На правах рукописи

Колосов Алексей Павлович

МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ПОЛНОТЕКСТОВОГО ПОИСКА В БАЗАХ ДАННЫХ НА ОСНОВЕ КОНЦЕПТУАЛЬНОГО МОДЕЛИРОВАНИЯ

Специальность 05.13.11 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата технических наук

Тула 2012

Работа выполнена в ФГБОУ ВПО «Тульский государственный университет»

Научный руководитель: доктор технических наук, доцент
Богатырев Михаил Юрьевич

Официальные оппоненты: Есиков Олег Витальевич,
доктор технических наук, профессор,
ФГУП «Научно-исследовательский институт репрографии», советник по информатизации

Рыжков Евгений Александрович,
кандидат технических наук,
ООО «Системы программной верификации», генеральный директор

Ведущая организация: ФГБОУ ВПО «Российский государственный педагогический университет им. А.И. Герцена»

Защита состоится «28» мая 2012 г. в 12 час. 00 мин. на заседании диссертационного совета Д 212.271.07 при ФГБОУ ВПО «Тульский государственный университет» (300012, г. Тула, проспект Ленина, 92, 9-101)

С диссертацией можно ознакомиться в библиотеке ФГБОУ ВПО «Тульский государственный университет».

Автореферат разослан «25» апреля 2012 г.

Учёный секретарь
диссертационного совета Данилкин Федор Алексеевич

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы исследований. Полнотекстовые базы данных играют все более важную роль в современных информационных ресурсах. Поэтому совершенствование математического и программного обеспечения полнотекстовых баз данных является одним из ключевых направлений развития индустрии программирования. В рамках данного направления решение задач полнотекстового поиска имеет принципиальное значение.

Традиционно полнотекстовый поиск выполняется по всем текстам хранящихся в базе данных документов с целью нахождения документов, близких в смысле некоторой меры близости к поисковому запросу. При этом поисковый запрос представляется в виде набора ключевых слов, а для оценки близости документов запросу применяются алгоритмы, основанные на анализе статистики появления ключевых слов в документах базы данных.

В настоящее время актуальны задачи разработки систем полнотекстового поиска в базах данных для случаев, когда сам запрос является не словом или фразой, а осмысленным текстом. *Системы технической поддержки* являются характерными примерами систем, где применяются подобные полнотекстовые запросы. В настоящее время в большинстве подобных систем обработка запросов и подготовка ответов выполняются вручную, исключая некоторые возможности автоматического поиска по ключевым словам. При большой нагрузке системы падает ее производительность, поскольку при ручной работе в системе невозможно обработать за приемлемое время большие объемы данных. Автоматизация поиска документов, релевантных полнотекстовым запросам, поступающим в систему, является чрезвычайно актуальной задачей.

Применение полнотекстовых запросов требует учета семантики в решении задачи полнотекстового поиска, что невозможно при традиционном подходе, поскольку семантика полнотекстовых запросов не может быть описана ключевыми словами. В связи с этим тема данной диссертационной работы является актуальной, поскольку связана с разработкой математического и программного обеспечения полнотекстового поиска в базах данных, основанного на новых семантических моделях текстов.

Результаты, полученные в работе, опираются на известные ранее результаты в области информационного поиска, отраженные в работах российских (Н.Н. Леонтьева, С.О. Кузнецов, А.Е. Ермаков) и зарубежных (J.Sowa, S.Buttcher, S.Robertson) исследователей, и ориентированы на практическое применение в программном обеспечении полнотекстовых баз данных.

Объектом исследования является ПО систем полнотекстового поиска.

Предметом исследования являются алгоритмы полнотекстового поиска, концептуальные графовые модели, алгоритмы выделения ключевых словосочетаний из текстов, конкретные технологии полнотекстового поиска.

Целью диссертационной работы является повышение точности решения задач полнотекстового поиска в базах данных.

Поставленная цель достигается решением следующих задач.

1. Формализация задачи полнотекстового поиска с применением концептуальных графовых моделей.

2. Разработка метода выделения ключевых словосочетаний из текстов запросо с применением концептуальных графов.

3. Разработка сопутствующего алгоритма индексирования документо использующего обработку знаков препинания.

4. Разработка алгоритма полнотекстового поиска с контекстным окно плавающего размера, использующего при вычислении релевантност словосочетания и полнотекстовые индексы.

5. Разработка инструментального ПО системы полнотекстового поиска и е интеграция в существующие информационные системы.

6. Экспериментальная проверка эффективности разработанных алгоритмов и и сравнение с существующими аналогами.

7. Разработка технологии полнотекстового поиска, реализующе разработанные алгоритмы для конкретной СУБД.

