Московский государственный университет имени М.В.Ломоносова

Факультет Вычислительной математики и кибернетики

Кафедра системного программирования

Курсовая работа:
Исследование и разработка методов реализации вопросно-ответных систем.

Выполнил:
студент 428группы
Агаев Нурлан Закирович

Научные руководители:
с.н.с., к.ф.-м.н. Турдаков Денис Юрьевич

Москва

2012

Содержание

Аннотация

В данной работа рассмотрена задача вопросно-ответного поиска, исследована предметная область вопросно-ответного поиска и рассмотрены существующие решения для английского языка. Разработана и реализована модель простой вопросно-ответной системы для русского языка, проведена частичная оценка разработанной системы.

1 Введение

В последние годы наблюдается бурный рост объема общедоступной информации. Вследствие чего появилась проблема для современного общества – проблема поиска и получения нужной информации. Эту проблему усугубляет и то, что в настоящее время информация, доступная в сети Интернет, имеет очень высокий уровень динамики. В каждый момент времени появляются новые метриалы и факты. Постоянный рост объема информационных массивов и их обновление делают сложным, а зачастую практически невозможным , учёт всей информации. По причине этого данные, представляющие ценность, зачастую остаются невостребованными.

В XXI веке деятельность людей, коллективов, организаций и компаний в большей степени зависит от имеющейся у них информации, а так же способности быстро её найти. Имея доступ к представленной в сети Интернет информации пользователю хотелось бы получать только нужную ему её часть, в то время как поисковые системы представляют для этого малые возможности. Пользователю приходится самому продолжать искать информацию среди предложенной ему поисковой машиной. При использовании поисковиков пользователь получает большое количество ссылок на документы, и часто ему требуется продолжать поиск интересующей его информации, что затрудняет её восприятие. Таким образом получается противоречие между большим количеством доступной информации и ограниченными возможностями по её поиск и получением.

Когда мы хотим что-то узнать, мы спрашиваем – задаём вопрос, что, в общем, и естественно в процессе познания. Большинство систем по поиску информации, не имеют возможности отвечать на наши вопросы. Для поиска и получения человеку нужно сформировать запрос из ключевых слов и задать его поисковой машине.

В последнее время повысился интерес к разработке интеллектуальным и нетрадиционным механизмам поиска и получения информации. Интернет стал рассматриваться как потенциальная большая база знаний, для работы с которой требуются специальные инструменты. Сегодня термин «информационный поиск» (англ. Information retrieval) включает в себя поиск текстовых документов, поиск изображений, поиск видео, многоязыковой поиск, географически-зависимый поиск. Помимо этого к

информационному поиску можно причислить и поиск ответа на вопрос. В последние годы увеличилось количество проектов таких систем в данной области - поиска ответа на вопрос на естественном языке. Вопросно-ответные системы – это программные комплексы, которые умеют обрабатывать введенные пользователем вопросы на естественном языке и давать на них краткие ответы, состоящие из слов или предложений. В отличие от традиционных поисковых машин системы вопросно-ответного поиска могут обрабатывать вопрос на естественном языке и выдавать не список ссылок и документов, а ответ – сжатый и лаконичный. Вопросно-ответные системы имеют другую цельпо сравнению с традиционными система информационного поиска. Их задача – найти фрагмент документа, содержащий точный и краткий ответ на вопрос. Источником информации для таких систем обычно служит большая коллекция текстовых документов, например, общедоступные страницы сети Интернет. Таким образом, вопросно-ответные системы образуют класс интеллектуальных систем информационного поиска.

При разработке и реализации вопросно-ответных систем приходится иметь дело с естественным языком, а именно с фразами и предложениями, сформированные по опрделенным правилам этого языка. Поэтому создание систем вопросно-ответного поиск – далеко не простая задача. При проектировании таких систем используются относительно новые методы компьютерной лингвистики (англ. NLP – Natural Language Processing), требуется применение адекватных лингвистических средств по работе с естественным языком, при этом результат работы системы существенно зависит от качества из реализации.

В последние годы появилось немало проектов в данном направлении. Причем это проекты, в которых разработаны технологии обработки простых вопросов, ответы на которые состоят из одного слова или небольшого предложения. Эти проекты обходят стороной вопросы более сложного вида, например, вопросы причины, описания объектов и т.д.

Авторы большинства создаваемых в настоящее время систем вопросно-ответного поиска ориентируются в основном на английский язык. Типовая система состоит из большинства сложных частей, которые предназначены для анализа вопроса и обработки текстовых документов с учетом правил и особенностей естественного языка. Например, при анализе вопроса происходит синтаксический и семантический разбор предложения. Части и подпрограммы анализа разрабатываются обычно независимыми группами и, как уже было отмечено, работают с английским языком. Поэтому для русского языка

затруднительно применить готовые модули обработки предложений. Данные факты являются серьезным препятствием для российских разработчиков, желающих создать системы для русского языка. При анализе вопроса и текстовых документов, для извлечения из последних ответа используют различные анализаторы языка: синтаксические, морфорлогические, семантические. К сожалению, выбор русскоязычных систем анализа, представленных в свободном доступе довольно скуден.

Любая серьезная система вопросно-ответного поиска должна каким-либо образом производить анализ структуры вопросительного предложения, опираясь на знания о конкретном естественном языке, на котором сформулирован вопрос. Вследствие этого изучение и сравнение решений, рассчитанных на разные языки – практически невыполнимая задача. Однако для исследования принципов работы существующих систем, технологий решения задачи вопросно-ответного поиска, позволят сделать вывод о глубине анализа и обработки вопросительных предложений и текстовых документов, что будет использовано при построении системы для русского языка.

2 Постановка задачи.

Целью данной курсовой работы является исследование методов реализации современных вопросно-ответных систем и разработка и построение простой вопросно-ответной системы для русского языка. Для достижения данной цели поставлены следующие задачи:

1. Исследовать существующие решения в области вопросно-ответных систем.
2. Разработать модель вопросно-ответной системы для русского языка.
3. Реализовать разработанную модель.
4. Создать корпус вопросов для тестирования и разработать методы тестирования системы.
5. Протестировать и оценить систему.

3 Обзор предметной области.

Системы вопросно-ответного поиска в сравнении с традиционными поисковыми системами получают вопросительно предложение на естественном языке (на английском, на русском и т.д.), а не набор ключевых слов, и возвращают краткий ответ, а не список документов и ссылок. Современные системы информационного поиска позволяют нам получить список целых документов, которые могут содержать интересующую информацию, при этом оставляют пользователю работу по получению нужных данных из документов, упорядоченных по уровню релевантности запросу. Например, пользователь вводит следующий вопрос: «Кто является президентом России?» и в качестве ответа получает имя человека, а не список релевантных ссылок на документы. Таким образом, нахождение ответа на вопрос извлечением небольшого отрывка текста из документа, в котором непосредственно содержится сам ответ, в отличие от информационного поиска совсем другая задача.

Большая часть существующих проектов в области вопросно-ответного поиска предназначены для английского языка. Если сравнить несколько работ в данной сфере исследований, то можно прийти к стандартной схеме устройства вопросно-ответных систем. Как правило, работа типовой вопросно-ответной системы состоит из нескольких этапов:

1.этап анализа вопроса, введенного пользователем;

2.этап информационного поиска;

3.этап извлечения ответа.

