МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ М. В. ЛОМОНОСОВА

На правах рукописи

Голомазов Денис Дмитриевич

Методы и средства управления научной информацией с использованием онтологий

Специальность 05.13.17 — теоретические основы информатики

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата физико-математических наук

Москва — 2012

Работа выполнена на Механико-математическом факультете и в Научно-исследовательском институте механики Московского государственного университета имени М. В. Ломоносова.

Научный руководитель:	доктор физико-математических наук, профессор *Васенин Валерий Александрович.*
Официальные оппоненты:	доктор физико-математических наук, профессор *Кузнецов Сергей Олегович*;
	кандидат физико-математических наук, доцент *Бездушный Анатолий Николаевич.*
Ведущая организация:	Институт математики имени С.Л. Соболева СО РАН.

Защита состоится 29 февраля 2012 г. в 16 час. 45 мин. на заседании диссертационного совета Д 501.002.16 при Московском государственном университете имени М. В. Ломоносова по адресу: Российская Федерация, 119991, Москва, ГСП-1, Ленинские горы, д. 1, Московский государственный университет имени М. В. Ломоносова, Механико-математический факультет, ауд. 14-08.

С диссертацией можно ознакомиться в библиотеке Механико-математического факультета (14 этаж) Московского государственного университета имени М. В. Ломоносова.

Автореферат разослан 27 января 2012 г.

Ученый секретарь
диссертационного совета Д 501.002.16
при Московском государственном университете
имени М. В. Ломоносова
доктор физико-математических наук, профессор Корнев А. А.

Общая характеристика работы

Актуальность работы. Для улучшения работы научных организаций и, как следствие, повышения эффективности развития науки в масштабах государства необходимо перманентно анализировать информацию о результатах деятельности отдельных ученых и коллективов исследователей. Основными результатами деятельности организаций, входящих в научное сообщество, как правило, считаются публикации сотрудников, результаты патентных исследований, участие в конференциях, руководство курсовыми, дипломными и диссертационными работами, чтение лекций и ряд других. При этом, как показывает практика, далеко не все результаты такой деятельности представлены в открытом доступе в Интернет. Зачастую единственным источником подобной информации могут служить лишь годовые отчеты сотрудников научных организаций, представленные с той или иной степенью подробности. Естественным образом возникает необходимость автоматизированной (с участием человека) обработки данных из подобных научных отчетов в целях количественного и качественного анализа эффективности научной деятельности отдельного коллектива, вклада каждого его участника и возможной корректировки планов, мер и мероприятий на основе такого анализа. Целями проведения анализа могут быть следующие.

- Сравнение данных о работе подразделения с данными других подразделений, в том числе – зарубежных, которые работают на рассматриваемом направлении.
- Интеграция данных о работе подразделения с данными других подразделений в целях анализа развития науки в рамках структур корпоративного масштаба и по стране в целом.
- Корректировка финансирования подразделений, поощрения отдельных их членов на основе результатов научных достижений.
- Публикация обзорных аналитических статей, посвященных научным достижениям организации.
- Получение интегрированной информации о направлении исследований в отдельной области знания, например, список основных публикаций, задач, методов, уровень активности ученых, ключевые персоны и конференции на этом направлении.

  Такая информация может представлять интерес для исследователя, которому необходимо получить первое, общее представление о еще недостаточно знакомом научном направлении.

Инструментом аналитика, целью которого является получение адекватного представления о деятельности организации или объединения нескольких организаций, могут служить результаты выполнения аналитических запросов к системе, занимающейся загрузкой, обработкой и хранением информации о научной деятельности работающих в них сотрудников. Примером такого запроса может служить «найти все публикации сотрудников интересующего учреждения за последний год, посвященные вопросам выделения данных из неструктурированных текстов и включенные в материалы международных конференций».

В качестве предмета исследования и анализа в диссертации выступает *научная информация*, которая определяется как совокупность данных, характеризующих результаты деятельности отдельных научных сотрудников. К такой информации относятся данные о научных статьях, которые включают их названия, списки авторов, места публикации и другие сведения, книгах, патентах, докладах на конференции и других видах научной деятельности.

Побудительным мотивом и конечной целью исследований, результаты которых представлены в настоящей диссертации, является создание интеллектуальной программной системы для поиска, систематизации и анализа научной информации. **Актуальность поставленной задачи** определяется острой необходимостью контролировать и анализировать информацию, характеризующую деятельность научных организаций, а также большими объемами такой информации и низкой степенью автоматизации процессов ее сбора и анализа.

Целью диссертационной работы является исследование и разработка математических моделей, алгоритмов и программных средств поиска и систематизации, хранения и анализа информации, характеризующей деятельность научных организаций, с использованием онтологий. Такая деятельность соответствует областям исследований, отмеченным в пп. 2, 5, 9 Паспорта специальности 05.13.17 – теоретические основы информатики.

Научная новизна. Автором разработан новый алгоритм построения онтологии отдельной области научного знания на основе выделения терминов из анонсов научных конференций, а также путем использования информации из поисковых систем в Интернет. Математически доказана оценка вычислительной сложности его реализации. Отличительными особенностями разработанного алгоритма являются: мягкие требования к исходным данным; автоматическое выделение терминов области знания; возможность использования алгоритма для построения онтологий других областей научного знания без его модификации; отсутствие необходимости в большом объеме ручного труда экспертов. Автором разработан также новый, удовлетворяющий предъявляемым к нему требованиям алгоритм выделения терминов-пар слов из коллекций текстов с заданным тематическим делением, эффективность которого в сравнении с классическими алгоритмами продемонстрирована на задачах классификации и кластеризации текстов. Математически доказана оценка вычислительной сложности его реализации и тот факт, что базовая функция веса термина в рубрике удовлетворяет предъявляемым к ней требованиям.

Практическая значимость. Рассматриваемый в диссертации программный комплекс учета и анализа научной информации ИСТИНА, реализующий архитектуру и разработанные автором алгоритмы построения онтологии предметной области и выделения терминов представляет собой самостоятельный инновационно перспективный продукт. Вместе с тем, его математическое, алгоритмическое и программное обеспечение может найти эффективное применение при построении других информационно-аналитических систем, в том числе – систем подготовки принятия решений в организациях научно-технического профиля и высших учебных заведениях.

На защиту выносятся:

- разработанные на основе результатов исследования предметной области математические модели и алгоритмы, архитектурные и технологические решения, опирающиеся на онтологии, для создания системы пополнения и хранения, анализа и выдачи по запросу информации, характеризующей результаты деятельности научной организации;
- формальное описание запросов к системе с использованием онтологий и языка SPARQL, создающее гарантии вычисления запросов и дополнительные возможности для эффективной верификации кода системы на всех этапах ее жизненного цикла;
- алгоритмы построения онтологии отдельной области научного знания и выделения терминов-пар слов из коллекции текстов с заданным тематическим делением, удовлетворяющие предъявляемым к ним требованиям; аналитические оценки сложности их программной реализации, полученные с использованием математических моделей;
- прототип программного комплекса для учета и анализа научной информации, именуемый Интеллектуальной Системой Тематического Исследования НАучно-технической информации (ИСТИНА), тестовые испытания которого подтвердили справедливость аналитических оценок сложности реализации основных алгоритмов, а также тот факт, что комплекс в целом удовлетворяет предъявляемым к нему требованиям.

Апробация работы. Основные результаты диссертации докладывались на всероссийской конференции с международным участием «Знания–Онтологии–Теории (ЗОНТ-2011)»,

на научно-практической конференции «Актуальные проблемы системной и программной инженерии (АПСПИ-2011)», на международных конференциях «3rd International Conference on Language and Automata Theory and Applications (LATA 2009)» и «Ломоносовские чтения» (2008–2010), на научном семинаре РАН «Виртуальные научные сообщества и технологии нечетких распределенных вычислений (Cloud Computing)» (2010), на механико-математическом факультете МГУ имени М.В. Ломоносова на семинаре «Проблемы современных информационно-вычислительных систем» под руководством д.ф.-м.н., проф. В.А. Васенина (2008, 2010, 2011), на факультете бизнес-информатики НИУ ВШЭ на семинаре под руководством д.ф.-м.н., проф. С.О. Кузнецова (2011).

Публикации. По теме диссертации опубликовано 9 научных работ, в том числе – две в зарубежных изданиях. Три статьи [1–3] опубликованы в изданиях из перечня ВАК ведущих рецензируемых журналов.

Личный вклад автора. Результаты диссертации, которые выносятся на защиту, получены лично автором. Даны соответствующие ссылки на публикации, используемые в диссертации. В совместных работах отмечен вклад автора.

Структура и объем диссертации. Работа состоит из введения, пяти глав, заключения, списка литературы. Объем диссертации — 154 страницы, Приложений – 34 страницы. Список литературы включает 83 наименования. В текст диссертации входят 39 рисунков и 33 таблицы.

Содержание работы

Во **введении** описываются цели работы, обосновывается ее актуальность и практическая значимость, перечисляются основные результаты.

Первая глава является вводной и посвящена исследованию и систематизации подходов к учету и анализу научной информации. В разделе 1.1 ставится задача разработки системы управления научной информацией, которая включает перечень требований к качеству разработанного программного средства.

