наметить наиболее эффективные пути речевой коммуникации на базе новой техники, утвердить норму языка путём обучения, распространить новые виды речевой коммуникации путём обучения новым видам текстов (создания этих текстов, их передачи, хранения и применения)" [5, с. 215]. По его мнению, прикладная лингвистика включает три основных направления: лингводидактику, лингвосемиотику, информационное обслуживание [5, с. 299]. В информационное обслуживание входят такие области деятельности, как библиотечное, архивное, канцелярское дело, информационный поиск, реферирование, составление информационных словарей, двуязычный перевод, автоматизированные системы управления. Для обозначения "языковедческой части" теории информационного обслуживания Рождественский предлагает термин "лингвистическая информатика" [5, с. 354].

Очевидно, что такое понимание прикладной лингвистики значительно шире того, которое принято за рубежом и охватывает сферы, которые в англо-американской традиции относятся к другим предметным областям. Так, проблемы информационного поиска обычно включаются в предметную область информационной науки (*information science*), а проблемы разработки автоматизированных систем управления относятся к компьютерной науке (*computer science*) [6]. Вместе с тем, логопедия, с точки зрения интерпретации, предложенной Рождественским, не должна входить в прикладное языкознание, а является приложением языкознания, поскольку в этом случае языковые данные используются "для решения практической задачи, за которую ответственна другая область науки или практики" [5, с. 298]. В работе А.Н. Баранова, напротив, указывается, что под прикладной лингвистикой следует понимать "деятельность по приложению научных знаний об устройстве и функционировании языка в нелингвистических научных дисциплинах…" [7, с.7].

Кроме различных, в том числе и противоречивых, интерпретаций, термин "прикладная лингвистика" имеет и отмеченный ранее недостаток – использование слова "лингвистика" в качестве опорного слова в словосочетании. Особенностью данного термина также является то, что он обозначает целый ряд видов деятельности, не связанных напрямую с автоматической обработкой текстов, в том числе лингвистическую семиотику, лингводидактику, терминологию, указанные в паспорте специальности 10.02.21 "Прикладная и математическая лингвистика" [8].

В нашей работе [9] также был предложен термин "лингвистическая информатика" и была описана её предметная область, специфика которой определялась понятием информационно-лингвистической модели, а задачи научной дисциплины сводились к исследованию проблем информационного поиска и реферирования. Позднее японские специалисты использовали кальку *linguistic informatics* для обозначения предметной области, в сферу которой входит разработка программных средств, предназначенных для обучения иностранным языкам [10].

В данной статье мы попытаемся с новых позиций интерпретировать значение термина "лингвистическая информатика", рассмотреть основные понятия и структуру предметной области, раскрыть её междисциплинарную сущность. Особое внимание будет уделено сопоставительному рассмотрению понятий лингвистической информатики и теоретической лингвистики, что позволит выявить специфику рассматриваемой предметной области.

Как мы считаем, использование слова *информатика* в качестве опорного слова вполне правомерно, поскольку в рамках информатики создаются продук-

ты, прежде всего программные и аппаратные средства и технологии, предназначенные для различных, в том числе и непрофессиональных групп пользователей. В прототипическом представлении слово *информатика* связано с семантическими компонентами "компьютеры", "программы", "Интернет" и не соотносится с узкопрофессиональной областью, в отличие от слова *лингвистика*. Определение *лингвистическая* позволяет ограничить предметную область проблемами разработки лингвистического программного обеспечения, лингвистических аппаратных средств и технологий. Под лингвистическими аппаратными средствами мы понимаем вычислительное оборудование, специально предназначенное для обработки текстов на естественном языке. Такое оборудование широко используется в системах распознавания артикуляционных и акустических параметров устной речи, а также в оптических системах распознавания символов. Под лингвистическим программным обеспечением мы понимаем программы, приложения и системы, на входе которых – текст на естественном языке и которые функционируют на основе лингвистических алгоритмов – алгоритмов, применяющихся для обработки единиц естественного языка.

В основе функционирования многих лингвистических приложений лежит процесс лексической декомпозиции текста, в результате которой во входном тексте распознаются токены и на выходе генерируется их список. *Токен* можно определить как последовательность буквенных и/или цифровых символов, отделённую слева и справа знаками форматирования текста и/или препинания. Разбивка текста на токены называется *токенизацией*, а программы, выполняющие токенизацию – *токенайзерами.* В тексте *"By-by, dearie, he smiled."* токен *By-by* распознаётся по кавычкам слева и запятой справа, токен *dearie* – по пробелу слева и запятой справа, токен *he* – по пробелам слева и справа, токен *smiled* – по пробелу слева и точке справа.

Как видно из приведённого примера, токены обычно совпадают со словами, поэтому термину *токен* соответствует термин *слово* в теоретической лингвистике. В инструкциях для пользователей и в интерфейсах лингвистического программного обеспечения (ПО) достаточно часто используется термин *word*, а не *token*, поскольку он более понятен и привычен. Однако, с точки зрения теоретической интерпретации, между двумя терминами имеются существенные различия. Интерпретация термина *слово* в языкознании обычно даётся на основе соотношения между знаком, обозначаемым объектом и значением и представляется в виде известного семантического треугольника. В лингвистической информатике, как было показано выше, учитываются только знаки, выделяемые по формальным признакам. Соответственно, различаются и цели изучения слов и токенов. В лингвистике проводятся исследования, направленные на толкование значений слов, разграничение окказиальных и узуальных словоупотреблений, выявление новых значений и условий их актуализации. В лингвистической информатике рассматриваются статистические особенности распределения токенов в тексте, на основе которых разрабатываются формулы взвешивания, необходимые для выявления наиболее статистически значимых терминов. В этой связи проводится разграничение между уникальными токенами и общими токенами. Термин *уникальный токен* обозначает токен без учёта количества его повторов в тексте, а термин *общие токены* – количество токенов с учётом их частотностей. В Британском национальном корпусе уникальный токен *the* повторяется 5973437 раз, т.е. даёт 5973437 общих токенов, а в Корпусе современного американского английского [11] его частотность составляет 25063954. Таким образом, количество общих токенов, как правило, больше количества уникальных токенов. Это позволяет при взвешивании терминов использовать вероятностные величины и устранить зависимость весовых коэффициентов от размера текста. Вероятностный коэффициент для *the* в Британском национальном корпусе составляет 0.05973 (при размере корпуса 100000000 слов), а в Корпусе современного американского английского – 0.05569 (размер корпуса – 450000000 слов). Разница в вероятностных величинах составляет около четырёх тысячных, в то время как разница между сырыми частотностями – около девятнадцати миллионов. Процесс преобразования сырых частотностей с целью устранения зависимости от размера текста, а также приведения различных величин к единому виду в лингвистической информатике называется нормализацией. В качестве средства нормализации широко применяется логарифмирование по основанию 2. Двоичный логарифм от 5973437 равен 22,51013, а от 25063954 – 24,57911, что даёт различие в две целых, а не в девятнадцать миллионов.

Очевидно, что данное значение термина *нормализация* отличается от значения аналогичного термина, используемого в языкознании.

Для систем, связанных с представлением смысла текста, в процессе лексической декомпозиции важно распознавать в качестве одного токена такие словосочетания, как географические названия, личные имена, сокращения, устойчивые сочетания. При разбивке на отдельные токены сочетаний *New York* или *N.A.T.O.* может не воспроизвестись или даже исказиться смысл текста. Поэтому в ряде лингвистических программ для распознавания сочетаний применяются специальные списки и дополнительные правила. При обработке текста [12] предназначенное для автоматического реферирования приложение Essence [13] распознаёт в качестве токенов сочетания *Jamrul Hussain*, *Nilufa Begum*, *NEW YORK*. Программа статистического анализа AntConc [14], обрабатывая тот же самый текст, разделяет все эти сочетания на отдельные токены. Такое различие объясняется разной функциональностью и пользовательской аудиторией. Программы статистического анализа выдают данные о частотностях единиц текста; их пользователями являются специалисты в области автоматической обработки текста, а также лингвисты, которые используют эти данные в своей профессиональной деятельности. Системы реферирования относятся к программному обеспечению общего назначения и предназначены для намного более широкого круга пользователей.

В языкознании словарный состав языка классифицируется по семантическим, синтаксическим, этимологическим, стилистическим критериям. В лингвистиче-

ской информатике широко применяется классификация лексических единиц на знаменательные и служебные слова (стоп-слова). В литературе [15, 16] в качестве основного признака стоп-слов выделяется их равномерная распределённость по разным жанрам текстов. В любом достаточно большом тексте на английском языке наиболее частотными будут артикли, местоимения, предлоги, союзы. Как отмечает У. Фрэнсис [16], 10 наиболее частотных слов английского языка дают от 20 до 30% общих токенов. Удаление стоп-слов позволяет существенно (в ряде случае почти на 40%) уменьшить размеры лингвистических баз данных, повысить быстродействие и точность поиска. Вместе с тем стоп-слова используются в качестве одного из основных параметров в процессе автоматической классификации текстов: тот факт, что они встречаются в любых текстах, независимо от их жанрово-стилистических особенностей, позволяет провести сопоставительный анализ текстов и выявить особенности распределения стоп-слов, присущие отдельным группам, категориям, жанрам текстов [17]. Фильтрация стоп-слов является важной процедурой обработки текста в информационно-поисковых системах и системах автоматической классификации текстов, которая выполняется на основе специальных списков стоп-слов, либо алгоритмически. С целью фильтрации стоп-слов часто применяют формулу TF*IDF, предложенную Дж. Солтоном и Ч. Янгом в 1973 г. [18], а также её интерпретации [19]. В соответствии с формулой распределение терминов в анализируемом тексте сопоставляется с их распределением в коллекции текстовых документов; при этом наибольший вес получают термины, встречающиеся с наибольшей частотностью в данном документе, но редко встречающиеся в других текстовых документах коллекции, в то время как термины, встречающиеся в текущем документе и во всех текстах коллекции, получают нулевые коэффициенты. Таким образом, формула описывает определённую закономерность распределения текстовой информации. Основной проблемой, возникающей при использовании формулы TF*IDF, является неопределённость количественного и жанрово-стилистического состава коллекции текстов, с которой сопоставляется анализируемый текст. Одним из подходов к решению этой проблемы может быть использование зонального анализа текста на основе интерпретации закона Брэдфорда [20].

Мы подробно остановились на процессе обработки лексических единиц текста, поскольку он наглядно демонстрирует междисциплинарные особенности предметной области, а лексическая декомпозиция является фундаментальным алгоритмом, который лежит в основе ряда алгоритмов, выполняемых на различных уровнях системы языка. На основе токенизации проводится морфологический анализ, аннотирование, фразовая декомпозиция, разбивка на н-грамы, клаузальная декомпозиция, разрешение анафоры.

С помощью алгоритмов морфологического анализа распознаются элементы морфологической структуры слова – корни, основа, суффиксы, окончания. К алгоритмам, широко применяемым на морфологическом уровне, относятся стемминг и лемматизация.

Цель стемминга – отождествить основы различных словоформ, имеющих одно значение. На входе стеммера – список токенов, на выходе – список их основ (стемм). Стемминг позволяет существенно повысить показатели точности и полноты поиска и широко используется в информационно-поисковых системах различных типов. В теоретической лингвистике под основой слова понимается его неизменяемая часть, выражающая лексическое значение. Термином *стемма* мы обозначаем последовательность символов, остающуюся после удаления строк, содержащихся в определённых файлах данных, и выполняющую функцию отождествления токенов. Ланкастерский стеммер [21] в токене *daughter* удаляет *er*, так как такая строка есть в файле данных, на основе которого он работает. С точки зрения теоретической лингвистики в данном случае происходит ошибка, поскольку *er* входит в основу слова. С точки зрения лингвистической информатики ошибки не происходит, потому что с помощью стеммы *daught* можно отождествить токены *daughter* и *daughters*. Вместе с тем при отделении *er* от *cater* по стемме *cat* отождествятся не только токены *catered, caters, catering,* но также и *cat, cats, cat's*. Возникает ошибка избыточного стеммирования, поскольку по одной стемме отождествляются токены с разным значением. В разработанном нами стеммере [22] суффиксы и окончания соотносятся с соответствующими частями речи, что помогает снизить количество ошибок, однако требует предварительного аннотирования тегами частей речи.

Аннотирование проводится теггерами, на входе у которых – список токенов, на выходе – список, в котором каждому токену присвоен определённый тег – условное обозначение, указывающее на его лингвистические характеристики. Наиболее распространённым видом теггеров являются теггеры частей речи (*POS taggers*), которые распознают часть речи токена и приписывают ему соответствующий тег. Помимо информации о части речи обычно указывается и информация о лексико-грамматических и семантических характеристиках слова, например, *NN* – нарицательное существительное в единственном числе, *NNS* – нарицательное существительное во множественном числе, *AJC* – прилагательное в сравнительной степени и т.д.

Термин *тег* был введён в научный оборот в связи с разработкой электронных текстовых корпусов и не имеет аналогов в теоретической лингвистике, хотя широко используется в информатике, в частности для обозначения дескрипторов языков гипертекстовой разметки. В настоящее время аннотирование широко применяется в системах автоматической классификации текстов, а теги частей речи и их сочетания выступают в качестве параметров такой классификации. К другим видам тегов относятся семантические теги и теги когнитивных ролей (*knowledge roles*). Первые используются в фактографических информационно-поисковых системах, в то время как вторые – в системах интеллектуального анализа текста. В фактографической ИПС *InFact*, разработанной в компании *Insightful Corporation,* используются теги *Person*, *Location*, *Organization*. В ответ на запрос [*Organization/Name*] >*buy*> [*Organization/Name*] ^ *money* данная ИПС выдаст отрезки текста, в которых

содержится информация о покупке одной компанией другой компании за определённую сумму денег [23]. Следует отметить, что на входе у таких ИПС – текст на искусственном языке, что не позволяет отнести их к предметной области лингвистической информатики. То же самое относится и к криптографическим системам, которые следует рассматривать как часть предметной области информатики.

В [24] проводилось аннотирование текстов диагностических отчётов о состоянии электроизоляции высоковольтных ротационных устройств когнитивными ролями *Observed Object*, *Symptom*, *Cause*. В результате была создана система, с помощью которой инженер мог получать информацию о признаках неполадки в конкретном объекте, причинах и способах её устранения. В качестве лингвистической базы данных использовалась лексикографическая информация, разработанная в рамках проекта Framenet [25].

Аннотирование семантическими и когнитивными ролями предусматривает распознавание как отдельных слов, так и словосочетаний. Такое аннотирование требует предварительной разработки и применения специальных грамматик фразовой структуры на синтаксическом уровне языковой системы.

Одним из фундаментальных алгоритмов, применяемых на синтаксическом уровне, является синтаксическая декомпозиция, которая выполняется синтаксическими сплиттерами. На входе у сплиттера – текст, на выходе – список предложений текста. Алгоритмы синтаксической декомпозиции предусматривают распознавание предложений на основе символов форматирования текста: пробелов, знаков пунктуации, знаков конца строк. Таким образом, термин *предложение* в лингвистической информатике обозначает последовательность строк, отделённую справа и слева символами форматирования текста и знаками пунктуации. Распознавание предложений осложняется отсутствием стандартного форматирования текста; точки, восклицательные и вопросительные знаки, которые обычно применяются в качестве разделителей, могут использоваться не только в конце, но и в середине предложения. Целый ряд единиц текста, которые форматируются как предложения, на самом деле предложениями не являются. К ним относятся такие элементы, как оглавление, заглавия отдельных разделов, названия рисунков, таблиц, текст, использующийся внутри самих таблиц и рисунков, колонтитулы. Между тем именно предложения являются основной единицей анализа во многих системах, а в системах автоматического реферирования и выходной текст состоит из предложений. Ошибки в распознавании предложений существенно снижают эффективность таких систем в целом.

Нами была предложена дедукционно-инверсионная архитектура декомпозиции текста, в соответствии с которой вначале текст разбивается на абзацы, затем – на слова, затем из слов генерируются предложения. Таким образом, декомпозиция начинается с большей единицы (абзаца), затем осуществляется переход к меньшей единице (слову), затем – снова к большей (предложению). Дедукционно-инверсионная архитектура декомпозиции позволяет игнорировать такие компоненты текста как заголовки, подзаголовки, оглавления, поскольку они не входят в состав абзацев [22].

Синтаксическая декомпозиция является основой для выполнения целого ряда алгоритмов распознавания фразовой структуры предложения. Широко распространены алгоритмы выделения *n-gram*–словосочетаний, состоящих из двух (биграмы), трёх (триграмы) и более (тетраграмы, пентаграмы, гексаграмы, гептаграмы, октограмы) токенов [26]. Разбивка на словосочетания в данном случае проводится с учётом позиции токена в предложении. Например, предложение *John has a dog* включает 4 юниграма, 3 биграма (*John has*, *has a*, *a dog*), 2 триграма (*John has a*, *has a dog*), 1 тетраграм – всё предложение. Количество биграмов для каждого предложения ($ng_{(s)}$) будет составлять n-1; триграмов – n-2, где n – количество токенов в предложении, т.е. $ng_{(s)} = w_{i-(n-1)}$, $w_{i-(n-2)}\ldots w_{i-(n-n)}$, где w_i – порядковый уровень *n-gram*, начиная с биграмов. Распознавание *n-gram* проводится на основе соответствующих правил.

Анализ распределения *n-gram* позволяет выявить статистически значимые словосочетания и часто применяется в стохастических алгоритмах аннотирования тегами частей речи. Распределения *n-gram* используются с целью автоматической классификации и категоризации, поскольку выступают в качестве важного параметра, позволяющего определить принадлежность текста к определённой категории, типу, группе, жанру. При анализе на синтаксическом уровне в качестве основной единицы выступают биграмы и триграмы, поскольку рекуррентность словосочетаний с большим количеством токенов маловероятна. Анализ *n-gram* большего порядка применяется в системах автоматической коррекции орфографии, а также в системах оптического распознавания символов (*Optical Character Recognition*), где основной единицей выступают символы в токенах.

Для анализа морфологически значимых словосочетаний применяются программы фразовой декомпозиции – чанкеры, которые на выходе выдают списки фраз определённого типа (именных, глагольных, предложных, адъективных, адвербиальных). Наиболее распространены именные (*noun-phrase*) чанкеры, распознающие словосочетания с управляющим существительным. Именно этим типом словосочетаний обозначаются объекты, описываемые в тексте, а их ранжирование по весовым коэффициентам позволяет получить список ключевых слов, отражающих основное содержание текста. Распознавание словосочетаний этого типа выполняется на основе предварительного аннотирования тегами частей речи и объединения отдельных частей речи во фразы на основе правил грамматики.

Правила фразовой структуры были разработаны для английского языка в рамках концепции генеративной грамматики, предложенной Н. Хомским. Грамматические правила записываются в виде $NP \rightarrow NN; NP \rightarrow DetNN; NP \rightarrow DetANN$, где указывается состав словосочетания, в данном случае именного, а также порядок слов. В первом случае показано, что именное словосочетание может состоять только из одного существительного (*NN*); во втором случае оно состоит из детерминанта (*Det*) и

существительного, причём детерминант занимает позицию перед существительным, а обратный порядок слов неправилен; в третьем случае словосочетание состоит из детерминанта, прилагательного (*A*) и существительного, причем другие варианты словопорядка неправильны.

К настоящему времени на основе концепции Хомского создан целый ряд грамматик, которые делятся на два основных вида – деривационные и недеривационные. В деривационных грамматиках проводится разграничение между поверхностной и глубинной структурой словосочетания и предложения и формулируются дополнительные правила вывода (деривации) поверхностных структур их глубинных. Синтаксическая структура представляется в виде иерархического дерева зависимости. Недеривационные грамматики описывают поверхностные, как правило, линейные синтаксические структуры. Выбор того или иного типа грамматики обуславливается задачами конкретного исследовательского проекта. Деривационные грамматики лежат в основе функционирования синтаксических парсеров, которые выдают на выходе граф синтаксической структуры предложения. Так же как и теггеры частей речи, синтаксические парсеры обучаются на предложениях с размеченной вручную синтаксической структурой; в них применяются правила для определения наиболее вероятного варианта на основе скрытых моделей Маркова. В качестве примера можно привести *Lexparser*, разработанный в Стэнфордском университете США [27].

