УДК 004.6

ИНТЕЛЛЕКТУАЛЬНЫЙ ПОИСК СЛОЖНЫХ ОБЪЕКТОВ В МАССИВАХ БОЛЬШИХ ДАННЫХ

А.М. Гусенков[1]

Институт вычислительной математики и информационных технологий Казанского (Приволжского) федерального университета

[1] gusenkov.a.m@gmail.com

Аннотация

Предложен подход к интеллектуальному поиску сложных объектов в различных типах структурно размеченных текстов, который может быть применен для обработки Больших данных (Big Data). Исследуются два вида представления информационных объектов: реляционные базы данных (РБД), которые структурно размечены своими схемами, и полнотекстовые естественнонаучные документы, содержащие математические выражения (формулы). Для таких полнотекстовых документов предлагается дополнительная автоматизированная разметка для организации поиска формул. В обоих случаях источником информации для построения онтологии и, в дальнейшем, организации поиска являются тексты на естественном языке, которые относятся к слабоструктурированным данным. Для РБД это комментарии к наименованиям таблиц и их атрибутов, а для естественнонаучных документов (статей, монографий и т. д.) – текстовое содержимое размеченных документов.

Ключевые слова*: большие данные, семантический поиск, слабоструктурированные данные, онтологии, реляционные базы данных, естественнонаучные тексты, разметка математических выражений.*

ВВЕДЕНИЕ

Экспоненциальный рост накопленных массивов данных, объём которых в настоящее время измеряется в зеттабайтах, привел к качественным изменениям в IT-технологиях сбора, хранения, управления, обработки и анализа информации. Одновременно в научный оборот вошло понятие Big Data (Большие данные).

Например, международная исследовательская компания Forrester [1] определяет это понятие как технологию в области аппаратного и программного обеспечения, которая объединяет, организует, управляет и анализирует данные, характеризующиеся «четырьмя V»: объемом (Volume), разнообразием (Variety), изменчивостью (Variability) и скоростью (Velocity):

- Volume – очень большой объем информации, накопленный в базах данных, трудоемко обрабатывать и хранить традиционными средствами СУБД;
- Variety – разнообразие форматов данных: способность приложения обрабатывать большие массивы данных, поступающие из разных источников в различных форматах, является главным критерием отнесения его к технологии Big Data; обычно такие приложения объединяют данные из разных источников (как внутренних, так и внешних по отношению к организации) и разной степени структурированности (структурированные, слабоструктурированные и неструктурированные); многие бизнес-задачи и научные эксперименты требуют совместной обработки данных различных форматов – это могут быть табличные данные в СУБД, иерархические данные, текстовые документы, видео, изображения, аудиофайлы и т. д.;
- Variability – изменчивость информации: например, информация, непрерывно поступающая с датчиков некоторых устройств или из интернета, имеющая важное значение для анализа, прогнозирования и принятия решений;
- Velocity – скорость накопления и обработки данных; в ряде задач востребованы технологии обработки данных в реальном времени.

В настоящее время разработаны технологии работы с Big Data, наиболее известными из них являются следующие:

- NoSQL [2] – ряд подходов к реализации хранилищ баз данных, имеющих отличия от моделей, используемых в РБД; их удобно использовать при работе с данными, структура которых не может быть жестко определена;
- MapReduce [3]– модель распределения вычислений, используется для параллельных вычислений при обработке очень больших наборов данных;
- Hadoop [4] – фреймворк с набором утилит и библиотек для разработки и выполнения распределённых программ, работающих на кластерах из сотен и тысяч узлов; используется для реализации поисковых и контекстных механизмов

высоконагруженных сайтов; система защищена от выхода из строя любого из узлов кластера путем дублирования данных на других узлах;

- SAP HANA [5] – высокопроизводительная NewSQL-платформа для хранения и обработки данных; сочетание технологий OLAP и OTLP в базе данных SAP HANA создает унифицированный ракурс данных, полученных из систем обработки транзакций, систем анализа, принятия решений и планирования.

Проблема интеграции гетерогенных электронных ресурсов чрезвычайно многоаспектна и многообразна. М.Р. Когаловский предложил следующую классификацию систем интеграции данных [6]: интеграция информации на физическом, логическом и семантическом уровнях. Интеграция на физическом уровне сводится к конверсии данных из различных источников в требуемый единый формат их физического представления. Интеграция на логическом уровне предусматривает возможность доступа к данным, содержащимся в различных источниках, в терминах единой глобальной схемы, которая описывает их совместное представление с учетом структурных и, возможно, поведенческих (при использовании объектных моделей) свойств данных. Интеграция на семантическом уровне обеспечивает поддержку единого представления данных с учетом их семантических свойств в контексте единой онтологии предметной области. Достоинство семантического подхода заключается в том, что основой пользовательского интерфейса является при этом высокоуровневая модель данных, а возможность рассуждений в терминах онтологии выступает в качестве концептуальной модели. В качестве средства описания онтологий обычно используется язык OWL, разработанный рабочей группой Semantic Web Activity и рекомендованный консорциумом W3C [7].

Одной из основных целей интеграции ресурсов является возможность организации эффективного поиска информации в интегрированных электронных ресурсах.

В настоящее время активно развиваются исследования в области компьютерной лингвистики и обработки информации, представленной на естественном языке. Это работы и прикладные системы, связанные с созданием электронных словарей, тезаурусов, онтологий, а также алгоритмы автоматического извлечения фактов из текста. В рамках этого направления разработано большое количество специализированных систем поиска в различных предметных областях.

В статье рассматривается единый подход к интеллектуальному поиску сложных объектов в различных типах структурно размеченных текстов, который может быть применен для обработки Big Data.

Исследуются два вида представления информационных объектов: реляционные базы данных (РБД), которые структурно размечены своими схемами, и полнотекстовые естественнонаучные документы, содержащие математические выражения (формулы). Для таких полнотекстовых документов предлагается дополнительная автоматизированная разметка для организации поиска формул. В обоих случаях источником информации для построения онтологии [8] и, в дальнейшем, организации поиска являются тексты на естественном языке, которые относятся к слабоструктурированным данным [9]. Для РБД это комментарии к наименованиям таблиц и их атрибутов, а для естественнонаучных документов (статей, монографий и т. д.) – текстовое содержимое размеченных документов.

ИНТЕГРАЦИЯ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ

Предлагается следующий подход для интеграции реляционных баз данных с целью организации эффективного поиска в РБД больших информационных систем, насчитывающих несколько десятков локальных баз данных с различными логической структурой и физической организацией, но относящихся к одной предметной области. Для интеграции используются информация, извлекаемая из самих баз данных, а также более общая информация, относящаяся к предметной области в целом. Таким образом, для успешного решения задачи интеграции РБД использованы вспомогательные информационные ресурсы, содержащие физические модели баз данных, логические модели предметной области и тезаурусы пользовательской терминологии, представленные в формализме онтологий.

ОНТОЛОГИЯ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ

Для эффективной интеграции реляционных баз данных при организации поиска необходима информация о структуре самой базы данных. Анализ структурных проблем интеграции РБД приведен в работах [10–12]. Связность и близость значений атрибутов РБД определяются связностью кортежей через общие ключи. Таким образом, для выдачи пользователю осмысленной информации по-

исковая машина должна иметь информацию о возможных соединениях кортежей по ключам. Такую информацию достаточно легко получить непосредственно из схем баз данных, так как современные СУБД хранят информацию о ключах в своих системных данных. Восстановление всех возможных осмысленных комбинаций соединения таблиц РБД по ключам является достаточно простой задачей. Кроме того, большинство современных СУБД содержит информацию о возможных соединениях таблиц по ключам РБД для поддержки целостности данных.

Для описания структуры реляционных баз данных существует достаточное количество формализмов: DDL SQL, ER-диаграммы [13], UML [14] и другие диаграммные техники. Их мощности вполне достаточно для решения задач проектирования баз данных и поддержки доступа к базам данных, рассматриваемых изолированно, каждая в своей собственной системе атрибутов. При решении задач интеграции информации из двух или более реляционных баз данных возникают проблемы [15], связанные с особенностями наименований артефактов баз данных, в первую очередь, таблиц и столбцов. Эти особенности требуют рассмотрения наименований артефактов не просто как атомарных имен, а как самостоятельных объектов, возможно обладающих собственной структурой и имеющих между собой семантически нагруженные связи. Такие связи невыразимы в традиционных формализмах представления структуры реляционных баз данных. Таким образом, мы не можем не только решить задачу интеграции реляционных баз данных, но и сформулировать ее.

Известны подходы к описанию структуры конкретных реляционных баз данных с помощью онтологий. В целом варианты представления структуры реляционных баз данных сводятся к двум основным подходам.

Первый подход [16] предполагает преобразование множества таблиц конкретной базы данных во множество одноименных концептов со слотами, соответствующими столбцам определенной таблицы, и проецирование связей между таблицами (в виде мигрирующих ключей) в отношения между концептами. В набор концептов онтологии также включается множество доменов (встроенных или пользовательских типов данных). Экземплярами концептов онтологии в данном случае выступают записи таблиц, значения ключей и состав доменов перечисляемого типа.