Исходными для решения рассматриваемой далее задачи являются представленные в виде электронной коллекции результаты деятельности отдельных научных сотрудников. Такая информация может храниться в различном виде, например, в форме годового отчета или списка публикаций. Примером такой информации могут служить данные о научной статье, которые включают ее название, список авторов, место публикации и другие сведения. Кроме статей, в рамках настоящей работы анализу подлежат такие объекты, как книги, патенты, доклады на конференциях, тезисы докладов, научные проекты, научные отчеты, свидетельства о регистрации прав на программное обеспечение, диссертации, членство ученого в редколлегиях журналов, сборников, программных комитетах конференций и диссертационных советах, руководства диссертациями и дипломными работами, а также учебные курсы. Конечной целью исследований, результаты которых представлены в настоящей диссертации, является создание системы (программного комплекса), способной на основе анализа данных из коллекции отвечать на различные аналитические запросы. Результатами выполнения таких запросов могут быть: перечень направлений, которые активно исследуются в рамках интересующей области знания (в запросе направление исследования может задаваться набором ключевых слов); перечень задач, в которых используются методы интересующего направления; перечень направлений научных интересов отдельного ученого; список исследователей, работающих на интересующем направлении; список публикаций, похожих на заданную и другие, аналогичные им.

В соответствии со стандартом ГОСТ Р ИСО/МЭК 9126-93 к качеству системы управления научной информацией предъявляются следующие требования: функциональность; надежность; практичность; эффективность; сопровождаемость; мобильность. Каждое из этих требований детализировано в Приложении А к диссертации. В качестве детализированных требований к функциональности системы, например, рассматриваются: автоматизированный ввод данных, которые описывают результаты научной и учебной деятельности сотрудников; полуавтоматическое (с участием пользователя) разрешение неоднозначностей имен объектов при вводе данных; вычисление типовых запросов, примеры которых представлены ранее; задание поискового запроса на естественном языке без использования терминов онтологии; реализация логического вывода новых данных из существующих; возможность интеграции информации, которая содержится в системе, с другими хранилищами.

В разделе 1.2 приводится описание существующих подходов к учету и анализу научной информации, включающих следующие методы: количественный анализ результатов научной деятельности по информации из отчетов; экспертный анализ материалов конференций и журналов; анализ обзорных статей; поиск по ключевым словам; системы управления научной информацией.

В разделе 1.3 представлен краткий обзор трех основных классов существующих систем управления научной информацией, условно разделенных по назначению и способу обработки данных, а именно – крупные веб-сервисы, зарубежные исследовательские проекты и российские семантические системы, включающие платформу ИСИР[1] и комплекс, разработанный сотрудниками Института систем информатики имени А.П. Ершова Сибирского отделения РАН[2]. В обзоре отмечаются отличия решения, предлагаемого автором диссертации, от рассмотренных разработок, и обосновывается необходимость создания новой системы.

В заключение первой главы перечисляются основные недостатки известных на настоящее время систем обработки и анализа научных данных, которые могли бы рассматриваться как возможные решения основной задачи. К числу таких недостатков относятся: сложность ввода данных; сложность и малые возможности поиска информации; использование жестких и малоинформативных моделей области знания, нехватка гибкости систем; направленность на обработку информации из Интернет, а не на полуавтоматический ввод пользователем; недостаточное внимание к интеллектуализации алгоритмов загрузки, обработки и поиска информации. Отмечается, что программный комплекс, выступающий в качестве целевого в настоящей работе, с успехом может применяться во взаимодействии с другими системами, которые представлены выше, использовать отдельные их компоненты или информационные активы.

Во **второй главе** представлены разработанные автором архитектурно-технологические решения, которые используются в автоматизированной системе управления научной информацией. В разделе 2.1 описаны основы используемого онтологического подхода, включающие формальные положения онтологий, которые определяются на языке дескриптивной логики. Согласно классическому определению Т. Грубера[3], онтология - это «формальная, явная спецификация общей концептуализации». Другими словами, это способ формального представления знаний с помощью конечного множества понятий и отношений между ними. Понятие, или сущность - это

[1]Бездушный А.А., Нестеренко А.К., Сысоев Т.М., Бездушный А.Н., Серебряков В.А. Возможности технологий ИСИР в поддержке Единого Научного Информационного Пространства РАН // Электронные библиотеки. — 2004. — Т. 7, №6.

[2]Боровикова, О.И. Онтологический подход к построению систем информационной поддержки научной и производственной деятельности / О.И. Боровикова, Ю.А. Загорулько, Е.А. Сидорова // Материалы Всероссийской конференции с международным участием «Знания – Онтологии – Теории» (ЗОНТ–09). — Т. 2. — Новосибирск: Институт математики им. С.Л. Соболева СО РАН, 2009. — С. 93–102.

[3]Gruber, Thomas R. A translation approach to portable ontology specifications / Thomas R Gruber // Knowledge Acquisition. — 1993. — Vol. 5, №2. — Pp. 199–220.

класс индивидуальных объектов, или экземпляров. Связи между понятиями бывают следующих типов: иерархические (собаки являются животными); свойства (учитель обучает ученика); ограничения значений (только человек может быть родителем человека); определяющие непересекаемость понятий (кошка или собака); конкретизирующие логические отношения (в статье должен быть как минимум один автор).

В разделе 2.1 даны определения основных сущностей, которые используются в базовой дескриптивной логике $\mathcal{ALC}$[4]. Далее перечислены некоторые из них (с нумерацией по тексту диссертации).

Определение 2.1. Пусть N_C – множество имен понятий и N_R – множество имен отношений. Множеством *$\mathcal{ALC}$-понятий* называется такое наименьшее по мощности множество, что

- $\top$ (универсальное понятие), $\bot$ (пустое понятие) и все имена понятий $A \in N_C$ являются $\mathcal{ALC}$-понятиями;
- если C и D – $\mathcal{ALC}$-понятия и $r \in N_R$, то выражения $C \sqcap D$, $C \sqcup D$, $\neg C$, $\forall r.C$, $\exists r.C$ являются $\mathcal{ALC}$-понятиями.

Семантика дескриптивных логик задается с помощью *интерпретации*, определяемой для логики $\mathcal{ALC}$ следующим образом.

Определение 2.2. *Интерпретацией* называется пара $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, состоящая из непустого множества $\Delta^{\mathcal{I}}$, называемого *доменом* интерпретации и функции $\cdot^{\mathcal{I}}$, которая отображает каждое $\mathcal{ALC}$-понятие в подмножество домена $\Delta^{\mathcal{I}}$, а каждое имя отношения из N_R – в подмножество декартова произведения $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ так, что для любых $\mathcal{ALC}$-понятий C, D и произвольного имени отношения r справедливо:

$$
\begin{aligned}
&\top^{\mathcal{I}} = \Delta^{\mathcal{I}}, \quad \bot^{\mathcal{I}} = \varnothing, \\
&(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}, \quad (C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}, \quad \neg C^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}, \\
&(\exists r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}, \\
&(\forall r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}} : \text{если } (x, y) \in r^{\mathcal{I}}, \text{то } y \in C^{\mathcal{I}}\}.
\end{aligned}
$$

Определение 2.3. *Аксиомой вложенности понятий* называется утверждение вида $C \sqsubseteq D$, где C, D – произвольные $\mathcal{ALC}$-понятия.

Определение 2.4. Конечное множество аксиом вложенности понятий называется $TBox$, или *терминологической* частью онтологии.

Определение 2.7. Пусть N_x – множество имен экземпляров. Тогда *утверждениями об экземплярах* называются выражения вида $x : C$ и $(x, y) : r$, где C – произвольное $\mathcal{ALC}$-понятие, r – произвольное имя отношения, а $x, y \in N_x$.

Определение 2.8. Конечное множество утверждений об экземплярах называется $ABox$, или *фактологической* частью онтологии.

Определение 2.11. *Базой знаний (онтологией)* называется пара $(\mathcal{T}, \mathcal{A})$, где $\mathcal{T}$ является $TBox$, а $\mathcal{A} - ABox$.

Далее в разделе отмечаются преимущества использования онтологий для представления знаний, включающие общее видение области знания, возможность логического вывода, выполнение сложных структурированных запросов, сравнительная легкость объединения баз знаний, гибкость модели данных и возможность повторного использования существующих онтологий.

В разделе 2.2 представлена **математическая модель** разработанной системы. Пусть задана область научного знания D (например, «информатика»). Пусть I – множество описаний единиц (атомарных гранул) научно-технической информации в рамках этой области знания. К таким единицам относятся: научные статьи; патенты; отчеты; доклады на конференциях;

[4]Schmidt-Schauß, Manfred. Attributive concept descriptions with complements / Manfred Schmidt-Schauß, Gert Smolka // Artificial Intelligence. — 1991. — Vol. 48, №1. — Pp. 1–26.