Иерархические синтаксические структуры применяются в системах машинного перевода для установления эквивалентности синтаксических структур в двух языках.

На синтаксическом уровне может проводиться декомпозиция не только на словосочетания и предложения, но и на клаузы – элементарные предикативные структуры, выражающие суждение. Понятие клаузы в какой-то степени соответствует понятию пропозиции в лингвистике, однако клаузы выделяются по формальным признакам, к которым может относиться, например, наличие именной группы и следующей за ней глагольной группы. Разбивка на клаузы применяется в системах интеллектуального анализа для более адекватной передачи содержания текста (см., например, [24]).

Наиболее распространенными алгоритмами, применяемыми на дискурсивном уровне, являются алгоритмы разрешения анафоры, которые предусматривают замену анафорических местоимений предшествующими кореферентными именами объектов. Под дискурсом в лингвистической информатике понимается текст, связи между компонентами которого (клаузами, предложениями) манифестируются повторами лексических и/или синтаксических единиц. Как мы полагаем, исследование логико-семантических связей между единицами текста и проблема моделирования его логико-семантической структуры выходят за рамки предметной области лингвистической информатики и являются частью проблем исследования искусственного интеллекта.

В таблице представлены сгруппированные по уровням системы языка алгоритмы и программы, характеризующие специфику предметной области лингвистической информатики. Названия программ даны на английском языке, поскольку в качестве русских эквивалентов обычно используются их транслитерации.

Представленные в таблице алгоритмы и программы лежат в основе лингвистического программного обеспечения, которое можно классифицировать по целому ряду критериев. По материальной форме входного текста выделяются системы обработки устных текстов и письменных текстов. В первом случае обычно говорят об обработке речи (*speech processing*), а во втором – об обработке текста (*text processing*). Начало развития лингвистической информатики связано с проблемами обработки письменных текстов, создания ИПС, систем реферирования и машинного перевода в конце 1950-х и в 1960-х гг. Системы обработки устной речи стали интенсивно разрабатываться в 1990-х гг., когда появились бытовые системы распознавания речи. В настоящее время они широко применяются в автоответчиках, системах распознавания индивидуальных характеристик личности, таких как возраст, пол и даже уровень алкогольного опьянения [28], в системах голосового управления техническими объектами, в том числе и наносистемами [29]. По форме речевой деятельности можно выделить алгоритмы, предназначенные для обработки монологической и диалогической речи. Долгое время объектом автоматического анализа текста были монологические тексты, в основном тексты научных работ. Развитие Интернета обусловило появление жанров диалогической письменной речи: чатов, блогов, форумов. Обработка таких текстов имеет свою специфику и требует применения специальных алгоритмов, учитывающих их паралингвистические особенности. Интенсивно развиваются и системы обработки диалогической устной речи: вопросно-ответные системы, системы машинного перевода.

По степени интеллектуальности получаемых пользователями результатов можно выделить в отдельную группу алгоритмы, с помощью которых выдаётся информация, содержащаяся в тексте имплицитно, либо новая информация, которой нет в обрабатываемом тексте. Такие алгоритмы разрабатываются в процессе интеллектуального анализа текста (*text mining*) и существенно отличаются от традиционных алгоритмов информационного поиска и реферирования, в результате применения которых выявляется наиболее значимая информация, содержащаяся в тексте. Интеллектуальный анализ текста широко применяется в технике и медицине как средство обмена опытом. В медицине перевод историй болезней пациентов в электронную форму и их аннотирование тегами когнитивных ролей позволяет врачу с помощью поисковых систем находить диагнозы, соответствующие определённым симптомам, методики лечения, применявшиеся другими врачами, назначавшиеся медикаменты и препараты, результаты лечения [30]. Успешно развивается интеллектуальный анализ мнений пользователей о коммерческих продуктах [31], позволяющий фирмам-производителям выявлять достоинства и недостатки продукции и проводить эффективную маркетинговую политику.

Таблица

Алгоритмы и программы автоматической обработки текста

Алгоритмы	Программы	Распознаваемая/ обрабатываемая единица	Лингвистический термин	Уровни языка
Распознавание символов	OCR	Символ	Графема	Графемный
Стемминг	Stemmers	Стемма	Основа слова	Морфологический
Лемматизация	Lemmatizers	Лемма	Лексема	
Токенизация	Tokenizers	Токен	Слово	Лексический
Аннотирование	Taggers	Тег	-	
Взвешивание терминов	Weighting filters	Весовой коэффициент	-	
Разбивка на н-граммы	N-gram splitters	Н-грам	-	Синтаксический
Фразовая декомпозиция	Chunkers	Фраза	Словосочетание	
Парсинг	Parsers	Предложение	Предложение	
Синтаксическая декомпозиция	Text splitters	Предложение	Предложение	
	Clause splitters	Клауза	Пропозиция	
Разрешение анафоры	Resolvers	Анафора	Анафора	Дискурсивный

По целевым группам пользователей можно выделить универсальное, специальное и профессиональное лингвистическое ПО. Системы универсального типа предназначены для любых групп пользователей, независимо от их профессии, возраста, социального положения. Типичный пример – ИПС Интернета, которыми каждый день пользуются миллиарды людей в мире. Специальное лингвистическое ПО предназначено для определённых групп пользователей. Системы интеллектуального анализа текста обычно позиционируются как системы, предназначенные для поддержки принятия решений представителями определённой профессиональной группы. Профессиональное лингвистическое ПО предназначено для специалистов в области лингвистической информатики и поддержки исследований в области автоматического анализа текста. Существует ряд программ статистического анализа, предоставляющих информацию о количестве уникальных и общих токенов, количестве н-грамов, контексте использования лексических единиц, вероятностных и статистических показателях их совместной встречаемости [14; 32].

В зависимости от режима функционирования лингвистические приложения и системы можно разделить на автоматические и автоматизированные. Автоматизированные системы работают в дискретном режиме; к этому виду относится большинство разрабатываемого в настоящее время ПО. В качестве примера можно привести ИПС, функционирование которых начинается с запроса пользователя и заканчивается выдачей результата. Автоматические системы функционируют в непрерывном режиме, как, например, системы реферирования устной речи, позволяющие отслеживать новостные события. Заметим, что в названиях конкретных видов лингвистического ПО нет строго разграничения между терминами "автоматический" и "автоматизированный". Информационно-поисковые системы совершенно верно характеризуются как автоматизированные, в то время как системы реферирования по традиции называются автоматическими (ср. название известного сборника "Advances in automatic text summarization"), хотя на самом деле имеются в виду системы, работающие в дискретном режиме. Наряду с автоматическими и автоматизированными системами и

приложениями разрабатывается также ПО для компьютерно-опосредованной (*computer-assisted/aided*) обработки текстов. Системы этого типа наиболее широко используются в практике двуязычного перевода и в обучении иностранным языкам, повышая эффективность деятельности преподавателя и переводчика. Системы типа переводческой памяти (*translation memory*) содержат базы данных, включающие ранее переведённые тексты, словари, корпуса. Выполняя перевод, его автор может вставлять в текст слова, фразы, предложения из переводов, сделанных ранее, а также с помощью словарей и корпусов проверять контекст использования лексических единиц.

Функционирование лингвистического ПО поддерживается лексикографическими ресурсами, к которым относятся списки терминов, терминологические словари, терминолого-статистические словари, тезаурусы, онтологии. Списки терминов содержат лингвистические единицы, необходимые для реализации программным обеспечением определённой функции. К ним можно отнести списки суффиксов и окончаний, на основе которых функционируют стеммеры, а также списки стоп-слов, использующиеся в соответствующих фильтрах. Терминологические словари, помимо самих терминов, содержат метаинформацию. В морфологических словарях, необходимых для поддержки функционирования теггеров частей речи, каждому токену приписывается тег части речи (или два и более возможных тегов). В терминолого-статистических словарях для каждой лингвистической единицы даётся информация о её распределении в текстах или файлах. В нелемматизированном словаре, составленном А. Килгарифом [33] на основе Британского национального корпуса, для каждого токена указывается тег части речи, частотность в корпусе и количество файлов, в которых он встречается. Статистические данные, содержащиеся в словарях этого типа, имеют существенное значение для определения вероятностных характеристик, которые необходимы при разработке ряда систем автоматического анализа теста. Например, вероятностные параметры можно учитывать при разработке стохастических теггеров частей речи, причём можно игнорировать случаи маловероятного использования токенов с определёнными тегами. Частотность использования в корпусе *frequent* как прилагательного почти в 60 раз выше его использования в качестве глагола, в вероятностных величинах разница составит 0,002321 - 0,000039 = 0,002282 (при 1000000 общих токенов в корпусе). В данном случае вполне возможно игнорировать глагольные формы, и всем токенам *frequent* приписывать тег прилагательного, поскольку вероятность ошибки крайне мала.

Тезаурусы предоставляют информацию о терминах, связанных с данным термином структурно-семантическими связями: синонимическими, антонимическими, гипонимо-гиперонимическими. Наиболее широко известным тезаурусом для английского языка является *WordNet*, разработанный в Принстонском университете США и распространяющийся с открытым исходным кодом, локализованным для различных языков программирования [34]. По проблемам применения этого тезауруса в целях автоматического анализа текста даже проводятся международные конференции, что свидетельствует об актуальности разработок словарей этого типа.

Основным понятием, лежащим в основе архитектуры *WordNet*, является понятие синонимических рядов (*synsets*) – группы семантически связанных терминов, распределённых по частям речи, которые различаются в зависимости от степени смысловой близости. Смысловая близость определяется расстоянием от исходного (текущего) слова. Если взять в качестве исходного, например, слово *courage*, то синонимический ряд первого уровня составят слова, через которые непосредственно толкуется данное слово: *courageousness, bravery, braveness*. К синонимическому ряду второго уровня относятся синонимы, в данном случае это слово *spirit*. К синонимическому ряду третьего уровня будут относиться синонимы *spirit*: *character, fiber, fibre*. К синонимическому ряду четвёртого уровня относятся синонимы слов, находящихся на третьем уровне и т.д. Таким образом, создаётся типичная гипертекстовая структура с непрерывными переходами от одного кластера синонимов к другому. Тезаурусы применяются в информационно-поисковых системах и системах автоматического реферирования, системах автоматической классификации и категоризации текстов, системах интеллектуального анализа текста. В ИПС применение тезаурусов является эффективным средством расширения поисковых запросов.

Под онтологиями понимаются сложно структурированные словари, моделирующие структуру определённой предметной области на основе функциональных отношений между компонентами, и используемые для поддержки систем интеллектуального анализа текста. Сложная структура проявляется в многоуровневой иерархии, причём на определённых уровнях компоненты онтологии – понятия и категории – соотносятся с конкретными терминами – инстанциациями (*instantiations*). Онтологии классифицируются на формальные и лингвистические. Специфика лингвистической онтологии состоит в том, что она связана с грамматикой, правила которой позволяют распознавать компоненты онтологии и отношения между ними в тексте. В [31] описывается шестиуровневая лингвистическая онтология, разработанная для поддержки системы автоматического анализа мнений пользователей о коммерческих продуктах. На верхнем уровне онтологии находятся категории семантических и синтаксических терминов, которые, соответственно, либо выражают положительную или отрицательную оценку, либо изменяют её интенсивность. Онтология связана с линейной грамматикой, благодаря которой имена продуктов, указанные в поисковом запросе, соотносятся с компонентами онтологии.

Вышеизложенное позволяет определить лингвистическую информатику как дисциплину, изучающую закономерности распределения текстовой информации, проблемы, принципы, методы и алгоритмы разработки лингвистического программного обеспечения и аппаратных средств.

Лингвистическая информатика представляет собой междисциплинарную науку, развитие которой детер-

минируется математическими, техническими, лингвистическими основами. Математика и языкознание выполняют методологическую роль, которая проявляется разнопланово. Методологическая роль математики возрастает по мере повышения уровня разработок и исследований. Разработка прикладного лингвистического ПО требует знания общих для всех видов программирования элементов булевой алгебры и исчисления высказываний, в то время как выполнение теоретических и фундаментальных исследований невозможно без знания соответствующих разделов математики, например, теории множеств, теории графов, теории вероятностей, статистического анализа, а также закономерностей и законов распределения текстовой информации. В настоящее время одной из фундаментальных проблем предметной области, решение которой невозможно без применения математического аппарата, является разработка критериев репрезентативности текстовых корпусов. Методологическая роль лингвистики возрастает по мере возрастания сложности обрабатываемых и распознаваемых лингвистических единиц. Если токенизация может проводиться по символам форматирования текста, то стемминг требует знания морфологической структуры слова, чанкинг и парсинг – знания фразовой структуры предложения, клауз-сплиттинг – знания структуры предикативных конструкций, разрешение анафоры – знания межфразовых связей между предложениями. Актуальной теоретической задачей, требующей серьёзного лингвистического анализа, является разработка ролевых грамматик для поддержки систем интеллектуального анализа текста.

Одной из основных проблем, влияющих на развитие предметной области, является сложность подготовки специалистов, которые должны владеть сочетанием гуманитарных, технических и математических знаний. Подготовка таких специалистов в зарубежных университетах проводится в рамках магистерских программ, содержание которых включает технические, математические и лингвистические компоненты. В качестве примера можно привести магистерскую программу лингвистического факультета (Department of Linguistics) Вашингтонского университета в Сиэтле [35]. Технический компонент предусматривает хорошие навыки программирования на C++ и Java (рекомендуется также знание Perl и/или Python); знание структур баз данных и алгоритмов, конечных автоматов и измерительных преобразователей; умение использовать серверные кластеры на платформе UNIX. Математический компонент включает теорию вероятностей и статистический анализ. Лингвистический компонент включает введение в фонетику и синтаксис с акцентом на изучение артикуляционных и акустических коррелятов фонологических единиц и разработку формальных грамматик, необходимых для создания прикладных программ; изучение методов поверхностной (*shallow*) обработки единиц естественного языка, включая токенизацию, аннотирование, морфологический анализ, парсинг; изучение методов глубокой обработки единиц естественного языка, включая алгоритмы и грамматики, необходимые для соотнесения глубинных структур с поверхностными синтаксическими структурами. На заключительном этапе обучения осваиваются методы разработки информационно-поисковых и вопросно-ответных систем, систем машинного перевода, а также приложений и программ обучения языкам, проверки орфографии и грамматики, распознавания рукописного ввода и оптического распознавания символов, кластеризации документов, распознавания и синтеза речи. Во время обучения студенты проходят практику в крупнейших компаниях, занимающихся разработкой лингвистического программного обучения, таких как Microsoft, Google, InXight.

Данная магистерская программа представляет интерес, так как даёт представление о междисциплинарной природе и структуре предметной области. Из трёх компонентов ведущим и наиболее объёмным является лингвистический, что вполне естественно, и не случаен тот факт, что такие программы обучения реализуются именно на лингвистических факультетах. В предметную область входят информационно-поисковые и вопросно-ответные системы, системы распознавания речи и текста, машинного перевода, обучения иностранным языкам.

Предлагаемая магистерская программа, как утверждается на сайте Вашингтонского университета, относится к числу немногих лучших (*top-notch*) программ международного уровня, по которым готовятся специалисты в области компьютерной лингвистики.

Очевидно, что развитие лингвистических технологий и создание аналогичных программ обучения является актуальным и для России.

СПИСОК ЛИТЕРАТУРЫ

1. Grishman R. Computational linguistics: An Introduction. – Cambridge (USA): Cambridge University Press, 1986. – 195 p.
2. Richter F. Introduction to computational linguistics. – 2005. – URL: http://www.sfs.uni-tuebingen.de/~fr/teaching/ws05-06/icl/slides/lecture2.pdf.
3. Naves T. Applied linguistics what it is and the history of the discipline. – 2002. – URL: http://diposit.ub.edu/dspace/bitstream/2445/4701/1/Naves2008ALDisciplinePartIonGrabe2002.pdf.
4. Definition of applied linguistics // Oxford dictionaries. – 2012. – URL: http://www.oxforddictionaries.com/definition/english/applied-linguistics?q=applied+linguistics.
5. Рождественский Ю.В. Лекции по общему языкознанию. – М.: Высшая школа, 1990. – 381 с.
6. Information science – definition and more // Merriam-Webster dictionary. – 2012. – URL: http://www.merriam-webster.com/dictionary/information%20science.
7. Баранов А.Н. Введение в прикладную лингвистику: учебное пособие. – М.: Эдиториал УРСС, 2001. – 360 с.
8. Паспорта Номенклатуры специальностей научных работников / ФГАУ ГНИИ ИТТ "Информика". – 2002–2012. – URL: http://www.edu.ru/db/portal/spec_pass/vuz_ds_pasport.php?spec=10.02.21.
9. Яцко В.А. Лингвистические аспекты предмета информатики // Научно-техническая информация. Сер. 1. – 1996. – № 2. – С. 1–7.

10. Linguistic informatics – state of the art and the future / ed. by Yuji Kawaguchi et al. – Amsterdam; Philadelphia: John Benjamins publishing company, 2005. – 363 p.
11. The corpus of contemporary American English (COCA) / Brigham Young University. – 2012. – URL: http://corpus.byu.edu/coca/.
12. Missing tot's trail goes cold after three month. – 2009. – January 19. http://edition.cnn.com/2009/CRIME/01/13/grace.coldcase.hussain/index.html?eref=rss_crime.
13. Sillanpää M. Lost knowledge – DM Partner's Essence. – 2009, June 4. – URL: http:// bigmenoncontent.com/2009/06/04/lost-knowledge-%E2%80%93-dm-partner%E2%80%99s-essence/.
14. Laurence Anthony's software. – 2011. – URL: http://www.antlab.sci.waseda.ac.jp/software.html.
15. Tsz-Wai L.R., He B., Ounis I. Automatically building a stopword list for an information retrieval system // Journal on digital information management: special issue on the 5th Dutch-Belgian Information Retrieval Workshop (DIR'05). – 2005. – Vol. 3, № 1. – P. 3–8.
16. Francis W.N. Frequency analysis of English usage lexicon and grammar. – Boston: Houghton Mifflin, 1982. – 561 p.
17. Santini M. Automatic identification of genre in Web pages. PhD thesis. – Brighton (UK): University of Brighton, 2007. – URL: http://www.itri.brighton.ac.uk/~Marina.Santini/MSantini_PhD_Thesis.zip.
18. Salton G, Yang C.S. On the specification of term values in automatic indexing // Journal of documentation. – 1973. – Vol. 29, Issue 4. – P. 351–372.
19. Yatsko V.A. TF*IDF Revisited // International journal of computational linguistics and natural language engineering. – 2013. – Vol. 2, Issue 6. – P. 385–387.
20. Яцко В.А. Метод зонального анализа данных // В мире научных открытий. – 2013. – № 6.1. – С. 166–182.
21. Paice C.D. Another Stemmer // SIGIR Forum . – 1990. – Vol. 24, № 3. – P. 56–61.
22. Яцко В.А., Стариков М.С., Ларченко Е.В., Вишняков Т.Н. Алгоритмы предварительной обработки текста: декомпозиция, аннотирование, морфологический анализ // Научно-техническая информация. Сер. 2. – 2009. – № 11. – С. 8–18.
23. Marchisio G., Dhillon N., Liang J. et al. A case study in natural language based Web search // Natural language processing and text mining / Eds. A. Kao, S. Poteet. – London, 2007. – P. 69–90.
24. Mustafaraj E., Hoof V., Freisleben D. Mining diagnostic text reports by learning to annotate knowledge roles // Ibid. – P. 45–68.
25. Baker C.F., Fillmore C.J., Lowe J.B. The Berkeley framenet project. – 1998. – http://acl.ldc.upenn.edu/C/C98/C98-1013.pdf?origin=publication_detail.
26. Bickel S., Haider P., Scheffer T. Predicting sentences using n-gram language models. – 2005. – URL: http://delivery.acm.org/10.1145/1230000/1220600/p193-bickel.pdf?ip=2.61.106.175&id=1220600&acc=OPEN&key=4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E6D218144511F3437&CFID=304034823&CFTOKEN=75172216&__acm__=1395046160_ee1d17fe0ade60cc1447d85533fc168c.
27. The Stanford parser: a statistical parser / The Stanford natural language processing group. – 2014. – URL:http://nlp.stanford.edu/software/lex-parser.shtml.
28. Levit M., Huber R, Batliner A., Nöth E. Use of prosodic speech characteristics for automated detection of alcohol intoxication // Proceedings of the workshop on prosody and speech recognition. – Red Bank, NJ, 2001. – P. 103–106.
29. Потапова Р.К. Нанотехнологии и лингвистика: прогнозы и перспективы взаимодействия // Нанотехнологии в лингвистике и лингводидак-тике: миф или реальность? Опыт создания общего образовательного пространства стран СНГ. Тезисы Международной научно-практической конференции. – М., 2007. – С. 9–11.
30. Li Q., Zhai H., Deleger L., et al. A sequence labeling approach to link medications and their attributes in clinical notes and clinical trial announcements for information extraction // Journal of the American Medical Informatics Association. – 2013. – Vol. 20, Issue 5. – P. 915–921.
31. Яцко В.А., Стариков М.С. Опыт разработки онтологии для автоматического анализа мнений пользователей о коммерческих продуктах // Научно- техническая информация. – 2011. – № 7. – С. 9–14.
32. Yatsko's Computational Linguistics Laboratory. – 2013. – URL: http://yatsko.zohosites.com/linguistic-toobox-a-concordancer.html.
33. Kilgarriff A. BNC database and word frequency lists. – 1998. – URL: http://www.kilgarriff.co.uk/bnc-readme.html.
34. About WordNet. – 2012. – URL: http://wordnet.princeton.edu.
35. UW professional master's in computational linguistics / University of Washington. –2014. – URL: http://www.compling.uw.edu/about.