На первом этапе производится ввод вопроса на естественном языке и первичная обработки и формализация предложения различными анализаторами (синтаксическим, морфологическим, семантическим), определяются соответствующие его атрибуты для дальнейшего их использования. Далее на втором этапе происходит поиск и анализ документов - отбираются документы и их фрагменты, в которых может содержаться ответ на исходный вопрос. На третьем этапе происходит извлечение ответа: система, получая текстовые документы или их фрагменты, извлекает из них слова, предложения или отрывки текста, которые могут стать ответом.

Следует отметить, что важную роль в результатах и разработке играет использование различных словарей-тезаурусов. Применение данных словарей решают задачу определения типов сущностей для выявления ответов, нахождение начальной формы слов для использования их в поисковых запросах. Также данные словари используются для нахождения синонимов слов.

С точки зрения архитектуры практически любую вопросно-ответную систему можно разделить на три модуля (рис.1):

1.модуль обработки вопроса;

2.модуль поиска документов и получения текстовых фрагментов;

3.модуль формулировки и получения ответа.

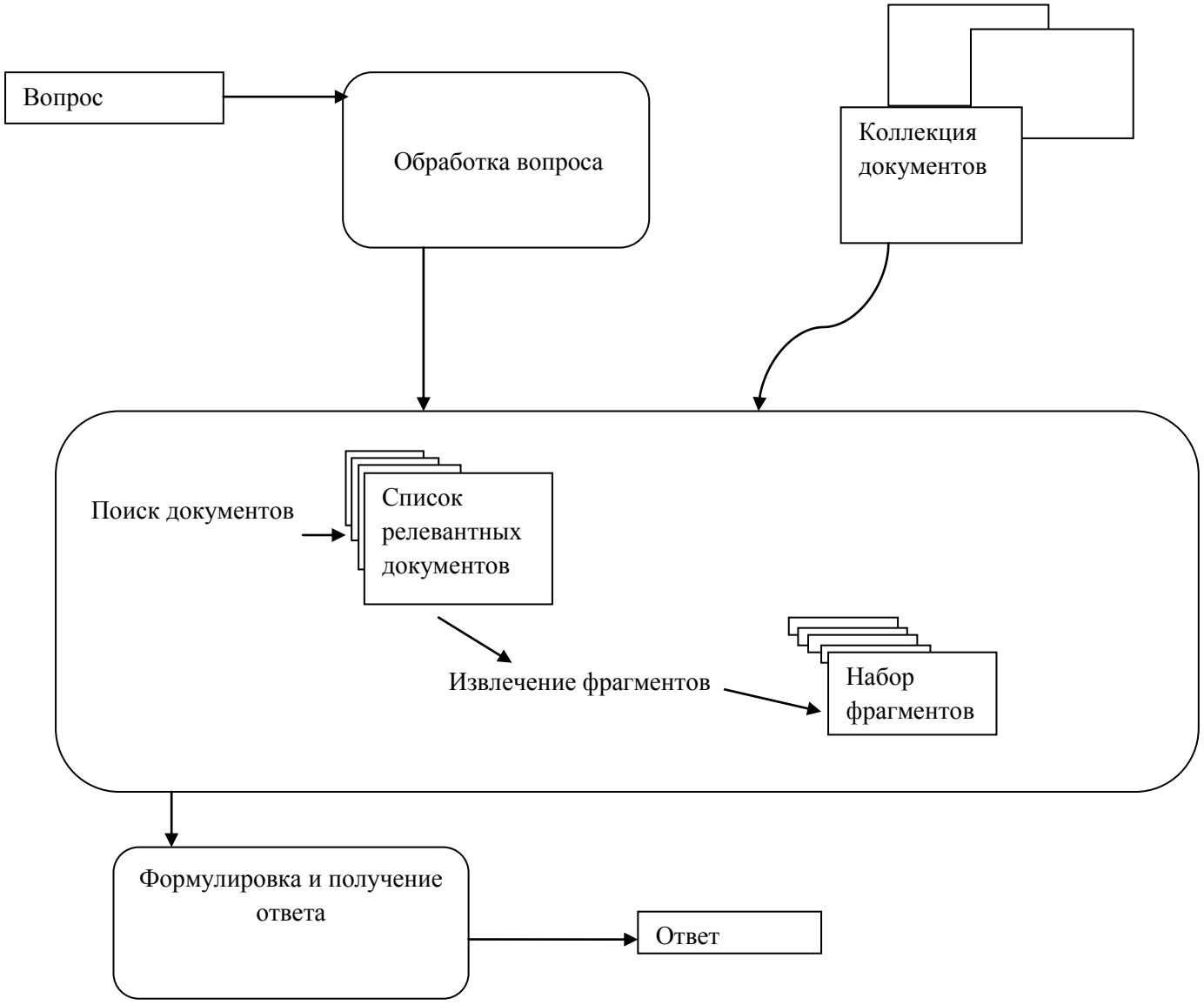


Рис.1 - Общая схема работы вопросно-ответной системы (недетализированная диаграмма).

При этом система может иметь и другие компоненты для улучшения производительности, в виде анализаторов текста, распознавателей имён, автоматических переводчиков и других подпрограмм. Все эти дополнения могут быть включены в три основных модуля, упомянутых выше, или быть отдельными так называемыми «черными ящиками», способствующих в решении общей задачи вопросно-ответного поиска.

Вопросно-ответная система способна обрабатывать некоторые предопределённые классы вопросов. Наиболее успешно решается задача ответа на вопросы об определениях (англ.: definitional) и фактографические (англ.: factoid). Сегодня системы ограничиваются поиском текста ответа и не занимаются логическим выводом неявной информации.

Далее рассматриваются принципы работы каждой из частей типовой вопросно-ответной системы.

3.1 Анализ вопроса.

На вход системе от пользователя поступает запрос в форме вопросительного предложения. Рассмотрим анализ вопроса на примере система LASSO [5]. Вопросно-ответная система LASSO была разработана в лаборатории компьютерной лингвистики Южного Методического университета, штат Даллас, США. Данная система является в настоящее время одной их наиболее развитых. В LASSO методы нахождения ответа на вопрос основаны на использовании относительно новых способов компьютерной лингвистики. При обработке вопроса учитывается синтаксическая и семантическая информация, характеризующая запрос.

Модуль анализа вопроса в системе LASSO выполняет следующие основные задачи:

1) определение типа вопроса;

2) определение типа ожидаемого ответа;

3) определение фокуса вопроса;

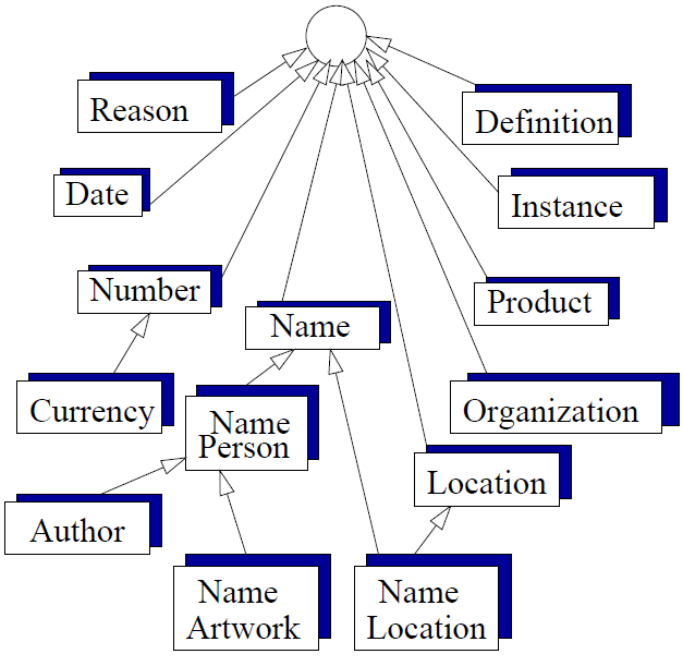
4) определение ключевых слов для формирования поискового запроса.