тезисы докладов; монографии; учебные пособия и иные авторские разработки (рефераты, переводы). Каждый элемент множества I представляет собой некоторое текстовое описание соответствующего объекта. Основной целью системы является выполнение поисково-аналитических запросов, примеры которых представлены выше. Обозначим множество типовых запросов символом Q. Задача состоит в построении отображения $r_I : Q \to 2^I$, которое сопоставляет запросу $q \in Q$ подмножество описаний единиц научно-технической информации $I_q \subseteq I$. В диссертации предлагаются методы и средства решения поставленной задачи, которое включает следующие пять этапов:

- выделение терминов, которые характеризуют область научного знания D, из текстовых описаний научно-технических конференций, посвященных этой области знания;
- построение онтологии рассматриваемой области научного знания D;
- загрузка данных о результатах научной деятельности сотрудников;
- установление связей между загруженной информацией о результатах научной деятельности и экземплярами построенной онтологии области знания;
- выполнение аналитических запросов к полученной информации.

Общая архитектура разработанной системы представлена на рис. 1. Далее в разделах 2.3–2.7 каждый из пяти этапов рассматривается подробнее.

Рис. 1. Общая архитектура разрабатываемой системы управления научной информацией.

Раздел 2.3 посвящен **задаче выделения** из текстовых документов **терминов**, которые характеризуют заданную область научного знания. В разделе представлены лингвистический и статистистический подходы к ее решению. В диссертации выделение терминов используется как этап построения онтологии области знания, однако, и это следует отметить, задача выделения терминов имеет и самостоятельную ценность.

В диссертации предлагается новый алгоритм выделения терминов, получивший название *Brainsterm* [1]. Алгоритм решает задачу извлечения терминов, содержащих пары слов, из текстовых документов, разделенных на рубрики. Подробное описание алгоритма приведено в разделе 3.1.

Определение 2.13. Под *термином* в настоящей диссертации понимается пара слов, характеризующая документ, в котором она встречается, с точки зрения его принадлежности к одной или нескольким рубрикам.

Раздел 2.4 посвящен **задаче построения онтологии** области научного знания. Формально задача построения онтологии на основе коллекции текстов определяется следующим образом.

Определение 2.14. *Задача построения онтологии* $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ состоит в выборе экспертом (или группой экспертов) коллекции текстов Doc, которая адекватно характеризует интересую-

щую разработчика онтологии предметную область и в формировании на основе ее формального (автоматизированного) анализа: множества имен понятий N_C; множества имен отношений N_R; множества имен экземпляров N_x; конечного множества $\mathcal{T} = TBox$ аксиом вложенности понятий (терминологической части онтологии); конечного множества $\mathcal{A} = ABox$ утверждений об экземплярах (фактологической части онтологии).

В разделе 2.4 дано описание близкой по постановке задачи заполнения онтологии и пример, иллюстрирующий отличие этих двух задач. Перечисляются причины, по которым построение онтологии является в настоящее время очень актуальной задачей. Для учета и анализа научной информации необходимо построить онтологию $\mathcal{O}_D = (\mathcal{T}_D, \mathcal{A}_D)$ заданной области научного знания D. В разделе дается краткая характеристика метода построения онтологии *Sonmake* (Science Ontology Maker), разработанного автором. Подробное описание алгоритма *Sonmake* представлено в разделе 3.2.

Раздел 2.5 посвящен **задаче загрузки данных** в систему. Предложен метод загрузки данных, который используется в целевой системе. Процесс загрузки данных включает первичную обработку вводимых пользователем данных и выделение из них необходимой информации, например, списка публикаций сотрудника с указанием даты, места, названия публикации, информации о конференциях и проектах, в которых участвовал сотрудник, а также некоторых других сведений. Для описания научной деятельности используется онтология Semantic Web for Research Communities (SWRC)[5], включающая такие концепты, как «человек», «организация», «публикация», «конференция», «проект», а также связи между ними. Отметим, что для использования в разрабатываемой системе онтология SWRC была расширена путем добавления понятий и отношений, которые характеризуют ранее не предусмотренные ею типы результатов научной деятельности. К их числу относятся патенты, свидетельства о регистрации прав на программное обеспечение, членство в редколлегиях журналов и другие, аналогичные им. В соответствии с требованиями к системе в разработанном автором ее прототипе предполагается следующие четыре возможных способа ввода данных:

- разбор библиографических ссылок;
- разбор BibTeX-записей;
- импорт страницы автора или страницы публикации в системе eLibrary.ru;
- заполнение полей вручную.

Разбор библиографических ссылок производится с помощью программного комплекса FreeCite[6], разработанного в университете Брауна, США. Этот комплекс использует библиотеку CRF++[7], реализующую алгоритм классификации Conditional Random Fields[8]. Авторами комплекса FreeCite было проведено обучение программы на размеченных библиографических ссылках. Однако со ссылками на русском языке комплекс не работал. В ходе работ по подготовке настоящей диссертации код программы FreeCite был модифицирован в целях поддержки русского языка. Было проведено также ее дополнительное обучение на размеченных библиографических ссылках на русском языке. В результате работы алгоритма из входной строки выделяются необходимые поля. Отметим, что в разработанном автором диссертации прототипе системы пользователь может редактировать извлеченные ею данные.

[5]The swrc ontology - semantic web for research communities / York Sure, Stephan Bloehdorn, Peter Haase et al. // Proceedings of the 12th Portuguese Conference on Artificial Intelligence - Progress in Artificial Intelligence (EPIA 2005), volume 3803 of LNCS. — Covilha: Springer, 2005. — Pp. 218–231.

[6]http://freecite.library.brown.edu

[7]http://crfpp.sourceforge.net

[8]Lafferty, J. Conditional random fields: Probabilistic models for segmenting and labeling sequence data / J. Lafferty, A. McCallum, F.C.N. Pereira // Proc. 18th International Conf. on Machine Learning. — Morgan Kaufmann, 2001. — Pp. 282 – 289.

Разбор BibTeX-записей[9] осуществляется с помощью библиотеки pybtex[10], написанной на языке Python. Для нее автором создана небольшая обертка, которая позволяет использовать ее в общем интерфейсе системы. Благодаря тому, что BibTeX является структурированным форматом, эта процедура не требует использования интеллектуальных алгоритмов.

Импорт информации из eLibrary.ru[11] выполняется следующим образом. Входящими данными при этом способе ввода являются URL-адрес страницы автора или URL-адрес страницы публикации на портале eLibrary.ru. Автором разработан модуль, который копирует необходимые данные с сайта eLibrary.ru и предоставляет пользователю заполненные поля для проверки и редактирования, как и в случае использования других способов ввода. Отметим, что при вводе адреса страницы автора система сама копирует информацию о *всех* публикациях автора на портале eLibrary.ru, сводя объем необходимой ручной работы к минимуму.

Ручной ввод данных необходим в том случае, когда пользователь добавляет в систему данные не о публикациях, а о других результатах научной деятельности, например, о докладах на конференциях, патентах, об участии в редколлегиях журналов и отчетах. Ему необходимо вручную заполнить поля формы (такие как «название», «авторы» и подобные им). Вместе с тем, система облегчает пользователю «ручную» работу, подсказывая по мере набора фамилии авторов и имена сущностей, содержащихся в ней, например, названия конференций, журналов, организаций и других объектов.

Раздел 2.6 посвящен **задаче установления связей** между загруженными данными, содержащими результаты научной деятельности сотрудников, и построенной онтологией области научного знания. До этого этапа из исходных документов выделяется лишь общая информация о научной деятельности сотрудника, например, в каких конференциях он участвовал и какие работы опубликовал. Этап связи необходим для получения информации о содержательной стороне деятельности сотрудника, например, каким областям знания посвящены его работы, какие задачи в этих областях он решал, какие методы и средства применял для решения поставленных задач.

В диссертации для определения степени семантической близости Sim между экземпляром $e \in N_x^S$ онтологии $\mathcal{O}_S$ (например, статьей) и экземпляром $t \in N_x^D$ онтологии $\mathcal{O}_D$ (термином области знания) используется следующая формула:

$$Sim(e,t) = sim_{edit}(title(e), t),$$

где $title(e)$ – название публикации e, а $sim_{edit}(s_1, s_2) = \frac{1}{1+editDist(s_1,s_2)}$ – функция похожести строк[12] на основе расстояния Левенштейна $editDist(s_1, s_2)$[13], равного количеству правок (вставок, удалений и замен), необходимых для превращения строки s_1 в строку s_2. Если значение функции $Sim(e,t)$ превышает значение константы C_{sim}, то между научной публикацией и экземпляром онтологии ставится связь $isAbout$.

Раздел 2.7 посвящен **задаче выполнения запросов** к информации в онтологии. Выполнение аналитических запросов к данным обеспечивается в процессе взаимодействия конечного пользователя системы с программной реализацией модели, описывающей область знания. Такая модель, построенная автором, включает как общую информацию об области знания, так и данные о результатах научных исследований сотрудников организации в этой области. При

[9]Lamport, Leslie. Latex: a document preparation system / Leslie Lamport. — Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1986.

[10]http://pybtex.sourceforge.net

[11]http://elibrary.ru

[12]Lin, Dekang. An Information-Theoretic Definition of Similarity / Dekang Lin // Quality / Ed. by Jude W Shavlik; Citeseer. — Vol. 1. — Citeseer, 1998. — Pp. 296–304.

[13]Левенштейн, В.И. Двоичные коды с исправлением выпадений, вставок и замещений символов / В.И. Левенштейн // Доклады Академий Наук СССР. — 1965.