Второй подход, предложенный в работах [11, 12] и описанный в данной статье, предполагает более высокую степень абстракции концептов. При представлении структуры РБД выделяются универсальные концепты (не зависящие от конкретной базы данных): ТАБЛИЦА, СТОЛБЕЦ, КЛЮЧ, ДОМЕН, соответствующие основным объектам баз данных, и универсальные отношения между ними:

ТАБЛИЦА содержит СТОЛБЕЦ

ТАБЛИЦА имеет первичный КЛЮЧ

ТАБЛИЦА имеет внешний КЛЮЧ

КЛЮЧ содержит СТОЛБЕЦ

СТОЛБЕЦ имеет тип ДОМЕН

Объекты (таблицы, столбцы, ключи и домены) конкретной базы данных представляются как экземпляры универсальных концептов соответствующего типа.

Вводятся две функции интерпретации (ФИ) [8]:

ФИ1: Если ТАБЛИЦА1 имеет первичный КЛЮЧ1 и ТАБЛИЦА2 имеет внешний КЛЮЧ1, то существует ТАБЛИЦА3, содержащая столбцы, принадлежащие ТАБЛИЦА1 и ТАБЛИЦА2.

ФИ2: Если ТАБЛИЦА1 содержит СТОЛБЕЦ1, то существует ТАБЛИЦА2, содержащая все остальные столбцы ТАБЛИЦА1, кроме СТОЛБЕЦ1.

Первая функция интерпретации соответствует операции соединения по ключу, вторая – операции проекции реляционного отношения, необходимой для сокращения множества столбцов, получаемого при соединении таблиц, до искомого.

Для РБД задача интеграции сводится к нахождению способов извлечения заданных атрибутов (столбцов) из таблиц интегрируемых БД. Для задачи построения совместного представления это будет собственно целевая таблица, для задачи переноса информации – это таблицы целевой базы. Таким образом, задача сводится к нахождению такой последовательности применения ФИ1 и ФИ2, которая даст в результате искомое множество столбцов {C} из онтологии O, т. е. принадлежит известному классу задач о блуждании по ориентированному графу, где вершинами являются экземпляры концепта ТАБЛИЦА, а дугами – наличие общего ключа, ориентированного посредством отношений «имеет первичный» и «имеет

вторичный». Подходы к решению данного класса задач хорошо известны [17] и представляют лишь вычислительную сложность при большой размерности задачи.

Оба подхода имеют как преимущества, так и недостатки. В частности, первый подход позволяет сохранить в онтологии полный экстенсионал базы данных, во втором случае это нереализуемо. Вместе с тем, первый подход подразумевает создание для каждой конкретной базы данных уникальной онтологии, что не дает возможности построить универсальные аксиомы.

Отмеченные особенности получаемых онтологий определяют необходимость выбора второго подхода при построении онтологий больших баз данных для задач их интеграции, так как мы имеем дело с базами данных произвольной и заранее неизвестной структуры.

ОНТОЛОГИЯ ПРЕДМЕТНОЙ ОБЛАСТИ

Создание онтологии предметной области является трудоёмким процессом, который, как правило, выполняется командой высококвалифицированных специалистов, как в конкретной предметной области, так и в области компьютерной лингвистики. В этом направлении проводится большое количество исследований, в том числе и в Казанском университете, среди которых можно выделить следующие [35–38]. В настоящей статье обсуждается подход, предложенный в работах [11, 12, 18, 19, 34], а именно, методика построения онтологии предметной области на основе логической модели «сущность–связь», представленной в виде ER-диаграмм, для нефтедобывающей корпорации. В качестве прототипа онтологии предметной области выбрана логическая модель данных Epicentre нефтетехнической корпорации Petrotechnical Open Software Corporation (POSC) [20], имеющая статус отраслевого стандарта. Модель представлена в виде ER-диаграмм, а также набора текстовых файлов на языке EXPRESS. В модели данных Epicentre определено более 1000 реально существующих технических и бизнес-объектов, связанных с разведкой и добычей нефти. В терминологии POSC эти объекты названы сущностями (entities). В модели определены характеристики, которые могут содержать сущности, названные атрибутами сущностей (attributes). Наиболее важными являются атрибуты, определяющие взаимосвязи между сущностями. Объ-

ектно-ориентируемая концепция наследования классов является частью архитектуры Epicentre. Так как модель данных достаточно велика, концепция наследования классов обеспечивает эффективный способ организации всех сущностей в логически связанную структуру.

В подходе, рассматриваемом в статье, в качестве единого средства представления онтологий выбран язык OWL [21], разработанный рабочей группой Semantic Web Activity и рекомендованный консорциумом W3C, а именно, диалект языка OWL-DL (Description Logic). Выбор языка OWL-DL обусловлен чисто практическими аспектами, в частности, его поддержкой в существующих сегодня системах описания знаний и системах логического программирования. Разработана схема конвертации модели Epicentre в язык описания онтологий OWL. Применялись следующие основные подходы:

- любой сущности Epicentre соответствует простой именованный класс OWL-онтологии с сохранением в именах классов приставок, позволяющих идентифицировать сущности-свойства и сущности-справочники; все эти классы располагаются в корне таксономического дерева онтологии;
- степени связности сущностей (один-к-одному, один-ко-многим, многие-ко-многим) в OWL соответствует определение простых свойств-атрибутов, если связанная сущность не является типом данных, и свойств-значений в противном случае; указание степени связи между классами реализовано с помощью понятия кардинальности в OWL.

В OWL отсутствуют структурные элементы, которые в полном объеме описывают определение уникальности Epicentre. Поэтому в определение каждого класса на языке OWL добавлено новое предопределенное свойство, в котором перечислены все атрибуты, образующие уникальный ключ. Аналогичным образом решена проблема сохранения условий ограничений.

Для каждой категории данных Epicentre в OWL-онтологии построены отдельные классы, в свойствах которых использовались встроенные типы данных языка OWL. Построена формальная LR(1)-грамматика модели Epicentre, на основе которой реализовано семантическое преобразование модели Epicentre, описан-

ной на языке EXPRESS, в онтологию на языке OWL. Выполнена русификация описания сущностей и атрибутов модели Epicentre, а также соответствующих им классов и свойств на OWL.

ЛИНГВИСТИЧЕСКИЙ ТЕЗАУРУС

Для создания лингвистической онтологии природно-технических объектов выбран подход, основанный на построении тезаурусов WordNet [22]. Словарь предметной области построен путем объединения словоформ из описаний сущностей и атрибутов модели Epicentre со словоформами из описаний атрибутов таблиц и доменов таблиц-справочников реляционных баз данных нефтедобывающей корпорации ОАО «Татнефть». Лексико-семантические характеристики баз данных, использованных при построении словаря, описаны в [11, 23]. Словарь содержит около 6000 словоформ. При построении тезауруса для каждого слова определен входной синонимический ряд (синсет). На лексико-семантических вариантах слов и синсетах определены следующие отношения: гипонимия, часть – целое, несовместимость, антонимия, конверсивность, омонимия.

Описаниям синсетов построенного таким образом тезауруса присущи особенности, вытекающие из специфики описания атрибутов реальных баз данных: наличие коротких фраз, сокращений, технических аббревиатур, орфографических ошибок. Однако в отличие от анализа сплошных текстов для баз данных имеется возможность уточнения распознавания входных слов путем просмотра содержимого доменов атрибутов таблиц и сопоставления их с онтологией предметной области.

ПОИСК НА ЕСТЕСТВЕННОМ ЯЗЫКЕ. ГЕНЕРАЦИЯ SQL-ЗАПРОСОВ

В данном разделе излагается методика автоматической генерации SQL-запросов, требующая от пользователя знаний только в своей предметной области. Данный подход использует онтологию предметной области, экземпляр онтологии РБД и тезаурус пользовательской терминологии, описанные в предыдущих разделах. Следует отметить, что автоматизация построения онтологии предметной области и языка предметной области (лингвистического тезауруса), а также способ создания экземпляра онтологии произвольной РБД обуславливают применимость данного подхода к его использованию технологиями сбора, хранения

и обработки информации больших данных [1, 2]. Автоматизация построения перечисленных объектов интеграции позволяет строить системы, более устойчивые к модификациям в процессе их эксплуатации. При изменениях логической модели предметной области онтологию предметной области можно перестроить, а лингвистический тезаурус пополнить. Все это относится не только к нефтедобывающей предметной области, но и к любой другой, имеющей представление в виде ER-диаграмм логической модели «сущность–связь».

Из существующих реализаций можно выделить систему InBase, разработанную школой А.С. Нариньяни [24], как наиболее близкую к предложенному подходу. Отличительной особенностью InBase является применение опережающего семантического анализа при разборе и понимании запросов. Разбор производится на основе объектной модели предметной области, которая привязывается проектировщиком к модели РБД. Данный подход обладает рядом существенных преимуществ по сравнению с применением синтаксических шаблонов. В частности, он избавляет от необходимости задавать в шаблонах запросов все возможные словоформы и порядки следования слов в запросе. Вместе с тем, следует признать, что InBase не перешла в стадию широкого применения.

Как нам представляется, подход, развиваемый в системе InBase, имеет ряд проблем. Наиболее существенной из них является трактовка столбцов РБД как некоторых атрибутов классов объектной модели предметной области. В реальных базах данных довольно часто используются конструкции, которые не могут быть адекватно отображены на объектные модели, так как имеют смысл скорее подзапроса, чем классического атрибута. Этот вопрос подробно рассмотрен в [15, 23], поэтому ограничимся одним примером. Так, в одной из исследованных промышленно эксплуатируемых РБД встречались столбцы с комментариями «Дебит скважины до ремонта» и «Дебит скважины после ремонта». Очевидно, что такие конструкции в объектных моделях соответствуют не атрибуту класса, а достаточно сложному выражению.