Вопросы, задаваемые системе можно разделить на несколько видов. Таким образом, на первом этапе работы производится определение типа вопроса. Для этого вопросительное предложение классифицируется системой и относится к одному из заранеее определенных типов вопроса. . В таблице 1 приведен список некоторых классов вопросов, используемых в системе TEQUESTA для английского языка [2]. Например, примитивным способом является определение типа вопроса по вопросительным словам, которые содержит запрос к системе: «кто», «что», «какой», «где», «когда» и т.д. Конкретная вопросно-ответная система может работать с вопросами одного-двух определенных классов, а некоторые расширенные системы могут работать с вопросами конечного набора классов.

Таблица 1

Класс вопроса	Описание
capital	столица страны или штата «What is the capital of Kentucky?»
date	дата события «When did the story of Romeo and Juliet take place?»
location	географическое местонахождение предмета или место события «Where did Golda Meir grow up?»
date-birth	дата рождения какой-либо персоны «When was King Louis XIV born?»

Как можно заметить, класс date-birth является подклассом класса date. То есть подразумевается, что имеется некая встроенная предопределенная иерархия классов вопросов, которая обычно разрабатывается авторами системы – так называемая таксономия классов. На рис.1 приведен пример таксономии классов вопроса в системе для английского языка QaLasie. Таксономия типов представлена в виде древовидной структуры. Каждому узлу дерева соответствует тип вопроса, тип ответа, фокус вопроса, ключевые слова для вопроса (по которым можно распознать тип вопроса). Структура и вид дерева зависит от функций и методов разработки системы.



Классификация вопроса может быть выполнена различными способами. Одним из простых и достаточно эффективных способов является использование шаблонов с применением регулярных выражений. Определение типа ответа зависит от порядка применения шаблонов регулярных выражений. Данный способ используется в вопросно-ответной системе для английского языка (Min-Yuh DAY, 2003). В Таблице 2 приведен список классов с соответствующими шаблонами в виде регулярных выражений.

Метод шаблонов с регулярными выражениями успешно использовался в системах, участвовавших в TREC-2002 (2003), в котором организаторы подготовили вопросы для дорожки QA вручную. Однако, уже в TREC-2004 были предложены задания на основе реальных запросов пользователей и те системы, которые не применили иные методы анализа вопроса, заметно отстали от адаптировавшихся лидеров.

Таблица 2. Примеры шаблонов для классификации вопросов на английском языке

Класс вопроса	Примеры шаблонов
date	/[Ww]hen /, /[Ww](hat|hich) year /
location	/[Ww]here(\'s)? /, / is near what /
capital	/[Ww]hat is the capital /, /[Ww]hat is .+\'s capital/
date-death	/[Ww]hen .* die/, /[Ww](hat|hich) year .* die/
date-birth	/[Ww]hen .* born/, /[Ww](hat|hich) year .* born/
expand-abbr	/stand(s)? for(what)?\s*?/, /the abbreviation .+ mean\s*?/

Для того чтобы найти нужный ответ на вопрос среди большого количества фрагментов текста, требуется знать, что именно искать. Поэтому вводится такое понятие, как тип ожидаемого ответа. Тип ожидаемого ответа – это класс запрашиваемой пользователем информации согласно некоторой ранее заданной таксономии. Обычно тип ответа можно определить по самому вопросу. Хорошая классификация вопросов способствует более точному распознанию типа ожидаемого ответа.

При разработке системы LASSO было выяснено, что определения типа вопроса недостаточно для эффективного нахождения ответа. Было предложено еще определять такое понятие, как фокус вопроса. Фокус вопроса (англ.: question focus) – это такие сведения, содержащиеся в вопросе, которые несут в себе информацию об ожидаемом ответе. Например, в вопросе «Какое озеро является самым глубоким в России?» фокусом вопроса будет «самое большое озеро». Понятие фокуса вопроса было введено и в ряде других вопросно-ответных систем. Оно используется на последующих этапах работы вопросно-ответной системы. Экспериментально было доказано, что фокус вопрос облегчает определение и формулировку ответа, то есть улучшает точность системы.

В Таблице 3 приведены примеры результатов классификации вопросов на английском языке, которые были представлены для тестирования вопросно-ответных систем на конференции TREC (Text Retrieval Conference — серия конференций, сконцентрированных на исследовании различных областей информационного поиска и их задач). Представлены вопросы, тип вопроса, тип ответа, фокус вопроса.

Таблица 3 – примеры из 200 вопросов, представленных на конференции TREC-11

Q-class	Q-subclass	Nr.Q	Nr. Q answ.	Answer type	Example of question	Focus
what		64	54			
	basic what	40	34	MONEY/NUMBER/ DEFINITION/TITLE/ NNP/UNDEFINED	*What was the monetary value of the Nobel Peace Prize in 1989?*	monetary value
	what-who	7	7	PERSON/ ORGANIZATION	*What costume designer decided that Michael Jackson should only wear one glove?*	costume designer
	what-when	3	2	DATE	*In what year did Ireland elect its first woman president?*	year
	what-where	14	12	LOCATION	*What is the capital of Uruguay?*	capital
who		47	37	PERSON/ ORGANIZATION	*Who is the author of the book "The Iron Lady: A Biography of Margaret Thatcher"?*	author
how		31	21			
	basic how	1	0	MANNER	*How did Socrates die?*	Socrates
	how-many	18	13	NUMBER	*How many people died when the Estonia sank in 1994?*	people
	how-long	2	2	TIME/DISTANCE	*How long does it take to travel from Tokyo to Niigata?*	–
	how-much	3	2	MONEY/PRICE	*How much did Mercury spend on advertising in 1993?*	Mercury
	how-much-<modifier>	1	0	UNDEFINED	*How much stronger is the new vitreous carbon material invented by the Tokyo Institute of Technology compared with the material made from cellulose?*	new vitreous carbon material
	how-far	1	1	DISTANCE	*How far is Yaroslavl from Moscow?*	Yaroslavl
	how-tall	3	3	NUMBER	*How tall is Mt. Everest?*	Mt. Everest
	how-rich	1	0	UNDEFINED	*How rich is Bill Gates?*	Bill Gates
	how-large	1	0	NUMBER	*How large is the Arctic refuge to preserve unique wildlife and wilderness value on Alaska's north coast?*	Arctic refuge
where		22	16	LOCATION	*Where is Taj Mahal?*	Taj Mahal
when		19	13	DATE	*When did the Jurassic Period end?*	Jurassic Period

Одним из успешных приемов при анализе вопроса является использование синтаксических шаблонов. В основе метода лежит предположение, что фокус вопроса часто находится в определённом синтаксическом отношении с вопросительным словом, может быть не в одном, но набор вариантов этих отношений ограничен. В результате синтаксического разбора можно получить дерево разбора (Рис. 4).