Материал поступил в редакцию 13.02.14.

Сведения об авторе

ЯЦКО Вячеслав Александрович – доктор филологических наук, профессор, Хакасский государственный университет им. Н.Ф. Катанова, г. Абакан
e-mail: viatcheslav-yatsko@rambler.ru

УДК 81’322.2 : (047.31)

Э.С. Клышинский, Я.Б. Калачёв, В.В. Жаднов

Методика автоматизации проверки полноты технической отчетной документации*

Рассматривается новый метод автоматизации определения соответствия технического задания и итогового отчета в ходе его приемки. Предложенный метод позволяет экспертам получить предварительную оценку степени соответствия отчета техническому заданию. Используются выделение значимых фрагментов технического задания, поиск соответствующих им элементов отчета и проверка степени его покрытия. Разработанный метод, в отличие, например, от косинусной меры сходства, дает лучшее разделение отчетов по критерию хорошего и плохого изложения материала.

Ключевые слова: *информационные технологии, электронный документооборот, проверка документации, автоматическая обработка текстов*.

Оформление корректной документации при проектировании изделий является залогом успешного выполнения проекта. Качественно написанное техническое задание (ТЗ) на проект и проектная документация снижают шансы на срыв поставки изделия. Важную роль играет и отчет о выполненных работах, так как он позволяет повторить проделанные работы и (или) разобраться в них. Особую роль документация играет при работе в соответствии с принципами ИПИ (CALS)-технологии [1].

Существует несколько стандартов на оформление технической документации. В первую очередь – это ГОСТы, по которым оформляется документация в государственных учреждениях, на промышленных предприятиях и т.д. В качестве альтернативы можно привести стандарты International Standard Organization, которые дают рекомендации по составу документации. В области электротехники и телекоммуникаций – серия рекомендаций European Telecommunications Standards Institute, в области разработки программного обеспечения – рекомендации Rational Unified Process от IBM [2] и Microsoft Solutions Framework [3]. Хотя последние не утверждены в качестве государственных стандартов, договор может регламентировать работу в соответствии с этими рекомендациями или по стандартам предприятий, описывающим состав и содержание документации.

Текст отчета проверяется экспертами, что занимает много времени. Недобросовестный исполнитель может пытаться спрятать низкое качество отчета за его объемом, цитатами, уходом в смежную область, применением бюрократического стиля изложения текста и т.д. Для создания «первого эшелона обороны» в помощь экспертам необходимо создать автоматизированную систему, проверяющую степень соответствия отчетов или технической документации тексту ТЗ и помогающую принимать решения о необходимости проведения детальной экспертизы документации. Подобная система выявит наиболее очевидные несоответствия, которые должны быть проверены специалистом. На вход система должна получать тексты документов, содержащие постановку задачи и требования к проекту, а также отчетные документы. Выходом системы является оценка степени сходства и связности фрагментов текста, их смысловой нагруженности.

Для определения смысловой близости документов или их фрагментов применяется, например, метод шинглирования, который основывается на выделении последовательностей слов длины k с последующей оценкой вероятности совпадения текстов документов [5, 6]. Этот метод обладает высокой скоростью, однако, может быть применен лишь для выявления заимствованных фрагментов отчетов. Для сравнения текстов ТЗ и отчетов более подходят методы определения тематической близости документов. Пусть для двух документов вычислены векторы частот встречаемости слов **a** и **b**. Эти векторы определены на множестве всех слов, встречающихся в обоих документах. В этом случае косинусная мера сходства двух документов определяется следующим образом [7]:

$$\cos(\mathbf{a},\mathbf{b}) = \frac{\sum a_i * b_i}{\sqrt{\sum a_i * a_i} * \sqrt{\sum b_i * b_i}}$$

Развитием этого метода является использование векторов частот для словосочетаний различной длины. В этом случае точность определения степени сходства документов возрастает [8]. Косинусная мера сходства документов хорошо зарекомендовала себя при решении различных задач, но определяет лишь сходство тематики документов в целом.

К настоящему времени хорошо проработаны методы хранения технической документации с использованием PDM/PLM технологий [9]. Также существуют методики, учитывающие специфику

* Работа выполнена при поддержке РГНФ (грант № 12-04-00060) и РФФИ (грант № 11-01-00793).

обработки технических документов [10]. Современные достижения в области ИПИ-технологий дают основу для создания системы контроля качества документации, но не позволяют решить задачу определения ее полноты.

Еще одной группой методов, для которой проводятся исследования, является разработка формальных моделей текста документов [11], но для создания подобных систем необходимы большие онтологии предметных областей, не всегда доступные разработчикам. В настоящее время проводятся работы по автоматизированному составлению онтологий [12], однако и эти работы еще не доведены до программного обеспечения, удобного для применения. Наиболее разработанными в теоретическом и практическом плане являются работы в области формализации выделения спецификаций систем [13], однако и они далеки от широкого распространения.

Из сказанного становится очевидным, что существует потребность в создании теоретического метода автоматизации определения соответствия технического задания и итогового отчета, его практической реализации. Кратко изложим основные фрагменты предлагаемого нами метода.

Из текста ТЗ выделяются предложения, содержащие характеристики разрабатываемого изделия. Из выделенных фрагментов извлекаются все пары стоящих рядом слов (коллокации), незначимые коллокации удаляются.

Текст отчета разбивается на фрагменты, для которых также выделяется список коллокаций, после чего находится максимум меры сходства абзацев со значимыми фрагментами ТЗ. Это значение и будет мерой соответствия абзаца тексту технического задания. Полученный результат выдается лицу, принимающему решение (ЛПР), в визуальной форме. С его помощью ЛПР определяет меру соответствия отчета и технического задания, а также определяет необходимость дальнейшего анализа текста отчета или его фрагментов.

Теперь рассмотрим каждый из этапов более подробно.

Представим текст как упорядоченное множество предложений $\mathbf{t} = \langle \mathbf{s}_i \rangle$, предложение – как упорядоченное множество слов: $\mathbf{s}_i = \langle w_{ij} \rangle$. Под словосочетанием будем понимать упорядоченное множество слов: $\mathbf{c} = \langle w \rangle$. Будем считать, что словосочетание входит в i-е предложение ($\mathbf{c} \subset \mathbf{s}_i$), если в $\mathbf{s}_i$ существует контактное подмножество, эквивалентное **c**.

Пусть $\mathbf{K} = \{\mathbf{c}\}$ – список ключевых коллокаций, вводящих требования к изделию, поставленные заказчиком (например, «должно обладать / состоять / …», «обеспечивает», «служит»). Тогда предложение **s**, входящее в текст ТЗ, называется значимым, если $\exists \mathbf{c} \in \mathbf{K}$: $\mathbf{c} \subset \mathbf{s}$. Значимое предложение входит в значимый фрагмент, содержащий в себе одно или несколько предложений:

$$\mathbf{f} = \langle \mathbf{t}, s, e \rangle,$$

где **t** – текст, в который входит фрагмент, s – номер начального предложения фрагмента, e – номер последнего предложения фрагмента.

По результатам анализа текстов ТЗ были разработаны следующие правила, определяющие границы значимых фрагментов:

- если ключевая фраза встречается в предложении, после которого идет перечисление, то выделяется и весь текст до конца перечисления (например, «система должна состоять из следующих подсистем: …»);
- если ключевая фраза встречается в предложении, находящемся в связанном тексте, то выделяется предложение целиком;
- если фраза встречается отдельно (например, заголовок «необходимо»), то выделяется весь следующий абзац.

Эксперименты показали, что качество результата, полученного с помощью метода, возрастает, если помимо значимого предложения в фрагмент включается одно предложение до и два предложения после значимого, так как они чаще всего связаны по смыслу. Предыдущее предложение часто вводит некоторые определения или определяет общее направление, последующие – расшифровывают требования.

На первом шаге по тексту ТЗ ищутся ключевые фразы, к которым применяются приведенные правила. Если условие выполняется, выделяется очередной значимый фрагмент, который заносится в список $\mathbf{F} = \{\mathbf{f}\}$. Два значимых фрагмента могут быть объединены вместе, если их границы пересекаются или между ними нет значимого текста: если $\mathbf{f}_m = \langle \mathbf{t}, s_1, e_1 \rangle$ и $\mathbf{f}_{m+1} = \langle \mathbf{t}, s_2, e_2 \rangle$: $e_1 >= s_2$, то $\mathbf{f}_m = \langle \mathbf{t}, s_1, e_2 \rangle$, а $\mathbf{f}_{m+1}$ удаляется.

На втором шаге выделяются коллокации из значимых фрагментов, для значимых фрагментов рассчитывается вектор признаков:

$$\mathbf{a} = \langle \{w, f\} \rangle,$$

где $w \in \mathbf{f}$ – коллокация, а f – ее частота встречаемости.

Из вектора признаков отсеиваются коллокации с частотами выше 0,75 и ниже 0,25 от максимальной частоты. Эти меры позволяют избавиться от служебных слов и авторских особенностей текста, отсеять редко встречающиеся сочетания. В итоге будет сформировано множество векторов признаков ТЗ $\mathbf{S_1} = \{\mathbf{a}\}$.

На третьем шаге текст отчета разбивается на абзацы, для которых формируется список коллокаций с частотами их встречаемости (вектор признаков **b**: $\mathbf{S_2} = \{\mathbf{b}\}$). Значимость абзаца с номером j вычисляется как максимум косинусной меры сходства вектора **b** с векторами **a** ТЗ или равна нулю, если найденная значимость ниже заданного порога:

$$v_j = \max_i \cos(\mathbf{a}_i, \mathbf{b}_j),$$

где $\mathbf{a}_i \in \mathbf{S_1}$, а $\mathbf{b}_j \in \mathbf{S_2}$.

На четвертом шаге лицо, принимающее решение, получает информацию о покрытии отчета фрагментами ТЗ в виде точечной диаграммы. Так как в работе метода возможны ошибки при выделении свойства или значимого фрагмента, эксперт может получить более подробную информацию о фрагментах отчета и технического задания: соответствие значимых фрагментов, список коллокаций и т.д.

По текстам ТЗ и отчетов формируются точечные диаграммы. На них точка соответствует 100, а строка – 10 000 символов текста. Темные точки показывают части текста, содержащие ключевые слова.

Проверка метода проводилась в два этапа. На первом этапе экспертам давали ознакомиться с содержанием ТЗ и отчета, и высказать свое мнение относительно их содержания, затем документы проверялись в соответствии с разработанным методом. На втором этапе проводилась кросс-проверка документации. Все ТЗ проверялись со всеми отчетами, чтобы проверить гипотезу о том, что максимум совпадения для качественно написанных отчетов должен находиться на соответствующих им ТЗ.

На рис. 1а представлена диаграмма разбора ТЗ, содержащего ненужную информацию. В нем большая часть текста говорит о составе и планах организации, проводимой ею научной работе. Техническое задание, соответствующее рис. 1б, написано в строгом стиле и по требованиям ГОСТа. Ключевые предложения найдены в середине текста в разделе, описывающем требования к изделию. В заключительной части ТЗ формируются сроки разработки, требования к рабочим местам и интерфейсу. Хотя число ключевых фрагментов в первом и втором случае почти одинаково, второй текст выигрывает из-за сжатости и точно поставленных требований.

На рис. 2а показана диаграмма для отчета полного «воды». Блоки из компактно расположенных 5-10 темных точек описывают заявленные в ТЗ требования. Отдельно стоящие темные точки соответствуют единичным коллокациям (например, в заголовке). Здесь на более чем 130 000 знаков отчета было найдено лишь 470 коллокаций, относящихся к ТЗ (считая единичные вхождения в заголовках). Максимальная длина связного текста, имеющего отношение к одному из значимых фрагментов ТЗ, – 700 символов.

На рис. 2б представлен качественно написанный отчет, в котором ключевые коллокации встречаются везде, за исключением начала (содержание, авторы, введение) и конца отчета (юридический и экономический разделы). При длине отчета свыше 130 000 знаков в нем найдено более 3500 коллокаций. Максимальная длина текста, имеющего значимые фрагменты ТЗ – 1500 знаков.

Для кросс-проверки были использованы тексты шести ТЗ и девяти отчетов. Отчетам присвоен номер соответствующих ТЗ. ТЗ с номерами 1-3 написаны по одной тематике, отчеты 5 и 6 написаны по близким тематикам. Отчет с номером 0 не связан ни с одним из ТЗ. Отчеты 3 и 6 представлены в двух версиях. Вторая версия отчетов, отмеченная знаком «+», содержит исправления найденных заказчиком недостатков.

Результаты проверки показаны в табл. 1. Соответствия ТЗ и отчетов выделены рамкой. Результаты удачных проверок выделены темным фоном. Успешные проверки с другими отчетами показаны светло-серым фоном.

Как видно из табл. 1, разработанный метод и программное обеспечение определило высокое качество отчетов, написанных для технических заданий 1-3 и 6. При этом результат работы системы для отчета 3 и 6 совпал с мнением заказчика. Отчет 0 не показал совпадений ни для одного из ТЗ.

Технические задания 4 и 5 не предполагали подробного описания результатов работы и требований к ним. Также в ТЗ 5 требовалось дать рекомендаций по улучшению изделия, это усложнило поиск соответствия. Отчет 4 содержал информацию по предметной области ТЗ 5, в связи с чем их сходство выше.

Для проверки метода были вычислены значения косинусной меры сходства между ТЗ и отчетами. Для этого использовались частоты встречаемости отдельных слов, отсеивание слов не проводилось. Результаты кросс-проверки для косинусной меры сходства сведены в табл. 2.

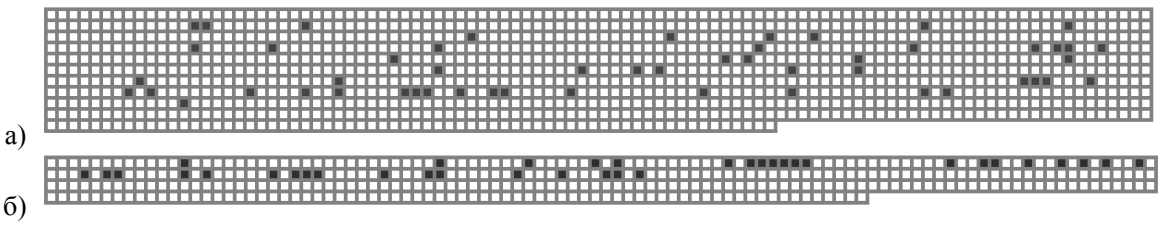

Рис. 1. Визуализация результатов анализа неудачного (а) и удачного (б) ТЗ

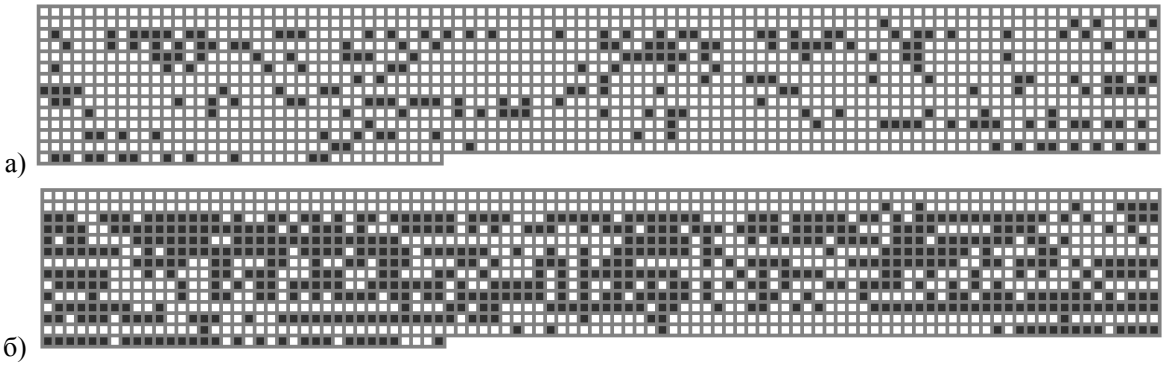

Рис. 2. Точечная диаграмма для неудачного (а) и качественного (б) отчета

Таблица 1

Результаты кросс-проверки для предложенного метода

		Технические задания					
		1	2	3	4	5	6
Отчеты	1	0,521	0,157	0,192	0,032	0,025	0,072
	2	0,394	0,592	0,543	0,056	0,054	0,062
	3	0,37	0,39	0,158	0,05	0,049	0,05
	3+	0,494	0,45	0,535	0,045	0,051	0,054
	4	0,032	0,032	0,066	0,032	0,002	0,031
	5	0,032	0,009	0,02	0,307	0,057	0,095
	6	0,006	0,011	0,007	0,002	0,031	0,638
	6+	0,006	0,009	0,006	0,002	0,016	0,725
	0	0,011	0,043	0,035	0,006	0,006	0,017

Таблица 2

Результаты кросс-проверки для косинусной меры сходства

		Технические задания					
		1	2	3	4	5	6
Отчеты	1	0,533	0,546	0,546	0,198	0,263	0,292
	2	0,532	0,554	0,880	0,151	0,169	0,162
	3	0,554	0,779	0,579	0,183	0,205	0,191
	3+	0,534	0,763	0,612	0,191	0,204	0,192
	4	0,331	0,238	0,326	0,189	0,212	0,167
	5	0,418	0,302	0,331	0,182	0,317	0,243
	6	0,161	0,091	0,116	0,089	0,091	0,443
	6+	0,163	0,091	0,117	0,089	0,091	0,443
	0	0,257	0,174	0,207	0,139	0,175	0,185

Как видно из табл. 2, косинусная мера успешно определяет тексты с общей тематикой, но не показывает качество отчетной документации: максимальные значения достигаются при сравнении ТЗ, не соответствующих данному отчету; разделимость хороших и плохих отчетов отсутствует.

Проверка результатов показала повышение точности работы метода по сравнению с методами, разработанными ранее. Проблему представляют ТЗ, описывающие лишь основные цели работы. Кроме того, даже при соответствии 70% и выше, детальная экспертиза необходима, так как метод не гарантирует полностью достоверных результатов. Для решения этой задачи необходимо применять специализированные методы (например, для экспертизы отчёта по надёжности электронных средств – методику, описанную в [14]).

Тем не менее, метод может применяться как часть автоматизированной системы ведения и хранения документации по проекту и помогать в принятии решений о доработке отчета или о его детальной экспертизе.

СПИСОК ЛИТЕРАТУРЫ

1. Яблочников Е.И., Молочник В.И., Миронов А.А. ИПИ-технологии в приборостроении: учебное пособие. – СПб: СПбГУ ИТМО, 2008. – 128 с.
2. Кролл П., Крачтен Ф. Rational Unified Process - это легко. Руководство по RUP для практиков:пер. с англ. – М.: КУДИЦ-ОБРАЗ, 2004. – 432 с.