Второй проблемой является необходимость ручной привязки столбцов к атрибутам объектной модели. Помимо большой сложности этого процесса при первоначальном внедрении данной системы, это создает сложности при ее сопро-

вождении. При изменении РБД, что в реально эксплуатируемых системах – достаточно частый случай, требуются квалифицированная корректировка объектной модели и ее проекция на столбцы таблиц РБД.

В корпоративных информационных системах РБД могут содержать сотни таблиц и тысячи столбцов. Изучение вопроса, в какой конкретно РБД и в каких конкретно столбцах и таблицах содержится требуемая информация, может занять несколько дней, тогда как формулировка собственно запроса на естественном языке потребует нескольких минут.

В предлагаемом подходе обработка информации строится на следующих принципах:

- ведение диалога между системой и пользователем в табличном виде;
- использование семантических подходов для поиска столбцов;
- автоматический поиск возможных соединений по ключам БД;
- использование визуальных процедур для задания операций селекции.

Пользователь формулирует запрос в виде таблицы на естественном языке, используя термины тезауруса (языка предметной области). В табл. 1. приведён пример модельного запроса. Очевидно, что подобный запрос, сформулированный в виде предложения на естественном языке, был бы весьма громоздким не только для машинного анализа, но даже для понимания человеком.

Название столбца	Условие
Номер скважины	10*
Дата ввода в эксплуатацию	> 1.06.2001
Дата КРС	
Дебит нефти ожидаемый	>0
Дебит нефти фактический	

Таблица 1. Модельный запрос

Семантическая информация представлена в системе в виде семантической сети (онтологии) предметной области, которая содержит набор возможных связей между концептами. Операция семантического разбора комментариев к названиям столбцов РБД почти полностью автоматическая, что позволяет доста-

точно быстро привязать к предметной области большую корпоративную базу данных. Семантический разбор запроса пользователя заключается в выявлении подграфов на семантической сети, которые связывают концепты, именуемые словами и словосочетаниями, употребленными в названии столбцов. На искомые подграфы наложено условие минимальности либо по количеству связей, либо по их суммарной длине.

Поиск столбцов БД, наиболее релевантных названиям, задаваемым пользователем в своем запросе, производится путем сопоставления подграфов семантического разбора столбцов запроса и столбцов БД. При сопоставлении могут использоваться различные метрики, например, разность подграфов или расстояние между подграфами.

Для выявленных релевантных кандидатов, которые могут содержаться в различных таблицах, автоматически выявляются подмножества, для которых возможно построение соединений по ключам. Это алгоритмически несложная, но ресурсоемкая операция, в результате которой среди кандидатов выявляются столбцы, имеющие между собой связь по миграции ключей. Если комбинаций связанных релевантных столбцов возникает несколько, то дополнительно учитывается количество записей в построенном соединении. Построенный запрос предъявляется пользователю для задания дополнительных условий в режиме визуального конструктора.

Предложенная в статье методика и реализованная по этой методике система прошли тестирование на реальных базах данных ОАО «Татнефть». По большинству несложных запросов система показала достаточно хорошую релевантность результатов.

Реактивность системы сильно зависит от количества столбцов, задаваемых в запросе. Основное время тратится на поиск связей столбцов в БД, которые выполнялись в данной реализации простым перебором. Таким образом, предлагаемая технология в текущей реализации эффективно работает с запросами, содержащими небольшое количество столбцов, что вполне достаточно для запросов справочного характера. Для ограничения вариантов перебора соединений таблиц

используются связи, сохраненные в контексте пользователя, что позволяет повысить реактивность системы. Более подробно система описана в работах [11, 12, 25].

ПОИСК В ЕСТЕСТВЕННОНАУЧНЫХ ТЕКСТАХ, СОДЕРЖАЩИХ МАТЕМАТИЧЕСКИЕ ВЫРАЖЕНИЯ (ФОРМУЛЫ)

Подход, представленный в настоящей работе, направлен на интеграцию функциональных возможностей полнотекстового поиска и поиска по математическим формулам, при котором конечному пользователю предлагается формулировать поисковый запрос на поиск математической формулы в форме ключевых слов или словосочетаний (именных групп). Наиболее близкий подход предложен в математической поисковой системе EgoMath (доступна по URL: http://egomath.projekty.ms.mff.cuni.cz), которая в данный момент предоставляет возможности традиционного формульного поиска в синтаксисе LATEX и механизм переформулирования запроса (от ключевых слов к символьным обозначениям), однако алгоритм этого связывания не раскрыт в оригинальной статье авторов EgoMath [26]. Достаточно близкие подходы были предложены в работах [38–40]

Новизна подхода, предлагаемого в данной статье, состоит в том, что в качестве поискового объекта рассматривается сложный нелинейный нетекстовый объект, включающий собственно математическое выражение формулы в нотации LATEX и набор определений символьных обозначений, участвующих в математическом выражении, которые извлекаются из всего анализируемого текста [12, 28, 29].

Данная постановка позволяет рассматривать предлагаемый подход как новый тип запросов к коллекциям математических документов. Действительно, в отличие от известной задачи поиска математической формулы по ее фрагменту в формулировке запроса должны использоваться не математические конструкции, а словесные наименования переменных, входящих в искомую формулу. В предлагаемом подходе используется дополнительная разметка естественнонаучных текстов, смысл которой заключается в следующем.

В естественнонаучных текстах выделяются виды сущностей: естественнонаучные термины, символьные условные обозначения терминов (переменные), ма-

тематические фрагменты (формулы). Для них определяются отношения: «термины – переменные» и «переменные – формулы». Первое отношение есть текстовое определение значения символа в некотором контексте с помощью терминов, второе отношение указывает на вхождение символа в формулу. Предполагается, что появление текстового определения переменной в окрестностях её символьного представления указывает на семантическую связь между ними.

Все перечисленные сущности и отношения между ними составляют контекст формулы.

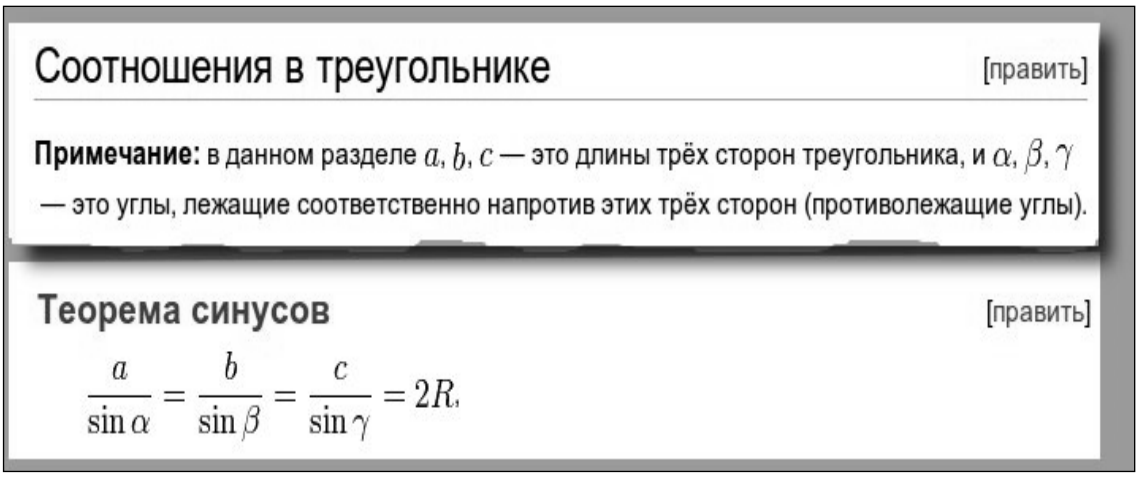

Рис. 1. Пример формульного контекста (фрагмент статьи Википедии)

Пример расширенного формульного контекста приведен на рис. 1, где определения переменных даны в сплошном тексте, а формула является нетекстовым объектом.

Введем метод разметки математических выражений.

Шаг 1. Классификация математических выражений (МВ). МВ считается любой текст между специализированными тегами разметки $и$ (рис. 2). В качестве инструмента анализа использован язык регулярных выражений. В МВ выделяются: символы арифметических и логических операций, переменные, переменные с индексами, ключевые слова, числа. Если МВ содержит только одну переменную или переменную с индексом, то оно классифицируется как переменная. Иначе МВ классифицируется как формула.

Рис. 2. Структура математической формулы в Википедии

Шаг 2. Связывание формул с переменными. Во всем анализируемом тексте для каждой переменной производится поиск вхождения этой переменной в каждую формулу. Пусть {F} – множество формул, а {P} – множество переменных. Для $\forall p_i \in P$, если $p_i \subset f_k \in F$, устанавливается отношение $<p_i,f_k>$. Для каждого отношения в качестве атрибута запоминаются позиции формул и переменных в тексте.

В результате разметки получаем отношение много-ко-многим между формулами и переменными, входящими в состав этих формул.

ПОИСК В ВИКИПЕДИИ

Для проверки базовых концепций предложенного выше метода разметки математических выражений была реализована система семантического поиска математических выражений, специализированная для поиска математических формул в статьях русскоязычной Википедии. Выбор Википедии связан, с одной стороны, с тем, что последняя является одной из крупнейших коллекций научных текстов из различных областей знаний, с другой стороны, математические выражения в Википедии имеют унифицированную форму представления.

Для поиска формул в статьях Википедии была выполнена программная реализация предложенного подхода. Пользовательский интерфейс системы представляет собой веб-приложение, доступ к которому осуществляется через любой современный браузер с поддержкой JavaScript.