Вот пример синтаксического шаблона представленного в строковом виде для распознавания фокуса, используемого в системе OpenEphyra [11] :

(ROOT (SBARQ (WHNP (WP What)) (SQ (VP (VBZ is) (NP (NP (DT the) (NN name)) (PP (IN of) (*NP !)))))))

Здесь в скобочной нотации задано синтаксическое дерево со словами или их синтаксическими/морфологическими метками в узлах. Такой шаблон дерева сравнивается с реальным деревом вопроса и, в случае совпадения, фокусом считаются члены предложения, соответствующие позиции "!" в шаблоне.

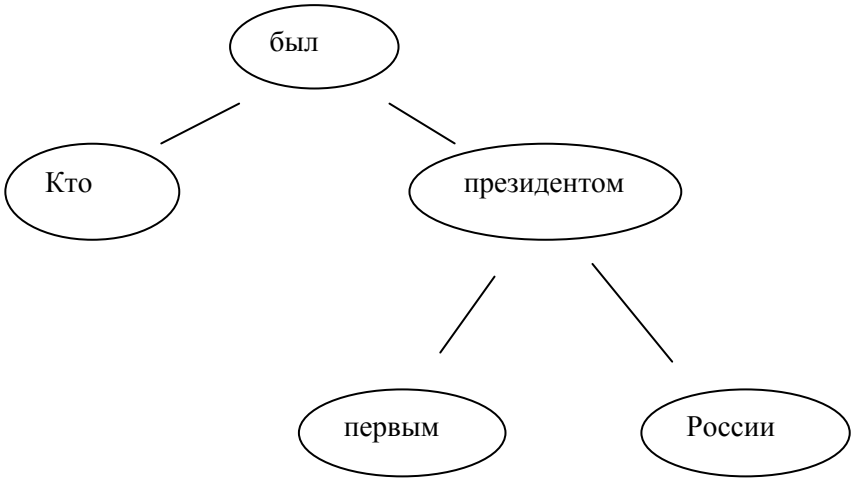


Рис. 4 - Синтаксическое дерево, построенное системой Dwarf «Кто был первым президентом России?»

Также многие современные классификаторы вопросов основываются на машинном обучении с учителем. Такие классификаторы обучаются на больших коллекциях вопросов, которые размечены вручную такими атрибутами, как тип вопроса и тип ответа. Похожие методы используются в системах Joost [3] и [4]. Такой подход подразумевает применение заранее приобретенных знаний. Рассмотрим похожий прием на примере системы Joost. Перед тем как системе буден задан вопрос от пользователя, она полностью просматривает коллекцию текстовых документов в поисках потенциальных ответов на вопросы определенных типов – таких, как названия географических мест, даты, аббревиатуры и т.д. Далее такие ответы извлекаются в офф-лайн корпус и хранятся в виде табличных структур для быстрого обращения и получения при обработке вопроса от пользователя. Такие офф-лайн методы при тестах показали высокую эффективность (Fleischmann, 2003).

Далее при анализе вопроса извлекаются ключевые слова. Ключевые слова– слова, используемые в качестве запроса для информационного поиска. Данные слова будут для поиска и выбора документов. При разработке многих систем было доказано, что выбор ключевых слов очень сильно влияет на результат информационного поиска. Набор релевантных документов очень зависит от слов, выбранных для поиска. Рассмотрим, как выявляются слова в качестве поискового запроса в некоторых вопросно-ответных системах.

Ключевыми словами во многих системах становятся слова из вопросительного предложения, являющиеся именами существительными, глаголами. Во множество ключевых слов включаются в первую очередь имена собственные слова и словосочетания, заключенные в кавычки. Если в вопросе есть словосочетание, заключенное в кавычки, то слово из него может быть отдельным ключевым словом и это словосочетание включается в поисковый запрос как одно целое.

Синтаксический анализ вопросительного предложения позволяет рассматривать отдельные слова из предложения. Так, например, системе Lasso процесс получения ключевых слов состоит в следующем алгоритме - при последовательном анализе предложения в множество ключевых слов добавляет следующие слова:

1. все слова и выражения, заключенные в кавычки;

2. все имена собственные;

3. все пары имя существительное – имя прилагательное;

4. все остальные существительные;

5. все глаголы;

6. фокус вопроса;

В таблице 4 показан пример извлечения ключевых слов.

Таблица 4 – пример ивзлечения ключевых слов для вопросов с конференции TREC-9

What is the name of the “female” counterpart to el Nino, which results in cooling temperatures and very weather?	How much could you rent a Volkswagen bug in 1966?
female female El Nino female El Nino female El Nino dry weather female El Nino dry weather cooling temperatures ...	Volkswagen Volkswagen bug Volkswagen bug rent

Также перед тем, как выявлять ключевые слова, из вопроса могут быть удалены вопросительные слова (например: «кто», «что», «какой»), так как они обычно не встречаются в ответах. Удалением этих слов можно повысить полноту (англ. recall). результата информационного поиска. Для этого можно опять же использовать регулярные выражение, как это делается в системе Lamp.

Другим способом формирования поискового запроса является переформулирование вопросительного предложения. Согласно определенным правилам вопрос перефразируется в утвердительную форму части предложения, в котором содержится ответ. Например, вопрос «Где находятся горы Анды?» переформулирутся в часть предложения-ответа так: «Анды находятся в». Таким образом, можно применять несколько правил такого рода и расширять множество поисковых запросов. Далее представлены примеры правил перефразирования вопрос в вопросно-ответной системе Lin для английского языка:

Wh-word did A verb B ? → A verb+ed B

Where is A ? → A is located in

Для выявления ключевых слов для формирования поисковых запросов также используются различные словари-тезаурусы. В вопросно-ответных системах широкое применение словарь WordNet. WordNet - диалоговая лексическая справочная система, разработанная в Принстонском университете. Она представляет собой семантическую сеть, в которой английские существительные, глаголы, и прилагательные, организованы в наборы синонимов, каждый из которых представляет одну лексическую единицу. Базовой словарной единицей в WordNet является не отдельное слово, а так называемый

синонимический ряд («синсеты»), объединяющий слова со схожим значением и по сути своей являющимися узлами сети. Для удобства использования словаря человеком каждый синсет дополнен дефиницией и примерами употребления слов в контексте. Слово или словосочетание может появляться более чем в одном синсете и иметь более одной категории части речи. Каждый синсет содержит список синонимов или синонимичных словосочетаний и указатели, описывающие отношения между ним и другими синсетами. Слова, имеющие несколько значений, включаются в несколько синсетов и могут быть причислены к различным синтаксическим и лексическим классам.

Таким образом, при использовании WordNet может быть создан целый набор поисковых запросов, получаемых с помощью расширения множества ключевых слов различными синонимами. Например, в вопросе «Где находится озеро Байкал?» глагол «находится» можно заменить синонимичным «располагается». Использование синсетов при применении тезауруса позволяет увеличить вероятность нахождения документов, содержащих ответ на вопрос.

После того, как определены ключевые слова, то есть сформулирован поисковой запрос, работа системы переходит на следующий этап – этап информационного поиска.

3.2 Информационный поиск

На данном этапе производится поиск релевантных документов, а также получение текстовых фрагментов, содержащих ответ. В вопросно-ответных системах данный модуль представляет собой обычную поисковую машину, на вход которой поступает запрос из ключевых слов.

Большинство современных систем вопросно-ответного поиска используют готовые решения для поиска релевантных документов. После того как получен набор текстовых документов, релевантных запросу, извлекаются фрагменты, в которых велика вероятность получения ответа на вопрос.