3. Тернер М. Основы Microsoft Solutions Framework. – СПб.: Питер, 2008. – 336 с.
4. Клышинский Э.С., Антонова А.Ю. Об использовании мер сходства при анализе документов // Сб. трудов 13-й Всероссийской научной конференции RCDL'2011, с. 246-250.
5. Broder S., Glassman M. Manasse and G. Zweig. Syntactic clustering of the Web // Proc. of the 6th International World Wide Web Conference, April 1997.
6. Зеленков Ю.Г., Сегалович И.В. Сравнительный анализ методов определения нечетких дубликатов для WEB-документов // Сб. трудов 9-й Всероссийской научной конференции RCDL'2007, с. 166-174.
7. Маннинг К.Д., Рагхаван П., Шютце Х. Введение в информационный поиск. – М.: Вильямс, 2011. – 528 с.
8. Клышинский Э.С. Анализ комплексных мер тематического сходства документов // Научно-техническая информация. Сер. 2. 2011. – № 9. – С. 6-11.
9. Колчин А. Что такое PDM? // PC Week. – 2001. – .№ 38.
10. Черников Б. В. Технологии подготовки документов на основе кибернетических методов. – М.: Финансы и статистика, 2009. – 206 с.
11. Тарасенко А.В. Разработка и исследование методов и моделей автоматической проверки текстов на соответствие требованиям технической документации: автореф. дис. … д-ра техн. наук. – Таганрог, 2009.
12. Волкова Г.А. Создание «онтологии всего». Проблемы классификации и решения // Сб. трудов научно-практического семинара «Новые информационные технологии в автоматизированных системах». – М., 2013. – С. 293–300.
13. Заболеева-Зотова А.В., Орлова Ю.А. Автоматизация процедур семантического анализа текста технического задания // Известия Волгоградского гос. технического университета. – 2007. – Т. 9, № 3. – С. 52-55.
14. Жаднов В.В. Методические указания по проведению экспертизы конструкторского документа РР01 «Расчет надежности» для электронных модулей первого уровня с использованием технологии прогнозирования надежности АСОНИКА® – М.: МИЭМ НИУ ВШЭ, 2012. – 16 с.

Материал поступил в редакцию 21.02.14.

Сведения об авторах

КЛЫШИНСКИЙ Эдуард Станиславович – кандидат технических наук, доцент Московского института электроники и математики Национального исследовательского университета – Высшая школа экономики (МИЭМ НИУ ВШЭ)
e-mail: eklyshinsky@hse.ru

КАЛАЧЁВ Ярослав Борисович – аспирант МИЭМ НИУ ВШЭ
e-mail: Kalachyov-YB@sac.minenergo.gov.ru

ЖАДНОВ Валерий Владимирович – кандидат технических наук, доцент, профессор МИЭМ НИУ ВШЭ
e-mail: vzhadnov@hse.ru

УДК 004.89 : [025.4.025 : 81’322.2]

Н.Д. Лыфенко

Об одной концептуальной модели системы автоматической классификации документов на естественном языке

Предлагается концептуальная модель системы, позволяющей решать задачу автоматической классификации текстовых документов на естественном языке, т. е. определения принадлежности нового текстового документа к заранее заданному классу. Приводятся функциональные требования к будущей системе. Рассматриваются различные представления текстов на естественном языке, а также статистические и логико-комбинаторные методы анализа текстов. Данная работа будет интересна специалистам по обработке естественного языка, информационному поиску, компьютерной лингвистике.

Ключевые слова: *системы классификации текстов, машинное обучение, интеллектуальный анализ данных, обработка естественного языка*

ВВЕДЕНИЕ

В статье рассматривается задача автоматической классификации текстов с применением различных моделей текстового документа и разнообразных методов интеллектуального анализа данных. Задача решается с помощью системы классификации текстов *ADC* (Automatic Document Classifier), для которой описана структурная схема и к которой представлены различные требования.

Новизна нашей работы обусловлена следующими обстоятельствами:

- предлагается не просто программа для решения прикладных задач по классификации текста, а платформа для проведения научных экспериментов по сравнению эффективности различных методов классификации текстовых документов, снабженная удобным пользовательским интерфейсом и развитыми средствами формализации результатов экспериментов для последующей их оценки;
- среди реализуемых с помощью предлагаемой платформы методов классификации наряду с широко применяемыми методами (наподобие метода опорных векторов и *k*-ближайших соседей) присутствуют и некоторые варианты ДСМ-метода автоматического порождения гипотез, который для анализа текстов ранее почти не использовался, хотя данные отдельных экспериментов свидетельствуют о хороших возможностях этого метода для анализа текстовых данных.

Описать задачу классификации документов или рубрикации текстов (text categorization, text classification) можно следующим образом.

Пусть заданы множества D и C, такие что $D = \{d_1, d_2, ..., d_{|D|}\}$ – конечное множество документов; $C = \{c_1, c_2, ..., c_{|C|}\}$ – конечное множество заранее заданных классов. Тогда задача автоматической классификации текстов – это задача сопоставления каждой паре $\langle d_j, c_i \rangle \in D \times C$ булевского значения. Документ d_j принадлежит классу c_i, когда значение пары $\langle d_j, c_i \rangle$ равно истине, в противном случае документ не принадлежит этому классу.

Более формально: необходимо построить максимально близкую аппроксимацию неизвестной функции $\breve{\varphi} : D \times C \to \{0,1\}$, описывающей, как документы должны быть помещены в классы, с помощью функции (классификатора) $\varphi : D \times C \to \{0,1\}$ или $\varphi : D \times C \to [0,1]$. В первом случае получим точную классификацию, во втором — ранжирование.

В общем случае классы могут либо пересекаться, либо нет, т. е.

$$\begin{bmatrix} C_i \cap C_j = \varnothing \\ C_i \cap C_j \neq \varnothing \end{bmatrix}, \; i = 0,...,|C|, \; j = 0,...,|C|, \; i \neq j \; .$$

Задано некоторое размеченное множество документов $S \subset D \times C$, для которого известны значения $\breve{\varphi}$. При этом имеются два подмножества $L \subset S$, и $T \subset S$, такие что $L \cap S = \varnothing$ и $L \cup T = S$. В терминах машинного обучения L называется обучающей, а T – тестовой выборкой. На первой происходит обучение классификатора, на второй – его испытание. Обычно для оценки качества классификационной мо-

дели (различные метрики информационного поиска: точность, полнота, f-мера и др.) и ее поведения на независимых данных проводят серию измерений, т. е. перекрестную проверку (cross-validation), используемую в рассматриваемой системе.

КОНЦЕПТУАЛЬНАЯ СХЕМА СИСТЕМЫ ADC

Основной алгоритм работы системы включает три этапа: получение данных, их обработка и анализ результатов (рис 1).

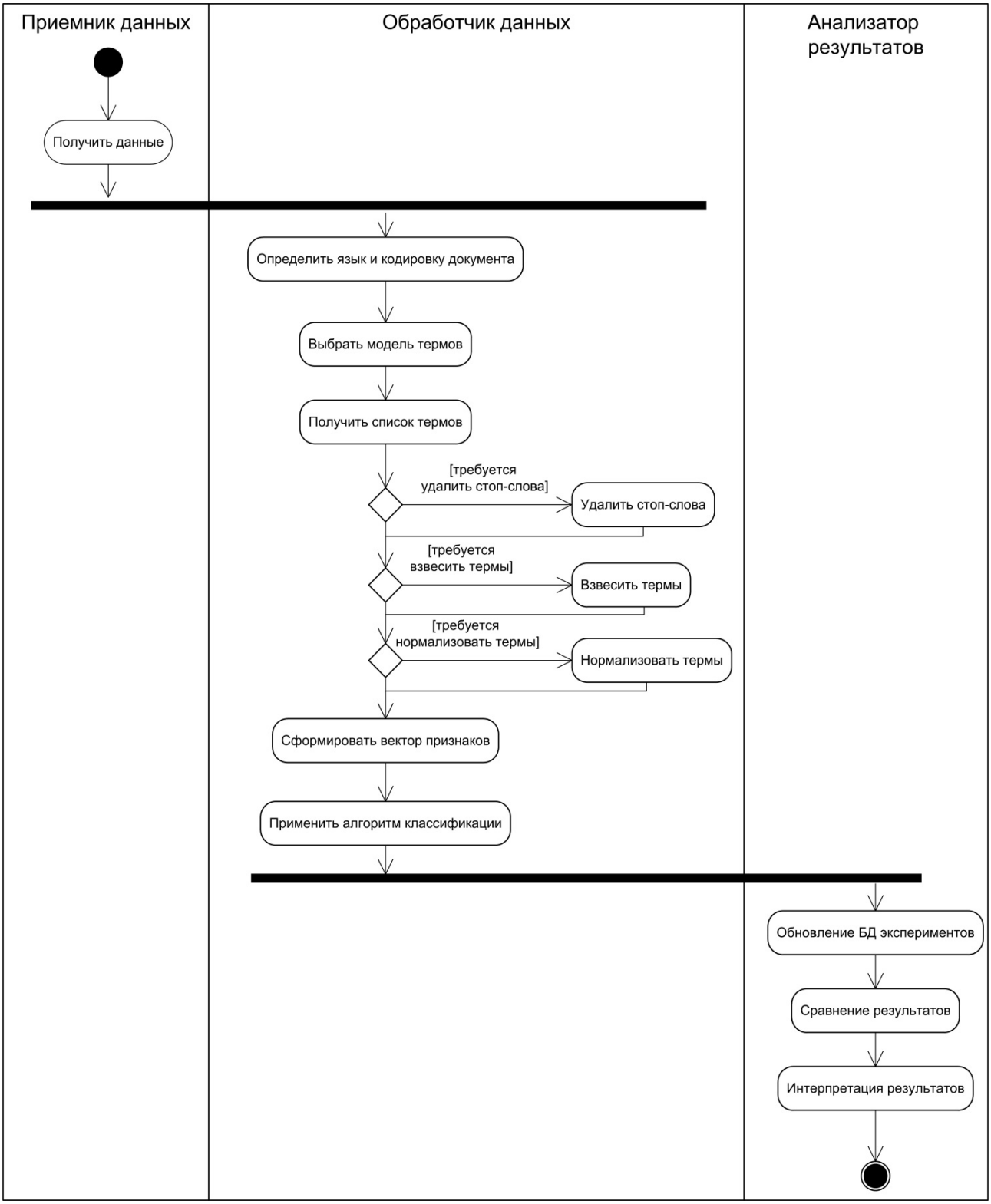

Рис. 1. Концептуальная модель системы ADC

Получение данных

Первый этап состоит в конвертации документов в текстовый файл, содержащий только значимую текстовую информацию. Зачастую объектами интереса являются различные новостные статьи, расположенные в сети Интернет и представленные html-страницами. Очевидно, что теги в html-документе не несут семантической нагрузки и их можно не учитывать. Но обычно удаление тегов не приводит к безошибочному получению чистого текста (plain text) из-за наличия большого количества рекламы, новостного мусора, комментариев, которые вносят большой шум в обучающую выборку. Поэтому был предложен и реализован механизм более точного извлечения чистого текста с учетом тегов, который повысил качество получения текстовых данных из html-документов. Идея механизма состоит в том, что на основе html-страницы строится гистограмма, учитывающая длину строки текста (количество слов/букв) и тег, в который заключен текст. Анализируются только те части документа, которые получат максимальное значение в результате такой композиции.

Обработка данных

Обработка данных состоит из двух модулей: предварительная обработка текстового документа и классификация. В первом модуле решаются лингвистические задачи: использование морфологии для нормализации термов, выбор способов увеличение веса термов и др. Второй модуль работает только с векторами, выполняя математические операции.

Предварительная обработка документа

Документы могут быть представлены с помощью различных кодировок, заданных списком в настройках системы, поэтому на нулевом шаге предварительной обработки документа нужно определить кодовую страницу.

Поддержка по крайней мере двух и более естественных языков приводит к тому, что нужно иметь несколько списков стоп-слов и механизмов нормализации термов. Вследствие этого на первом шаге определяется язык документа.

Описанные выше этапы используют статистический анализ для формирования гистограммы классов (язык, кодировка) и косинус угла между текстами, представленными с помощью векторов, как меру близости. Каждый документ, представленный соответствующим вектором, сравнивается с эталонным вектором, характеризующим класс (кодировку или язык), и выбирается наиболее похожий (т.е. имеющий наибольшую меру близости с эталоном).

Для работы с текстовыми документами обычно используют их векторное представление (vector space model), т. е. отображают текст в вектор. Данную процедуру можно осуществить несколькими способами. Наиболее популярными в области интеллектуальной обработки текста (text mining) являются: *мешок слов* (Bag of Words) и *учет взаимного положения слов*.

В первом представлении не учитывается порядок следования термов[1] и их связь. Второе использует семантические сети или синтаксические деревья разбора, элементы в которых связаны семантическими или синтаксическими отношениями.

В рассматриваемой системе используется модель мешка слов, под которыми понимаются *n*-граммы, т. е. словосочетания длиной не более *n* (*n*=[1…5]) с фиксированным порядком следования. Данное представление хорошо зарекомендовало себя в задачах автоматической классификации [1, 2] и использует простую математическую модель, которую можно эффективно реализовать. Также можно воспринимать текст в виде множества термов, не обязательно стоящих рядом.. При этом количество термов вырастет с $2k - n$ до $2^k - 1$, где k – количество слов в тексте, n – значение пар параметра n в n-грамме. Но в данном случае мы не получим большего значения веса для неделимых словосочетаний и фразеологизмов. Список термов (tokenization) получается путем выделения термов, между которыми есть разделитель (например, пробел).

Зачастую получение всех термов из текста приводит к очень большой размерности вектора признаков, и некоторые слова вносят шум в выборку. Поэтому стоп-слова, т.е. слова, несущие в себе небольшую семантическую нагрузку, имеет смысл не учитывать. Чаще всего это служебные части речи (союзы, предлоги) и наречия (например, вводные слова). Но, скажем, в задаче определения авторства документа, вероятно, имеет смысл оставить такие термы.

Отображение нескольких словоформ в одну лексему также уменьшает размерность пространства, хотя несколько снижает качество классификации. Например, словоформы *столы, столе* отображаются в одну лексему *стол.* Данный процесс называется лемматизацией. Она используется наряду со стеммингом (отображением нескольких словоформ в одну основу). Это два стандартных варианта нормализации терма, которые применяются в задачах интеллектуальной обработки текста.

Построение всех *n*-грамм даже при небольших значениях *n* (2 или 3) также увеличивает пространство признаков, поэтому разумно получать не все термы, а только пары (при $n = 2$) *существительное + существительное,* или *существительное + прилагательное,* или *прилагательное + существительное*. Возможно, учет частеречной принадлежности повысит качество классификации. Данные гипотезы будут проверены в предстоящих экспериментах.

Под этапом взвешивания подразумевается приписывание большего значения некоторым термам, характеризующим некоторую предметную область.

Будет реализовано несколько способов взвешивания термов:

- *бинарный*

$$W_i = \begin{cases} 1, t \in d_i \\ 0, t \notin d_i \end{cases}$$

1 Термами в данной работе мы называем объекты анализа – словосочетания длиной не более *n*.

Вес нулевой, если терм ни разу не встретился в тексте, иначе истина.

- *частотный*

$$W_i = \begin{cases} n, t \in d_i \\ 0, t \notin d_i \end{cases}$$

Представление текста в виде *n*-мерного вектора, *n* – частота встречаемости терма *t* в документе D_i. Соответственно, вес нулевой, если терм ни разу не встретился в тексте.

- *TF-IDF*

$$W_i = tf_i * idf_i \ ,$$

где W_i – вес *i*-го терма; tf_i – частота встречаемости *i*-го терма в данном документе (term frequency); $idf_i = Log\frac{N}{n}$, т.е. это логарифм отношения количества всех документов в коллекции к количеству документов, в которых встречается *i*-й терм (inverse document frequency). Вес признака определяется тем, что чем больше локальная частота терма в документе (term frequency) и чем больше «редкость» терма в коллекции (inverse document frequency), т.е. чем реже он встречается в других документах, тем выше вес данного документа по отношению к терму. Большой вес в *TF-IDF* получат термы с высокой частотой употребления в пределах конкретного документа и с низкой – в других документах.

- *LOWBOW*, локально взвешенная модель мешка слов (Locally Weighted Bag-of-Words Framework).

Все документы представлены множеством локальных гистограмм, каждая из которых сглажена ядрами, но центрирована в разных точках документа [3].

- Самая частотная последовательность (Maximal Frequent Sequences).

Документ представлен в виде последовательности «слов» нефиксированной длины, в которой могут быть пропуски. В работе [3] было эмпирически установлено, что для задачи определения авторства эффективным оказалось использование локально взвешенной модели представления текста, а не классической схемы TF-IDF.

- Частота встречаемости документа и совместная частота встречаемости терма (Document Occurrence Representation (DOR) & Term Co-occurrence Representation (TCOR)).

Семантику терма можно рассматривать как функцию от модели «мешка документов», в которых он встречается (DOR).

Идея TCOR состоит в том, что значимость терма проявляется в контексте других термов, с которыми он встречается и входит в последовательность [4].

Последние три представления (LOWBOW, MFS, DOR/TCOR) не получили (пока) большого распространения в силу их недавнего появления, но в работе [5] показывают хорошие результаты по сравнению с базовым TF-IDF.

Классификация

Наиболее популярными при решении задачи автоматической классификации являются методы машинного обучения, описанные ниже*: деревья принятия решений, нейронные сети, линейные классификаторы, вероятностные алгоритмы и др.* Данный обзор составлен на основе [6].

1. *Метод K-ближайших соседей (k-nearest neighbours, k-NN)*

Для того чтобы найти классы, релевантные документу *d*, этот документ сравнивается со всеми документами из обучающей выборки. Для каждого документа из обучающей выборки находится мера близости – косинус угла между векторами признаков. Далее из обучающей выборки выбираются *k* документов, ближайших к документу *d* . Для каждого класса вычисляется релевантность по формуле:

$$R(c_i, d) = \sum_{a \in K_i} \cos(d, a) \ ,$$

где $R(c_i, d)$ – релевантность документа d классу c_i, K_i – множество текстов класса c_i, наиболее близких к документу *d*. Пусть $r_i = \max\{\cos(d,a) \,|\, a \in c_i\}$, тогда $K_i = \{a \,|\, \cos(d,a) = r_i\}$.

Данный метод, в отличие от классификатора Роше, находит границы классов локально и не требует фазы обучения.

2. *Классификатор Роше*

Для векторной классификации документов необходимо определить границы между классами, поскольку именно они определяют результат классификации. В качестве границ используются центроиды класса:

$$\mu = \frac{1}{|D_c|} \sum_{d \in D_c} \vec{v}(d) \ ,$$

где D_c – множество документов из пространства *D*, принадлежащих классу *C*, $\vec{v}(d)$ – нормализованный вектор документа.

После вычисления взвешенных центроидов для каждой категории классификатор Роше определяет принадлежность документа классу при помощи вычисления меры близости между вектором обрабатываемого документа и центроидом каждой рубрики. Полученная мера близости сравнивается с заданным порогом. В качестве меры близости часто используется косинус угла между векторами.

Этот метод обладает полезной особенностью: взвешенные центроиды можно быстро пересчитать при добавлении новых примеров. Эта особенность применима, например, в задаче адаптивной фильтрации, когда пользователь постепенно указывает системе, какие документы выбраны правильно, а какие нет. В ответ система может уточнить результаты, учитывая новые документы.

Существует множество различных модификаций данного метода. В связи со своей простотой, данный метод часто используется в качестве базового для сравнения с другими методами.

3. *Нейронные сети (Neural network)*

Данный метод использует математическую модель искусственных нейронных сетей для обучения классификатора. Так, множеству входных нейронов соответствует набор термов входного документа, а выходному нейрону – собственно, класс. Веса на ребрах между элементами сети определяют степень связи термов. Обычно для обучения сети используют алгоритм обратного распространения ошибки. Входной нейрон или вектор документа $d_i = \left\langle t_i,...,t_{|D|} \right\rangle d_i \in D$ взвешивается и суммируется $S = \sum_{i=0}^{|D|} w_j * t_i$. Строится нелинейная функция активации нейрона $y = I(S)$. Здесь S и I – параметры процедуры классификации. Функциональность нейрона проста, поэтому для решения конкретных задач нейроны объединяются в сети.