этом появляется необходимость реализовать автоматическую перезапись запроса. Например, если пользователь ищет все публикации за последний год, посвященные группам Ли, то в случае, если их найдено мало, система должна предложить пользователю выдать все публикации по более широким направлениям, например, теории групп или алгебре в целом. И наоборот, при поиске публикаций или конференций, посвященных математическому анализу, система может предложить пользователю уточнить запрос, предоставив возможность выбора конкретного направления в рамках этой области знания.

Онтологический подход к представлению знаний позволяет применять существующие и прошедшие апробацию алгоритмы выполнения аналитических запросов. В частности, перезапись запроса при использовании онтологий может выполняться автоматически с помощью механизмов логического вывода. В качестве языка запросов к онтологиям в разрабатываемой системе используется язык SPARQL[14], получивший в 2008 году статус рекомендации консорциума W3C[15]. Автором показано, что с помощью описанных в главе 2 архитектуры системы управления научной информацией и технологических решений можно получать ответы на все принятые в системе типы запросов. Представленная в разделе 2.7 связь между запросами, формальной моделью разрабатываемой системы и кодом запросов на языке SPARQL создает дополнительные возможности для верификации программной системы на всех этапах ее жизненного цикла.

Третья глава посвящена разработанным автором алгоритмам построения онтологии области научного знания и выделения терминов из коллекции текстов с заданным тематическим делением. В разделе 3.1 приводится описание алгоритма *Brainsterm* выделения терминов, состоящего из четырех критериев.

Алгоритм выделения терминов *Brainsterm* опирается на следующую математическую модель. Пусть W – множество всех слов, которые встречаются во всех документах заданной коллекции Doc, включая ε – пустое слово, а PW – множество всех упорядоченных пар слов, то есть $PW = W \times W$. Определим документ d как отображение $d\colon \mathbb{N} \to W$, которое сопоставляет каждому натуральному числу n слово, стоящее на n-той позиции в данном документе коллекции. Номера позиций, на которых нет слов (после конца документа), отображаются в пустое слово. Аналогично определим абзац p как отображение $p\colon \mathbb{N} \to W$, которое сопоставляет каждому натуральному числу n слово, стоящее на n-той позиции в данном абзаце. Номера позиций, на которых нет слов, отображаются в пустое слово. Обозначим множество всех абзацев в коллекции через P. Определим рубрику r как произвольное подмножество множества документов, а именно – $r \in 2^{Doc}$. Мощность рубрики, как количество документов в ней, будем обозначать через $|r|$. Обозначим множество всех заданных рубрик через R.

Определим еще несколько вспомогательных отображений:

- $\tau_1\colon PW \to W$, $\tau_2\colon PW \to W$ - проекции пары на множество слов, которые сопоставляют паре первое (соответственно, второе) слово пары;
- $Freq\colon PW \times Doc \to \mathbb{N} \cup \{0\}$ - функция, которая определяет число вхождений пары $pw \in PW$ в документ $d \in Doc$;
- $Freq\colon W \times Doc \to \mathbb{N} \cup \{0\}$ - функция, которая определяет число вхождений слова $w \in W$ в документ $d \in Doc$;
- $L(d) = |\{n \in \mathbb{N} \mid d(n) \neq \varepsilon\}|$ - длина документа d;
- $id(a) = a$ - тождественное отображение;
- $Av(f, A) = \frac{\sum\limits_{a \in A} f(a)}{|A|}$ - среднее значение функции f на конечном множестве A. Например, $Av(|\cdot|, R)$ - среднее количество документов в рубрике, $Av(L, Doc)$ - средняя длина доку-

[14] http://www.w3.org/TR/rdf-sparql-query

[15] http://www.w3.org

мента, $Av(id, A)$ - среднее арифметическое чисел из множества $A = \{a_1, \ldots, a_k\}$.

Исходные данные для алгоритма *Brainsterm* представляют собой таблицу, в которой строки соответствуют словам, встречающимся в документах коллекции. Отметим, что перед применением алгоритма рекомендуется провести первоначальную лингвистическую обработку документов – лемматизацию, то есть преобразование словоформ в нормальную (словарную) форму. Например, для существительных в русском языке такой формой является именительный падеж, единственное число. В каждой строке исходной таблицы записано четыре числа: номер рубрики; номер документа; номер абзаца; номер слова. Если слово А встречается в абзаце раньше слова Б, то и в таблице строка слова А будет выше строки слова Б. Таблица отсортирована по первым трем колонкам. Таким образом, о конкретном слове известно только то, в каких документах, сколько раз и на каких позициях оно встречается.

Алгоритм *Brainsterm* включает четыре этапа. На каждом из них с помощью некоторого правила выбирается подмножество M_i множества M_{i-1}, полученного на предыдущем шаге. На первом этапе выбор производится из множества PW (всех пар слов), то есть $M_0 = PW$. Множество M_4 и есть термины – пары, которые удовлетворяют всем четырем критериям.

Пространственный критерий основан на предположении о том, что слова, образующие термин, расположены в тексте достаточно близко (хотя и не обязательно рядом):

$$M_1 = \{pw \in M_0 \mid \exists p \in P : |p^{-1}(\tau_1(pw)) - p^{-1}(\tau_2(pw))| \leqslant \texttt{MAX\_DIST}\}.$$

Пару образуют два слова, находящиеся в одном абзаце, между которыми в тексте стоят не более `MAX_DIST-1` других слов.

Критерий частотности обеспечивает более высокую информативность базиса векторного пространства текстов путем исключения из множества M_1 пар, которые встретились во всей коллекции меньше, чем `MIN_FREQ` раз:

$$M_2 = \{pw \in M_1 \mid \sum_{r \in R} \sum_{d \in r} Freq(pw, d) \geqslant \texttt{MIN\_FREQ}\}.$$

Критерий характерности – основной критерий алгоритма. Его суть заключается в определении термина, «пара должна быть характерной для некоторых рубрик». Каждой паре сопоставляется набор чисел – весов пары в каждой из рубрик. Вес $Weight_r$ пары pw в рубрике r вычисляется по формуле, подробное обоснование выбора которой приведено в разделе 3.1.4. Пусть рубрика r состоит из k документов $d_1, \ldots, d_k$. Будем считать, что $k > 1$. К параметрам, от которых должна зависеть функция веса, относятся: число вхождений пары в документы рубрики $Freq(pw, d_i)$, $i \in \overline{1,k}$; относительные длины документов рубрики $\frac{L(d_i)}{Av(L,Doc)}$, $i \in \overline{1,k}$; относительное количество документов в рубрике $\frac{|r|}{Av(|\cdot|,R)}$. В итоге получаем, что функция $Weight_r$ должна зависеть от $2k+1$ параметров:

$$Weight_r(pw) = Weight_r\left(Freq(pw, d_1), \ldots, Freq(pw, d_k), \frac{L(d_1)}{Av(L, Doc)}, \ldots, \frac{L(d_k)}{Av(L, Doc)}, \frac{|r|}{Av(|\cdot|, R)}\right) = Weight_r(x_1, \ldots, x_k, y_1, \ldots, y_k, z) = Weight_r(\bar{x}, \bar{y}, z).$$

Индекс r у функции $Weight$ подчеркивает тот факт, что для рубрик, содержащих различное количество документов, функции $Weight_r$ зависят от разного числа аргументов. Отметим область определения функции $Weight_r$:

$$D(Weight_r) : x_i \in \mathbb{Z} \cap [0, +\infty),\ y_i \in (0, +\infty),\ z \in (0, +\infty),\ i \in \overline{1,k}.$$

Пользуясь введенными обозначениями, формализуем **требования** к функции $Weight_r$:

1. неотрицательное значение – $Weight_r(pw) \geqslant 0$;
2. прямая зависимость от частоты – $\forall i \in \overline{1,k}$ при $x'_i > x_i$

$$Weight_r(x_1, \ldots, x_{i-1}, x'_i, x_{i+1}, \ldots, x_k, \bar{y}, z) > Weight_r(x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_k, \bar{y}, z);$$

3. обратная зависимость от длины документа – $\forall i \in \overline{1,k}$ при $y'_i > y_i$

$$Weight_r(\bar{x}, y_1, \ldots, y_{i-1}, y'_i, y_{i+1}, \ldots, y_k, z) < Weight_r(\bar{x}, y_1, \ldots, y_{i-1}, y_i, y_{i+1}, \ldots, y_k, z);$$

4. обратная зависимость от мощности рубрики – при $z' > z$

$$Weight_r(\bar{x}, \bar{y}, z') < Weight_r(\bar{x}, \bar{y}, z);$$

5. прямая зависимость от числа документов, в которых встречается пара, а именно – при $y_1 = y_2 = \ldots = y_k$, $\forall n \in \mathbb{N}$, $\forall l \in \overline{1,k}$ выполняется

$$Weight_r(\underbrace{n, \ldots, n}_{k}, \bar{y}, z) > Weight_r(\underbrace{\overbrace{0, \ldots, 0, kn}^{l}, 0, \ldots, 0}_{k}, \bar{y}, z).$$

Для облегчения подбора функций, которые удовлетворяют сформулированным требованиям, выбран следующий общий вид функции $Weight_r$:

$$Weight_r(\bar{x}, \bar{y}, z) = f(g(f(x_1, y_1), \ldots, f(x_k, y_k)), z).$$

Внутренние функции $f(x, y)$ определяют веса пары в документах и зависят от числа вхождений пары в документы и относительных длин документов. После этого к вычисленным весам пары в документах применяется функция $g(x_1, \ldots, x_k)$, которая обеспечивает выполнение требования 5 (прямую зависимость веса от числа документов, в которых встречается пара). После этого снова применяется функция $f(x, y)$, которая выражает зависимость веса пары в рубрике от относительной мощности рубрики. Функции f и g должны иметь следующие области определения и значений:

$$D(f) : x \in [0, +\infty),\ y \in (0, +\infty),\ E(f) = [0, +\infty),$$
$$D(g) : x_i \in [0, +\infty),\ i = \overline{1,k},\ E(g) = [0, +\infty).$$

Следующая лемма позволяет проверять выполнение требований 1-4 к функции $Weight_r$, используя функции f и g.