Рассмотрим работу данного модуля у системы Joost. Авторы системы используют поисковую машину ZPrise IR System. Поисковик Zprise принимает запрос, который состоит из ключевых слов, полученных в первом модуле, и возвращает множество документов, в которых содержатся слова, входящие в запрос. После этого системой Joost отбираются только те текстовые документы, в которых содержатся именно все ключевые

слова. В ходе исследований авторами системы было выяснено, что это увеличивает полноту (recall) результата поиска.

Обычно в качестве элемента поиска современные вопросно-ответные системы используют обычные машина интернет-поиска. Это решение связано с тем, что не требуется заново разрабатывать решения информационного поиска. Поисковые машины предлагают интерфейс прикладного программирования (англ. application programming interface API) для разработчиков. Таким образом при поисковом запросе можно получить упорядоченный набор релевантных документов.

Для того, чтобы получить фрагменты из документов, которые могут содержать ответ с наибольшей вероятностью, текст документа делится на части – одним из способов является деление на абзацы. Затем выбирается тот фрагмент (абзац), который содержит все ключевые слова или наибольшее их количество. Например, в системе вопросно-ответноого поиска, разработанной в Южном Методистском Университете США (Moldovan, 2004), применяется немного другой способ получения фрагментов (параграфов) документов для последующего извлечения из них ответа, который рассматривает чуть более сложный случай выбора отрывка текста. Пусть имеется поисковой запрос, состоящий из следующего набора ключевых слов : {k1,k2,k3,k4}. Текст документа разделен на фрагменты (параграфы) и один из параграфов содержит включения k1,k2,k3, причем k1 и k2 встречаются два раза, k3 – один (см.рис 1). Вводится понятие окна параграфа – оно включает в себя, весь текст между двумя ключевыми словами – одним, расположенным выше остальных по тексту, вторым – ниже. Рассматриваются всевозможные включения ключевых слов во фрагмент документа (окно параграфа). Таким образом, можно для данного случая можно получить 4 случая окна параграфа:

[k1-1, k2-1, k3], [k1-2, k2-1, k3], [k1-1, k2-2, k3], [k1-2, k2-2, k3].

Каждое из окон параграфов оценивается - для каждого из них рассчитываются следующие величины:

1.Same_word_sequence_score – количество слов из вопроса, встречающихся во окне параграфа в таком же порядке;

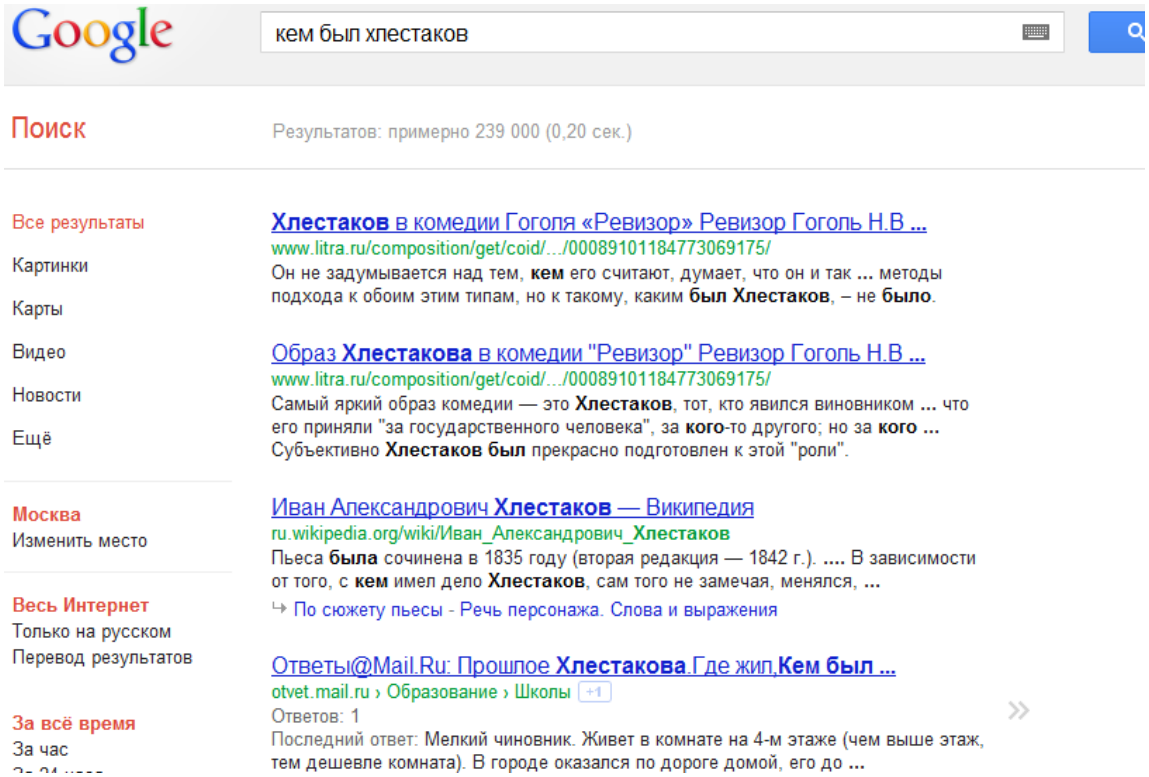
2.Distance_score – количество слов, разделяющее самые удаленные ключевые слова в окне параграфа;

3.Missing_score: количество слов из запроса, не встречающихся в окне параграфа.

Далее происходит сортировка и выбор фрагмента документа, причем сравниваются величины всех окон всех параграфов.

Но в связи с высоким развитием систем информационного поиска для общих (не узко специализированных) вопросно-ответных систем достаточно бывает использовать результаты поисковых запросов к ним, традиционным поисковым машинам. Так, например, в построении вопросно-ответной системы в данной курсовой работу применялась поисковая машина Google. Также отпадает задача выявления фрагментов, содержащих ответ на вопрос, так как мы можем использовать фрагменты, созданные поисковой машиной – сниппеты. Сниппет – это небольшой отрывок текста из найденной поисковой машиной веб-страницы, использующихся в качестве описания ссылки в результатах поиска. Как правило, они содержат контекст, в котором встретились ключевые слова в тексте на странице. Просмотрев сниппет, можно приблизительно понять, соответствует ли страница именно вашему запросу, даже не открывая самой этой страницы. Например, на рис.2 показан пример результата поиска со сниппетами для первых 4-х ссылок.

Рис.2



3.3 Извлечение ответа

Данный модуль распознает и извлекает из текстовых документов ответ на вопрос. Ключевую роль в распознании ответа играет тип ответа. Так как почти едва ли не всегда тип ответа не ясен ни из вопроса, ни из ответа, необходимо полагаться на лексико-семантическую информацию, предоставляемую синтаксический анализатором, который распознаёт в предложении именованные сущности (например, имена людей, организаций и т. д.).

Стратегия извлечения ответа из текстового фрагмента зависит от типа ожидаемого ответа. Например, для таких типов ответа, как географические местоположения, имена людей (PERSON, LOCATION, COUNTRY) в извлечении ответа будут использованы алгоритмы распознавания имён собственных.

Общая идея решения задачи извлечения ответов состоит в следующем: выявляются так называемые кандидаты для ответа – слова, которые могут рассматриваться как ответ на вопрос; затем все кандидаты оцениваются и выбирается самый подходящий, то есть тот, который имеет наивысшую оценку.