4. *Деревья принятия решений (Decision Trees)*

Классификатор на основе дерева принятия решений представляет собой дерево, на ребрах которого записаны веса, вершины помечены соответствующими термами, концевым вершинам приписан класс. Для классификации нужно обойти все вершины дерева до листьев, сравнив их и определив дальнейшее направление обхода. Большой популярностью пользуются алгоритмы ID3, C4.5, CART. Деревья принятия решений имеют наглядное представление, просты в понимании, используют модель белого ящика.

5. *Наивный байесовский классификатор (Naive Bayes Classifier)*

В основе данного метода лежит вероятностная модель документа, имеющая сильное предположение о независимости компонент вектора. Согласно теореме Байеса можно вычислить вероятность принадлежности документа d классу c: $P(c|d) = P(c)\prod_{1\le k\le n_d} P(t_k|c)$, где $P(t_k|c)$ - условная вероятность того, что термин t_k появится в документе из класса c, $P(c)$ – априорная вероятность принадлежности классу c. Последовательность $\left\langle t_1, t_2,..., t_{n_d} \right\rangle$ состоит из лексем документа d, n_d – количество таких лексем в документе d. Цель классификации текстов – найти наилучший класс для документа. В рассматриваемом методе таковым является наиболее вероятный класс, или класс c_{map}, имеющий максимальную апостериорную вероятность (maximum a posteriori – MAP)

$$c_{map} = \arg\max \hat{P}(c|d) = \arg\max P(c) \prod_{1\le k\le n_d} P(t_k|c)$$

В данном случае $\hat{P}$ – оценка значения P с помощью обучающих множеств. Этот метод показывает высокую скорость работы, имеет простую математическую модель и достаточно высокое качество классификации. Его можно рекомендовать для построения классификатора, когда существуют жесткие ограничения на время и в качестве базового при сравнении различных методов машинного обучения.

6. *Машина опорных векторов (Support Vector Machine, SVM)*

Основная идея метода в том, чтобы отобразить n-мерный вектор в пространство большей размерности и построить в нем гиперплоскость, такую, что расстояние от нее до ближайшей точки максимально. Данное расстояние называется *зазором*, а сами точки – опорными векторами. Метод был предложен В.Н. Вапником в работе [7].

Используемый метод

Многие даже классические методы не дают возможность осуществить достаточно релевантную классификацию или приводят к алгоритмам, имеющим большую вычислительную сложность. В качестве базовых для сравнения (base line) некоторые из них также будут реализованы и исследованы.

В настоящей работе наряду с уже зарекомендовавшими себя алгоритмами машинного обучения и автоматической классификации будет использован логико-комбинаторный метод интеллектуального анализа данных – *ДСМ-метод автоматического порождения гипотез* [8–10]. В его основе лежат формализованные правила правдоподобных рассуждений. ДСМ-метод реализует синтез трех познавательных процедур – индукции, аналогии и абдукции. Свое сокращенное название метод получил в честь английского философа и логика Д.С. Милля. Данный метод показывает неплохую эффективность в задаче определения тональности текста [11], существенно превосходит базовый классификатор, он на 4% лучше SVM без подбора оптимального числа характеристик и менее чем на 1% отличается от SVM с подбором характеристик. Поэтому есть все основания использовать этот метод и предположить, что и для другого круга задач автоматической классификации текстов (задачи определения темы текста и авторства) данный метод будет эффективен.

Анализ результатов

В силу специфики применения различных методов интеллектуального анализа данных следует выяснить, какие из имеющихся алгоритмов и моделей представления текста дают наиболее релевантную классификацию документов для конкретной предметной задачи, т.е. требуется инструментарий для хранения результатов экспериментов и их сравнения.

Для формирования базы данных экспериментов с различными опциями обработки текста и классификации необходимо иметь инструментарий, позволяющий быстро и удобно производить настройку экспериментов. Для решения данной задачи была разработана объектная модель проекта системы.

Проект состоит из имени (путь к файлу проекта), даты проведения, настройки проекта и серии экспериментов (рис. 2). Последнее поле включает имя эксперимента (уникальный идентификатор), дату проведения, список файлов для обучения и тестирования классификатора, результаты и настройки. Поля «результаты» и «настройка эксперимента» – это указатели на соответствующие поля в базе данных по экспериментам. Это своего рода краткое описание эксперимента: методы для предварительной обработки текста, взвешивания и классификации, список критериев эффективности классификации и их значений.

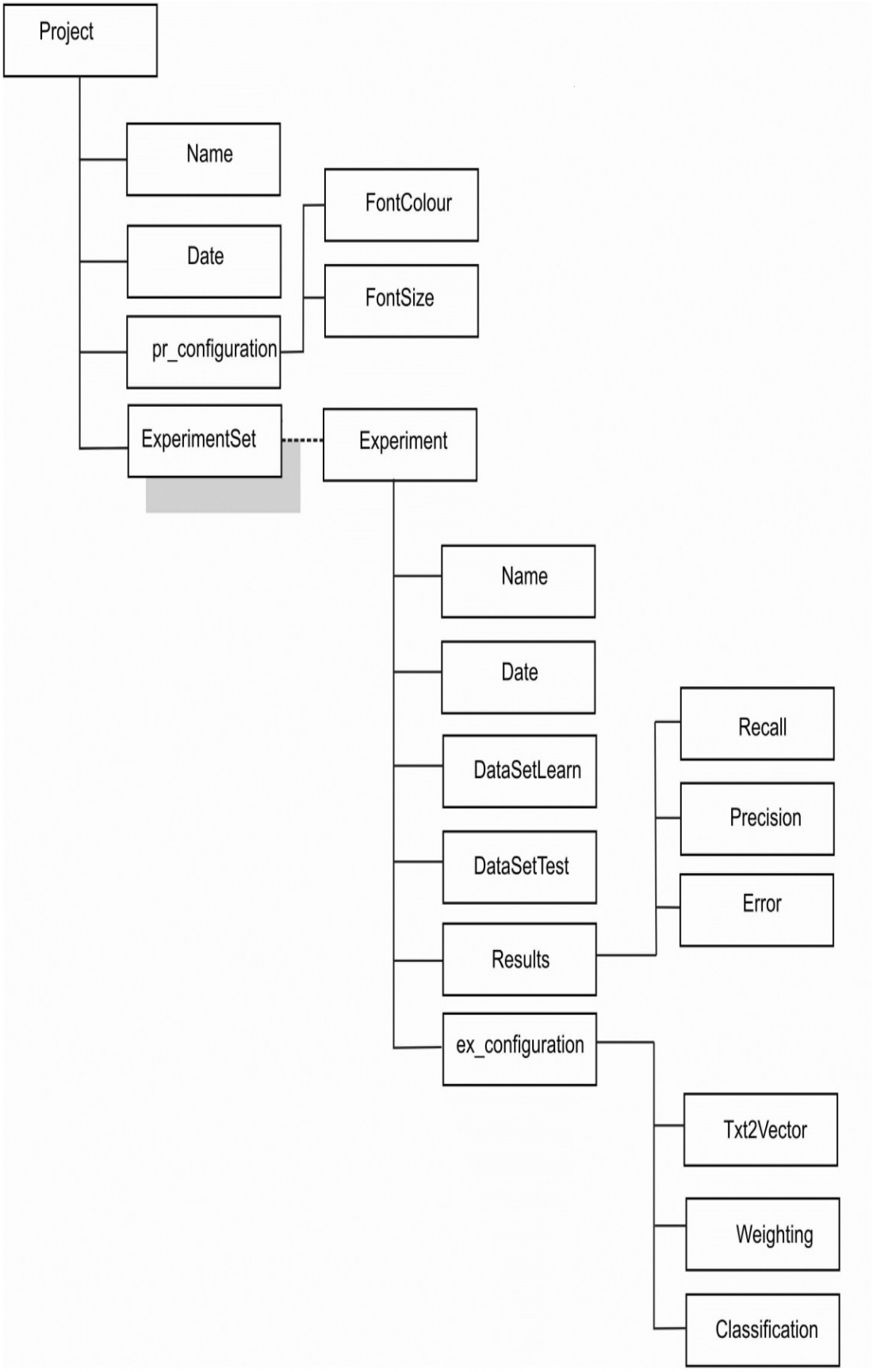

Рис. 2. Объектная модель проекта системы ADC

ТРЕБОВАНИЯ, ПРЕДЪЯВЛЯЕМЫЕ К СИСТЕМЕ

Бизнес-требования

Основная задача системы – осуществление релевантной классификации документов, т.е. отнесение их к заранее заданным классам, с помощью различных техник обработки и представления текстового документа и алгоритмов интеллектуального анализа данных (стиль, тема, автор). Эффективность решения последних предметных задач зависит от модели текста и правильности выбора применяемых к ней методов. Поэтому разумно реализовать систему, в настройках которой можно выбрать наиболее приемлемый алгоритм для конкретной предметной задачи классификации. К сожалению, в существующих системах обработки естественного языка не реализован используемый нами ДСМ-метод[2] автоматического порождения гипотез, поэтому создание собственной системы представляется логичным и востребованным.

Функциональные требования

Система состоит из трех основных модулей: получение данных, их обработка и анализ[3] и решает задачу автоматической классификации текстовых документов, представленных в виде неразмеченного текста, т.е. позволяет относить новый текст к заранее заданному классу. Особенностью системы является наличие множества моделей представления языка и методов интеллектуального анализа данных, применение различных комбинаций которых позволит (предположительно) решать различные предметные задачи более эффективно, чем в системах со строго заданным списком методов. Подобные системы полезны (и используются) при базовом сравнении композиций различных методик и алгоритмов обработки текста на естественном языке. Некоторые из них, например, RapidMiner Studio[4], позволяют создавать свои плагины на языке Java, внедрять код на языке R, ориентированный на статистическую обработку входных данных.

ТЕКУЩЕЕ СОСТОЯНИЕ СИСТЕМЫ

Реализован пользовательский интерфейс (рис. 3), позволяющий задавать файл проекта, который состоит из серии экспериментов. Файл настройки каждого эксперимента включает в себя такие характеристики, как: источник данных для обучения и тестирования, метрики схожести, метод предварительной обработки текстового документа, возможность использования дополнительного взвешивания терма, метод классификации и/или кластеризации, использование списка стоп-слов и др. Такое представление частично повторяет структурную схему проекта в системе ADC.

На рис. 4 изображен пользовательский интерфейс для добавления эксперимента в файл проекта и его настройки: выбор методов предварительной обработки текстового документа и методов классификации. Также представлены возможные метрики, результаты которых пользователь получит после проведения эксперимента в файле отчета.

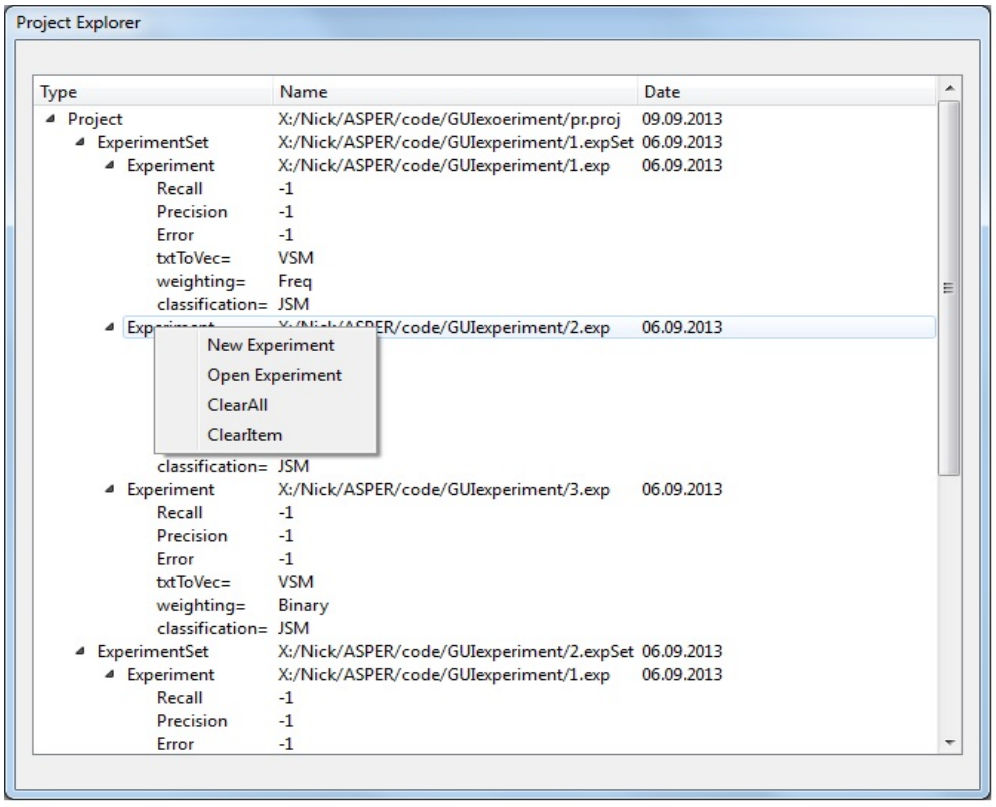

Рис. 3. Пользовательский интерфейс, представление проекта системы ADC

[2] См. ранее об опыте использования ДСМ.
[3] См. ранее: «Концептуальная схема системы ADC».
[4] http://rapidminer.com/products/rapidminer-studio/

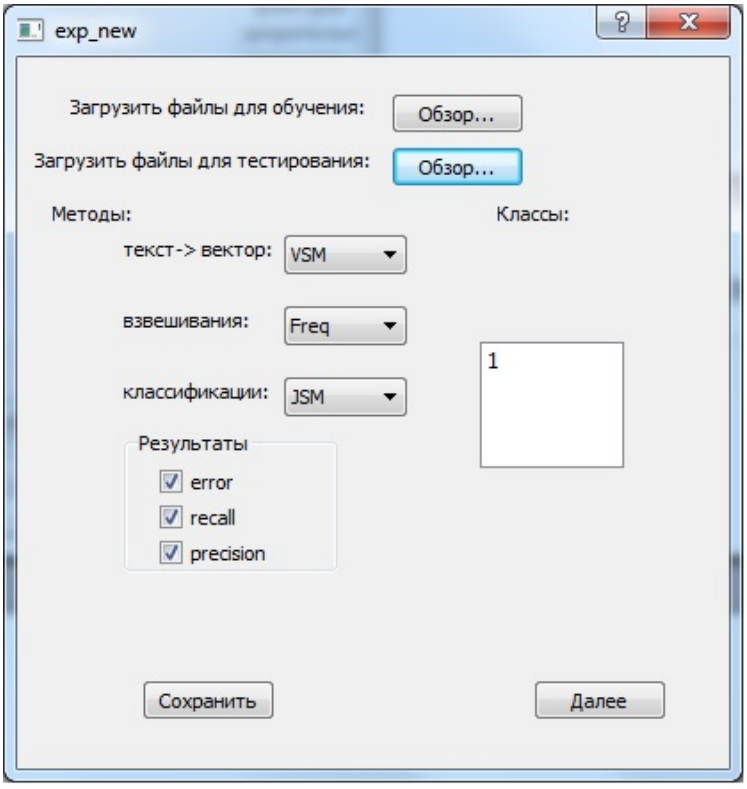

Рис. 4. Редактирование настроек эксперимента

В целом отражены все поля из описанной ранее объектной модели проекта, которая представлена в виде xml-файла. Для проекта реализованы методы загрузки и сохранения в данный файл.

Для представления текста используется модель мешка слов (*Bag of Words*), т.е. не учитывается порядок следования термов и их синтаксическая и семантическая связь. В качестве термов используются биграммы и словоформы, т. е. «слова» длиной не больше двух. Каждый терм при этом нормализуется. Удаляются стоп-слова, т.е. те словоформы, которые не несут сильной смысловой нагрузки, например, служебные части речи.

На данном этапе можно получить список термов, взвесить их и отобразить в вектор, после чего получить список термов с посчитанными весами

$$W_i = (tf_i * idf_i) * W$$

На текущей стадии увеличиваются веса W_i у термов, начинающихся с большой буквы, и термов, входящих в определенный список.

Таким образом, задачу предварительной обработки текстового документа (формирования вектора признаков) можно считать решенной.

* * *

В настоящей статье было дано краткое описание задачи автоматической классификации документа, предложена концептуальная модель системы ADC, решающей данную задачу, и представлены требования к ней, а также описаны используемые методы обработки текста и классификации.

Таким образом, описанная система позволит проводить эксперименты с большим набором опций и метрик, использовать различные корпуса текста для обучения классификаторов и их тестирования, сравнивать результаты, строить графики и диаграммы по характеристикам эффективности методов классификации (точность, полнота, f-мера и др.), выбирая тем самым наиболее подходящий алгоритм для конкретной задачи.

СПИСОК ЛИТЕРАТУРЫ

1. Náther P. N-gram based Text Categorization. Diploma thesis. Institute of Informatics, Comenius University, 2005.
2. Cavnar W.B., Trenkle J.M. N-Gram-Based Text Categorization // Proceedings of the Third Symposium on Document Analysis and Information Retrieval, 1994.
3. Lebanon G., Mao Y., Dillon M. The Locally Weighted Bag of Words Framework for Document Representation // Journal of Machine Learning Research. – 2007. – Vol. 8. – P. 2405–2441.
4. Ahonen-Myka H. Finding All Maximal Frequent Sequences in Text // Proceedings of the 16th International Conference on Machine Learning ICML-99 Workshop on Machine Learning in Text Data Analisys / eds. D. Mladenic and G. Grobelnik. – J. Stefan Institute, Ljubljana, 1999. – P. 11–17.
5. Cabera J.M., Escalante H.J., Montes-y-Gómez M. Distributional Term Representations for Short-Text Categorization // 14th International Conference on Text Processing and Computational Linguistics (CICLing 2013). – Samos, Greece, 2013.

6. Sebastiani F. Machine learning in automated text categorization // ACM Computing Surveys. – 2002. – Vol. 34(1). – P. 1–47.
7. Вапник В.Н. Восстановление зависимостей по эмпирическим данным. – М.: Наука, 1979. – 448 с.
8. Финн В.К. О возможностях формализации правдоподобных рассуждений средствами многозначных логик // VII Всесоюзный симпозиум по логике и методологии науки. – Киев: Наукова думка, 1976. – С. 82–83.
9. Финн В.К. Базы данных с неполной информацией и новый метод автоматического порождения гипотез // Диалоговые и фактографические системы информационного обеспечения. – М., 1981. – С. 153–156.
10. Финн В.К. О машинно-ориентированной формализации правдоподобных рассуждений в стиле Ф. Бэкона – Д.С. Милля // Семиотика и информатика. – 1983. – Вып. 20. – С. 35–101.
11. Котельников Е.В. Опыт применения ДСМ-метода для определения тональности текста // Труды Тринадцатой национальной конференции по искусственному интеллекту с международным участием КИИ-2012 (16–20 октября 2012 г., Белгород). – Т. 1. – Белгород: Изд-во БГТУ, 2012. – С. 135–142.

Материал поступил в редакцию 14.03.14.

Сведения об авторе

ЛЫФЕНКО Николай Дмитриевич – аспирант Российского Государственного Гуманитарного Университета, Москва
e-mail:LyfenkoNick@ya.ru

АВТОМАТИЗАЦИЯ ОБРАБОТКИ ТЕКСТА

УДК 811.581'322.2

Л.С. Холкина

Семантические поля *ПОЛНЫЙ* и *ПУСТОЙ* в китайском языке: системное описание как основа для словаря нового поколения *

Проводится системное описание прямых и переносных значений лексических единиц, относящихся к семантическим полям ПОЛНЫЙ и ПУСТОЙ в китайском языке. Описание выполнено с учетом типологически релевантных параметров и может служить основой для составления мультиязычного словаря нового типа.

Ключевые слова: *полный, пустой, лексикография, словарь, китайский язык*

Лексическая типология является одним из наиболее динамично развивающихся направлений современной лингвистики. Только в последние годы вышло несколько тематических обзоров [1-3] и даже специальный выпуск журнала Linguistics с обзорной статьей «Новые направления лексической типологии» [4].

Поскольку в основе исследований по лексической типологии лежит точное и систематичное сопоставление значений, одним из их практических применений может служить создание мультиязыковых словарей нового поколения, а также использование результатов этих исследований в работах по машинному переводу.