Методы исследований. Основные результаты работы получены применением методов обработки естественного языка, математической логики концептуального моделирования. Программные решения для систем техническо поддержки реализованы в парадигме объектно-ориентированно программирования.

Основные научные результаты диссертационной работы заключаются следующем.

1. Показано, что применение концептуальных графов в качестве семантическо модели полнотекстовых запросов в инструментальном ПО полнотекстового поиск обеспечивает извлечение из текста запроса словосочетаний, независимо от близост слов в них.

2. Разработан алгоритм индексирования документов, позволяющий, сохраня лишь позиции слов, в неявном виде хранить информацию о содержащихся в тексте знаках препинания, что позволяет делать предположения о наличии семантической связи между словами предложений уже на этапе индексирования.

3. Разработан эффективный алгоритм полнотекстового поиска, использующий в качестве запросов тексты на естественном языке с выделенным множеством ключевых словосочетаний и опирающийся, в отличие от существующих аналогов, на семантику текстов, а не на статистические данные.

Достоверность научных результатов подтверждена корректным использованием применяемых методов и экспериментальными исследованиями.

Результаты данной работы получены при выполнении следующих научных проектов:

- гранта РФФИ, № 11-07-97542-р_центр_а,
- проекта, поддержанного Фондом содействия развитию малых форм предприятий в научно-технической сфере, госконтракт № 9444р/15234.

Практическая значимость результатов работы состоит в следующем.

1. Применение концептуальных графов в качестве семантических моделей текстов запросов обеспечивает повышение точности решения задачи автоматического выделения ключевых словосочетаний за счет непосредственного моделирования их семантики. В результате повышается точность решения задачи полнотекстового поиска в целом.

2. Разработанное программное обеспечение позволяет снизить время получения ответа для пользователей систем технической поддержки, форумов и других ресурсов, посвященных ответам на вопросы, сформулированным в виде текстов на естественном языке, благодаря автоматическому поиску документов, которые могут содержать искомый ответ.

3. Разработанная система полнотекстового поиска может быть интегрирована с любыми информационными ресурсами: корпоративными базами данных, базами знаний, электронными библиотеками, системами технической поддержки и т.п., что позволяет расширять возможности существующих систем в области полнотекстового поиска.

Положения, выносимые на защиту. На защиту выносятся следующие результаты диссертационной работы:

1. Алгоритм индексирования документов с учетом знаков препинания.
2. Метод выделения ключевых словосочетаний из текстов на естественном языке, использующий концептуальные графы для моделирования смысла текстов.
3. Алгоритм полнотекстового поиска, запросы для которого представляются в виде множества словосочетаний.

Реализация и внедрение результатов диссертационной работы. Разработана система полнотекстового поиска, которая внедрена в системе технической поддержки ООО «Автоматизированное обеспечение качества», филиале компании SmartBearSoftware, и применяется на сайте компании. Система полнотекстового поиска также внедрена в программное обеспечение для разработки документации, разрабатываемое в ООО «Тульский Стандарт», что подтверждается актами о внедрении.

Результаты диссертационного исследования внедрены в учебный процесс на кафедре Автоматики и телемеханики ТулГУ в лекционные курсы «Сетевое программирование», «Базы данных и знаний» и их лабораторный практикум.

Апробация работы. Основные результаты работы докладывались на международных и всероссийских научно-технических конференциях, совещаниях и семинарах: **1.** 4-я международная конференция по распознаванию образов и искусственному интеллекту PReMI 2011 – Pattern Recognition and Machine Intelligence, Россия, Москва, 2011. **2.** 13-я всероссийская научная конференция «Электронные библиотеки: перспективные методы и технологии, электронные коллекции», Россия, Воронеж, 2011. **3.** 14-я всероссийская объединенная научная конференция «Интернет и современное общество» IMS-2011, Россия, Санкт-Петербург, 2011. **4.** Всероссийский семинар «Natural Language Processing», Россия, Санкт-Петербург, 2011.

Публикации. По теме диссертационного исследования опубликовано 7 печатных работ, в том числе 3 рекомендованных ВАК РФ, получено два свидетельства о регистрации программ для ЭВМ.

Структура и объем работы. Диссертационная работа изложена на 153 страницах, включает 5 таблиц и 27 рисунков. Состоит из введения, пяти глав, заключения, списка литературы из 101 наименования и 4 приложений.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении обосновывается актуальность разработки систем полнотекстового поиска, отмечены научная новизна и практическая значимост диссертационной работы.

В первой главе рассматриваются задачи и технологии полнотекстового поиска описываются модели и методы, используемые в известных алгоритмах. Н основании анализа современного состояния данной области науки ставятся задач исследования.