Лемма 1. Пусть функции $f = f(x, y)$ и $g = g(x_1, \ldots, x_k)$, $k > 1$ имеют области определения и значений, указанные выше. Пусть выполнены следующие условия на всей области определения этих функций:

- при $x' > x \quad f(x', y) > f(x, y)$;
- $\forall i \in \overline{1,k}$ при $x'_i > x_i \quad g(x_1, \ldots, x_{i-1}, x'_i, x_{i+1}, \ldots, x_k) > g(x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_k)$;
- при $y' > y \quad f(x, y') < f(x, y)$;

Тогда требования 1-4 к функции $Weight_r(\bar{x}, \bar{y}, z) = f(g(f(x_1, y_1), \ldots, f(x_k, y_k)), z)$ выполнены.

Для окончательного выбора функции веса пары слов в рубрике было проведено выборочное поэтапное тестирование комбинаций семи вариантов функции $f(x, y)$ и трех вариантов функции $g(x_1, \ldots, x_k)$. Для наглядности были протестированы еще 3 функции веса более простого вида, зависящие от меньшего числа аргументов. Тестирование показало, что самый высокий уровень точности классификации достигается при использовании следующей комбинации функций: $f(x, y) = \frac{\sqrt{x}}{\ln(1+y)}$; $\quad g(x_1, \ldots, x_k) = \sum_{i=1}^{k} \ln(1 + x_i)$.

С учетом изложенного выше, в качестве функции веса пары в рубрике была выбрана следующая функция:

$$Weight_r(pw) = \frac{\sqrt{\sum\limits_{d \in r} \ln\left(\frac{\sqrt{Freq(pw,d)}}{\ln\left(\frac{L(d)}{Av(L,Doc)}+1\right)}+1\right)}}{\ln\left(\frac{|r|}{Av(|\cdot|,R)}+1\right)}.$$

Аналогично определяется вес слова в рубрике $Weight_r(w)$, а именно – в представленной выше формуле $Freq(pw, d)$ заменяется на $Freq(w, d)$. Автором сформулирована и доказана следующая теорема, позволяющая использовать определенную таким образом функцию $Weight_r$ для вычисления веса пары в рубрике.

Теорема 1. Выбранная функция $Weight_r(pw)$ удовлетворяет требованиям 1-5.

К вычисленному набору весов пары в рубриках применяется функция характерности $Discr$, которая сопоставляет набору неотрицательных действительных чисел $A = \{a_1, \ldots, a_n\}$, не равных одновременно нулю, число – «показатель характерности набора»:

$$Discr(a_1, a_2, \ldots, a_n) = 1 - \frac{Av(id, A)}{\max_{a \in A} a \cdot (1 + \ln(1 + \max_{a \in A} a))}.$$

Эта функция принимает значения из отрезка $[0, 1]$, и чем более характерным является набор с точки зрения определения термина, тем ближе значение этой функции к 1.

Пусть $R = \{r_1, \ldots, r_n\}$. Тогда $M_3 = \{pw \in M_2 \mid$
$Discr(Weight_{r_1}(pw), \ldots, Weight_{r_n}(pw)) \geqslant \texttt{MIN\_DISCR}\}$.

Таким образом отбираются пары, значение функции $Discr$ на которых не меньше константы `MIN_DISCR`, принадлежащей отрезку $[0, 1]$.

Критерий значимых рубрик основан на идее о том, что слова, образующие термин, должны встречаться достаточно часто в паре и сравнительно редко по отдельности. В частности, её учет позволяет исключить следующую возможную ошибку. Пусть в документах рубрики некоторое число раз встретилось слово, уникальное для данной рубрики (например, «гидрофосфат» в рубрике «химия»). Тогда все слова, находящиеся на небольшом расстоянии от этого слова, вероятнее всего будут удовлетворять критерию характерности, так как в других рубриках эти пары не встречаются вообще. Вместе с тем, многие из них, очевидно, не являются терминами (например, «гидрофосфат считается» или «имея гидрофосфат»).

Определение 3.1. Рубрика r называется *значимой* для пары слов pw, если

$$Weight_r(pw) \geqslant Av(Weight_\cdot(pw), R).$$

Пусть $2^R = \{A \mid A \subseteq R\}$. Определим отображение $imp\colon M_3 \to 2^R$, сопоставляющее каждой паре слов множество значимых для нее рубрик:

$$imp(pw) = \{r \in R \mid Weight_r(pw) \geqslant Av(Weight_\cdot(pw), R)\}.$$

Тогда

$$M_4 = \{pw \in M_3 \mid \min_{r \in imp(pw)} \left(\frac{Weight_r(\tau_1(pw))}{Weight_r(pw)} + \frac{Weight_r(\tau_2(pw))}{Weight_r(pw)}\right) \leqslant \texttt{MAX\_FREQ\_RATIO}\}.$$

Таким образом отбираются пары, у которых среди значимых для них рубрик найдется рубрика, в которой сумма весов каждого из слов пары превышает вес пары не более чем в `MAX_FREQ_RATIO` раз.

Автором сформулирована и доказана следующая теорема, позволяющая оценить вычислительную сложность представленного алгоритма.

Теорема 2. Пусть дано множество документов Doc, разделенных на рубрики из множества R. Пусть W – множество всех различных слов, которые встречаются в этих документах, а $m = |W|$ – количество таких слов. Пусть $L(d)$, $d \in Doc$ – функция, выражающая длину документа (количество слов в нем). Обозначим $n = \sum\limits_{d \in Doc} L(d)$ – количество всех слов во множестве документов. Тогда для алгоритма *Brainsterm* справедливы следующие оценки:

- временная сложность алгоритма (в худшем случае) равна $O(|R||Doc|\min(m^2, n) + n)$;
- пространственная сложность алгоритма (в худшем случае) равна $O(|Doc|\min(m^2, n))$;
- количество терминов, извлеченных в результате работы алгоритма, равно $O(\min(m^2, n))$.

Под временной сложностью алгоритма понимается максимальное количество элементарных операций (арифметических и операций сравнения), которые необходимо выполнить для решения задачи. Под пространственной сложностью понимается максимальное количество ячеек памяти, которые необходимо выделить для работы алгоритма.

В разделе 3.2 представлен разработанный автором **алгоритм *Sonmake* построения онтологии** области научного знания на основе коллекции анонсов научных конференций, разделенных на рубрики, а также информации из поисковых систем в Интернет. В качестве основного источника данных для построения онтологии используются анонсы конференций, называемые в научной среде *call for papers* (CFP). Этот подход обладает важными достоинствами. В частности, он позволяет получить достаточно надежную, актуальную и полную информацию об области научного знания. Документы CFP содержат основные сведения о конференциях, в том числе – название, место и даты проведения, состав программного комитета, описание конференции, список направлений области знания, работы по которым принимаются на рассмотрение. Вторым инструментарием, который используется в алгоритме для получения информации, является поисковая система в Интернет. Алгоритм *Sonmake* построения онтологии области знания состоит из следующих семи этапов:

- построение множества имен понятий N_C^D;
- выделение терминов, которые характеризуют заданную область научного знания D;
- фильтрация терминов;
- выделение ассоциативных связей между терминами;
- построение иерархии терминов;
- перевод терминов на русский язык;
- классификация терминов по понятиям онтологии.

Список имен понятий онтологии N_C^D, содержащий 60 элементов, фиксирован и является общим для всех областей научного знания. В него входят такие слова, как направление, понятие, алгоритм, парадигма, метод и другие, аналогичные им, а также их эквиваленты на английском языке.

На **этапе выделения терминов** из коллекции анонсов конференций извлекаются ключевые слова. Исходя из требований алгоритма *Brainsterm*, документы должны быть разделены на рубрики. В случае анонсов конференций это требование легко выполняется, например, при использовании списков рассылок анонсов конференций, посвященных различных областям знания. Каждый список рассылки будет представлять одну рубрику. Отметим, что одна из рубрик должна соответствовать области знания D, онтология которой строится. Назовем эту рубрику «целевой». Применим к документам, разделенным на рубрики, алгоритм *Brainsterm* выделения терминов. Результатом работы алгоритма является множество терминов $Terms$. Выделим из него подмножество $Terms_1 \subseteq Terms$, состоящее из терминов, которые встречаются в документах целевой рубрики r_D хотя бы один раз, и отсортируем его элементы по убыванию веса термина в этой рубрике. Веса терминов в рубриках вычисляются в процессе работы алгорит-

ма *Brainsterm*. Полученное множество $Terms_1$ содержит ключевые слова, которые описывают тематики конференций в рамках области знания D.