В настоящей статье мы рассмотрим достаточно хорошо исследованные с типологической точки зрения семантические поля ПОЛНЫЙ и ПУСТОЙ [5-7] на новом – китайском – материале. Нашей задачей будет, опираясь на релевантные с типологической точки зрения противопоставления, представить систематичное описание этих полей в сопоставлении с русским материалом.

ПРЯМЫЕ ЗНАЧЕНИЯ

Базовое значение поля ПУСТОЙ – 'отсутствие содержимого в контейнере'. Прототипическим контекстом для данного значения является *пустой X (от У-а),* где *X* – существительное, обозначающее контейнер (например, бокал, бутылка, коробка и т.д.), а *У* – отсутствующее содержимое (ср. [5]).

В языках мира существуют две основные синтаксические стратегии употребления прилагательных с этим значением. В некоторых языках *У*, т. е. отсутствующее содержимое, может выражаться на поверхностном уровне[1], а в некоторых – нет (ср. русск. ***пустая от зерна** коробка*). Даже когда *У* не выражается на поверхностном уровне, он осознается как говорящим, так и слушающим, и отсутствие его в предложении не затрудняет понимания: у каждого «контейнера» существует некий коррелят – либо его прототипическое содержимое, либо важный компонент, в нем ожидаемый. Иными словами, прилагательное «пустой» обозначает отсутствие ожидаемого содержимого.

Базовым значением для поля ПОЛНЫЙ является 'содержащий в себе что-либо до возможных пределов'. Прототипическими контекстами для данного поля являются *полный X* и *X, полный У-а*, где *X* – существительное, которое обозначает контейнер, а *У* – его содержимое. Актант, обозначающий содержимое, может выражаться или не выражаться на поверхностном уровне в зависимости от конструкции.

Структуру семантических полей ПУСТОЙ и ПОЛНЫЙ в русском и китайском языках в основном определяют следующие противопоставления [6][2]:

– форма vs. функция

– функция: тип содержимого

– функция: прототипический контейнер vs. плоскости vs. пространства vs. кронштейны

– посессивная зона

* Исследование выполнено при поддержке гранта РФФИ №14-06-00343

[1] Ср. англ. конструкцию *smth empty of smth* (букв. 'нечто, пустое от чего-либо'):

What half of China is almost empty of people?

'Какая часть Китая практически 'пуста от людей'?' [wiki.answers.com]

[2] В рамках данной статьи мы остановимся только на тех параметрах, которые могут вести к лексическим противопоставлениям в русском или китайском языках.

Форма vs. Функция

Противопоставление формы и функции - основное противопоставление в поле ПУСТОГО, которое часто служит основанием для использования разных лексем в языках мира. В русском по этому признаку различаются лексемы *пустой* и *полый*. *Пустой*, как уже говорилось, обозначает отсутствие (ожидаемого) содержимого в контейнере - в этом случае пустое пространство может быть использовано функционально. *Полый* также описывает ситуацию отсутствия содержимого, но не внешней сущности, а части самого предмета. В таком случае функциональное использование пространства не предполагается - мы говорим не о контейнере и функции, а о предмете и его форме.

В китайском языке семантическое поле ПУСТОЙ обслуживается двумя лексемами, тоже противопоставленными по этому параметру: 空 *kōng* 'пустой' и 空心 *kōngxīn* 'полый'. При этом слово 空心 *kōngxīn* (букв. 'пустой-сердцевина') этимологически производно от 空 *kōng*. Мы включаем его в рассмотрение, потому что в современном языке слово 空 *kōng* не может описывать форму предмета, оно описывает только пустоту функциональную, которую можно использовать: ср. **空** 箱子 ***kōng*** *xiāngzi* 'пустой ящик/чемодан'

Слово же 空心 *kōngxīn* описывает полые предметы: например, 'пустотелый кирпич' **空心** 砖 ***kōngxīn*** *zhuān* или дерево:

老	槐树	**空心**	了。	(1)
lǎo	huáishù	**kōngxīn**	le	
старый	софора японская	полый	MOD[3]	

'В старой софоре японской (дерево) сердцевина стала полой'.(XHCD [18]*)*

Параметр «форма vs. функция», основной в семантическом поле ПУСТОЙ, релевантен и для семантического поля ПОЛНЫЙ (ср. *цельный* и *полный* в русском). В китайском языке это противопоставление тоже актуально, в нем используются две основные лексемы: 满 *mǎn* 'полный' и 实 *shí* 'цельный'.

При этом, если 'полый' описывает особую форму объекта, то 'цельный' – более сложный концепт: отсутствие внутренней полости является не основным значением, а следствием внутренней однородности описываемого объекта. Ср. определение Малого академического словаря [8]: 'состоящий, сделанный из одного вещества, из одного куска, не составной'. Эта несимметричность проявляется в китайском языке: слово 'полый' (空心 *kōngxīn*) является производным от 'пустой' (空 *kōng*), а слово для 'цельного' (实 *shí*) этимологически не связано с 'полным' (满 *mǎn*).

Идеи «отсутствия полости» и «внутренней цельности» в китайском языке разделяются. В первом случае обычно употребляется слово 实心 *shíxīn* (букв. 'цельный-сердцевина'), симметричное по своей структуре слову 'полый': 'цельный шар' 实心球 *shíxīnqiú*. Во втором случае используется просто 实 *shí*: ср. 'цельнодеревянный пол' 实木地板 *shímù dìbǎn*. Различие между значениями 'полый' и 'цельный' проявляется и в метафорических употреблениях соответствующих прилагательных: в отличие от лексем 空 *kōng*, 实 *shí* и 满 *mǎn*, ни 空心 *kōngxīn*, ни 实心 *shíxīn*, специализирующиеся на описании формы, переносных значений не развивают.

Тип содержимого и тип контейнера

Ситуации с разными типами контейнеров могут концептуализовываться в языках мира по-разному. Прототипическим контейнером является пространство, ограниченное с пяти сторон, но в языках мира аналогичным образом могут описываться предметы, соответствующие этому прототипу только по функции «содержать/переносить/хранить что-либо», но не по топологическим характеристикам. Это плоские предметы (как *поднос, лист бумаги* и др.), кронштейны (*вешалка, крючок* и др.), неограниченные пространства. Например, в корейском по данному параметру противопоставлены три лексемы: *thengpita* ('пустой о контейнерах'), *pita* ('пустой о плоских предметах') и *konghehata* ('пустой о больших неограниченных пространствах')[4], в английском – две *empty* ('пустой о контейнерах, пространствах и кронштейнах') и *blank* ('пустой (о листе бумаги)').

В поле ПУСТОГО и русский, и китайский языки использует одну базовую лексему во всех случаях, ср. *пустой бокал, пустое поле, пустой поднос, пустой лист бумаги, пустой крючок* в русском. В китайском слово 空 *kōng* также может характеризовать непрототипические контейнеры: например, плоские «вместилища» типа неисписанного «листа бумаги (2), пустого крючка (3), пустой площадки (4)».

Этот тип объектов может описываться также сочетанием 空无一字 *kōng wú yí zì* (букв. 'пустой-без-один-иероглиф').

При этом, как и русская лексема *пустой*, китайское слово 空 *kōng* может описывать отсутствие как предметов (5), так и людей (6).

Интересно, что в «Словаре современного китайского языка» [9] специально указывается на двусмысленность выражения «пустой дом»: 【空房】 *kōngfáng* 没有放东西或无人居住的房子 *méiyǒu fàng dōngxī huò wú rén jūzhù de fángzi* (букв. 'дом без сложенных вещей или проживающих людей').

То, что случай, когда «содержимым» являются люди, – особый, подтверждается тем, что для него возможны дополнительные противопоставления. Например, в русском, когда речь идет об отсутствии людей, а не предметов, ограниченные пространства (*дом, квартира* и др.) описываются словом *пустой*, а неограниченные – *пустынный.*

[3] MOD — модальная частица. Далее по тексту: ATR — маркер определения к существительному или глаголу; CL — счетное слово — классификатор; LOC — пространственные предлоги и послелоги; NEG — показатель отрицания; RDP — редупликация; PCL – конечная частица.

[4] Данные предоставлены А.С. Сорокиной.

请教,	打印机	无故	走	**空**	纸,	怎么	办?	(2)
qǐng jiào	dǎyìnjī	wúgù	zǒu	**kōng**	zhǐ,	zěnme	bàn	
скажите	принтер	не.иметь причина	идти	пустой	бумага	как	делать	

‘Подскажите, пожалуйста, что делать, если принтер безо всякой причины выдает пустые листы?’ (baidu)[5]

Пустые кронштейны также могут описываться этим словом:

那么，	我	就	占用	帽架上	的	一个
nàme	wǒ	jiù	zhànyòng	màojià shàng	de	yí gè
тогда	я	сразу	занимать	шляпа.вешалка LOC	ATR	один CL

空	挂钩	了。”	(3)
kōng	guàgōu	le	
пустой	крючок	MOD	

‘Тогда я займу пустой крючок на вешалке для шляп’. (ccl)

空 *kōng* может употребляться и для описания обширных неограниченных пространств:

操场上	**空**无一人	(4)
cāochǎng shàng	**kōng** wú yì rén	
спортплощадка LOC	пустой не.иметь один человек	

‘На спортплощадке никого нет’. (инф.)

各种	粮食	袋子	堆	在四周，	中间
gè zhòng	liángshí	dàizi	duī	zài sì zhōu	zhōngjiān
каждый вид	зерновые	мешок	складывать	LOC четыре сторона	середина

留下	一块	宽大的	**空**	地方。	(5)
liúxià	yí kuài	kuāndà de	**kōng**	dìfang	
оставлять	один CL	широкий ATR	пустой	место	

‘Мешки с разными видами зерна свалили в кучи по четырем сторонам, посередине осталось большое свободное место’. (ccl)

一间	屋子里	到处	都	在	漏雨，	可是	谁
yī jiān	wūzi lǐ	dàochù	dōu	zài	lòuyǔ	kěshì	shuí
один CL	комната LOC	везде	все	PRG	протекать	но	кто

也	没	被	淋湿，	为什么？	答案：	**空**	房子	(6)
yě	méi	bèi	línshī	wèi shénme	dá’àn	**kōng**	fángzi	
тоже	NEG	PASS	промокнуть	почему	ответ	пустой	дом	

‘В комнате сильно протекает крыша, но никто не промок, почему?
Ответ: пустой домик’ (baidu)

В семантическом поле ПОЛНЫЙ разные типы контейнеров получают разное осмысление. Так, например, в русском слово *полный* не сочетается с кронштейнами (**полный крючок*) и плохо сочетается с плоскими (вертикальными) поверхностями и неограниченными пространствами (**полный лист, *полная стенка, *полное поле*).

В китайском языке употребление 满 *mǎn* в этих контекстах возможно в особой синтаксической позиции результативного компонента (7-8):

挂钩	挂**满**	了	(7)
guàgōu	guà**mǎn**	le	
крючок	вешать-полный	MOD	

‘Крючки полностью завешаны’. (baidu)

[5]baidu - поисковая система Байду (URL: http://www.baidu.com). Далее по тексту: ccl - корпус Пекинского университета (URL: http://ccl.pku.edu.cn:8080/ ccl_corpus /index.jsp?dir=xiandai\); НКРЯ - Национальный корпус русского языка (URL: www.ruscorpora.ru), инф. – пример получен от информанта.

两天	半	的	时间,	留言	已经	写**满**	了	3 本。	(8)
liǎng tiān	bàn	de	shíjiān	liúyán	yǐjīng	xiě**mǎn**	le	3 běn	
два дня	половина	ATR	время	отзыв	уже	писать. полный	MOD	3 CL	

‘За два с половиной дня отзывами исписали три тетради’. (ccl)

行	文	到	此，	**满**	纸	泪	痕。	(9)
xíng	wén	dào	cǐ	**mǎn**	zhǐ	lèi	hén	
следовать	письмена	до	этот	полный	бумага	слеза	след	

‘Письмо здесь заканчивается, весь лист в следах слез’. (ccl)

满	书	的	错别	字,	怀疑	不	是
mǎn	shū	de	cuòbié	zì	huáiyí	bù	shì
полный	книга	ATR	неверный	иероглиф	сомневаться	NEG	быть

正规	出版	的	吧…	(10)
zhèngguī	chūbǎn	de	ba	
нормальный	издавать	ATR	PCL	

‘В книге полно неправильных иероглифов, подозреваю, что это не официальное издание...’ (baidu)

В ситуациях, когда на поверхностном уровне выражены оба участника, – и контейнер, и содержимое, употребление 满 *mǎn* тоже возможно (9).

Однако в примерах (9-10) значение конструкции меняется: речь идет не о том, что лист бумаги заполнен, а о том, что содержимого (следов слез, опечаток) много. В русском языке сочетания типа *полная работа ошибок*, *полная книга опечаток* тоже более приемлемы, чем **полная работа*, **полная книга*.

Посессивная зона

К семантическим полям ПУСТОЙ-ПОЛНЫЙ примыкает зона «наличие/отсутствие посессора». Совмещение в одной лексеме значений ‘отсутствие временного посессора’ и ‘пустой’ является в языках мира довольно распространенным явлением [7]. В русском языке эти два значения разделены между лексемами *свободный* и *пустой*. Китайское же слово 空 *kōng* может употребляться и в «зоне посессора»:

空座 (11)
kōngzuò
‘*свободное место (напр. в театре)’*

空车 (12)
kōngchē
‘*свободное такси’*

Для поля ПОЛНОГО такое совмещение в целом нетипично: в выборке [7] сербский является единственным языком, использующим одну и ту же лексему в значениях ‘занятый’ и ‘полный’. Значение наличия временного посессора маркируется особым способом как в русском (*занятый*), так и в китайском языках (сочетанием 有人 *yǒurén* (букв. ‘иметь человек’)) (13).

Таким образом, в соответствии с типологическими прогнозами, зона отсутствия посессора обслуживается в китайском языке той же лексемой, что и значение ‘пустой’, а ‘наличие посессора’ описывается иначе, чем ситуация ‘полный’.

Можно видеть, что в области прямых значений материал русского и китайского языков хорошо соотносится. Так, в китайском, как и в русском, значимо противопоставление по параметру «форма-функция» и по типу контейнера.

При этом китайская система является более доминантной, так как основная функциональная лексема 空 *kōng* используется и в ситуации отсутствия временного посессора, в то время как в русском эта зона обслуживается отдельной лексемой *свободный*. Однако, как мы уже видели, поля ПОЛНЫЙ и ПУСТОЙ несимметричны: для них оказываются значимы разные параметры (ср. противопоставление по типу контейнера), разное толкование получает фрейм «формы». Несимметричность особенно ярко проявляется в зоне переносных значений: их мы рассмотрим по-отдельности.

对不起，	这个	座位	**有人**	坐	吗？	(13)
duì bù qǐ	zhè gè	zuòwèi	**yǒu rén**	zuò	ma	
извините	этот CL	место	иметь человек	сидеть	PCL	

‘Извините, здесь кто-то сидит?’ (baidu)

ПЕРЕНОСНЫЕ ЗНАЧЕНИЯ

пустой / 空 kōng

Как уже отмечалось, ‘пустой’ имеет два семантических актанта: контейнер и содержимое, причем первый выражается обязательно, а второй, в зависимости от языка, факультативно или же не выражается вовсе, но всегда подразумевается. При метафорическом употреблении этого признака нечто, прототипически не являющееся контейнером, уподобляется контейнеру и получает возможность описываться с помощью прилагательных со значением ‘пустой’ (ср. *пустая голова, пустые слова*). При этом отсутствующее содержимое, как правило, не называется эксплицитно.

В большинстве исследованных языков такие метафоры имеют сугубо отрицательную оценку [7]. Механизм ее развития следующий: отсутствие чего-либо в месте, для этого специально предназначенном (контейнере), связано с идеей недостатка и означает, что контейнер не выполняет свое основное предназначение – это делает его нефункциональным, «плохим».

В разных языках в качестве контейнера могут осмысливаться слова разных семантических классов. Наиболее частым источником для таких переносов являются:

1) части тела
2) речевые акты
3) временные промежутки
4) действия
5) эмоции.

Рассмотрим их последовательно.

1. Части тела: глаза, грудь, сердце, голова

Как контейнеры могут осмысливаться только такие части тела, которые в наивном представлении носителей должны обладать некоторым «содержимым»: так голова – это «контейнер» для ума и мыслей, сердце – «контейнер» для чувств и пр.: *пустая голова*, *пустые глаза*. Метафора отсутствия предполагающегося содержимого очень широко представлена в китайском языке. Так, 空 *kōng* может характеризовать голову, в которой нет необходимых умных мыслей (14) и сердце в ситуации «пусто на сердце», когда в нем отсутствуют какие бы то ни было эмоции (15).

В китайском языке в рассматриваемом контексте представлен еще один интересный путь метафоризации, с другим словом ‘пустой’, 虚 *xū*, которое в современном языке в прямом значении не употребляется[6]. Это значение ‘скромный’, т. е. пустой в хорошем смысле, способный вместить в себя мнения других людей (16).

脑袋	**空空**的，	什么	都	想不起来。	(14)
nǎodài	**kōng kōng** de	shénme	dōu	xiǎng bù qǐlái	
голова	пустой RDP ATR	что	все	думать NEG подниматься	

‘Голова совсем пустая, ничего не могу вспомнить’. (инф.)

最近	心里	感觉	特	**空**。	(15)
zuìjìn	xīn lǐ	gǎnjué	tè	**kōng**	
последнее время	сердце LOC	ощущение	особенно	пустой	

‘Последнее время на сердце очень пусто’. (инф.)

君子	以	虚	受	人	(16)
jūnzǐ	yǐ	xū	shòu	rén	
благородный муж	использовать	пустой	принимать	человек	

‘Благородный муж привлекает к себе людей своею скромностью’.

[6]В древнекитайском такое употребление было возможно, ср. пример из «Исторических записок» Сыма Цяня (II-I вв. до н.э.):

良	賈	深	藏	若	**虛**，	君子	盛	德
liáng	gǔ	shēn	cáng	ruò	**xū**	jūnzǐ	shèng	dé
хороший	торговец	глубокий	прятать	будто	пустой	благородный муж	сильный	добродетель

容	貌	若	愚。
róng	mào	ruò	yú
облик	вид	будто	глупый

‘Хороший торговец прячет товар так, что кажется, что все пусто, благородный муж великих добродетелей на вид кажется глупым’.

Это значение слова 虚 *xū* сохранилось в составе сложных слов, например, 谦虚 *qiānxū* ‘скромный’ (букв. ‘скромный-пустой’) и 虚心 *xūxīn* ‘скромный’ (букв. ‘пустой-сердце’), и устойчивых выражений, например, 虚怀若谷 *xū huài ruò gǔ* ‘исключительно скромный’ (букв. ‘пустой-грудь-будто-ущелье’). Употребления же 虚 *xū* в качестве самостоятельного слова имеют отрицательные коннотации, при чем не все из них близки к 空 *kōng*. Мы не будем здесь на них останавливаться, поскольку в современном языке 虚 *xū* не употребляется в прямом значении.

В русском языке слово *пустой* может характеризовать и человека в целом, также передавая значение отсутствия смыслообразующего признака: ср. *пустой человек* ‘глупый, беспутный’ (главная функция человека – ум). В китайском языке такая метафора не представлена.

2. Речевые акты: слова, обещания, фразы

Основное содержание слов и других речевых актов – их смысл и стоящее за ними намерение: сочетания типа *пустые слова, пустые обещания* приобретают значение бессмысленных, бесполезных или не соответствующих реальности слов.

В китайском языке 空 *kōng* тоже может описывать продукты речевой деятельности, не нагруженные смыслом, ср. 空谈 *kōngtán* ‘пустые разговоры’, 空话 *kōnghuà* ‘пустые слова, болтовня’, 空论 *kōnglùn* ‘пустые рассуждения, доктринёрство’:

话	说	得	很	**空**	(17)
huà	shuō	de	hěn	**kōng**	
речь	говорить	EV	очень	пустой	

‘говорить очень пусто (без реального смысла)’ (инф.)

В этом значении 空 *kōng* может характеризовать и письменные продукты умственной деятельности – статьи, диссертации и т.п. (18).

В значении ‘несоответствующий реальности’ 空 *kōng* имеет более широкую сочетаемость, чем русское слово *пустой*. Так, возможно сочетание 空名 *kōngmíng* (букв. ‘пустое имя’) ‘одно название’, которое используется, когда реальное положение дел не соответствует заявленному или ожидаемому. С этим значением соотносится и устойчивое сочетание 空头支票 *kōngtóu zhīpiào* ‘необеспеченный вексель’, которое может обозначать пустые, ничем не подкрепленные обещания.