Задачи полнотекстового поиска возникают при развитии информационны поисковых систем. Их решение позволяет поисковым системам выйти на новы уровень обработки текстов, используя их смысловое содержание - семантику Задача полнотекстового поиска может рассматриваться как задача классификаци текста запроса на известном множестве текстов информационного ресурса системы.

Показано, что для создания систем полнотекстового поиска формальног решения задачи классификации недостаточно, поскольку все известные числовы меры близости текстов далеки от представления в них смыслового содержания Поэтому в подобных системах используется комплексная оценка близости текстов известная как *релевантность*. Алгоритмы определения релевантности строятся ка с привлечением формальных методов, так и некоторых эвристик.

Для целей программного решения задач полнотекстового поиска необходимо выделить в них две составляющие: построение семантической модели обрабатываемого текста и собственно поиск. Выполнен анализ современны методов анализа семантики текста и их применений, включая вероятностные и алгебраические методы.

Среди рассмотренных подходов выделено *концептуальное моделирование*, обладающее рядом преимуществ по сравнению с другими подходами к построению семантических моделей. Концептуальные модели позволяют описать семантику текста и достаточно формальны для применения их в программном обеспечении информационных поисковых систем. Проанализирована простейшая концептуальная модель предложения – *концептуальный граф*. На основе анализа сделан вывод о перспективности применения концептуальных графов для моделирования полнотекстовых запросов к базам данных.

В целях снижения вычислительной сложности алгоритмов поиска документы, составляющие информационный ресурс полнотекстовых баз данных, проходят предварительную обработку, называемую индексированием. В индексировании документов принципиальное значение имеет выделение терминов и понятий, отраженных в текстах документов, поскольку именно термины и понятия чаще всего обсуждаются в текстах запросов пользователей. В текстах термины и понятия задаются словосочетаниями, поэтому идентификация словосочетаний в текстах является основной задачей диссертационного исследования.

К другим задачам относятся алгоритмическое и программное обеспечение полнотекстового поиска, реализация подсистемы полнотекстового поиска в выбранной архитектуре, экспериментальная проверка эффективности алгоритмических решений.

В качестве области применения, где наиболее востребованы разрабатываемое математическое и программное обеспечение, выбраны системы технической поддержки, использующие полнотекстовые базы данных и полнотекстовые запросы.

Во второй главе рассматривается концептуальное моделирование как метод, применяемый для решения поставленных в работе задач. Исследуется математический аппарат концептуальных графов с целью применения его в задачах полнотекстового поиска.

Концептуальные графы являются одной из семантических моделей, применяемых для анализа текстов, и относятся к классу семантических сетей. Концептуальный граф – это двудольный ориентированный граф, состоящий из узлов двух типов: концептов и концептуальных отношений. На рисунке 1 показан пример концептуального графа, соответствующего одной из типичных фраз, встречающихся в полнотекстовых запросах пользователей систем технической поддержки: «*программа требует виртуальную или какую-то другую машину*». На рисунке 1 концепты графа показаны в виде прямоугольников, а отношения – в виде эллипсов.

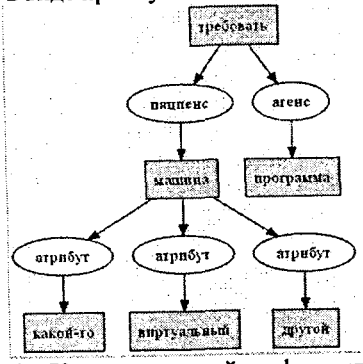

Рисунок 1 – Концептуальный граф предложения.

Отмечено, что семантика концептуальных графов определяется, с одной стороны, системой связей, устанавливаемых между словами-концептами, с другой стороны, - формальными моделями логики предикатов первого порядка.

Хотя концептуальный граф использует только бинарные отношения, показано, что семантическая выразительность концептуальных графов достаточна для идентификации словосочетаний.

Логическая модель концептуального графа строится как дизъюнкция конъюнктивных форм вида:

$$Q(x_1, \dots, x_n)(P_c(x_1) \wedge \dots \wedge P_c(x_n) \wedge \wedge_{i,j=1}^{m} P_r(x_i, x_j)),$$

где $Q(x_1, \dots, x_n)$ – сочетание кванторов переменных $x_1, \dots, x_n$, соответствующих концептам графа, $P_c(x_i)$ - предикат, соответствующий i-му концепту, $P_r(x_i, x_j)$ – предикат, соответствующий отношению, связывающему концепты i, j, n - число концептов в графе, m - число отношений. Данная форма соответствует каждому связному подграфу несвязного концептуального графа. Если граф связный, то он полностью задается указанной формой.

Применение концептуальных графов в качестве семантической модели предложений текстов запросов требует автоматизации их построений. Логическая модель концептуального графа применяется как основа алгоритмизации автоматического построения концептуальных графов в программном обеспечении.