На странице ввода запроса пользователь может указать в текстовых полях одно или более названий параметров, которые должны присутствовать в искомой формуле, и после этого инициировать поиск.

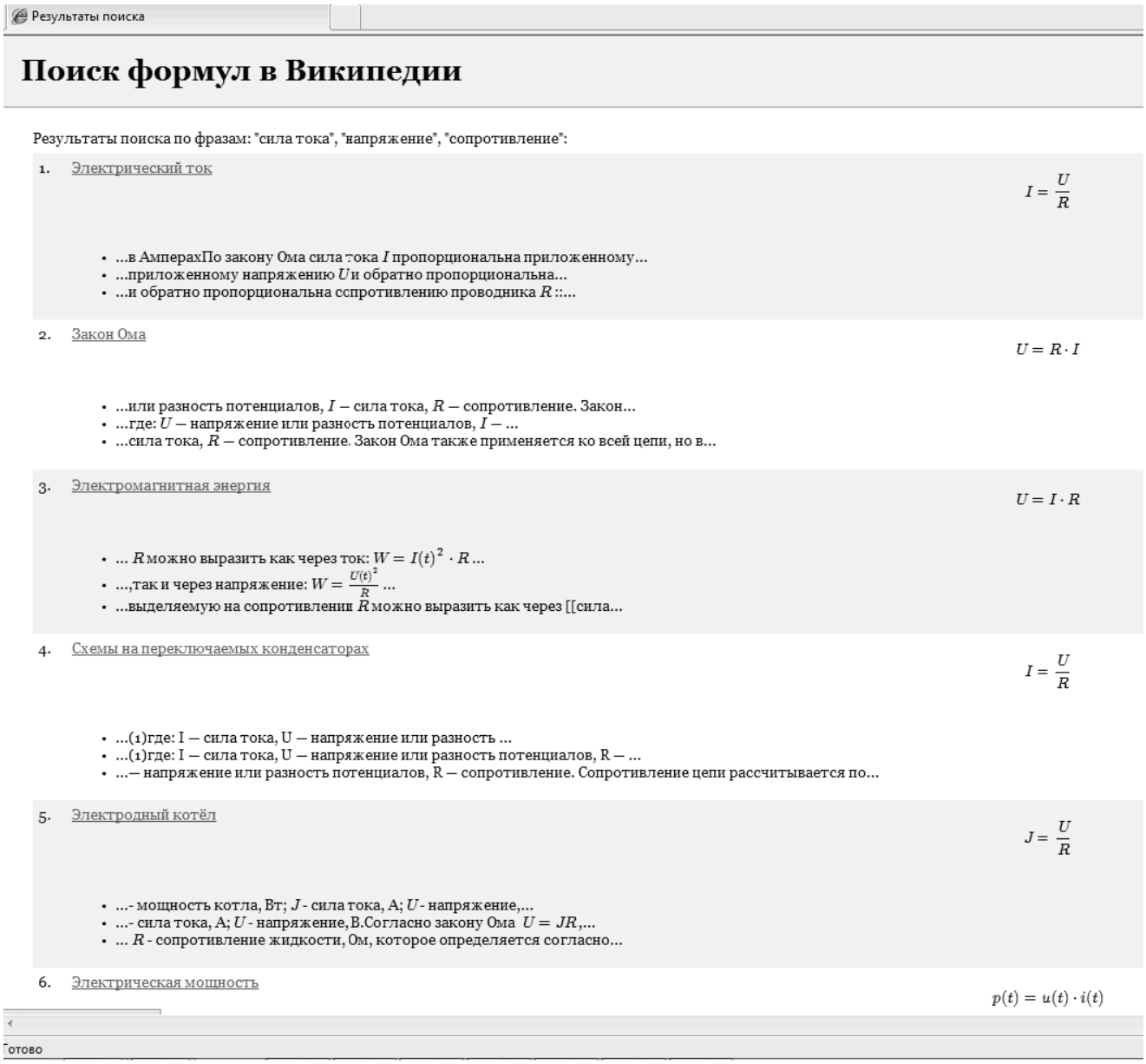

Рис. 3. Закон Ома. Поиск по параметрам: сила тока, напряжение, сопротивление

Результаты поиска отображаются на странице результатов, где выводятся запрос, а также ранжированный по релевантности список ссылок на страницы Википедии. Результаты поиска представляются наборами фрагментов, содержащих нетекстовый объект (формулу), и фрагментов, связывающих переменные формулы и их текстовые определения вне зависимости от их местонахождения в тексте (см. рис. 3).

Система поиска в Википедии включает следующие взаимосвязанные подсистемы: загрузки и анализа данных Википедии; полнотекстового индексирования; индексирования математических формул и переменных; поиска и ранжирования.

Производится импорт данных русскоязычной Википедии из предварительно полученного архива. Единицей анализируемой и загружаемой информации является html-страница. Страницы, содержащие формулы, сохраняются в базе данных для дальнейшего использования, остальные отбрасываются.

В системе производятся индексирование входных документов как текстовых данных (библиотека Apache Lucene [27]), а также дополнительное позиционное индексирование, построенное в результате применения метода разметки математических выражений.

Поиск и ранжирование выполняются в два этапа. На первом этапе производится полнотекстовый поиск всех вхождений ключевых словосочетаний (терминов) в тексты. Для каждого вхождения определяется, существует ли в некоторой окрестности ключевой фразы переменная. Для определения окрестности введено понятие максимально допустимого расстояния (МДР) – расстояния в символах влево и вправо от термина, в пределах которого может находиться переменная. По найденным переменным определяется соответствующая формула. Для каждой формулы строится группа текстовых фрагментов, включающих термины и переменные.

На втором этапе производится поиск наилучшей группы текстовых фрагментов для всей совокупности введенных ключевых фраз. Для этого составляются и проверяются по критерию близости все возможные сочетания полученных текстовых фрагментов в документе. В качестве критерия близости использован минимум среднеквадратичного отклонения найденных фрагментов с определениями переменных от позиции соответствующей формулы. В результате для каждого документа получим оптимальную группу текстовых фрагментов (потенциальных определений) и относящуюся к ним формулу. Результаты для всех документов сортируются по критерию близости и дополнительным критериям релевантности, связанным с полнотой вхождения выделенных переменных в формулу.

Реализованный в системе алгоритм поиска математических формул по введенным ключевым фразам (названиям параметров формулы) показал достаточную релевантность в сочетании с высокой скоростью поиска. Проведенное тестирование выявило, что выдаваемые результаты практически всегда имеют непосредственное отношение к задаваемому запросу, и на первой странице поиска находится формула, отвечающая запросам пользователя. Для дальнейшего улучшения релевантности необходимо модифицировать механизм поиска, а не механизм ранжирования.

Наиболее очевидным путем представляется синтаксический анализ текста с целью выделения терминологических словосочетаний – именных групп (ИГ) – и дальнейшего анализа отношения выделенных ИГ и заданных пользователем наименований параметров, также являющихся ИГ. С этой целью был разработан подход к разметке для поиска по онтологии с использованием связывания формул с ИГ.

РАЗМЕТКА ДЛЯ ПОИСКА ПО ОНТОЛОГИИ

Для организации поиска в связной коллекции математических текстов автоматизировано строится онтология, включающая в себя, кроме самих компонентов онтологии, связанные с ними переменные и формульные выражения [30, 31]. Поисковый запрос пользователя на естественном языке переводится в термины онтологии, по которым затем формируется запрос на языке запросов к RDF-документам SPARQL [32]. Автоматизация построения онтологии упрощает пополнение коллекции математических текстов, что играет важную роль при работе с Big data [1, 2].

В данной статье рассматривается только дополнение связной коллекции математических текстов, размеченных для построения онтологии, разметкой формул и входящих в них переменных, а также их связей с ИГ [12, 29, 33].

Модуль формульной разметки реализован на языке Java в формате плагина к текстовому процессору GATE [41], для разметки используются средства работы с аннотациями библиотеки Gate и оригинальные алгоритмы.

Формульная разметка применяется к XML-документу, предварительно размеченному стандартными аннотациями (Token, Sentence, Math и др.) и NLP-аннотациями (TERM, ENDS). Ключевыми аннотациями для работы алгоритма являются аннотации Math, размечающие формульные фрагменты, и аннотации TERM, соответствующие именным группам.

На основе аннотаций Math строится внутренняя модель документа, содержащая набор разобранных, классифицированных, связанных между собой формульных фрагментов.

Обработка XML-документа включает в себя следующие действия:

- выделение и анализ формульных фрагментов;
- определение связей между переменными и формульными фрагментами;
- определение связей между формульными фрагментами и ИГ;
- дополнение аннотаций Math атрибутами формульной разметки.

В первых двух действиях использован метод разметки математических выражений.

На третьем шаге вводится понятие максимально допустимого расстояния (МДР) между аннотациями Math и TERM, которое определяется как наибольшее расстояние в символах между концом левой аннотации и началом правой, при котором может быть выполнено связывание. МДР является параметром, который оказывает непосредственное влияние на точность связывания и может различаться для разных коллекций документов.

В документах встречается различное взаимное расположение формул и ИГ:

- ИГ содержит формулу, тогда она – единственный кандидат для связывания. В простейшем случае ИГ состоит из единственного главного слова. В более сложном случае она содержит более одного слова, и рассматривается расстояние между формулой и главным словом. Если это расстояние составляет более трех символов, формула считается дополнением и не связывается;
- Формула и ИГ следуют друг за другом (в пределах одного предложения). В этом случае основой анализа является концепция МДР. Алгоритм позволяет связывать формулу только с одной именной группой, но с одной и той же ИГ может быть связано более одной формулы.