Следующим шагом алгоритма *Sonmake* является **фильтрация** полученных терминов $Terms_1$, состоящая из двух этапов. На первом этапе фильтрации исключаются пары слов, которые не соответствуют критериям термина. Для этого используются перечисленные далее четыре критерия. Пусть $A \in Terms$ – термин-кандидат, состоящий из двух слов – A_1 и A_2, тогда эти критерии формулируются следующим образом:

- в онлайн-энциклопедии Википедия[16] есть статья с названием A;
- $\frac{hits(\text{“A is a term”})}{hits(A)} > C_1$;
- $\frac{hits(\text{“A is a concept”})}{hits(A)} > C_2$;
- $\frac{hits(\text{“}A_1 \text{ AND } A_2\text{”})}{\min(hits(A_1), hits(A_2))} > C_3$.

Функция $hits(x)$ обозначает количество страниц, найденных поисковой системой в Интернет в ответ на запрос x. Термин-кандидат считается прошедшим первый этап фильтрации, если он удовлетворяет *хотя бы одному* из перечисленных четырех критериев. Числа $C_1, C_2, C_3 \in [0, 1]$ являются параметрами алгоритма.

Целью второго этапа фильтрации является исключение пар слов, которые не связаны с заданной областью знания D. Для этого используется критерий

$$\frac{hits(\text{“A AND D”})}{hits(A)} > C_4,$$

где A – термин-кандидат, D – название заданной области знания, а C_4 – параметр алгоритма. Обозначим через $Terms_2$ множество всех терминов из $Terms_1$, успешно прошедших оба этапа фильтрации. Полученная совокупность терминов образует множество имен экземпляров N_x^D онтологии $\mathcal{O}_D$: $N_x^D = Terms_2$.

Целью следующего этапа является **выделение пар связанных терминов**, то есть выбор из всех возможных пар терминов, отобранных на предыдущих этапах, тех пар, которые являются семантически близкими. Для определения степени семантической близости между двумя терминами используется широко распространенная мера *Normalized Google Distance (NGD)*[17]. Пусть A и B – термины, а N – общее число страниц, индексируемых поисковой системой. Тогда степень семантической близости NGD между A и B определяется по формуле:

$$NGD(A, B) = \frac{\max\{\log hits(A), \log hits(B)\} - \log hits(\text{“A AND B”})}{\log N - \min\{\log hits(A), \log hits(B)\}}.$$

После этого из всех пар терминов во множество $Terms_s$ отбираются те, степень близости между которыми превышает пороговое значение.

Следующим этапом алгоритма является **построение иерархии терминов**. Классический алгоритм построения иерархии понятий с помощью лингвистических шаблонов, разработанный Херст[18], оказывается неэффективным для построения иерархии научных направлений. В рамках настоящей работы специально для решения этой задачи были разработаны лингвистические шаблоны. Основной шаблон выглядит как

$$A \ is \ * \ keyword \ * \ prep \ (\ aux)?\ B,$$

[16]http://www.wikipedia.org

[17]Cilibrasi, R L. The Google Similarity Distance / R L Cilibrasi, P M B Vitanyi // IEEE Transactions on Knowledge and Data Engineering. — 2007. — Vol. 19, №3. — Pp. 370– 383.

[18]Hearst, M.A. Automatic acquisition of hyponyms from large text corpora / M.A. Hearst // Proceedings of the 14th conference on Computational linguistics-Volume 2. — Association for Computational Linguistics, 1992. — Pp. 539–545.

где A, B – термины, между которыми происходит поиск иерархической связи, $keyword$ – ключевое слово из построенного словаря $keywords$ связей между научными терминами, $prep$ – предлог, содержащийся в построенном словаре предлогов, а aux – артикль или квантор из созданного словаря вспомогательных слов. Словарь ключевых слов $keywords$ содержит 40 слов, например, field, component, discipline, step, domain, method и другие. Шаблон применяется к результатам-сниппетам (небольшим фрагментам текстов найденных документов), которые возвратила поисковая система в ответ на запрос "A AND B". Следует подчеркнуть, что эти запросы выполняются только для тех пар терминов, которые вошли в множество $Terms_s$. Использование данных о семантической близости, полученных на предыдущем этапе, позволяет существенно сократить количество запросов к поисковой системе. Примером фразы, найденной по шаблону, служит "text categorization is a fundamental task in document processing". Здесь A = "text categorization", B = "document processing", $keyword$ = "task", $prep$ = "in", а элемент aux не используется.

Описанные выше этапы используются для построения онтологии на английском языке. Для того, чтобы добавить в онтологию термины на русском языке, применяется следующий **метод перевода терминов на русский язык**. Его идея заключается в использовании ручного труда людей, составляющих энциклопедию Википедия. Во многих статьях Википедии есть ссылки на версии этой же статьи на других языках. Используя эти ссылки, можно найти термины на русском языке, которые соответствуют термину на английском. Для каждого выделенного термина выполняется автоматическая проверка на существование статьи в Википедии с таким же названием, а потом проверяется факт наличия у этой статьи ссылки на русскую версию. Если такая ссылка есть, то название соответствующей статьи на русском языке добавляется в онтологию.

Для получения дополнительной информации об экземплярах онтологии, используется **алгоритм классификации терминов** по классам онтологии. Для определения подмножества классов, к которым относится термин $A \in Terms_2$, рассчитывается степень его принадлежности к каждому классу $C \in N_C^D$ по формуле

$$score(A, C) = \frac{hits(\text{"A is a C"})}{hits(A)}.$$

Если $score(A, C) > C_5$, то между термином A и классом C строится отношение $rdf : type$, которое означает, что экземпляр принадлежит классу. Число C_5 является параметром алгоритма. Отметим, что термин может принадлежать нескольким классам одновременно.

Автором сформулирована и доказана следующая теорема, позволяющая оценить вычислительную сложность представленного алгоритма.

Теорема 3. Пусть дано множество документов Doc, разделенных на рубрики из множества R. Пусть W – множество всех различных слов, которые встречаются в этих документах, а $m = |W|$ – количество таких слов. Пусть $L(d)$, $d \in Doc$ – функция, выражающая длину документа (количество слов в нем). Обозначим $n = \sum\limits_{d \in Doc} L(d)$ – количество всех слов во множестве документов. Тогда для алгоритма *Sonmake* справедливы следующие оценки:

- временная сложность алгоритма (в худшем случае) равна

$$O(\min(m^4, n^2) + |R||Doc|\min(m^2, n) + n);$$

- пространственная сложность алгоритма (в худшем случае) равна

$$O(\min(m^4, n^2) + |Doc|\min(m^2, n)).$$

Под временной сложностью алгоритма понимается максимальное количество элементарных операций (арифметических, операций сравнения и операций разового обращения к поисковой системе), которые необходимо выполнить для решения задачи. Под пространственной сложностью понимается максимальное количество ячеек памяти, которые необходимо выделить для работы алгоритма.

Разработанные алгоритмы составляют основу соответствующих модулей системы учета и анализа научной информации, описанию которой посвящена настоящая диссертация. Следует подчеркнуть, что эти алгоритмы могут быть использованы и в других системах, в которых возникает необходимость извлекать термины и строить онтологии областей научного знания.

В **четвертой главе** представлены результаты исследования эффективности (тестирования) программных реализаций разработанных автором алгоритмов выделения терминов и построения онтологии. Далее для краткости изложения будем под алгоритмом понимать его программную реализацию. В разделе 4.1 рассматривается **эффективность алгоритма выделения терминов *Brainsterm***.

Алгоритм *Brainsterm* реализован на языке C++. Программа составляет около 1200 строк. Программный код алгоритма находится в открытом доступе[19].

Эффективность алгоритма *Brainsterm* оценивается с использованием полученных терминов как базиса векторного пространства в задачах классификации и кластеризации текстов. Проведено сравнение алгоритма *Brainsterm* с двумя широко распространенными в области обработки данных алгоритмами, основанными на векторной модели, а именно – метода TF-IDF (term frequency – inversed document frequency) и алгоритма LSI (latent semantic indexing) в трех модификациях (с булевой матрицей, с матрицей частот и матрицей из весов tf-idf), которые не учитывают разделения документов на рубрики. Каждый из алгоритмов может быть использован для отображения документов в точки векторного пространства заданной размерности, при котором исходные рубрики переходят в кластеры точек. После этого конфигурация кластеров формально оценивается с помощью алгоритмов классификации и кластеризации. Классификация показывает, насколько алгоритм сохраняет исходное разделение на рубрики, а кластеризация – насколько компактными получились кластеры. Оценка с помощью классификации проводится следующим образом. Выполняется классификация точек тестовой выборки с помощью метода «K ближайших соседей», для этого используется разбиение на кластеры точек обучающей выборки. Каждая точка тестовой выборки, представляющая документ одной из рубрик, попадает в один из кластеров – образов исходных рубрик. Вычисляется процентное отношение количества документов тестовой выборки, попавших после классификации в образ той же рубрики, к которой они принадлежали изначально, к общему числу документов выборки. Чем больше полученное число, тем эффективнее рассматриваемый алгоритм. Оценка с помощью метода кластеризации производится следующим образом. К кластерам, образованным точками тестовой выборки, применяется один из стандартных методов оценки кластеризации точек, оценивающий относительный разброс точек внутри кластеров. Чем меньше относительный разброс, тем выше качество кластеризации.