Интересно, что антонимичное значение ‘соответствующий реальности’ описывается в китайском языке лексемой 实 *shí* ‘цельный’.

这篇	论文	很	**空，**	没	有	内容。	(18)
zhè piān	lùnwén	hěn	**kōng**	méi	yǒu	nèiróng	
этот CL	статья	очень	пустой	NEG	иметь	содержание	

‘*Это пустая статья, бессодержательная’. (инф.)*

3. Временные промежутки (метафора пространство <-> время): день, год, урок…

Переносы с пространства на время являются одной из базовых метафор [10: С. 218]. В случае признака ‘пустой’ происходит перенос с ограниченного участка пространства (контейнера) на ограниченный промежуток времени (отрезки типа *час, год, месяц, урок* и пр.). Отсутствующим содержимым в таких метафорах становятся события или занятия, которые могут (или должны, как в случае слова *урок* – специализированного названия временного промежутка по его наполнению) заполнять собой эти промежутки. Подобные употребления представлены в русском языке:

*— У каждого человека бывает **пустой день**, который проходит просто так, уходит незаметно из памяти и из жизни, — говорил Вегов.* (НКРЯ) (19)

*Мой маленький друг! Сегодня **пустой день**. И если бы не Вы, и не то, что можно Вам послать цветы, а потом думать об этом целый день, то было бы очень пусто и скучно.* (НКРЯ) (20)

Китайская лексема 空 *kōng* не имеет подобного употребления.

Ситуация отсутствия содержимого в контейнере может получать и некоторые дополнительные компоненты значения.

4. Действия: ‘напрасный, не ведущий к результату’

С идеей отсутствия в контейнере ожидаемого содержимого связана идея отсутствия ожидаемого результата от некоторых действий, значение ‘напрасный, не ведущий к результату’. Ср. русск. *впустую пробегать.*

Это значение развивает и китайская лексема 空 *kōng*: 空忙 *kōngmáng* ‘пустые заботы’ и 空想 *kōngxiǎng* ‘пустые размышления’ не ведут к решению никаких проблем.

那么	弄	点	小菜	使	朋友们	乐乐，
name	nòng	diǎn	xiǎocài	shǐ	péngyǒumen	lè lè
тогда	делать	немного	закуска	CAUS	друг.PL	радоваться RDP

也	省得	人家	**空**	跑	一趟。	(21)
yě	shěngde	rénjiā	**kōng**	pǎo	yī tang	
тоже	во избежание	люди	пустой	бегать	один CL	

‘Приготовил немного закусок, чтобы порадовать друзей, чтобы не вышло, что все напрасно пришли’. (ccl)

Значение ‘бесполезный’ реализуется и в составе фразеологических сочетаний, таких, например, как 竹篮子打水一场空 *zhú lánzi dǎ shuǐ - yī chǎng kōng* (букв. ‘в бамбуковой корзинке носить воду – одно поле пустое’) – ‘заниматься бесполезным делом’.

5. Эмоции: ‘напрасный, необоснованный’

Этот, более редкий, вид переноса возможен для слов-названий эмоций (гнев, злоба, ревность): в таком случае речь идет об отсутствии основания, причины:

Кто же докажет, когда бьют за дело, а когда по ***пустой злобе****?* (НКРЯ) (22)

Что означает ***пустой гнев*** *неумной женщины? Только то, что человеку не повезло в жизни.* (НКРЯ) (23)

Китайская лексема 空 *kōng* с названиями эмоций не сочетается.

Таким образом, основные лексемы семантического поля ПУСТОЙ и в русском, и в китайском языках могут сочетаться с названиями некоторых частей тела и речевых актов, передавая значение ‘лишенный основного (метафорического) содержимого’, а также описывать некоторые действия как бесполезные и напрасные. В значении ‘лишенный основного (метафорического) содержимого’ русская лексема *пустой* обладает более широкой сочетаемостью, описывая как человека в целом, так и временные отрезки. Кроме того, она может описывать некоторые эмоции как необоснованные.

Китайская лексема 空 *kōng* имеет более широкую сочетаемость в значении ‘не соответствующий реальности’. Отдельный интерес представляет значение ‘скромный’, зафиксированное в исторических памятниках и сохранившееся в устойчивых сочетаниях в современном языке, для лексемы 虚 *xū* ‘пустой’, исторически тоже относившейся к интересующему нас семантическому полю.

полный / 满 *mǎn*

При метафорическом употреблении, лексемы, в прямом значении не относящиеся к топологическому классу «контейнер», осмысливаются как имеющие свойства контейнеров «содержать в себе что-либо» и описываются при помощи слова *полный*. Таким образом чаще метафоризуются конструкции с выраженным «содержимым», которое в таком случае также является не материальным, а метафорическим, абстрактным. Наиболее частотными источниками метафоры являются:

1) временные промежутки

Итак, позади большой, полный интересных впечатлений день. (НКРЯ) (24)

2) слова и другие речевые акты

Как мог в немощном старческом теле храниться такой молодой, полный силы голос! (НКРЯ) (25)

3) результаты умственной и творческой деятельности человека: *фильм, полный режиссерских находок*

Китайская лексема 满 *mǎn* подобных употреблений не имеет[7].

Еще один вид метафорических употреблений признака ‘полный’ – при котором невозможно выделить двух участников ситуации – контейнер и содержимое, но появляется один участник – «контейнер», «содержимым» которого является как будто он сам. Таким образом метафоризуются только конструкции без выраженного второго участника «содержимое». В первую очередь такие переносы происходят на две зоны, смежные с полем ПОЛНОГО:

1. ‘X включает все необходимые части’
2. ‘X взят в полном объеме/достиг наивысшей точки развития/максимальный’

‘X включает все необходимые части’. Прототипическим контекстом для этого значения является прилагательное + X, где X – предметное имя со значением множества или совокупности объектов.

Например, *полный набор*, *полное собрание сочинений*.

В китайском языке подобные сочетания с 满 *mǎn* невозможны*:* его использование предполагает невозможность добавить что-либо, так как все существующее место заполнено, ср.:

满额 *mǎn'é* (букв. ‘полный-квота’)
‘полный штат; быть укомплектованным’,

满员 *mǎnyuán* (букв. ‘полный-служащий’)
‘быть полностью укомплектованным (например, личным составом)’.

满 *mǎn* предполагает наличие заранее заданного ограниченного места, превысить которое нельзя (26), или процесс постепенного наполнения, накопления (27)[8].

假期	已	满	(26)
jiàqī	yǐ	**mǎn**	
каникулы	уже	полный	

‘Каникулы уже закончились’. (инф.)

他	满	十五	岁	了	(27)
tā	**mǎn**	shíwǔ	suì	le	
он	полный	15	год	MOD	

‘Ему исполнилось пятнадцать лет’. (инф.)

Это подтверждается и диахроническими данными. Согласно «Словарю древнекитайского языка Ван Ли» [11], первое значение 满 *mǎn* ‘наполнить(ся)’: неслучайно смысловым детерминативом (ключом) в этом иероглифе является вода.

Для передачи значения ‘все существующие части (участники) включены’ используется лексема 全 *quán*:

全体 *quántǐ* (букв. ‘весь-тело’)
‘весь коллектив; в полном составе’,

全家 *quánjiā* (букв. ‘весь-семья’)
‘вся семья’,

全民 *quánmín* (букв. ‘весь-народ’)
‘весь народ; всенародный’,

全面 *quánmiàn* (букв. ‘весь-сторона’)
‘всеобщий; всесторонний’.

[7]Соответствующее значение может передаваться при помощи глагола ‘исполниться, преисполниться’ 充满 *chōngmǎn*.

[8]В этих предложениях 满 *mǎn* используется в глагольном значении, ср. также русск. *исполниться* (15 лет).

Значение полного комплекта (*собрание сочинений*) также может выражаться только при помощи лексемы 全 *quán*:

全副 *quán fù* (букв. ‘весь-комплект’)
‘полный комплект’,
全集 *quán jí* (букв. ‘весь-собрание’)
‘полное собрание (сочинений)’,
全套 *quán tào* (букв. ‘весь-комплект’)
‘полный набор (например, инструментов)’.

‘X взят в полном объеме/достиг наивысшей точки развития/максимальный’. Ср. *полная скорость, полный прилив, полный рабочий день*.

Прототипическим контекстом для этого значения мы будем считать *прилагательное + X,* где *X* – параметрическое имя или промежуток времени. Нужно заметить, что помимо параметрических имен и временных промежутков, в интересующую нас зону входят и иные абстрактные имена, в семантику которых встроен компонент «возможность проявления в разной степени».

С параметрическими именами в китайском языке также употребляется только лексема 全 *quán*:

全速 *quán sù* (букв. ‘весь-скорость’)
‘полная скорость’,
全力 *quán lì* (букв. ‘весь-сила’)
‘все силы; всеми силами’.

Как представляется, когда мы говорим о полной скорости или полной силе, идея контейнера отсутствует, мы акцентируем внимание на верхней точке шкалы, смотрим сразу на всю шкалу целиком (например, мы можем говорить о *полной скорости*, но не можем говорить о **неполной (частичной) скорости*).

Однако для явлений, которые обязательно проходят все стадии развития, т. е. в которых присутствует идея постепенного накопления (заполнения), употребляется исключительно лексема 满 *mǎn*:

满朝 *mǎn cháo* (букв. ‘полный -прилив’)
‘полный прилив’,
满月 *mǎn yuè*
‘полная луна’.

Здесь прослеживается связь с исходным значением 满 *mǎn*, проявляющаяся в идее вместилища, контейнера (ср. также [12-13]).

Интенсификатор. Прилагательные, одновременно покрывающие зону ‘X взят в полном объеме/достиг наивысшей точки развития/максимальный’ и ‘полный (X содержащий в себе что-либо до возможных пределов)’ часто расширяют сочетаемость на неградуируемые имена, семантика которых изначально такой возможности не предполагает. В таких случаях можно говорить о вымывании значения прилагательного и превращении его в лексическую функцию Magn (о лексических функциях см. [14], о прилагательных со значением Magn и ребрендинге степени см. [15]).

Так, например, в русском языке слово *полный* сочетается с такими изначально неградуируемыми понятиями, как *провал, победа, хаос, кошмар* и другими. В китайском языке, поскольку для основной лексемы 满 *mǎn* важна идея постепенного развития (наполнения) и ограниченного свободного места, в значении Magn ожидаемо используется лексема 全 *quán*, которая делает акцент на том, что все существующее содержимое включено, а идея контейнера и ограниченного объема отсутствует вовсе:

全胜 *quán shèng* (букв. ‘весь-победа’)
‘полная победа’,
全责 *quán zé* (букв. ‘весь-ответственность’)
‘полная ответственность’.

Таким образом, переносные значения основных функциональных лексем семантического поля ПОЛНЫЙ в русском и китайском языках значительно различаются. Русское слово *полный*, в отличие от китайской лексемы 满 *mǎn,* может употребляться в конструкциях с двумя выраженными актантами и передавать значения наполненности некоторым абстрактным содержимым временных промежутков, слов, произведений искусства, а также достаточно свободно употребляться в значениях с одним выраженным актантом ‘X включает все необходимые части’, ‘X взят в полном объеме/достиг наивысшей точки развития/максимальный’ и интенсификатора. Сфера употребления китайской лексемы 满 *mǎn* уже – она связана с идеей заполнения какого-либо контейнера, и потому может употребляться только в тех случаях, когда речь идет о постепенном заполнении некоторого объема, градуированном развитии признака. В значениях с одним выраженным актантом, возможных для *полный*, в китайском языке используется 全 *quán*, которая в части своих значений соответствует слову *полный*, а также передает значения, сходные с русским *весь*.[9]

цельный / 实 shí

В русском языке значение ‘обладающий внутренним единством, однородный’ слова *цельный* может относиться и к некоторым жидкостям, в значении ‘натуральный, неразбавленный, необезжиренный’, ср. *цельное молоко*, *вино*.

В переносных употреблениях это значение может распространяться не только на физические предметы, но характеризовать и произведения, и впечатление от них как лишенные внутренних противоречий, раздвоенности (ср. определение МАС [2]), например, *цельное произведение*, *впечатление*, *цельный характер*, *цельная натура*. Интересно, что эти переносные значения связаны не со значением формы, а со значением внутренней однородности.

В китайском языке лексема 实 *shí* ‘цельный’ таким образом не употребляется и развивает другие переносные значения: ‘соответствующий действительности, настоящий, подлинный’. В этом значении она антонимична лексеме 空 kōng ‘пустой’:

[9] В китайском языке существует также лексема 整 *zhěng* ‘целый’, которая в некоторых контекстах передает значения, близкие к рассмотренным значениям 全 *quán* ‘весь’ (ср. русск. *весь день – целый день*). При этом 整 *zhěng* делает акцент на целостности, неразложимости на отдельные компоненты, поэтому преимущественно используется, когда речь идет о полном включении в рассмотрение некоторого объекта, обладающего внутренним единством. Ср. [16]

少	发	**空**	论，	多	做	**实**	事。	(28)
shǎo	fā	**kōng**	lùn	duō	zuò	**shí**	shì	
мало	испускать	пустой	рассуждение	много	делать	полный	дело	

‘Поменьше пустых рассуждений, побольше реальных дел’. (инф.)

В силу ограничений на самостоятельное употребление односложных слов, в современном китайском языке это значение 实 *shí* часто реализуется в составе сложных слов. Например:

现实 *xiànshí* (букв. ‘сейчас-цельный’) ‘реальный’,
实在 *shízài* (букв. ‘цельный-существовать’) ‘действительно’,
实际 *shíjì* (букв. ‘цельный-граница’) ‘фактический’,
其实 *qíshí* (букв. ‘его-цельный’) ‘в реальности’,
事实 *shìshí* (букв. ‘дело-цельный’) ‘факт’,
调查属实 *diàochá shǔ shí* (букв. ‘проверять-относиться-цельный’) ‘проверка подтвердила факт’.

Как уже отмечалось, лексема 实心 *shíxīn*, специализирующаяся на описании формы, не развивает переносных употреблений.

ВЫВОДЫ

Таким образом, если в зоне прямых значений для полей ПОЛНЫЙ и ПУСТОЙ в китайском и в русском языках значимы одни и те же параметры, то переносные значения соответствующих прилагательных различаются достаточно сильно. Так, многие из переносных значений, характерных для русского *полный*, обслуживаются в китайском языке отдельной лексемой 全 *quán*, русскому слову *пустой* в переносных значениях тоже соответствуют две лексемы – 空 *kōng* и 虚 *xū*. Различие в метафорических употреблениях прослеживается и на синтаксическом уровне: так, употребление русского *полный* возможно в конструкциях с двумя выраженными актантами (контейнер и содержимое), а китайского 满 *mǎn* – нет. При этом данные китайского языка не нарушают типологических прогнозов – в них просто реализуется стратегия метафоризации, отличающаяся от русской.

Понимание такого рода различий в употреблении лексем важно как при изучении иностранного языка, так и при практической работе с ним. Мы надеемся, что результатом лексико-типологических исследований в области качественных признаков станет создание мультиязыковой базы данных, включающей в себя исчерпывающий список контекстов, в которых возможно проявление того или иного признака, с указанием ограничений на употребление соответствующих лексем.

* * *

Автор выражает глубокую признательность М.Г. Тагабилевой за помощь в подготовке статьи и информантам-студентам Пекинского Университета Вэй Хан, Ван Вэньин и Сун Фэйфэй за предоставленные примеры и плодотворные обсуждения.

СПИСОК ЛИТЕРАТУРЫ

1. Рахилина Е. В., Плунгян В. А. О лексико-семантической типологии // Глаголы движения в воде: лексическая типология / ред. Т. А. Майсак, Е. В. Рахилина – М. 2007. – С. 9-26.
2. Koptjevskaja-Tamm M. Approaching lexical typology // From Polysemy to Semantic Change / ed. Vanhove M. – Amsterdam: Benjamins, 2008.
3. Evans C.N. Semantic typology // The Oxford handbook of linguistic typology / ed. J.J. Song. – Oxford: New York, 2011.
4. Koptjevskaja-Tamm M. 2012 New directions in lexical typology // Linguistics. – 2012. – № 50(3). – P. 373-394.
5. Толстая С.М. Языковой образ пустого // Шездесет година Института за српски језик САНУ. Зборник радова I. – Београд, 2007. – С. 471–480.
6. Тагабилева М.Г., Холкина Л.С. Качественные признаки ‘пустой’ и ‘полный’ в типологическом освещении // Материалы VII конф. по типологии и грамматике для молодых исследователей. – СПб.: Наука, 2010. Tagabileva M., Kholkina L., Kiryanov D. Semantic domains ‘full’ and ‘empty’: a cross-linguistic study // Association for Linguistic Typology. The 10th Biennal Conference, 2013 August 15-18. – Leipzig: Abstracts, 2013. – URL: http://www.eva.mpg.de/lingua/ conference/ 2013_ALT10/pdf/abstracts/Abstracts_ALT10_complete.pdf.
7. Словарь русского языка в 4-х т. / под ред. А. П. Евгеньевой – М., 1999. – URL: http://feb-web.ru/feb/mas/.
8. XHCD – «Xiandai hanyu cidian» – Словарь современного китайского языка. – Пекин: Коммерческая пресса, 2012.
9. Lakoff G. The contemporary theory of metaphor // Metaphor and Thought: Second edition / ed. A. Ortony. – Cambridge: Cambridge University Press, 1993. – P. 202-251
10. Wáng Lì. Gǔ Hànyǔ zìdiǎn – Словарь иероглифов древнекитайского языка Ван Ли. – Пекин: Zhōnghuá shūjú, 2000

11. Chǔ Zéxiáng. “mǎn+N” yǔ “quán+N” péng. (Пара “mǎn+N и “quán+N”) // Zhōngguó yǔwén. – 1996. – №05. – P. 339-344.
12. Lǐ Wénhào. “mǎn+NP” yǔ “quán+NP” de tūxiǎn chāyì jí qí yǐnyù móshì. (Разница в значениях “mǎn+NP” и “quán+NP” и модели их метафоризации). // Yǔyán kēxué. – 2009. – № 04. – P. 396-404.
13. Мельчук И.А., Жолковский А.К. Толково-комбинаторный словарь современного русского языка. – Wien: Wiener Slawistischer Almanach, 1984.
14. Карпова О.С., Резникова Т.И., Архангельский Т.А., Кюсева М.В., Рахилина Е.В., Рыжова Д.А., Тагабилева М.Г. База данных по многозначным качественным прилагательным и наречиям русского языка // Компьютерная лингвистика и интеллектуальные технологии (по материалам ежегодной Международной конференции «Диалог», Бекасово, 26-30 мая 2010 г.). Вып. 9 (16). – М., 2010. – С. 163-168.
15. Dīng Xuěhuān. Cóng rènzhī shìjiǎo kàn “zhěng(gè)+N” hé “quán+N”. (Анализ конструкций “zhěng(gè)+N” hé “quán+N” с когнитивной точки зрения) // Zhōngnán mínzú dàxué xuébào. – 2007. – № 03. – P. 166-169.

Материал поступил в редакцию 21.02.14.

Сведения об авторе

ХОЛКИНА Лилия Сергеевна – преподаватель Российской государственного государственного университета; научный сотрудник Школы гуманитарных исследований Российской академии народного хозяйства и Государственной службы при президенте РФ, Москва
e-mail: kholkina@gmail.com

СПРАВОЧНО-ИНФОРМАЦИОННЫЙ РАЗДЕЛ

УДК [004.89 : 001.4] (049.32)

С.Д. Шелов

Разработка компьютерных терминологических систем – новый этап обеспечения качества*

В 2013 г. издательство «Unigrafia» (Хельсинки, Финляндия) выпустило пособие на английском языке «Quality Assurance in Terminology Management: Recommendations from the TermFactory project» (Обеспечение качества в терминологическом менеджменте: Рекомендации на основе опыта проекта «TermFactory»), подготовленное известным финским и российским специалистом в области терминологии и терминографии И.С. Кудашевым, доцентом Хельсинского университета, автором монографии «Проектирование переводческих словарей специальной лексики» (Helsinki, 2007), а также нескольких терминологических словарей, включая «Финско-русский лесной словарь», получивший в 2008 г. от Европейской терминологической ассоциации Международную премию за прикладные исследования и проекты в области терминологии.