На заключительном этапе связи, построенные на внутренней модели, переносятся в обрабатываемый документ и используются при построении онтологии связанной коллекции документов.

Рассмотрим вышесказанное на примере аннотирования математического текста, изображенного на рис. 4.

Пусть $\overline{\alpha}$ — вторая фундаментальная форма n-поверхности $\overline{M}$, $\overline{\nabla}$ — связность Леви-Чивита метрики $\overline{g}$. Имеет место [1] равенство

$$\partial_X df Y - df \overline{\nabla}_X Y = \overline{\alpha}(X, Y), \qquad (2)$$

Рис. 4. Фрагмент математического текста

В результате обработки документа, фрагмент из которого приведён на рис. 4, аннотации, соответствующие фрагменту, будут дополнены атрибутами формульной разметки (выделены жирным шрифтом):

Math Id=960 mode=inline, **termid=965**, tex=\overline{\alpha}, text=overline@(alpha), **varid=3**, xml:id=p10.m1

TERM Id=965 Form=вторая фундаментальная форма $$$-поверхности $$$, HeadBegin=0, HeadEnd=27

Math Id=966 mode=inline, tex=n, text=n, **varid=10**, xml:id=p10.m2

Math Id=969 mode=inline, **termid=965**, tex=\overline{M}, text=overline@(M), **varid=0**, xml:id=p10.m3

Math Id=974 mode=inline, **termid=979**, tex=\overline{\nabla}, text=overline@(nabla), **varid=8**, xml:id=p10.m4

TERM Id=979 Form=связность Леви-Чивита метрики $$$, HeadBegin=0, HeadEnd=20

Math Id=980 mode=inline, tex=\overline{g}, text=overline@(g), **varid=7**, xml:id=p10.m5

TERM Id=987 Form=равенство $$$, HeadBegin=0, HeadEnd=8

Math Id=989 mode=display, **termid=987**,

tex=\partial _{{X}}dfY-df\overline{\nabla}_{{X}}Y=\overline{\alpha}(X,Y),,

text=(partial-differential _ X)@(d * f * Y) - d * f * (overline@(nabla)) _ X * Y = overline@(alpha) * open-interval@(X, Y), **vars=3;8**, xml:id=S0.E2.m1

Выделенная в тексте формула очищается от служебных символов языка разметки и лишних пробельных символов. Затем формульный фрагмент разбивается на элементы: разделителями считаются различные символы скобок, символы арифметических и логических операций, знаки пунктуации, пробельные символы и т. п. Полученные элементы анализируются на принадлежность к специальным группам – ключевые слова (начинаются с символа «\»), нижние индексы (начинаются с символа «_»), числа и т. п. Если элемент на этом этапе не классифицирован, то с большой долей вероятности его можно считать переменной. К таким элементам дополнительно применяется проверка на соответствие правилам именования переменных – не начинается с цифры, может быть буквой греческого алфавита (например, “\alpha”), может содержать индекс. В результате все формульные фрагменты документа будут разделены на три типа: переменные, формулы, служебные (содержащие только символы разметки).

Одиночным переменным, выявленным в тексте, сопоставляются уникальные идентификаторы, которые будут использованы при дополнении аннотаций. Следует отметить, что алгоритм различает строчные и прописные буквы (X и x), индексированные и неиндексированные переменные (Y и Y_i), при этом варианты обозначения индекса не сказываются на идентификации переменной (Y_0 и Y_{i+1} определяются как одна и та же переменная). Для переменных, входящих в состав формульных фрагментов, строится индекс вхождения переменных в формулы. Для формул также определяются связи между фрагментами вида “содержит” и “содержится в”. Служебные фрагменты не рассматриваются как не несущие смысловой нагрузки.

Так, фрагмент текста на рис. 4 содержит переменные \overline{\alpha}, n, \overline{M}, \overline{\nabla}, \overline{g} и формулу, в которую входят идентифицированные переменные \overline{\alpha}, \overline{\nabla}. Переменные X, Y не присутствуют в тексте вне формул, поэтому идентификаторы им не сопоставлены.

Формулы и именные группы могут идти последовательно друг за другом, не пересекаясь. Другой вариант – когда формула содержится внутри именной группы. Частичное пересечение формул и именных групп не встречается, так как это противоречило бы структуре XML. Алгоритм связывания построен с учётом особенностей взаимного расположения формул и именных групп.

Для каждого формульного фрагмента с помощью средств работы с аннотациями библиотеки Gate определяется тип расположения и набор именных групп – кандидатов на связывание, из которых по заданным критериям отсеиваются неподходящие и отбираются наиболее близкие.

Если именная группа содержит формулу внутри себя, она становится единственным кандидатом на связывание. В простом случае, когда именная группа не содержит других слов, кроме главного слова, связывание будет проведено с высокой степенью достоверности. Например, в примере, приведённом на рис. 4, во втором предложении выделена ИГ “равенство $$$”, где “$$$” обозначает вхождение соответствующей формулы в именную группу. В данном случае формула, обозначенная (2), будет связана с ИГ “равенство $$$”.

В более сложных случаях анализируется расстояние между формулой и главным словом именной группы (атрибуты HeadBegin, HeadEnd аннотации TERM). Если оно оказывается больше допустимого интервала между словами (3 и более символа), считается, что формула является дополнением к основному понятию именной группы. В этом случае связывание не производится. Например, переменная \overline{g} из текста на рис. 4 не будет связана с ИГ “связность Леви-Чивита метрики $$$”, так как главное слово здесь – “связность Леви-Чивита”. Также не связываются формулы, входящие в конструкции с дефисом, как не имеющие самостоятельного значения. Например, для фразы “вторая фундаментальная форма n-поверхности \overline{M}” формула “n” останется не связанной, а формула “\overline{M}” будет связана с ИГ “вторая фундаментальная форма $$$-поверхности $$$”.

Если формула не содержится внутри именной группы, задача определения набора ИГ, с которыми возможно связывание, усложняется. Сначала определяются границы области, в которой должны располагаться аннотации-кандидаты. За левую границу принимается позиция в документе, соответствующая ближайшей левой аннотации ENDS и отстоящая от начала формулы влево не более чем на МДР. Аналогично за правую границу принимается позиция в документе, соответствующая ближайшей правой аннотации ENDS и отстоящая от конца формулы вправо не более чем на МДР. Кроме того, если формула входит в группу равенств

(аннотация equationgroup), отсчёт ведётся от начала и конца всей группы. Это необходимо, чтобы все уравнения группы были привязаны к одной и той же ИГ.

По границам области связывания определяются левый и правый наборы аннотаций TERM, из которых затем выбирается одна аннотация, находящаяся на минимальном расстоянии от формулы. Для повышения достоверности связывания при обнаружении распространённой в математических текстах конструкции “<формула> – <ИГ>” приоритет отдаётся правому набору (например, определение \overline{\alpha} и \overline{\nabla} на рис. 4).

Следует отметить, что алгоритм позволяет связывать формулу только с одной именной группой, но с одной и той же ИГ может быть связано более одной формулы. Это позволяет учитывать перечисления формул, относящихся по семантике к одной ИГ, но в то же время может давать некоторое количество недостоверных связываний.

На заключительном этапе связи, построенные на внутренней модели, переносятся в обрабатываемый документ. В аннотации Math в зависимости от типа формульного фрагмента и наличия связанных ИГ добавляются новые атрибуты. Обозначим через <variable_id> уникальный идентификатор переменной, присвоенный ей во внутренней модели документа. Отметим, что всем вхождениям какой-либо переменной в формульные фрагменты любого типа будет соответствовать один и тот же идентификатор. Переменные, встречающиеся только в составе сложных формул, идентификатора не имеют. Для формульной разметки используются следующие атрибуты:

varid=<variable_id> – идентификатор переменной, добавляется в аннотации к одиночным переменным;

vars=<variable_id1>;<variable_id2>;... – список идентификаторов переменных, входящих в формулу, добавляется в аннотации к формулам;

termid=<annotation_id> – идентификатор аннотации TERM в документе, соответствующей именной группе, с которой связана формула, добавляется к переменным и формулам при наличии связи с ИГ.

Реализованный модуль формульной разметки был протестирован на коллекциях XML-документов, предварительно размеченных стандартными аннота-

циями GATE. Была проведена оценка релевантности и полноты связывания математических выражений и именных групп, выявленных в тексте, которая основана на экспертной оценке качества связывания на корпусах математических текстов. В качестве документов для разметки использовались статьи журнала «Известия вузов. Математика» за 1997–2009 гг. Результаты анализа связывания при изменении МДР от 15 до 40 симметрично в обе стороны представлены в табл. 2.

МДР	Math	Terms	VirOK%	NotVirOK%	TotalOk%	VirBad%	Others%	TotalBad%
15	1247	1357	36,33	30,47	66,80	23,90	9,30	33,20
20	1247	1357	42,34	25,50	67,84	25,66	6,50	32,16
25	1247	1357	40,98	20,69	61,67	23,02	15,32	38,33
30	1247	1357	41,38	21,49	62,87	27,83	9,30	37,13
35	1247	1357	41,86	21,01	62,87	29,03	8,10	37,13
40	1247	1357	42,02	19,65	61,67	29,67	8,66	38,33

Таблица 2. Статистика связывания в зависимости от МДР

Для каждого заданного значения МДР на всем корпусе текстов определялись следующие параметры: Math – количество выделенных формул; Terms – количество выделенных ИГ; VirOK – процент правильных связываний формул с ИГ (полнота связывания); NotVirOK – процент правильных несвязываний формул с ИГ (т. е. констатация того факта, что математическое выражение находится в контексте, не содержащем его определения); TotalOk – общий процент правильно обработанных формул (сумма VirOK и NotVirOK); VirBad – процент неправильных связываний формул с ИГ (из возможных кандидатов на связывание была выбрана не подходящая по семантике ИГ или произошло связывание математического выражения в контексте, не предполагающем связывания); Others – другие ошибки связывания (отсутствие связывания там, где оно должно было быть; неправильное выделение ИГ; нераспознанные ИГ; влияющие на связывание особенности оформления текста автором); TotalBad – общий процент неправильно обработанных формул (сумма VirBad и Others).