Есть два основных класса алгоритмов, применяемых для решения задачи задачи извлечения ответа: одни основываются на применении специальных шаблонов ответов, другой на применении N-грамм.

В извлечении ответа при помощи шаблонов используются информация о типе ожидаемого ответа, полученная на первом этапе работы системы, и правила регулярных выражений. Например, в системах для английского языка в вопросах с типом ожидаемого ответа HUMAN будет логично искать в качестве ответа именованые сущности. Поэтому в данном случае к фрагменту применяется распознаватель именованных сущностей. Тогда ответом на данный вид вопроса будут слова, представляющие собой имена собственные. Также при поиска ответа из фрагмента после его анализа выявляются все слова, имеющие метку с типом ожидаемого ответа.

Таким образом, в примерах, представленных ниже, подчеркнутые слова рассматриваются как кандидаты для ответа

-*Who is president of Russia?*

-*Vladimir Putin, president of Russia, visited three asian countries last week.*

-*Какова глубина озера Байкал?*

- *Современное значение максимальной глубины озера — 1642 м*

В более сложных случаях, где не требуется поиск именованных сущностей или числе, используются специально созданные шаблоны из регулярных выражений для извлечения кандидатов. В таблице 5 представлены несколько шаблонов из системы Pasca (2003).

ТАБЛИЦА 6 – Примеры шаблонов из системы Pasca

Шаблон	Вопрос	Ответ
<AP> such as <QP>	What is autism?	“,developmental discorders such as autism”
<QP>, a <AP>	What is a caldera?	“the Long Valley caldera, a volcanic crater 19 miles long”

Шаблоны можно создавать как вручную, так и автоматическими оучаемыми алгоритмами. Так в работе Ravichandran и Rovy (2002) используются автоматические методы для выявления шаблонов для последующего их применения. Целью обучения является выявления и построение связей между конкретным типом ответа (например,DATE_OF_DEATH) и конкретным фокусом вопроса (для этого случая - персона). Таким образом нужно выявить шаблоны, связывающие два вида этих фраз (PERSON/DATE_OF_DEATH). Приведем примерный алгоритм обучения для выявления шаблонов:

1.для создания связи между двумя сущностями создаётся список из правильных пар;

2.производится запрос поисковой машине из частей этих пар

3.далее выбираются предложения из релевантных документов, содержащих обе части пар;

4.извлекается шаблон, содержащий слова и знаки пунктуации между этими частями пар;

5.далее оцениваются и выбираются шаблоны.

Рассмотрим идею оценки шаблонов. Например, оценка и выбор шаблонов можно произвести следующим образом: составляется запрос поисковой машине из фраз, входящий в вопрос и подходящих по паре (PERSON); далее к найденным фрагментам документов применяется шаблон, и так как правильное значение ответа уже известно, то просто выбираются такие шаблоны, которые имеют высокий процент правильно найденных ответов.

Далее представлены примеры шаблонов, найденные с помощью этого алгоритма.

<NAME> (<DATE_OF_BIRTH> – <DATE_OF_DEATH>)

<NAME> was died on <DATE_OF_DEATH>

После получения кандидатов для ответа нужно выбрать тот, который будет представлен пользователю как ответ на вопрос. Для этого каждый кандидат оценивается специальной функцией и выбирается кандидат с максимальной оценкой. Например, оценивать можно, вычисляя следующие величины:

-answer type match - булевская величина, равная true, если кандидат содержит слова с типом ожидаемого ответа

-keyword-score - количество ключевых слов, содержащихся в кандидате для ответа

-keyword-distance-score – величина, равная расстоянию между кандидатом для ответа и ключевыми словами во фрагменте

-punctuation-score – булевская величина, равная true, если после кандидата для ответа во фрагменте стоит знак пунктуации.

Также при оценке может учитываться встречаются ли слова из вопроса в кандидате для ответа в том же порядке, что и вопросе.

Другим способом извлечения ответов из фрагментов является выявление кандидатов применением n-грамм. N-грамма – это подпоследовательность из n элементов, слудующих друг за другом в данной последовательности. Данный алгоритм эффективно применять к сниппетам при поисковом запросе, полученном при перефразировании

вопросительного предложения. На первом этапе из сниппета извлекаются униграммы, биграммы и триграммы. Далее им присваются веса, равные количеству сниппетов, в которых встретилась данная n-грамма. Следующий этапе – оценивание и сбор кандидатов из n-грамм. При оценивании преследуется цель определения того, насколько данная n-грамма соответствует типу ожидаемого ответа. Далее n-граммы ранжируются, выбирается определенное их количество с высокими оценками и строится кандидат ответа, путем конкатенации n-грамм. Кандидат для ответа с высокой оценкой выбирается в качестве ответа.

4 Построение простой вопросно-ответной системы для русского языка.

При разработке вопросно-ответной системы за основу была взята типовая архитектура вопросно-ответной системы. В части информационного поиска использовалась поисковая система Google. Были реализованы методы получения ответа не некоторые виды вопросов.

4.1 Анализ вопроса.

После ввода вопроса пользователем система распознаёт тип вопроса. Распознавание происходит по вопросительным словам, всего в настоящее время распознаётся только несколько видов вопросов. Вопросительными словами могут являться только те, слова, которые стоят в начале предложения. Разработана следующая типология вопросов, используемая в данной системе:

1.вопрос типа “Person” – вопросительные слова – «кто» «кем»;

2.вопрос типа “Location” – вопросительное слово «где»;

3.вопрос типа “Date” – вопросительное слово «когда»;

4.вопрос типа “Definition” – вопросительное слово «что».

В таблице 7 приведены примеры определения типа вопроса данной системой.

Таблица 7

Вопрос	Тип
«Кто такой Ленин?»	Person
«Где был первый полет на самолете?»	Location
«Кем был Суворов?»	Person
«Что такое любовь?»	Definition
«Когда была Куликовская битва?»	Date

Также были созданы шаблоны для распознавания некоторых типов вопросов. Шаблоны разработаны для типов вопроса «Person» и «Definition». В таблице 8 приведены примеры шаблонов для данных двух типов вопросов.

Таблица 8

Person	Definition
«Кто такой () ?» «Кем был ()?» «Кем являлся ()?»	«Что такое ()?»

Вопросительное предложение также анализируется синтаксически. После данного анализа строится синтаксическое дерево разбора. Дерево разбора представляется в виде набора бинарных направленных отношений зависимости между словами в предложении. В каждом отношении задаются главное слово и зависимое слово) Тем самым получается дерево, которое можно использовать в дальнейшем для выбора и получения ответов. Также происходит и разметка частей речи в предложении.

После анализа вопроса каждому слову в предложении сопоставляется его начальная форма.

4.2 Формирование запросов к поисковой системе.

При работе программы к поисковой системе производится несколько запросов. Первым запросом к поисковой системе является полностью само вопросительное предложение. Дело в том, что современные поисковые системы настолько развиты, что позволяют получать релевантные документы таким образом. Это было протестировано, для примера смотрите рис.1

РИС.1

После этого формируются другие поисковые запросы. В предложении убираются вопросительные слова в начале предложения, таким образом, формируется еще один готовый запрос.