В целях сравнения алгоритмов была использована выборка, содержащая около 1.4 миллиона слов в примерно 7 тысячах документов. Обучающая выборка содержала 3591 документ, остальные документы составили тестовую выборку. Заметим, что алгоритм LSI не допускает размерности выше, чем количество документов в обучающей выборке. Именно поэтому на иллюстрациях графики показателей модификаций алгоритма LSI обрываются на точке 3591.

Общее представление о сравнительной производительности алгоритмов можно получить из результатов тестирования, оценивающих время их работы. Параметры используемой вычислительной установки: CPU AMD Opteron 2 Ghz, 8 Gb RAM. На рис. 2 представлены показатели

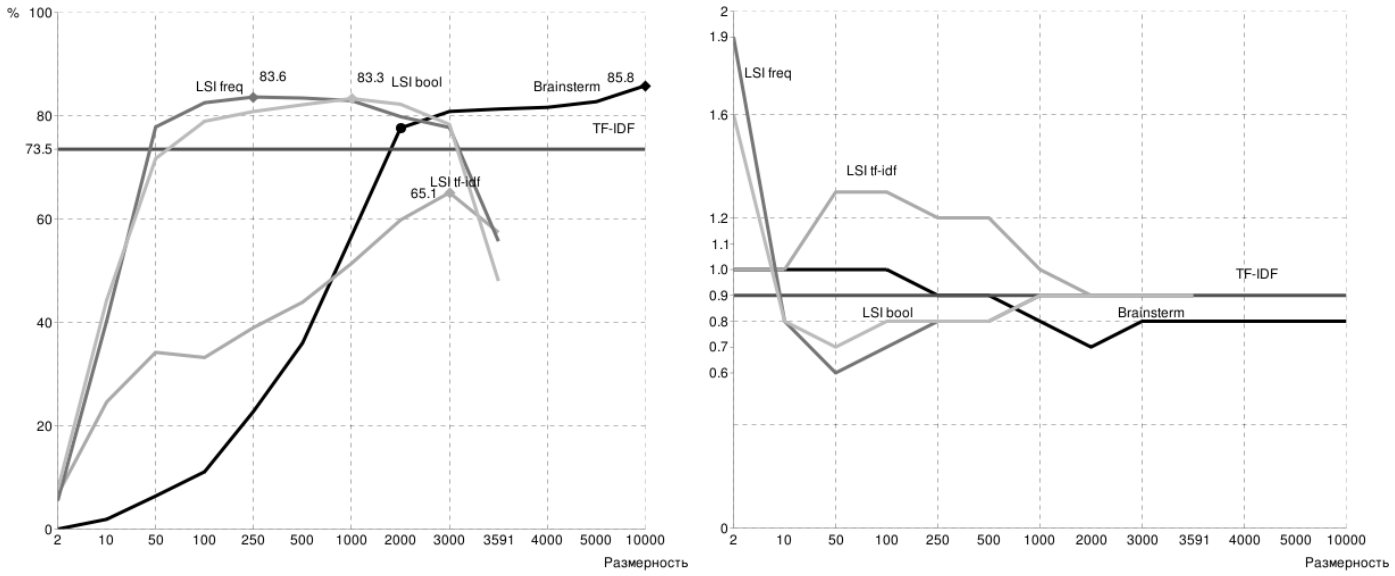
[19]https://bitbucket.org/goldan/brainsterm

времени работы алгоритмов *Brainsterm*, LSI и TF-IDF. Каждый алгоритм был запущен с параметром размерности целевого пространства, имеющим оптимальное значение, при котором данный алгоритм достигает максимальных результатов точности классификации. Для алгоритма *Brainsterm* это значение равно 10000, для алгоритма LSI – 250. Алгоритм TF-IDF не позволяет выбрать размерность целевого пространства. Фактически она равна общему количеству уникальных слов в коллекции, то есть в данном случае около 21000.

Рис. 2. Сравнение времени работы алгоритмов.

Как видно из графика, алгоритмы *Brainsterm* и TF-IDF работают гораздо быстрее алгоритма LSI. Отметим, что при тестировании на имеющейся выборке алгоритму *Brainsterm* потребовалось около 1 Gb оперативной памяти, алгоритму TF-IDF – около 550 Mb, а алгоритму LSI – около 2.2 Gb. Показатели времени работы программы, реализующей алгоритм *Brainsterm*, и используемой ею памяти подтверждают справедливость полученных ранее (теорема 2) аналитических оценок вычислительной сложности алгоритма и свидетельствуют о его практической применимости.

На рис. 3a и 3b приведены результаты сравнения точности классификации и качества кластеризации. Большее значение точности классификации и меньшее значение относительного разброса кластеров соответствуют более высокой эффективности алгоритма. Качество кластеризации у всех алгоритмов получилось примерно одинаковым (при этом у алгоритма *Brainsterm* нет таких скачков ухудшения качества, как у алгоритма LSI), а самые высокие показатели точности классификации (85.8%) продемонстрировал алгоритм *Brainsterm*. Следует отметить, что этот результат был достигнут на достаточно высокой размерности – 10000. Алгоритм LSI показал результат 83.6% на размерности 250, а алгоритм TF-IDF, не зависящий от размерности, позволил добиться точности 73.5%.



(a) Точность классификации

(b) Качество кластеризации

Рис. 3. Сравнение показателей эффективности алгоритмов Brainsterm, LSI и TF-IDF.

Как показали результаты анализа, алгоритм *Brainsterm* сочетает в себе высокую скорость

работы, значительно превышающую скорость работы алгоритма LSI и высокую эффективность, сравнимую с эффективностью алгоритма LSI и превышающую показатели TF-IDF. Изложенные факты подтверждают хорошие перспективы применения алгоритма *Brainsterm* на практике.

В разделе 4.2 представлены **результаты тестирования алгоритма *Sonmake* построения онтологии** области научного знания на основе информации из анонсов научных конференций и Интернет, разработанного автором. Программный код алгоритма *Sonmake* находится в открытом доступе[20]. В качестве исходной коллекции анонсов конференций использовалась база портала WikiCFP[21], содержащая 13098 документов. Разделение документов на рубрики производилось на основе меток, присвоенных им пользователями сайта. В результате применения алгоритма выделения терминов *Brainsterm* было получено 1793 термина, из которых на этапах фильтрации было выбрано 874. В таблице 2 приведены первые 15 терминов, полученных алгоритмом *Brainsterm*, отсортированные по весу в целевой рубрике.

№	термин	№	термин	№	термин
1	data mining	6	signal processing	11	intelligent systems
2	software engineering	7	data management	12	software development
3	machine learning	8	computational intelligence	13	communication systems
4	artificial intelligence	9	network security	14	access control
5	knowledge discovery	10	wireless networks	15	formal methods

Таблица 2. Первые 15 терминов, выделенных с помощью алгоритма Brainsterm в области «Computer Science».

В таблице 3 представлены результаты анализа эффективности выделения и фильтрации терминов. Отметим, что при вычислении точности и локальной полноты[22] термином считалась пара слов, характерная для области «информатика» с точки зрения эксперта. В таблице 4 приведены результаты анализа эффективности выявления отношений между терминами.

№	шаг алгоритма	терминов выделено	точность	локальная полнота	F-мера
1	*Brainsterm*	1793	70.5%	-	-
2	фильтрация-1	1403	73.5%	81%	77.1%
3	фильтрация-2	874	77%	63.6%	69.6%
2-3	фильтрация в целом	874	77%	51.7%	61.9%

Таблица 3. Оценки эффективности этапов выделения и фильтрации терминов.

№	тип отношений	отношений выделено	точность
4	ассоциативные	3771	-
5	иерархические	85	89.4%
6	категориальные	135	85.2%

Таблица 4. Оценки эффективности этапов выделения отношений между терминами.

[20]https://bitbucket.org/goldan/sonmake

[21]http://wikicfp.com

[22]Sánchez, D. Domain ontology learning from the web / D. Sánchez // The Knowledge Engineering Review. — 2009. — Vol. 24, №4. — Pp. 413–413.

На этапе перевода терминов на русский язык для 401 из 874 терминов (45.9%) в Википедии была найдена статья. Из них для 212 терминов (24.2%) были найдены русские эквиваленты, которые были добавлены в онтологию. В результате работы алгоритма *Sonmake* была построена онтология, содержащая 61 класс, 1086 экземпляров и 4203 отношения. Ее фрагмент изображен на рис. 4.