Из табл. 2 видно, что процент правильно обработанных формул TotalOk и процент ошибок всех типов TotalBad изменяются незначительно, что свидетельствует об устойчивости применяемого алгоритма. Тем не менее, процент правильно связанных математических выражений VirOK имеет тенденцию к возрастанию с увеличением МДР, что вполне ожидаемо. Общий процент ошибок связывания TotalBad также растет с увеличением МДР. Вместе с тем, изменения этих параметров имеют нелинейную зависимость, что позволяет сделать выбор оптимального МДР, при котором отношение VirOK/TotalBad максимально. Для данного корпуса текстов оптимальное МДР составляет 20 символов.

В целом описанные реализации показали принципиальную работоспособность и хорошую устойчивость результатов при применении предложенных подходов, а также выявили ряд проблем с релевантностью поиска. Для повышения релевантности и полноты связывания можно определить следующие направление дальнейшего развития предлагаемого подхода: дополнительный анализ контекста документа; разработка шаблонов, типичных для математических текстов; использование Байесовских методов; обучение распознаванию ключевых слов и конструкций на основе экспертного связывания.

ЗАКЛЮЧЕНИЕ

Рассмотренные подходы реализации поиска сложных объектов в структурно размеченных текстах, опыт их реализации и проведенные эксперименты показали, что такие тексты представляют собой многосвязные графы, неявно отражающие семантику предметной области.

Эксперт-пользователь, формулирующий запрос, знает, что между введенными им ключевыми словами должна быть некоторая связь, благодаря чему в достаточно большом наборе данных такие связи находятся и имеют содержательный смысл. Этим, вероятно, объясняется высокая эффективность простого переборного механизма поиска, реализованного в описываемом подходе. Таким образом, как структура хранения текстов, так и сформулированный запрос несут в себе неявную семантику, комбинация которых позволяет поисковому механизму найти полное и достаточно релевантное множество ответов, не имея априорной информации о предметной области. Вместе с тем, рассматриваемый поисковый механизм имеет несколько очевидных направлений дальнейшего развития.

Интересным частным результатом является фактическое выявление, при анализе особенностей структуры транзакционных баз данных, нового механизма нормализации данных. Известные нормальные формы, включая наиболее общую доменную нормальную форму, предложенную R. Fagin в 1981 году, не меняют состава атрибутов базы данных, а лишь перераспределяют их по кортежам. Вместе с тем, в реальных базах данных достаточно часто ключ с малым числом значений разворачивается в группу имен атрибутов или таблиц, т. е. множество значение атрибута и множество атрибутов могут трансформироваться друг в друга. Дефакто такая практика распространена в реальных структурах транзакционных БД, что нашло отражение в некоторых практических конструкциях механизмов хранения промышленных БД, например, сегментация объемных таблиц по ключу, поддерживаемая в Oracle, PostgreSQL и ряде других систем. До некоторой степени эта практика легитимизирована в хранилищах, при интеграции данных из различных источников, при версионности данных, их непрогнозируемом изменении во времени и ряде других случаев. Для поддержания хотя бы первой нормальной формы все атрибуты кортежей, обладающих данными особенностями, хранятся в отдельных таблицах с дополнительными атрибутами времени, источника, а также первичным ключом родительского кортежа. В результате форма хранения ряда атрибутов приобретает совершенно унифицированный вид и их можно хранить в одной таблице, введя дополнительный атрибут «Имя атрибута». Довольно часто такой прием используется при хранении группы однотипных по структуре справочников. Таким образом, практика трансформации имени атрибута в значение его супертипа существует, но в теоретическом плане пока получила слабое освещение. На наш взгляд, детальное изучение этого феномена может дать новый вклад в теорию реляционных баз данных.

Применительно к рассматриваемой теме дальнейшая формализация данного механизма может дать ключ к автоматизации построения запросов и репликации данных между транзакционными системами с разной структурой баз данных, а также с хранилищами, многомерными OLAP системами и другими форматами хранения, в том числе на базе XML. Весьма актуально может быть развитие данного направления для поиска в полнотекстовых базах данных, содержащих табличные данные. Учитывая возрастающий интерес к развитию направления Big

Data, для которых в принципе характерна разнородность и разноуровневость информации, формализация поисковых механизмов, учитывающая возможные преобразования имен атрибутов в их значение, может иметь хорошие перспективы дальнейшего развития.

Другим интересным результатом, вытекающим их рассмотренного подхода, является применение комбинаторного механизма к поиску математических зависимостей между параметрами, для которых в анализируемом корпусе текстов нет формулы, содержащей все заданные в запросе параметры. Пока реализован поисковый механизм для формул, ищущий явно прописанную формулу, объединяющую заданные в запросе параметры. В конструкторе запросов к базам данных в сконструированном запросе может участвовать несколько отношений, связь между которыми извлекается из схемы БД. Перенос данного подхода на поиск формул может дать фактически генератор новых знаний в виде цепочки формул, выстраивающих такие связи между переменными запроса, которые семантически подразумеваются, но в анализируемом корпусе текстов не присутствуют явно. Не исключено, что комбинаторный подход позволит получать такие цепочки вывода без предварительного обучения. Возможным препятствием к применению данного подхода является его большая комбинаторная сложность, однако непрерывно возрастающие вычислительные мощности и, возможно, некоторые новые закономерности в естественнонаучных текстах, полученные в новых исследованиях, позволят преодолеть этот барьер.

Неиспользованным резервом является подключение к комбинаторным механизмам поиска путей на графах статистических подходов. Очевидно, что в большом корпусе текстов будет содержаться несколько вариантов искомой информации. Статистический анализ, примененный к множеству результатов, позволяет решать несколько задач. В первую очередь можно повысить релевантность выдаваемой информации путем статистического анализа результатов, отсеивая редко встречающиеся варианты. Это позволит работать на недостаточно выверенных текстах, так как редко встречающиеся варианты скорее всего ошибочны. Опыт работы с естественнонаучными текстами показывает, что для обозначения конкретных физических и технических параметров применяется ограниченный набор имен переменных. Так, например, буквой P может быть обозначено давление или

объем промышленного производства, но никогда не обозначаются расстояние или денежный агрегат. Статистический анализ имен переменных, ассоциированных с запрашиваемыми параметрами, и их сочетаемость позволят, вероятно, еще более повысить релевантность. Статистический анализ сочетаемости также, возможно, позволит автоматизировать построение онтологий предметной области и получить ряд других полезных результатов.

Таким образом, комбинаторный поиск на графах, неявно задаваемых структурой хранения данных, обладает определенной универсальностью в силу того, что он позволяет решать различные поисковые задачи, и является развиваемым, так как имеет очевидные перспективы объединения с лексико-синтаксическими системами разбора текстов и статистическим анализом, а также в связи с бурным развитием таких областей, как Big Data и искусственный интеллект. Автоматизация построения объектов интеграции особенно актуальна при работе с Большими данными и позволяет строить системы, более устойчивые к модификациям в процессе их эксплуатации.

Подход, изложенный в статье, применялся к задачам, совершенно разным как по семантике, так и по формам представления данных, что позволяет говорить о достаточной его универсальности.

СПИСОК ЛИТЕРАТУРЫ

1. *Hopkins B., Evelson B.* Expand your digital horizon with Big Data. URL: http://www.asterdata.com/newsletter-images/30-04-2012/resources/forrester_ expand_your_digital_horiz.pdf, 2011.

2. URL: http://nosql-database.org/.

3. URL: https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html.

4. URL: http://hadoop.apache.org/.

5. URL: https://hana.sap.com/abouthana.html.

6. *Когаловский М.Р*. Методы интеграции данных в информационных системах // Институт проблем рынка РАН, Москва, 2010. URL: http://www.ipr-ras.ru//articles/kogalov10-05.pdf.

7. URL: https://www.w3.org/TR/2004/REC-owl-features-20040210/.

8. *Гаврилова Т.А., Хорошевский В.Ф.* Базы знаний интеллектуальных систем. СПб.: Питер, 2001. 384 с.

9. *Винетап P.* Semistructured data // Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. Tucson, Arizona, United States, May 11–15, 1997. P. 117–121.

10. *Биряльцев Е.В., Гусенков А.М., Косинов Я.Г.* Представление структуры реляционных баз данных в формализме онтологий // Труды Казанской школы по компьютерной и когнитивной лингвистике TEL-2006. Казань: Изд-во «Отечество», 2007. С. 32–37.

11. *Биряльцев Е.В., Гусенков А.М.* Интеграция реляционных баз данных на основе онтологий. // Ученые записки Казанского государственного университета. Серия Физико-математические науки. 2007. Т. 149, Кн. 2. С. 13–25.

12. *Гусенков А., Биряльцев Е., Жибрик О.* Интеллектуальный поиск в структурированных массивах информации // LAP LAMBERT Academic Publishing, 2015. 129 с.