Известна алгоритмическая сложность и неоднозначность решения задачи автоматического построения концептуальных графов. Для исключения неоднозначности выбрана *вербоцентрическая концепция* построения концептуальных графов, основанная на выделении главного глагола в тексте (глагол «требовать» на рисунке 1). При построении концептуальных графов применяется также известное решение лингвистической задачи *разметки семантических ролей*.

Алгоритм построения концептуального графа состоит из четырех этапов:

- языковой анализ - определение языка представленного текста;
- графематический анализ – разбиение текста на предложения, предложений – на слова и знаки пунктуации;
- морфологический анализ - определение морфологических признаков элементов предложения;
- концептуальный анализ - выделение концептов из элементов предложения, определение отношений между концептами.

Алгоритмические решения для этапов 1-3 известны. В частности, известны достаточно надежные морфологические анализаторы текста. Четвертый этап алгоритма – наиболее сложный и наименее разработан в настоящее время. В работе новые программные решения предлагаются именно на данном этапе.

Каждый элемент текста представляется в виде тройки вида:

$$w = \langle f, N, A \rangle,$$

где w – элемент текста, в исходном варианте соответствующий морфологическому словарю W^*, f – естественная форма элемента текста, N – нормализованная форма элемента, A – список морфологических атрибутов элемента. Упорядоченная последовательность элементов текста представляется в следующем виде:

$$P_n = \{w_1, w_2, \dots, w_n\}$$

где P_n – последовательность элементов текста длиной n, w_i – текстовый элемент, при этом элемент w_i предшествует элементу w_j, если $i<j$.

Для построения концептов и отношений применяются *грамматические шаблоны*. Каждый шаблон представляется в виде четвёрки:

$$s_n = \langle P_s^n, C, d, n_s \rangle$$

где P_s^n – последовательность текстовых элементов длиной nдля сравнения, C – список относительных атрибутов, d – действие (из набора действий D^*), применяемое к фразе, найденной в тексте и подходящей к шаблону, n_s – длина фразы (число текстовых элементов в шаблонной фразе).

Формируется функция сравнения, которая сопоставляет шаблон с выбранной последовательностью элементов текста. Определяется функция сравнения следующим выражением:

$$\left\{ \begin{array}{c} n = n_s \\ P_n \cap P_s^n = P_s^n \\ \{A_{i,j} \in C\} \cap \{A \in w_i\} = \{A_{i,j} \in C\} \cap \{A \in w_j\}, i = 1..n, j = 1..n \end{array} \right\} \Rightarrow l(P_n, s_n)$$

где n – число элементов исследуемой текстовой последовательности, n_s – число элементов последовательности шаблона сравнения, P_n – исследуемая последовательность элементов текста, $A_{i,j}$ – список атрибутов для сравнения i и j элемента исследуемой последовательности.

Показано, что разработанное программное обеспечение обеспечивает построение концептуальных графов для грамматически корректных текстов ограниченного лексикона, к которым относятся тексты запросов к полнотекстовым базам данных. Множество грамматических шаблонов является расширяемым и настраиваемым на определенные грамматические обороты текста. Используемые в технических текстах специальные термины распознаются при помощи дополнительного словаря.

В третьей главе разрабатывается алгоритмическое обеспечение инструментальных программных средств полнотекстового поиска. Информационные ресурсы полнотекстовых баз данных снабжены специально построенными индексами. Современные системы индексирования текстовых данных включают информацию о терминах и понятиях, применяемых в хранимых текстах. Эти термины и понятия задаются словосочетаниями.

Словосочетания из текста запроса к системе предложено извлекать из концептуальных графов, строящихся на тексте запроса. Условие выбора концептов c_i и c_j из графа в качестве части словосочетания, являющегося N-граммой, формулируется в виде выражения импликации:

$$\left\{\begin{array}{c} c_i \notin W_s \wedge c_j \notin W_s, W_s \subset W^* \\ r \in G(c_i) \wedge r \in G(c_j) \\ type(r) = attribute \end{array}\right\} \Rightarrow t(c_i, c_j)$$

где W_s – множество шумовых слов (стоп-слов), определяемое словарем, r – вершина отношения, связывающего концепты c_i и c_j, $type(r)$ – тип отношения, $G(c_i)$ - множество вершин, связанных с вершиной c_i. Объединение пары существующих множеств T_i и T_j делается в случае, когда выполняется следующее выражение импликации:

$$\left\{\begin{array}{c} t(c_i, c_j) \\ c_i \in T_i \\ c_j \in T_j \end{array}\right\} \Rightarrow join(T_i, T_j)$$