Пособие состоит из пяти разделов, заключения, библиографии, приложений и списка публикаций по проекту «TermFactory». Ввиду прикладной направленности этого издания приведем названия всех его разделов:

Введение; 1. Обзор основных особенностей компьютерной оболочки TermFactory; 2. Инфраструктура обеспечения качества компьютерных терминологических систем; 3. Методологические аспекты обеспечения качества совместных терминологических работ; 4. Структурные аспекты обеспечения качества в терминологическом менеджменте; 5. Описательные метаданные и их роль в обеспечении качества компьютерных терминологических систем; 6. Заключение; Библиография; Публикации по проекту TermFactory.

Пособие имеет ярко выраженную практическую направленность, в связи с чем почти каждый его раздел снабжен одним или двумя приложениями; суммарно эти пять приложений занимают более трети объема этого издания, а их список таков: Приложение 1. Классификация терминологических данных; Приложение 2. Классификация административных данных; Приложение 3. Классификация тематических областей (Domain classificaion); Приложение 4. Документирование источников; Приложение 5. Документация экспертизы разработчиков.

Как отмечается во введении, пособие не ставит своей целью представить в полном виде теоретические основы терминологической работы и терминологического менеджмента. В нем, скорее, представлены результаты исследования, построенного «снизу-вверх», т.е. идущего от целей и задач конкретных проектов до теоретических обобщений. В то же время сделанные рекомендации сопровождаются их научным обоснованием, которое, как надеется автор, имеет и теоретический, и практический интерес. Не все из сделанных в пособии рекомендаций реализованы в проекте «TermFactory», – с одной стороны, и имеются рекомендации, подсказанные опытом развития других проектов, с другой (их список также представлен в одном из приложений). Все рекомендации отражают точку зрения автора и продиктованы либо результатами завершившихся проектов, либо настоящим этапом их развития.

Небольшим по объему, но важным по своему содержанию является первый раздел пособия, содержащий описание архитектуры и основных принципов проекта «TermFactory». К этим принципам относятся: а) онтологическая структура компьютерной системы, б) ориентация на создание контента компьютерной системы совместно с другими организациями и странами, в) распределенная «облачная» структура системы (cloud-ready system), означающая, что данные не хранятся в одном месте, но распределяются между многочисленными хранилищами различных типов.

Второй раздел «Инфраструктура обеспечения качества терминологических систем» открывается определением основных понятий, которые используются в пособии и к которым относятся: «терминологический менеджмент», «терминологическое описание», «терминологический банк (данных)», «метаданные», «качество», «обеспечение качества». Дефиниции терминов «качество», «обеспечение качества» заимствуются из международных

* Kudashev I. Quality Assurance in Terminology Management: Recommendations from the TermFactory project (Обеспечение качества в терминологическом менеджменте: Рекомендации на основе опыта проекта «TermFactory»). – Helsinki, 2013. – 246 p.

стандартов ISO (Международной организации по стандартизации), дефиниции других терминов – составлены автором. Здесь элементы компьютерных терминологических систем (качество которых и надлежит улучшать) группируются следующим образом: методологические данные, структурные метаданные, описательные метаданные. Эта группировка и обусловливает композицию всего пособия (см. названия разделов пособия выше).

Третий раздел пособия важен для специалистов в области информатики, профессионально занимающихся разработкой терминологических систем, и лингвистов. Автор останавливается на технологии и порядке разработок в этой области, сравнивая традиционные методы создания терминологической продукции и совместных разработок информационных коллективов, разделенных в пространстве и времени, оценивает преимущества и сложности создания соответствующего контента тем и другим способами.

Для терминолога-лингвиста весьма интересен будет подраздел этого раздела, озаглавленный «Объекты терминологического описания в терминологическом банке данных». Автор пособия полностью согласен с мнением А.С. Герда о том, что для специалиста-терминографа вопрос, как и где следует представлять в словаре обозначение специального понятия данной области, важнее, чем вопрос о том, можно ли считать это обозначение термином. Поэтому, в связи с трудностями однозначного и общеупотребительного использования термина «термин», автор в ряде случае предпочитает обращаться к «зонтичному» понятию и к термину «обозначение ЯСЦ», т.е. обозначение языка для специальных целей (LSP designation), и формулирует требования, предъявляемые к такому обозначению, при выполнении которых оно может претендовать на включение в терминологический банк данных. Помимо собственно терминов и, в частности, так называемых амбисемичных терминов, автор рассматривает языковые единицы, часто обсуждаемые в российской лингвистической и терминологической науке – прототермины, предтермины, терминоиды, квазитермины (псевдотермины) и др. Автор не выносит здесь окончательного решения, надеясь на дальнейшие исследования вопроса о природе и составе таких единиц, как правило, обладающих размытыми концептами (vague concepts).

Более определенно можно высказаться относительно специальных номинаций, апеллятивов, т.е имен собственных типа *галактика Андромеды* или *Большой адронный коллайдер,* а также номенклатурных обозначений, т.е обозначений типа *револьвер Smith-and-Wesson 13-й модели* или *револьвер Smith-and-Wesson 27-й модели.* Что касается первых, то, хотя они обычно не включаются в терминологические банки данных, нет ни теоретических ни практических препятствий для того, чтобы это делать, – тем более, что, как показывают опросы, потребителям часто не достает именно данных об объектах, обозначенных именами собственными. Что касается вторых, то и их включение приветствовали бы многие компании-потребители и частные лица, но при этом приходится иметь в виду возможные негативные последствия:

а) номенклатура превосходит по количеству терминологию в тысячи раз и терминологический банк может быть очень быстро наводнен номенклатурными обозначениями, что вряд ли рационально;

б) номенклатура и ее описание устаревают значительно быстрее, чем терминология и, следовательно, требуется более частый ее пересмотр и обновление, чем терминологии;

в) номенклатуру вряд ли следует включать в терминологические банки, если о ее единицах нет релевантной информации (здесь необходимо иметь в виду, что номенклатурные обозначения фактически не поддаются дефиниции, а только достаточно пространным описаниям);

г) установление происхождения и другой информации, идентифицирующей содержание номенклатурного имени, должно быть проведено ясно и недвусмысленно, что бывает весьма затруднительно.

Кроме того, несмотря на многолетнюю дискуссию о составе номенклатуры, как полагает автор, вопрос все еще не может считаться окончательно решенным. Поэтому, хотя включение номенов в терминологические банки данных теоретически возможно и в ряде случаев желательно, разработчик компьютерных систем должен быть в курсе связанных с реализацией подобных пожеланий проблем и, во всяком случае, осознавать, что номенклатурные наименования не могут быть так же точно идентифицированы дефинициями, как термины.

В четвертом разделе пособия «Структурные аспекты обеспечения качества в терминологическом менеджменте» обсуждаются такие вопросы терминологических систем, как указание на язык системы, принятые в них способы кодировки, сравнения и сортировки буквенно-цифровых последовательностей, используемые классификации научных и тематических областей и дисциплин и классификации категории данных.

Остановимся здесь подробнее на классификации научных и тематических областей (domain classification), которой автор придает большое значение. С помощью параметра научной области фиксируется сфера употребления единиц языков для специальных целей, в частности, терминов; часто он важен и для разрешения омонимии этих единиц. Этот параметр также очень существенен для автоматического объединения знаковых единиц, относящихся к той или иной научной дисциплине. Наконец, тот же параметр играет важную роль в управлении правами и ролью потребителей компьютерных систем, ибо они могут зависеть от того, с какой тематической областью работает этот потребитель. В связи с этим в пособии представлен обзор тех классификаций тематических областей, которые используются в ряде компьютерных систем Финляндии – Общий финский тезаурус (http://fi.Wikipedia.org/wiki/YSA) и его обновленная и расширенная версия (http://onki.fi/en/browser/overview/yso), а также некоторые класси-

фикации финских библиотек. В области экономической деятельности в Финляндии функционируют Экономическая классификация (http://www.stat.fi/meta/luokitukset/ index_taluos_en.html) и Отраслевая классификация наук и технологий (http://www.stat.fi/meta/luokitukset/tieteenala/001-2007/kuvaus_en.html). Среди различных международных классификаций наибольшей популярностью пользуются классификации на английском языке, в частности, английская версия УДК (http://www.udcc.org). Однако в терминологических банках данных могут использоваться совсем другие классификации – либо разработанные для собственного, внутреннего пользования (например, собственная классификация, используемая в старейшем и самом большом в мире терминологическом банке данных Termium (Канада)), либо внешние, существующие независимо от терминологических задач, классификации (например, многоязычный и многодисциплинарный тезаурус Eurovoc, функционирующий в межинституциональной терминологической базе данных Европейского союза); в некоторых случаях, для не очень больших терминологических банков данных (подобных финскому TEPA или шведскому Rikstermbanken) подобные классификации не используются вовсе.

В то же время анализ «внешних» классификаций, проведенный в рецензируемом пособии, выявляет их неприспособленность к задачам, стоящим перед терминологическими банками данных, а наличие отдельных внутренних «собственных» классификаций делает их несопоставимыми друг с другом и затрудняет обмен компьютерной информацией. К классификации областей знания, используемой в терминологических банках данных, автор формулирует следующие общие требования:

1) классификация должна быть свободной и доступной онлайн. На практике это означает, что она должна быть частью собственного электронного ресурса, а не ресурса третьей стороны;

2) классификация должна быть многоязычной;

3) категории классификации должны быть широко распространенными;

4) классификация не должна быть культурноспецифичной;

5) классификация должна быть дружелюбной к пользователю, иметь простую нотацию и правила использования этой нотации;

6) классификация должна быть легко расширяемой, т.е. пользователи должны быть в состоянии сами добавлять в нее новые классы и подклассы;

7) классификация должна иметь управление своими более ранними и более поздними версиями с тем, чтобы более старые версии были совместимыми с более новыми.

В связи с этими специфичными требованиями автор пособия останавливается на сложностях составления подобных классификаций, к которым он относит: а) множественность оснований для классификации, поскольку различные области знания допускают разные критерии классификации, например, астрономия может быть подразделена в зависимости от изучаемых объектов (ср. астрономия солнца, астрономия звезд и астрономия галактик), а может подразделяться в зависимости от диапазона электромагнитного спектра, в пределах которого проводятся наблюдения (ср. радиоастрономия, астрономия инфракрасного излучения, оптическая астрономия); б) неясность вопроса о глубине классификации, которая в настоящее время колеблется от одного до девяти уровней, причем неглубокие, мелкие классификации малоинформативны или даже неправильно ориентируют пользователя, а слишком подробные классификации сложно использовать, трудно поддерживать в актуальном состоянии и, кроме того, они часто оказываются субъективными; в) неясность вопроса о возможностях изменения классификации в соответствии с жизненным циклом дисциплин. Так, по некоторым данным, в конце двадцатого века количество научных дисциплин удваивалось каждые двадцать пять лет и пока нет ответа на вопрос, как в этих условиях обеспечить полноту классификации в момент ее разработки и как – актуальность ее состояния по мере функционирования; г) национально-языковая специфика различных классификаций в разных странах, при которой классы, выраженные как будто бы в идентичных терминах, на самом деле имеют разное содержание и различное положение в иерархии, определяемой классификацией (так, соотношения, скажем, между русским обозначением «Машиностроение», немецким «*Maschinenbau*», финским «*Metallititeollisuus*» и английским «*Mechanical engineering*» весьма сложны и могут занимать разное место в классификационной иерархии, хотя конечный продукт этих промышленных областей чаще всего один и тот же).

В пособии подробно описывается, как решаются эти задачи в проекте TermFactory, классификация тематических областей которого характеризуется следующими параметрами: число классов, глубина уровней иерархии, трактовка научных дисциплин очень широкого характера (типа «Философия», «История», «Политика» и т.п.), возможная множественность отнесения специальных языковых единиц к разным классам, трактовка комплексных дисциплин (типа «Морское дело»), трактовка культурно- и лингвоспецифических областей, учет синонимии и вариативности при обозначении классов и подклассов и т.п. Завершая рассмотрение этой темы, автор приводит в Приложении 3 классификацию тематических областей верхнего уровня (core domain classification), разработанную в рамках проекта TermFactory, и обсуждает возможности ее расширения и дополнения. Классификация представлена на четырех языках (английском, немецком, русском и финском), причем каждая языковая версия считается идентичной другой, а объем всех разноязычных версий этой классификации занимает более четверти объема всего рецензируемого пособия. Разработчики компьютерной терминологической продукции могут непосредственно воспользоваться любой из этих языковых версий или любой их совокупности.

Подраздел «Классификация категорий данных» четвертого раздела пособия любопытен обзором

имеющихся категорий данных в различных компьютерных терминологических системах, рассмотрением и некоторыми критическими замечаниями в адрес принципиального в этой области стандарта ISO 12620, классификацией, сравнением и отображением (mapping) информационных категорий этого стандарта в лингвистической классификации параметров, характеризующих термин; сжато и конструктивно содержание этого раздела представлено в Приложении 1.2.

Последний, пятый раздел пособия «Описательные метаданные и их роль в обеспечении качества компьютерных терминологических систем» состоит из двух подразделов – «Документация источников» и «Административные данные». В первом из них обсуждается адекватное для целей создания и ведения современных компьютерных терминологических систем библиографическое описание источников, его состав, формат и те нормативные документы, согласно которым такая запись должна удовлетворять требованиям, в частности, Международного стандарта ISO 12615:2004 «Библиографические ссылки и идентификаторы источников в терминологической работе». Здесь развивается подход, при котором в ходе совместной терминологической работы на международном уровне формат описания должен включать единый предусмотренный минимум реквизитов для всех источников, но не исключать дополнительных данных, если разработчики по любым причинам хотели бы его расширить; такой минимальный формат приводится в Приложении 4.1. Базовый шаблон документирования источников.

Во втором подразделе «Административные данные» обсуждается единообразное описание данных, которые не относятся к собственно единицам языков для специальных целей (терминам, номенклатурным наименованиям и т.п.), а относятся к хранилищам информации о них и к их функционированию (протоколы появления или передачи, идентификаторы и т.п.). В связи с этим автор рассматривает ряд международных стандартов и нормативных документов в этой области и предлагает собственную классификацию этих данных.

В заключении пособия обсуждается перспектива развития терминологических систем и отмечаются «старые» и новые стоящие перед ними задачи. Не сняты с повестки дня такие известные и неоднократно описанные в литературе требования, как бóльшая содержательность терминологических систем, бóльшая модификация в соответствии с требованиями заказчика, увеличение их гибкости, дружелюбности по отношению к пользователю и интерактивности, бóльшая доступность и более широкая связь с другими системами информации. К этим требованиям автор добавляет два принципиально новых: 1) развитие национальных терминологических банков, которые, обладая всеми вышеперечисленными свойствами, перестают быть продуктом для узких групп профессионалов, а становятся крупномасштабными электронными продуктами, по кругу пользователей и стоящих перед ними задач сопоставимыми с академическими словарями, национальными корпусами языков и т.п.; 2) развитие онтологий терминологических компьютерных продуктов, которые, как и собственно терминоведение, коренясь в концептуальном анализе языковых знаков, позволяют сделать их многократно используемыми с разными целями, гибкими, машиночитаемыми и пригодными для обмена в различных компьютерных системах. Автор выражает надежду, что изданное пособие является одним из первых вкладов в методологию и теорию решения этих масштабных задач (с нашей точки зрения, особенно важны в этом отношении приложения).

Оценивая в целом рецензируемое издание, необходимо отметить, что с выходом его в свет российский читатель получил чрезвычайно актуальное пособие по разработке терминологической компьютерной продукции на национальном и международном уровнях, полезное, по крайней мере, в четырех отношениях: во-первых, оно описывает для разработчика компьютерной терминологической продукции (электронные терминологические словари и энциклопедии, банки данных и т.п.) если не алгоритм, то, во всяком случае, порядок действий и технологию решения подобной задачи; во-вторых, вводит в круг возникающих при этом разнообразных информационных, лингвистических и технических проблем; в-третьих, знакомит и пользователей, и разработчиков соответствующей интеллектуальной продукции с теми требованиями международных стандартов, рекомендаций и просто практики, выполнение которых позволяет считать эту продукцию конкурентоспособной, и наконец, в-четвертых, в нем представлены как уже существующие, так и новые, предлагаемые автором, решения практических задач, связанных с выполнением этих требований, решений, которые непосредственно или с незначительными изменениями могут быть использованы в практической работе терминолога.

Сведения об авторе

ШЕЛОВ Сергей Дмитриевич – доктор филологических наук, руководитель Терминологического центра Института русского языка им. В.В. Виноградова
e-mail: Volehs@mail.ru

УВАЖАЕМЫЕ КОЛЛЕГИ!

ВИНИТИ РАН предлагает Вашему вниманию Реферативный Журнал в электронной форме

РЖ в электронной форме (ЭлРЖ) выпускается по всем разделам естественных, технических и точных наук.

Каждый номер ЭлРЖ является полным аналогом печатного номера РЖ по составу описаний документов, их оформлению и расположению. Он сопровождается оглавлением, указателями.

ЭлРЖ представляет собой информационную систему, снабженную поисковым аппаратом и позволяющую пользователю на персональном компьютере:

- читать номер РЖ, последовательно листая рефераты;
- просматривать рефераты отдельных разделов по оглавлению;
- обращаться к рефератам по указателям авторов, источников, ключевых слов;
- проводить поиск документов по словам и словосочетаниям;
- выводить текст описаний документов во внешний файл.

ЭлРЖ в версии Windows Вы можете получить за текущий год с любого номера, а также за предыдущие годы.

Подробную информацию Вы можете получить:

Адрес: 125190, Россия, Москва, ул. Усиевича, 20, ВИНИТИ РАН

Телефон: 8 (499) 155-46-20

Телефон/Факс: 8 (499) 155-45-25

E-mail: zinovyeva@viniti.ru, davydova@viniti.ru

ВНИМАНИЮ ЧИТАТЕЛЕЙ!

С 2000 года ВИНИТИ РАН вошел в состав Управляющего совета Консорциума Универсальной десятичной классификации (УДК). Институт в качестве единственного в России владельца лицензии на распространение печатных и электронных (на CD-ROM) изданий УДК на русском языке возобновил полное издание таблиц УДК.

ВИНИТИ РАН предлагает издания:

1. Таблицы УДК

УДК. Том I Общая методика применения УДК. Вспомогательные таблицы. Основные таблицы. Общий отдел. Алфавитно-предметный указатель к Общему отделу (только электронное издание)

УДК. Том II **1/3** Философия. Психология. Религия. Богословие. Общественные науки (только электронное издание)

УДК. Том III **5/54** Математика. Естественные науки (только электронное издание)

УДК. Том IV **55/59** Геологические и биологические науки

УДК. Том V **6/61** Медицинские науки (только электронное издание)

УДК. Том VI (часть 1) 6/621 Прикладные науки. Технология. Инженерное дело (только электронное издание)

УДК. Том VI (часть 2) 622/629 Техника. Инженерное дело (только электронное издание)

УДК. Алфавитно-предметный указатель к т. VI (1 и 2 части) (только электронное издание)

УДК. Том VII **63/65** Сельское хозяйство. Домоводство. Управление предприятием

УДК. Том VIII **66** Химическая технология. Химическая промышленность. Пищевая промышленность. Металлургия. Родственные отрасли

УДК. Том IX **67/69** Различные отрасли промышленности и ремесел. Строительство

УДК. Том X **7/9** Искусство. Спорт. Филология. География. История.

УДК. Изменения и дополнения. Выпуск 2 (к т.т. 1-3) (только электронное издание)

УДК. Изменения и дополнения. Выпуск 3 (к т.т. 1-6) (только электронное издание)

УДК. Изменения и дополнения. Выпуск 4 (к т.т. 1-7)

УДК. Изменения и дополнения. Выпуск 5 (к т.т. 1-10)

2. Государственный рубрикатор научной и технической информации (ГРНТИ) в 2-х томах, издание шестое, 2007.

Для подписки необходимо направить заявку для оформления счета по адресу:

125190, Россия, Москва, ул. Усиевича, 20, НМО ВИНИТИ

Телефон: 8-499-155-42-52

Факс: 8-499-943-00-60 (для НМО)

E-mail: typo@viniti.ru