13. *Гарсиа-Молина Г., Ульман Дж., Уидом Дж.* Системы баз данных. Полный курс / Database Systems: The Complete Book // Вильямс, 2003. 1088 с.

14. *Буч Г., Рамбо Д., Джекобсон А.* Язык UML. Руководство пользователя. Пер. с англ. М.: ДМК, 2000. 432 с.

15. *Биряльцев Е.В., Гусенков А.М., Галимов М.Р.* Особенности лексико-семантической структуры наименований артефактов реляционных баз данных // Труды Казанской школы по компьютерной и когнитивной лингвистике TEL-2005. Казань: Изд-во Казанского ун-та, 2006. С. 4–12.

16. *Жучков А.В., Арнаутов С.А., Твердохлебов Н.В.* Новые технологии для понятийных сетей, создаваемых в рамках МНТП «Вакцины нового поколения и диагностические системы будущего» // Электронные библиотеки, 2003. Т. 6, Вып. 6. URL: http://www.elbib.ru/index.phtml?page=elbib/rus/journal/2003/part6/ZATGS.

17. *Андерсон Дж.* Дискретная математика и комбинаторика. Пер. с англ. М.: Изд. дом «Вильямс», 2003. 960 с.

18. *Биряльцев Е.В., Гусенков А.М., Хайруллина А.И*. Представление модели данных Epicenter POSC на языке онтологий OWL // Труды Казанской школы по

компьютерной и когнитивной лингвистике TEL-2006. Казань: Изд-во «Отечество», 2007. С. 38–49.

19. *Биряльцев Е.В., Гусенков А.М*. Построение онтологии предметной области на основе логической модели баз данных // Труды Всерос. конф. с международным участием «Знания-Онтологии-Теории» (ЗОНТ-07). Новосибирск: Ин-т математики им. С.Л. Соболева СО РАН, 2007. Т. 1. С. 176–183.

20. URL: http://www.energistics.org/energistics-standards-directory/epicentre-archive.

21. URL: https://www.w3.org/OWL.

22. *Fellbaum C. (ed.)* WordNet: An electronic lexical database. Cambridge: MIT Press, 1998. 423 p.

23. *Биряльцев Е.В., Гусенков А.М.* Онтологии реляционных баз данных. Лингвистический аспект. // Тр. межд. конф. Диалог'2007, «Бекасово», 30 мая – 3 июня 2007 г. М.: Изд. центр РГГУ, 2007. С. 50–53.

24. URL: http://www.dialog-21.ru/Archive/2001/volume2/2_21.htm.

25. *Биряльцев Е.В., Гусенков А.М., Миронов С.В.* Один подход к реализации нерегламентированного доступа к реляционным базам данных // Труды Казанской школы по компьютерной и когнитивной лингвистике TEL-2008. Казань, 10–13 декабря 2008 г. Казань: Изд-во Казанского ун-та, 2009. С. 10–23.

26. *Misutka J., Galambos L.* Extending full text search engine for mathematical content // Proceedings of DML. 2008. P. 55–67.

27. URL: http://lucene.apache.org.

28. *Биряльцев Е.В., Галимов М.Р., Гусенков А.М., Жибрик О.Н.* Некоторые подходы к повышению релевантности поиска математических выражений в естественнонаучных текстах // Труды Казанской школы по компьютерной и когнитивной лингвистике TEL-2012. Казань, 25–28 января 2012 г. Казань: Изд-во Фэн Академии наук РТ, 2012. С. 78–93.

29. *Биряльцев Е.В., Гусенков А.М., Жибрик О.Н.* Некоторые подходы к разметке естественнонаучных текстов, содержащих математические выражения // Ученые записки Казанского университета. 2014. Т. 156, Кн. 4. С. 133–148.

30. *Биряльцев Е.В., Галимов М.Р., Жильцов Н.Г., Невзорова О.А.* Подход к семантическому поиску математических выражений в научных текстах // Материалы межд. науч.-техн. конф. OSTIS-2012. Минск: БГУИР, 2012. С. 245–256.

31. *Nevzorova O., Zhiltsov N., Zaikin D., Zhibrik O., Kirillovich A., Nevzorov V., Birialtsev E.* Bringing Math to LOD: a semantic publishing platform prototype for scientific collections in Mathematics // The SemanticWeb – ISWC 2013. 12th Int. SemanticWeb Conference. Sydney, NSW, Australia, October 21–25, 2013. Springer, Lecture Notes in Computer Science, 2013. V. 8218. P. 379–394.

32. URL: http://www.w3.org/TR/rdf-sparql-query/.

33. *Биряльцев Е.В., Гусенков А.М., Жибрик О.Н.* Поиск математических выражений в естественно-научных текстах. Экспериментальная оценка релевантности // Интеллект. Язык. Компьютер. Труды Казанской школы по компьютерной и когнитивной лингвистике TEL-2014. Казань: Изд-во Фэн Академии наук РТ, 2014. С. 34–37.

34. *Биряльцев Е.В., Гусенков А.М., Елизаров А.М.* О доступе к электронным коллекциям в виде реляционных баз данных на основе онтологий // Труды 9-й Всерос. научн. конф. Электронные библиотеки: перспективные методы и технологии, электронные коллекции – RCDL-2007, Переславль-Залесский, Россия, 15–18 октября 2007 г. Переславль-Залесский, Ярославль: Изд-во Университет города Переславля, 2007. С. 211–216

35. *Elizarov A.M., Kirillovich A.V., Lipachev E.K., Nevzorova O.A., Solovyev V.D., Zhiltsov N.G*. Mathematical knowledge representation: semantic models and formalisms // Lobachevskii Journal of Mathematics. 2014. V. 35, No 4. P. 347–353.

36. *Биряльцев Е.В., Елизаров А.М., Жильцов Н.Г., Липачев Е.К., Невзорова О.А., Соловьев В.Д.* Методы анализа семантических данных математических электронных коллекций// Научно-техническая информация. Серия 2. Информационные процессы и системы. 2014. № 4. С. 12–16 (*Biryal'tsev E.V., Elizarov A.M., Zhil'tsov N.G., Lipachev E.K., Nevzorova O.A., Solov'ev V.D.* Methods for analyzing semantic data of electronic collections in mathematics// Automatic Documentation and Mathematical Linguistics. 2014. V. 48, No 2. P. 81–85).

37. *Елизаров А.М., Липачёв Е.К., Хохлов Ю.Е.* Семантические методы структурирования математического контента, обеспечивающие расширенную поисковую функциональность// Информационное общество. 2013. № 1–2. С. 83–92.

38. *Елизаров А.М., Липачев Е.К., Невзорова О.А., Соловьев В.Д*. Методы и средства семантического структурирования электронных математических документов // Докл. РАН. 2014. Т. 457, № 6. С. 642–645.

39. *Елизаров А.М., Липачев Е.К., Малахальцев М.А.* Веб-технологии для математика: основы MathML. Практическое руководство. М.: Физматлит, 2010. 192 с.

40. *Елизаров А.М., Липачев Е.К., Малахальцев М.А.* Сервисы электронных естественнонаучных коллекций, построенные на основе технологии MathML // Труды Всероссийской суперкомпьютерной конф. «Научный сервис в сети Интернет: суперкомпьютерные центры и задачи», г. Новороссийск, 20–25 сентября 2010 г. М.: Изд-во Московского ун-та, 2010. С. 533–534.

41. URL: http://gate.ac.uk/.

INTELLIGENT SEARCH OF COMPLEX OBJECTS IN BIG DATA

A.M. Gusenkov[1]

Institute of Computational Mathematics and Information Technologies.
Kazan (Volga Region) Federal University

[1] gusenkov.a.m@gmail.com

Abstract

This article considers approach to intelligent search of complex objects in different types of texts with structural markup which can be used for Big Data processing. We research two types of data entry: relational databases, which use their schemes as structural markup, and full-text scientific documents containing mathematical expressions (formulae). For such full-text documents we suggest additory automated markup to allow formula search. In both cases we use natural language texts, which are semi-structured data, as data source for building ontology and conducting search at a later stage. For relational databases those are comments to table and table attribute names;

for scientific documents (articles, monographs, etc.) it is a text content of marked up documents.

***Keywords**: Big Data, semantic search, semi-structured data, ontology, relational databases, science texts, mathematical expressions markup.*

REFERENCES

1. *Hopkins B., Evelson B.* Expand your digital horizon with Big Data. URL: http://www.asterdata.com/newsletter-images/30-04-2012/resources/forrester_expand_your_digital_horiz.pdf, 2011.

2. URL: http://nosql-database.org/.

3. URL: https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html.

4. URL: http://hadoop.apache.org/.

5. URL: https://hana.sap.com/abouthana.html.

6. *Kogalovskij M.R.* Metody integracii dannyh v informacionnyh sistemah // Institut problem rynka RAN, Moskva, 2010. URL: http://www.ipr-ras.ru/articles/ kogalov10-05.pdf.

7. URL: https://www.w3.org/TR/2004/REC-owl-features-20040210/.

8. *Gavrilova T.A., Horoshevskij V.F.* Bazy znanij intellektual'nyh sistem. SPb.: Piter, 2001. 384 s.

9. *Buneman P.* Semistructured data // Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. Tucson, Arizona, United States, May 11–15, 1997. P. 117–121.

10. *Birialtsev E.V., Gusenkov A.M., Kosinov Y.G.* Predstavlenie struktury relyacionnyh baz dannyh v formalizme ontologij // Trudy Kazanskoj shkoly po komp'yuternoj i kognitivnoj lingvistike TEL-2006. Kazan: Izdatel'stvo “Otechestvo”, 2007. S. 32–37.