Предложено производить добавление нового концепта c_i к существующему словосочетанию T_j в случае, если верно следующее выражение импликации:

$$\left\{\begin{array}{c} t(c_i, c_j) \\ c_j \in T_j \\ c_i \notin T_j \end{array}\right\} \Rightarrow add(c_i, T_j)$$

Полученное из концептуального графа словосочетание представляется в следующем виде:

$$T = \{t_1, t_2, \dots, t_n\}, t_i \notin W_s, i = 1,2, \dots, n$$
$$\forall t_i \exists t_j : t(c_i = t_i, c_j = t_j) \wedge ref(c_i) < ref(c_j), i < j$$

где t_i – i-ое слово из упорядоченного множества слов T, представляющего словосочетание, выделенное из запроса. Сортировка слов в словосочетаниях в

соответствии с порядком их возникновения в тексте проводится на последнем этапе, когда слова уже сгруппированы в N-граммы по принципу наличия отношения «атрибут» между ними. Показано, что слова, связанные именно этим отношением, то есть описывающие упоминаемые в тексте объекты и их атрибуты, составляют ключевые словосочетания, выделяемые для текстов экспертами.

Особенность применения терминов - словосочетаний в текстах хранимой документации по сравнению с текстами запросов состоит в том, что в них словосочетания могут иметь более разнообразные формы, в частности, быть разделенными знаками препинания. Выделение таких словосочетаний алгоритмически наиболее сложно. Поэтому в работе выполнена разработка сопутствующего алгоритма индексирования документов, ориентированного на обработку знаков препинания.

Отмечено, что разработанный алгоритм индексирования, учитывающий знаки препинания при определении позиций слов и сохраняющий тем самым информацию о них в индексах, обеспечивает наличие данных, необходимых для вычисления релевантности при поиске. Функция вычисления позиции слова w_i из индексируемого документа D_j с учетом символа-разделителя b_j представляется в следующем виде:

$$pos(w_i, b_j) = \begin{cases} pos(w_{i-1}) + \Delta_m, b_j \in B_m \\ pos(w_{i-1}) + \Delta_w, b_j \in B_w \\ pos(w_{i-1}) + \Delta_p, b_j \in B_p \\ pos(w_{i-1}) + \Delta_s, b_j \in B_s \\ 0, \quad i = 1 \end{cases}$$

где $pos(w_{i-1})$ - позиция предыдущего слова, вычисленная ранее, B_m, B_w, B_p и B_s – множества символов-разделителей, принадлежность к которым символа-разделителя b_j определяется с помощью регулярных выражений, аΔ_m, Δ_w, Δ_p и Δ_s- параметры метода, соответствующие различным классам символов-разделителей.

Условие вхождения искомого словосочетания в документ записывается в виде выражения импликации:

$$\left\{ \begin{matrix} n_j = n, n_j = |W_j|, n = |T| \\ w_k \in dict(t_k), w_k \in W_j, t_k \in T, k = 1,2..n \\ pos(w_k) - pos(w_{k-1}) < \Delta_s \end{matrix} \right\} \Rightarrow m(W_j, T)$$

где W_j – упорядоченное множество слов из документа, T – упорядоченно множество слов, т.е. словосочетание из запроса Q ($T \in Q$), $dict(p_k)$ - все формы k того слова из словосочетания T, полученные с помощью словаря, $pos(w_k)$ – позици k-того слова из W_j относительно начала документа, Δ_s - константа, характеризующа величину искусственного увеличения позиции слова после конца предложения.

Предложено релевантность каждого поля документа вычислять для пар (Q, D_j), где Q – множество словосочетаний из запроса, каждое словосочетани описывается парой $\langle T_i, u_{T_i} \rangle \in Q$, а D_j – проиндексированный документ из баз данных.

$$R_{field}(Q, D_j) = \frac{\sum_{i=1}^{N} R_T(T_i, D_j) u_{T_i}}{N}, N = |Q|$$

где T_l – словосочетание из запроса Q, u_{T_l} – вес этого словосочетания, а $R_T(T_l, D_j)$ - релевантность поля документа D_j словосочетанию из запроса, которую предлагается вычислять по следующей формуле:

$$R_T(T_l, D_j) = 2^{an} \sum_{k=1}^{n_m} r_k, n = |P_l|$$

где n_m – количество вхождений словосочетания в документ D_j, r_k – вклад в релевантность каждого из этих вхождений, а a - параметр алгоритма, регулирующий влияние количества слов на итоговую величину релевантности.