Многие вопросно-ответные системы для английского используют словари-тезаурусы для расширения множества своих запросов. Известной мощной системой для анализа слов является WordNet. WordNet - это мощный толковый словарь и тезаурус, выдающий результаты своей работы (справку по данному слову) в удобном для компьютерного анализа структурированном виде. Сами словарные статьи при этом, однако, являются именно словарными статьями в классическом смысле, то есть они представляют собой тексты на английском языке, вообще говоря предназначенные для чтения человеком. Безусловно, WordNet является очень полезным инструментом (независимо от его применимости в задаче построения вопросно-ответных системам). В настоящее время идет перевод данного словаря на русский язык с последующей его адаптацией. В процессе разработки были попытки использовать частично готовые элементы словаря, но словарь не полностью готов и не адаптирован для эффективной работы. Так, например, можно было расширить множество поисковых запросов применение синсетов – синонимический рядов. Далее приводится пример применения для создания запроса.

«Где будет находиться новое задние правительства?»

- *«Где будет располагаться новое здание правительства?»*

- *«Где будет размещаться новое здание правительства?»*

Также разработаны шаблоны для перефразирования некоторых типов вопросов, а именно для вопросов типа «Person», «Definition», «Location». В таблице 9 представлены данные шаблоны. Использование этих шаблонов позволяет получить при поиске фрагменты документов, с большей вероятностью содержащие ответы.

ТАБЛИЦА 9

4.3 Извлечение ответов.

После того как запросы были сформлулированы, они посылаются поисковой машине. Модуль информационного поиска целиком представляет собой элемент системы,

использующий возможности поисковой машины. При использовании поисковой машины получаются ссылки и сниппеты - фрагменты текстов, которые могут содержать ответ. Рассматриваются первые 16 результатов поиска каждого запроса.

Далее составляется список всех фрагментов и происходит поиск кандидатов для выдачи ответа. При извлечении ответа используются шаблоны, написанные вручную. Далее представлены примеры шаблонов для получения кандидатов ответа.

В качестве ответа выдается 5 разных кандидата, которые были получены при применении шаблонов. В общем случае выбор происходит без оценки – выбираются первые 5 ответов, полученных с помощью шаблонов.

В работе системы при извлечении ответа применяется следующий эвристический алгоритм, позволяющийдостаточно эффективно находить вполне точные ответы на вопросы типов «Person» и «Definition», точнее на вопросы, соответствующие следующим шаблонам:

«Кто такой <>?»,

«Кем был <>?»,

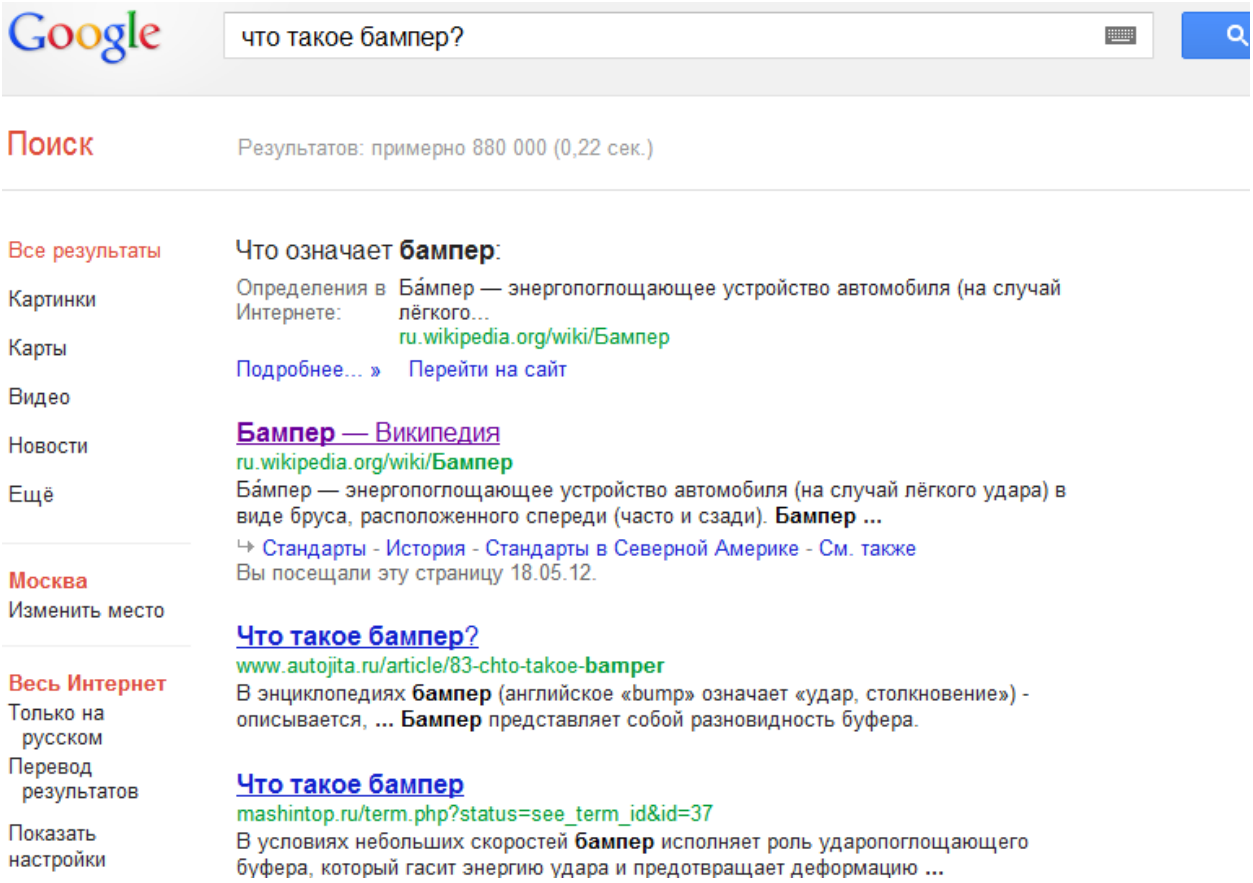
«Что такое <>?»,

то есть вопросы описания, определения и библиографического характера. На этапе получения ответа на вопросы такого рода, если в первых 10 результатах есть ссылки на статью в Википедии (англ. Wikipedia,) — свободную общедоступную универсальную энциклопедию (расположена на интернет-сайте http://www.wikipedia.org/), то ответом станет первый абзац статьи по ссылке сниппета. Первый абзац статьи представляет собой краткое и емкое представление ос статье. В вопросах о персоне, в первом абзаце кратко представлена информация о датах рождения и смерти, о роде деятельности и заслугах человека.

Для примера рассмотрим, как будет представлен ответ на вопрос «Что такое бампер?». На рис.3 показаны результаты для запроса «Что такое бампер?». Как видно, в

первых результатах есть ссылка на Википедию и ответ из Википедии, состоящий из первого абзаца текста статьи, наилучшим образом отражает смысл значения этого слова.

Рис.3



Для вопросов всех типов разрабатывался общий способ с применением дерева разбор предложения. Для начала отбираются только те сниппеты, которые содержат все ключевые слова в одном предложении. Затем рассматриваются только эти предложения для извлечения ответов. Строится дерево разбора для каждого из этих предложений. Затем специальная функция сравнивает два дерева на предмет их схожести. Далее описан способ оценки и сравнения двух деревьев.

1.Сравнение начинается с корня (глагола), и дальше методом поиска в глубину рассматривается каждый узел деревьев.

2.Если узлы слова в узлах идентичны, то к оценке начисляется 1 балл

3.Если совпадают только начальные формы слов в узлах – начисляется 0.5 балла

4.Если слово одного узла, является частью слова другого узла – начисляется 0.5 балла.