11. *Birialtsev E.V., Gusenkov A.M.* Integraciya relyacionnyh baz dannyh na osnove ontologij. // Uchenye zapiski Kazanskogo gosudarstvennogo universiteta. Seriya Fiziko-matematicheskie nauki. 2007. T. 149. Kn. 2. Kazan: Kazanskij universitet, 2007. S. 13–25.

12. *Gusenkov A., Birialtsev E., Zhibrik O.* Intellektual'nyj poisk v strukturirovannyh massivah informacii. // LAP LAMBERT Academic Publishing, 2015. 129 s. ISBN 978-3-659-76919-1.

13. *Garcia-Molina H., Ullman J., Widom J.* Database Systems: The Complete Book. Williams, 2003. 1088 s.

14. *Booch G., Rumbaugh J., Jacobson I.* The Complete UML Training Course // Per. s angl. Moskva: Izdatel'stvo DMK, 2000. 432 s.

15. *Birialtsev E.V., Gusenkov A.M., Galimov M.R*. Osobennosti leksiko-semanticheskoj struktury naimenovanij artefaktov relyacionnyh baz dannyh // Trudy Kazanskoj shkoly po komp'yuternoj i kognitivnoj lingvistike TEL-2005. Kazan: Izdatel'stvo Kazanskogo universiteta, 2006. S. 4–12.

16. *Zhuchkov A.V., Arnautov S.A., Tverdohlebov N.V*. Novye tekhnologii dlya ponyatijnyh setej, sozdavaemyh v ramkah MNTP «Vakciny novogo pokoleniya i diagnosticheskie sistemy budushchego» // Ehlektronnye biblioteki. 2003. T. 6, Vyp. 6. URL: http://www.elbib.ru/index.phtml?page=elbib/rus/journal/2003/part6/ ZATGS.

17. *Anderson J.* Discrete Mathematics with Combinatorics. Per. s angl. Moskva: Izd. dom Williams, 2003. 960 s.

18. *Birialtsev E.V., Gusenkov A.M., Khairullina A.I.* Predstavlenie modeli dannykh Epicenter POSC na iazyke ontologii OWL// Trudy Kazanskoi shkoly po kompiuternoi i kognitivnoi lingvistike TEL-2006. Kazan: Izdatel'stvo “Otechestvo”, 2007. S. 38–49.

19. *Birialtsev E.V., Gusenkov A.M.* Postroenie ontologii predmetnoj oblasti na osnove logicheskoj modeli baz dannyh // Trudy Vseros. konf. s mezhdunarodnym uchastiem “Znaniya-Ontologii-Teorii” (ZONT-07). Novosibirsk: In-t matematiki im. S.L. Sobolevа SO RAN, 2007. T. 1. S. 176–183.

20. URL: http://www.energistics.org/energistics-standards-directory/epicentre-archive.

21. URL: https://www.w3.org/OWL/.

22. *Fellbaum C. (ed.)* WordNet: An Electronic Lexical Database. Cambridge: MIT Press, 1998. 423 p.

23. *Birialtsev E.V., Gusenkov A.M.* Ontologii relyacionnyh baz dannyh. Lingvisticheskij aspekt. // Tr. mezhdunar. konf. Dialog'2007, Bekasovo, 30 maya – 3 iyunya 2007 g. Moskva: Izd. centr RGGU, 2007. S. 50–53.

24. URL: http://www.dialog-21.ru/Archive/2001/volume2/2_21.htm.

25. *Birialtsev E.V., Gusenkov A.M., Mironov S.V.* Odin podhod k realizacii nereglamentirovannogo dostupa k relyacionnym bazam dannyh // Trudy Kazanskoj shkoly

po komp'yuternoj i kognitivnoj lingvistike TEL-2008. Kazan, 10–13 dekabrya 2008 g. Kazan: Izdatel'stvo Kazanskogo universiteta, 2009. S. 10–23.

26. *Misutka J., Galambos L.* Extending full text search engine for mathematical content // Proceedings of DML. 2008. P. 55–67.

27. URL: http://lucene.apache.org.

28. *Birialtsev E.V., Galimov M.P., Gusenkov A.M., Zhibrik O.N.* Nekotorye podhody k povysheniyu relevantnosti poiska matematicheskih vyrazhenij v estestvennonauchnyh tekstah // Trudy Kazanskoj shkoly po komp'yuternoj i kognitivnoj lingvistike TEL-2012. Kazan, 25–28 yanvarya 2012 g. Kazan: Izdatel'stvo “Fehn” Akademii nauk RT, 2012. S. 78–93.

29. *Birialtsev E.V., Gusenkov A.M., Zhibrik O.N.* Nekotorye podhody k razmetke estestvennonauchnyh tekstov, soderzhashchih matematicheskie vyrazheniya. Uchenye zapiski Kazanskogo universiteta, Kazan: Kazanskij universitet, 2014. T. 156, kn. 4. S. 133–148.

30. *Birialtsev E.V., Galimov M.P., Zhiltsov N.G., Nevzorova O.A.* Podhod k semanticheskomu poisku matematicheskih vyrazhenij v nauchnyh tekstah // Materialy mezhdunarodnoj nauchno-tekhnicheskoj konferencii OSTIS-2012. Minsk: BGUIR, 2012. S. 245–256.

31. *Nevzorova Olga, Zhiltsov Nikita, Zaikin Danila, Zhibrik Olga, Kirillovich Alexander, Nevzorov Vladimir, Birialtsev Evgeniy*. Bringing Math to LOD: a semantic publishing platform prototype for scientic collections in Mathematics // The SemanticWeb – ISWC 2013. 12th International SemanticWeb Conference Sydney, NSW, Australia, October 21–25, 2013. Lecture Notes in Computer Science 8218. Springer. P. 379-394.

32. URL: http://www.w3.org/TR/rdf-sparql-query/.

33. *Birialtsev E.V., Gusenkov A.M., Zhibrik O.N.* Poisk matematicheskih vyrazhenij v estestvenno-nauchnyh tekstah. Ehksperimental'naya ocenka relevantnosti // Intellekt. Yazyk. Komp'yuter. Trudy Kazanskoj shkoly po komp'yuternoj i kognitivnoj lingvistike TEL-2014. Kazan: Izdatel'stvo “Fehn” Akademii nauk RT. 2014. S. 34-37.

34. *Birialtsev E.V., Gusenkov A.M., Elizarov A.M.* O dostupe k ehlektronnym kollekciyam v vide relyacionnyh baz dannyh na osnove ontologij // Trudy 9-j Vseros. nauchn. konf. Elektronnye biblioteki: perspektivnye metody i tekhnologii, ehlektronnye

kollekcii – RCDL-2007, Pereslavl'-Zalesskij, Rossiya, 15–18 oktyabrya 2007 g. Pereslavl'-Zalesskij, Yaroslavl': Izd-vo Universitet goroda Pereslavlya, 2007. S. 211-216.

35. *Elizarov A.M., Kirillovich A.V., Lipachev E.K., Nevzorova O.A., Solovyev V.D., Zhiltsov N.G*. Mathematical knowledge representation: semantic models and formalisms // Lobachevskii Journal of Mathematics. 2014. V. 35, No 4. P. 347–353.

36. *Birialtsev E.V., Elizarov A.M., Zhiltsov N.G., Lipachev E.K., Nevzorova O.A., Solovyev V.D.* Methods for analyzing semantic data of electronic collections in mathematics// Automatic Documentation and Mathematical Linguistics. 2014. V. 48, No 2. P. 81–85.

37. *Elizarov A.M., Lipachev E.K., Hohlov Yu.E.* Semanticheskie metody strukturirovanija matematicheskogo kontenta, obespechivajushhie ras-shirennuju poiskovuju funkcional'nost'// Informacionnoe obshhestvo. 2013. № 1–2. S. 83–92.

38. *Elizarov A.M., Lipachev E.K., Nevzorova O.A., Solovyev V.D.* Metody i sredstva semanticheskogo strukturirovaniya ehlektronnyh matematicheskih dokumentov // Dokl. RAN. 2014. T. 457, № 6. S. 642–645.

39. *Elizarov A.M., Lipachev E.K., Malakhal'tsev M.A.* Veb-tekhnologii dlya matematika: osnovy MathML. Prakticheskoe rukovodstvo. M.: FIZMATLIT, 2010. 192 s.

40. *Elizarov A.M., Lipachev E.K., Malakhal'tsev M.A.* Servisy ehlektronnyh estestvennonauchnyh kollekcij, postroennye na osnove tekhnologii MathML // Trudy Vserossijskoj superkomp'yuternoj konf. Nauchnyj servis v seti Internet: superkomp'yuternye centry i zadachi, g. Novorossijsk, 20–25 sentyabrya 2010 g. M.: Izd-vo Mosk. un-ta, 2010. S. 533–534.

41. URL: http://gate.ac.uk/.

СВЕДЕНИЯ ОБ АВТОРЕ

ГУСЕНКОВ Александр Михайлович – старший преподаватель Института вычислительной математики и информационных технологий Казанского (Приволжского) федерального университета.

Alexander Mikhailovich GUSENKOV – senior lecturer, Institute of Computational Mathematics and Information Technologies of Kazan Federal University. Current scientific interests: knowledge extraction technologies, big data, data mining, parallel computing.

email: gusenkov.a.m@gmail.com

Материал поступил в редакцию 24 декабря 2015 года