Вклад каждого вхождения словосочетания в документ предложено вычислять по следующей формуле:

$$r_k = \sum_{l=2}^{n} \frac{s(w_l, t_l)}{pos(w_l) - pos(w_{l-1})}, n = |P_l|,$$

где $pos(w_l)$ – позиция l-того слова из документа, соответствующего l-тому слову из искомого словосочетания, $s(w_l, t_l)$ – вес l-того слова из документа, вычисляемый в зависимости от степени совпадения формы слова в запросе и в документе.

Функция $s(w_l, t_l)$ представляется в следующем виде:

$$s(w_l, t_l) = \begin{cases} 4, & w_l = t_l \\ 1, & w_l \in dict(t_l) \\ 0, & w_l \notin dict(t_l) \end{cases}$$

Определено, что итоговое значение релевантности складывается из релевантности запросу каждого поля проиндексированного документа:

$$R(Q, D_j) = \sum_{l=1}^{N_f} u_f R_T(Q, D_j)$$

где N_f – общее количество проиндексированных полей (в текущей реализации равно 2), u_f – вес проиндексированного поля.

В четвертой главе приведено описание программного обеспечения систем полнотекстового поиска в базах данных на примере поисковой подсистемы реальной системы технической поддержки.

Система технической поддержки получает запросы от пользователей в виде текстов на естественном языке, например, в формате сообщений электронной почты, и помогает пользователям получать искомую информацию в виде ответов специалистов или ссылок на документы, хранящиеся в информационных ресурсах системы.

Архитектура разработанной поисковой подсистемы представлена на рисунке 2.

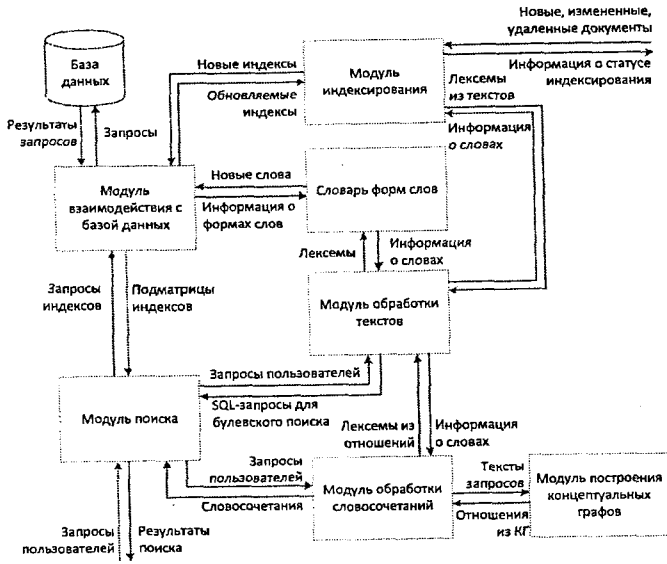

Рисунок 2 – Архитектура поисковой подсистемы.

Подсистема имеет следующие модули.

1. Модуль взаимодействия с базой данных обеспечивает доступ к базам данны различных платформ.

2. Словарный модуль обеспечивает обновление словаря и получени информации о словах (в частности, о связях между различными формами одного того же слова) по полученным из текста лексемам.

3. Модуль обработки текстов производит чтение документов различны поддерживаемых форматов (TXT, XML, HTML), разбиение текстов на предложени а предложений – на лексемы, после чего лексемы заменяются на слова из словар для каждого такого слова вычисляются, с учетом знаков препинания, его позиции тексте.

4. Модуль индексирования обеспечивает создание новых индексов, а так изменение существующих индексов при изменении документов и удалени индексов при удалении, либо исключении документов из публичного доступа.

5. Модуль обработки словосочетаний использует модули обработки текстов построения концептуальных графов для получения множества словосочетаний текстов запросов.

6. Модуль поиска принимает запросы пользователей системы и возвраща отсортированные в порядке уменьшения релевантности документы, релевантны заданному запросу.

Разработанные программные модули (язык C#, СУБД Microsoft SQL Serv 2008 R2, технология LINQ) предназначены для использования в любой верс операционной системы Windows, поддерживающей платформу .NET. Проведено внедрение в систему технической поддержки компании SmartBear Software с цел

обеспечения автоматического поиска по полным текстам пользовательских запросов до того, как новые вопросы попадут в базу данных. Данный подход позволил снизить нагрузку сотрудников службы технической поддержки компании и позволяет пользователям получить ответ на свои вопросы незамедлительно, не дожидаясь рассмотрения вопросов специалистами.

В пятой главе описываются эксперименты на реальных данных, подтверждающие эффективность разработанных алгоритмических решений, реализованных в инструментальном программном обеспечении. Результаты экспериментов позволили настроить алгоритм выделения словосочетаний так, что его эффективность превзошла эффективность существующих аналогов.