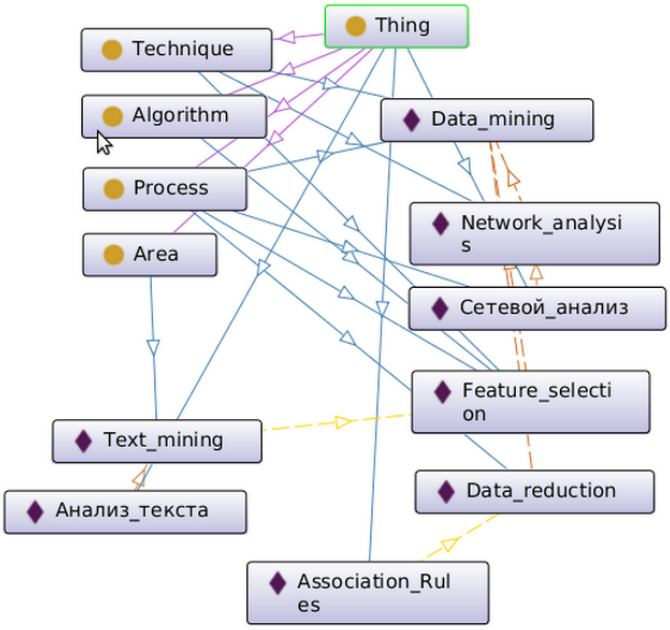


Рис. 4. Фрагмент построенной с помощью алгоритма Sonmake онтологии области «Computer Science».

Алгоритм *Sonmake* построения онтологии области знания обладает следующими недостатками: ограничение на вид термина (два слова); при определении связей между терминами не учитывается расширенный контекст; недостаточная полнота выделения именованных отношений; малое количество типов выделения именованных отношений; высокая вычислительная сложность. Результаты тестовых испытаний алгоритма *Sonmake* согласуются с аналитическими оценками его вычислительной сложности, доказанными в теореме 3. В связи с тем, что для каждого термина необходимо выполнять большое число запросов к поисковой системе, алгоритм построения онтологии обладает высокой вычислительной сложностью. При этом операции, выполняемые на локальном вычислительном узле, работают на порядок быстрее. Отметим, что обычно в информационных системах онтология строится и перестраивается сравнительно редко, что позволяет применять предложенный алгоритм на практике.

Перечислим достоинства алгоритма *Sonmake*: мягкие требования к исходным данным; автоматическое выделение терминов области знания; высокая точность выделения именованных отношений; применимость к любым областям научного знания; отсутствие необходимости в большом объеме ручного труда экспертов.

В целях реализации подхода к задаче управления научной информацией, предлагаемого автором, в настоящее время разрабатывается **программный комплекс ИСТИНА** (Интеллектуальная Система Тематического Исследования НАучно-технической информации)[23]. Описанию созданного автором прототипа этого комплекса посвящена **пятая глава**. Целью системы является учет и анализ информации о результатах научной деятельности в научных организациях с целью подготовки и принятия решений. Основными задачами системы ИСТИНА являются, во-первых, предоставление возможности сотрудникам структурных подразделений организа-

[23]http://istina.imec.msu.ru

ции перманентно вести учет результатов своей научной деятельности и в автоматизированном режиме формировать годовые научные отчеты, а во-вторых, предоставление руководителям отдельных структурных подразделений и организации в целом автоматизированного средства проведения количественного и тематического анализа научной деятельности каждого из сотрудников, отдельных подразделений и учреждения в целом. В настоящее время система ИСТИНА позволяет в удобном для конечного пользователя режиме вводить данные о публикациях путем автоматизированного разбора информации из библиографических ссылок, BibTeX-записей и импорта из портала eLibrary.ru. На основе введенных в хранилище системы данных о публикациях для каждого сотрудника автоматически создается отдельная «домашняя» страница, содержащая список его публикаций. В тестовом режиме применяется автоматическое определение научных интересов пользователей на основе введенных данных и онтологии. В разделе 5.1 перечислены особенности программной реализации системы. Раздел 5.2 посвящен описанию результатов практической апробации ее прототипа в НИИ механики МГУ имени М.В. Ломоносова[24]. В разделе 5.3 приводится анализ разработанного прототипа системы на предмет ее соответствия предъявляемым к системе требованиям. Делается вывод, что прототип соответствует основным требованиям, а предложенная архитектура позволяет построить целевую систему, которая при ее реализации в полном объеме будет удовлетворять всем предъявляемым к ней требованиям. В разделе 5.4 перечисляются направления дальнейшего развития системы.

В **заключении** представлены **основные результаты** диссертационной работы.

1. На основе исследования предметной области построены математические модели и алгоритмы, разработаны опирающиеся на онтологии архитектурные и технологические решения для создания системы пополнения и хранения, анализа и выдачи по запросу информации, характеризующей результаты деятельности научной организации. С использованием онтологий и языка SPARQL предложено формальное описание запросов к системе, создающее гарантии их вычисления и дополнительные возможности для эффективной верификации кода системы на всех этапах ее жизненного цикла.
2. Разработан алгоритм построения онтологии отдельной области научного знания, основанный на выделении терминов из анонсов научных конференций, а также на использовании информации из поисковых систем в Интернет. Получены аналитические оценки, характеризующие вычислительную сложность его программной реализации.
3. Разработан алгоритм выделения терминов-пар слов из коллекции текстов с заданным тематическим делением. Доказано, что предложенная автором в составе алгоритма базовая функция веса термина в рубрике удовлетворяет предъявляемым к ней требованиям. Получены аналитические оценки, характеризующие вычислительную сложность программной реализации алгоритма.
4. Создан прототип программного комплекса для учета и анализа научной информации, именуемый Интеллектуальной Системой Тематического Исследования НАучно-технической информации (ИСТИНА), тестовые испытания которого подтвердили справедливость аналитических оценок сложности реализации основных алгоритмов, а также тот факт, что комплекс в целом удовлетворяет предъявляемым к нему требованиям.

Благодарности. Автор выражает глубокую благодарность своему научному руководителю доктору физико-математических наук, профессору Валерию Александровичу Васенину за постановку задачи и постоянное внимание к работе. Автор благодарит кандидатов физико-математических наук С. А. Афонина и А. С. Козицына за ценные замечания с их стороны по ходу выполнения работы.

[24]Следует отметить, что система может использоваться не только в МГУ имени М.В. Ломоносова, но и в других научных центрах Российской Федерации.

Список литературы

[1] Голомазов Д. Д. *Выделение терминов из коллекции текстов с заданным тематическим делением* // Информационные технологии. — № 2, 2010. — С. 8–13.

[2] Afonin S., Golomazov D. *Minimal Union-Free Decompositions of Regular Languages* // Language and Automata Theory and Applications. Lecture Notes in Computer Science, volume 5457. — Springer, 2009. — pp. 83–92. (Д. Д. Голомазову принадлежат результаты по построению математической модели и доказательствам основных утверждений).

[3] Васенин В. А., Афонин С. А., Голомазов Д. Д. *Использование семантических технологий для обнаружения грид-ресурсов* // Программная инженерия. — № 7, 2011. — С. 2–8. (Д. Д. Голомазову принадлежат результаты исследования и анализа существующих подходов к обнаружению грид-ресурсов, а также их практического применения).

[4] Васенин В. А., Афонин С. А., Голомазов Д. Д. *К созданию системы управления научной информацией на основе семантических технологий* // Материалы Всероссийской конференции с международным участием «Знания - Онтологии - Теории» (ЗОНТ-2011), 3-5 октября 2011 г., г. Новосибирск, том 1. — Новосибирск, Институт математики им. С.Л. Соболева СО РАН, 2011. — С. 78–87. (Д. Д. Голомазову принадлежат результаты исследования и анализа существующих методов и средств управления научной информацией, описание архитектуры и программной реализации разработанных им системы ИСТИНА и алгоритма построения онтологии области научного знания).

[5] Afonin S., Golomazov D. *Calculating Semantic Similarity Between Facts* // Proc. of the Int. Conf. on Knowledge Discovery and Information Retrieval (KDIR 2010), Valencia, 2010. — pp. 514–517. (Д. Д. Голомазову принадлежат результаты, посвященные разработанному им алгоритму определения семантической близости между временными и пространственными частями фактов, а также исследованию эффективности предложенного алгоритма).

[6] Афонин С. А., Голомазов Д. Д. *Использование семантических технологий для решения задачи обнаружения Grid-ресурсов* // Материалы II Научно-практической конференции «Актуальные проблемы системной и программной инженерии». — М.: Издательство МЭСИ, 2011. — С. 108–116. (Д. Д. Голомазову принадлежат результаты исследования и анализа существующих подходов к обнаружению грид-ресурсов, а также их практического применения).

[7] Афонин С. А., Голомазов Д. Д. *Выделение терминов из коллекции текстов с заданным тематическим делением* // Тезисы докладов научной конференции «Ломоносовские чтения». Секция механики. — М.: Издательство Московского университета, 2008. — С. 27–28. (Д. Д. Голомазову принадлежат результаты по построению математической модели и разработке алгоритма выделения терминов).

[8] Голомазов Д. Д. *Перспективы применения семантических технологий при построении информационных систем* // Тезисы докладов научной конференции «Ломоносовские чтения». Секция механики. — М.: Издательство Московского университета, 2010. — С. 61.

[9] Афонин С. А., Голомазов Д. Д. *Минимальные разложения регулярных языков на языки без объединения* // Тезисы докладов научной конференции «Ломоносовские чтения». Секция механики. — М.: Издательство Московского университета, 2009. — С. 22. (Д. Д. Голомазову принадлежат результаты по построению математической модели и доказательствам основных утверждений).