5.Если есть поддеревья в дереве для ответа, а в вопросе нет – они игнорируются как дополнительная информация.

Данный метод предложен для рассмотрения общего случая вопроса и находится в роцессе дальнейшего построения и тестирования. При его разработке выяснено, что примерно 1 вопрос из 10 может быть обработан правильно.

5 Описание практической части.

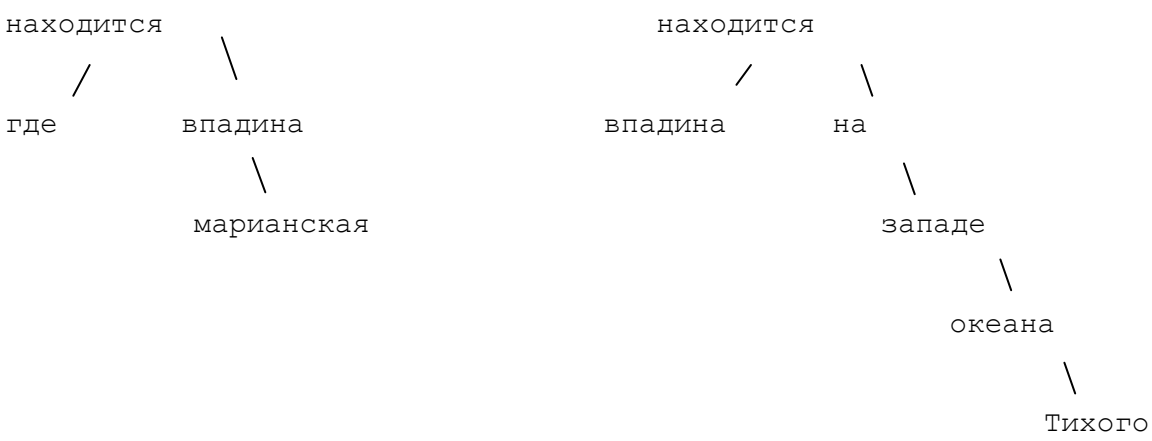
5.1 Использованный инструментарий

В качестве языка был выбран Java по причине того, что решаемая задача не предъявляет высоких требований к производительности и следовательно нет необходимости в непосредственной работе с памятью. Также выбор определен тем, что Java является кросслатформенным объектно-ориентированным языком, для которого разработано большое количество библиотек для дальнейшей разработки, что также повлияло на выбор данного языка.

В части информационного поиска использовались возможности поисковой системы Google. Для этого применялась библиотека Java Google Ajax API. При использовании данной библиотеки можно послать запрос поисковой системе и получить результаты. Результаты поиска выдаются в виде структурного массива. Элемент массива представляет собой каждый пункт результатов, имеет поля ссылки, сниппета и заголовка. То есть получаются готовые фрагменты релевантных документов и нет необходимости анализировать сам документ и выделять из него фрагмент, содержащий ответ. Сниппеты представляются в виде текста с выделением ключевых слов html тегами, что также очень удобно для поиска предложений с ключевыми словами. Необходимость поиска ключевых слов и их форм в сниппете отпадает.

Для работы с веб-страницами используется java-библиотека URLconnection. С её помощью можно получить html представление интернет-страницы. Библиотека использовалась при получении первого абзаца статей в Википедии.

При анализе вопросов и предложений, содержащих ответ использовался программный пакет синтаксического разбора для русского языка Dwarf. Пакет включает в себя синтаксический анализатор для русского и английского языков. Dwarf позволяет получать для предложения его синтаксическое дерево. Также это пакет применяется применялся для получения начальных форм слов из предложения, для последующего их использования в качестве ключевых слов для поискового запроса. Далее приведен пример разбора вопроса и предложения, взятого из сниппета.



Также данный пакет указывает дополнительную информация о каждой части речи и синтаксическую связь между словами в предложении, что может использоваться в будущих разработках.

5.2 Тестирование, оценка системы.

Так как разработанная система вопросно-ответного поиска простая, то для тестирования были необходимы несложные вопросы. Работа система оценивалась только на вопросах типа «Person» и «Definition», которые соответствовали использованию шаблонов «Кто такой ---?» и «Кем был ---?», где на месте «---» может стоять любая именованая сущность, с точки зрения формализации предложения. Таким образом, еще была возможность протестировать эвристический приём с использованием статей из Википедии.

Для частичной оценки системы был автоматически был создан небольшой корпус вопросов с последующим дополнением вручную. Для этого сначала был использован список имен известных людей, далее с применением шаблонов были созданы соответствующие вопросы. В корпусе всего 75 вопросов типа “Person” и 30 вопросов типа “Defintion”.

Система была запущена на данном корпусе вопросов и были получены результаты, представленные в таблице 9. Стоит отметить, что в случаях 28 и 10 верно отвеченных вопросов в качестве ответа было 24 и 7 фрагментов из Википедии.

Таблица 9

	Person	Definition
Верный ответ	28	10
Неверный ответ	11	8
Не найден ответ	26	22
Всего	75	40

6 Заключение.

В ходе работы была исследована предметная область вопросно-ответного поиска, рассмотрены существующие способы решения задачи поиска ответа на вопрос и разработана простая система для русского языка. В ходе частичного тестирования нескольких типов вопросов было показано, что система достаточно хорошо справляется с задачей поиска ответа на некоторые типы вопросов. Но обработка и анализ остальных типов вопросов требует дальнейшей разработки.

Таким образом была создана исследовательская база для будущих работ. При дальнейших разработках планируется увеличить количество типов вопросов, то есть расширить их типологию, также требуется более эффективный метод оценки и извлечения ответа с применением дерева синтаксического разбора. Метод использования шаблонов также может быть усовершенствован, а количество шаблонов увеличено.

Список литературы

[1] Dan Moldovan, Sanda Harabagiu. The structure and perfomance of open-domain question answering system . //Southern Methodist University, Dallas, 2005.

[2] C. Monz. From document retrieval to question answering. //Universiteit Van Amsterdam 2003.

[3] Gosse Bouma, Ismail Fahmi, Jori Mur, Gertjan van Noord. Linguistic knowledge and question answering. //University of Groningen, 2004

[4] Richard J.Cooper, Stefan M.Ruger. A Simple Question Answering System. //Imperial College of Science, Technology and Medicine, 2007.

[5] Dan Moldovan, Marius Pasca, Rada Milhalcea, Richard Goodrum, Roxana Girju and Vasile Rus. LASSO: A Tool for Surfing the Answer Net. //Southern Methodist University, 1999.

[6] Aaron Galea . Open-domain Surface-Based Question Answering System //Department of Computer Science and AI, University of Malta, 2005.

[7] Lucian Vlad Lita. Instance-Based Question Answering. // Carnegie Mellon University Pittsburgh, 2006.

[8] Sanda M. Harabagiu and Marius A. Pa_sca and Steven J. Maiorano. Experiments with Open-Domain Textual Question Answering. //Southern Methodist University, Dallas, 2006

[9] Dan Roth. Learning Components for a Question-Answering System. //University of Illinois at Urbana-Champaign, 2002

[10] Daniel Jurafsky, James H.Martin. Speech and Language Processing. //Upper Saddle River, New Jersey. Pearson Prentice Hall, 2009. 770-787 pp.

[11] Menno van Zaanen. Multi-lingual Question Answering using OpenEphyra //Tilburg University, 2003