Эксперименты выполнялись в полнотекстовых базах данных. Всего в экспериментах участвовали четыре базы, различающихся по количеству, стилю и формату содержащихся в них документов. Две из них представляли собой наборы аннотаций научных статей, еще две – ресурсы компании SmartBear: история переписки пользователей с сотрудниками службы технической поддержки, а также справочная документация к программным продуктам.

Для оценки качества алгоритма выделения словосочетаний использовалась база данных аннотаций научных статей с информацией о ключевых словосочетаниях, присвоенных профессиональными индексаторами. Оценка алгоритмов производилась с помощью трех известных мер: точности, полноты и сбалансированной *F*-меры. Правильными считались автоматически выделенные словосочетания, совпадающие со словосочетаниями, выделенными экспертами.

Результаты экспериментов приведены в табл. 1.

Табл. 1. Результаты экспериментов по оценке качества выделения словосочетаний

Алгоритм	Словосочетаний		Правильных		Точность	Полнота	F-мера
	Всего	Среднее	Всего	Среднее			
Новый							
Attribute, n-граммы, с фильтрацией стоп-слов	6 406	12,8	2 240	4,5	**35,0**	45,8	**39,7**
Attribute, n-граммы	6 517	13,0	2 256	4,5	34,6	46,2	39,6
Attribute, биграммы	7 851	15,7	1 793	3,6	22,8	36,7	28,1
Все отношения, биграммы	20 389	40,8	1 927	3,9	9,5	39,4	15,2
TextRank							
Undirected, window=2	6 784	13,7	2 116	4,2	31,2	43,1	36,2
Directed, forward, window=2	6 662	13,3	2 081	4,1	31,2	42,3	35,9
Hulth							
Ngram with tag	7 815	15,6	1 973	3,9	25,2	**51,2**	33,9
NP-chuncks with tag	4 788	9,6	1 421	2,8	29,7	37,2	33,0
Pattern with tag	7 012	14,0	1 523	3,1	21,7	39,9	28,1

Сравнение проводилось с алгоритмом с обучением (*Hulth*) и алгоритмом без обучения (*TextRank*). Оба алгоритма опираются при выделении словосочетаний на

позиции слов и применяют лингвистическую информацию лишь для фильтрации слов определенных частей речи (существительных и прилагательных).

Показано, что предлагаемый алгоритм дает на данной выборке более качественные результаты, чем другие. Это можно объяснить более глубоким лингвистическим анализом текстов, обеспечиваемым применением концептуальных графов. Такой анализ позволяет, во-первых, отфильтровывать слова, стоящие рядом, но не связанные друг с другом, а во-вторых, обнаруживать связи между словами, разделенные несколькими другими, что невозможно при чисто статистическом подходе.

Для полнотекстового поиска использовалась опубликованная в сети Интернет справочная документация к продуктам компании SmartBear и еще одна оценка релевантности: *nDCG*, учитывающая позицию результата в выдаче. Вычисляется величина *nDCG* по следующей формуле:

$$nDCG_p = \frac{rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}}{r_{max} + \sum_{i=2}^{p} \frac{r_{max}}{\log_2 i}}, rel = 0..2, p = 1..N, r_{max} = 2$$

где rel_i – оцененная асессором релевантность *i*-того результата, p – количество результатов поиска, r_{max} - максимально возможная величина релевантности.

График, показывающий изменение средней величины *nDCG* в зависимости от количества оцененных запросов представлен на рис.3(вертикальная ось соответствует среднему значению *nDCG*, горизонтальная – количеству запросов):

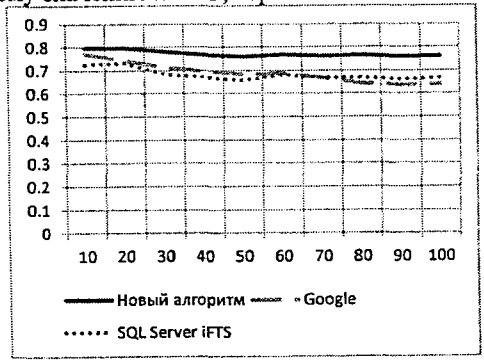

Рис.3. Зависимость величины *nDCG* от количества поисковых запросов

Показано, что по мере увеличения количества запросов предлагаемый алгорит дает для текущей выборки более качественные результаты, чем два других.

В диссертации для обобщенной оценки качества всего алгоритма в цело использовалась база данных аннотаций технических статей, разделенны экспертами по тематикам. Для оценки качества алгоритмов использовалась мер *P@N* (точность N наиболее релевантных результатов поиска), значения *N* бралис равным 5, 10 и 15 соответственно. Каждая статья корпуса поочередно выступала качестве запроса, релевантными документами для которого считались статьи из то же тематики.

Результаты проведенного эксперимента приведены в табл.2.

Табл.2. Величина P@N для четырех сравниваемых алгоритмов

Алгоритм	P@5	P@10	P@15
Новый алгоритм	**74,3%**	**69,4%**	**64,3%**
PATeR	65,8%	53,3%	42,7%
CTR	60,7%	50,4%	41,3%
TP	66,0%	53,9%	43,1%

Показано, что предлагаемый алгоритм дает на данной выборке существенно олее качественные результаты, чем другие аналогичные алгоритмы. Так точность ыделения словосочетаний по предлагаемому алгоритму превзошла точность звестного алгоритма *TextRank*, лежащего в основе недавно разработанного оискового алгоритма *PATeR*, поэтому и точность поиска предлагаемого алгоритма казалась выше. Аббревиатуры CTR и TP обозначают два других алгоритма, вляющихся модификациями алгоритма Okapi BM25,в которых при вычислении елевантности учитывается меньшее количество факторов, чем в разработанном горитме. Стоит также учесть более сложное условие плавающего контекстного кна, основанного на искусственном увеличении позиций слов в документах, что кже является усовершенствованием по отношению к существующим алгоритмам.

Таким образом, подтверждается эффективность разработанных горитмических решений.

В заключении сформулированы основные результаты и выводы по работе.

В приложения вынесены схемы алгоритмов выделения словосочетаний из екста и релевантности документов.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

1. Подтверждена эффективность применения концептуальных графов в ачестве семантических моделей полнотекстовых запросов к базам данных.
2. Разработан алгоритм индексирования тестов информационного ресурса истем технической поддержки, позволяющий неявно сохранять информацию о наках препинания в полнотекстовых индексах.
3. Разработан алгоритм, позволяющий выделять из текстов ключевые ловосочетания с использованием концептуальных графов.
4. Разработан алгоритм полнотекстового поиска с контекстным окном лавающего размера, использующий при вычислении релевантности полученные анее словосочетания и полнотекстовые индексы.
5. На основе экспериментальных исследований подтверждена эффективность редложенных алгоритмов.
6. Разработано инструментальное программное обеспечение, реализующее редложенные алгоритмы в виде системы полнотекстового поиска, которая может ыть интегрирована с существующими информационными системами.
7. Результаты диссертационной работы внедрены в практическое спользование в ООО «Автоматизированное обеспечение качества», тульский илиал компании SmartBear Software.

ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

1. **M. Bogatyrev, A. Kolosoff Using Conceptual Graphs for Text Mining in Technical Support Services // Pattern Recognition and Machine Intelligence 4th International conference, PReMI 2011. Proceedings. LNCS, vol. 6744. Springer-Verlag. Heidelberg, 2011.–p.p. 466-473**

2. **Колосов А.П. Выделение словосочетаний из текстов при помощи концептуальных графов // В мире научных открытий. Красноярск: НИЦ, 2012. №1.1(25). – с. 181-191**

3. **Колосов А.П. Алгоритм полнотекстового поиска с обучением на основе статистических данных // Известия ТулГУ: Технические науки. Вып. 6. Ч. 2 Тула: Издательство ТулГУ, 2011. –с. 462-471**

4. Колосов А.П., Богатырев М.Ю. Алгоритм полнотекстового поиска по длинным запросам // Труды XIII Всероссийской научной конференции RCDL'2011. Воронеж: Издательско-полиграфический центр Воронежского государственного университета, 2011. – с. 151-157

5. Колосов А.П., Богатырев М.Ю. Полнотекстовый поиск в порталах технической поддержки // Интернет и современное общество: сборник тезисов докладов. СПб: МультиПроджектСистемСервис, 2011. – с. 57-62

6. Колосов А.П., Николаев Д.С., Муравьев А.Н. и др. Библиотека полнотекстового поиска с обучением на основе статистических данных, использующая гибридный алгоритм поиска // Свидетельство о государственной регистрации программы для ЭВМ №2011612246. РОСПАТЕНТ 17.03.2011

7. Колосов А.П., Николаев Д.С., Муравьев А.Н. и др. ClickHelp, веб-приложение для разработки документации // Свидетельство о государственной регистрации программы для ЭВМ №2012613286. РОСПАТЕНТ 6.04.2012

Изд. лиц. ЛР №020300от 12.02.97. Подписано в печать 18.04.2012г.
Формат бумаги 60x84 1/16 .Бумага офсетная.
Усл. печ.л .0,93 Уч.-изд.л. 0,8 Тираж 100 экз. Заказ 020
Тульский государственный университет 300012, г. Тула, пр. Ленина, 92
Отпечатано в Издательстве ТулГУ 300012, г. Тула, пр. Ленина, 95