Отчет о проверке № 1

дата выгрузки: 09.05.2017 21:50:21
пользователь: dim3aaa@mail.ru / ID: 4242136
отчет предоставлен сервисом «Антиплагиат»
на сайте http://www.antiplagiat.ru

Информация о документе

№ документа: 40
Имя исходного файла: диплом.docx
Размер текста: 1314 кБ
Тип документа: Не указано
Символов в тексте: 63182
Слов в тексте: 7802
Число предложений: 413

Информация об отчете

Дата: Отчет от 09.05.2017 21:50:21 - Последний готовый отчет (ред.)
Комментарии: не указано
Оценка оригинальности: 92.11%
Заимствования: 7.89%
Цитирование: 0%

Оригинальность: 92.11%
Заимствования: 7.89%
Цитирование: 0%

Источники

Доля в тексте	Источник	Ссылка	Дата	Найдено в
2.23%	[1] Исследование лингво-статистических методов автоматического формирования ассоциативно-иерархического портрета предметной области	http://mggu-sh.ru	22.02.2017	Модуль поиска Интернет
1.22%	[2] Лекция 8 скачать документ doc, docx	http://tfolio.ru	20.01.2017	Модуль поиска Интернет
1.07%	[3] Применение методов text mining для классификации информации, распространяемой в социальных сетях | Публикация в журнале «Молодой ученый»	https://moluch.ru	01.10.2016	Модуль поиска Интернет
1.06%	[4] МЕТОДЫ И МОДЕЛИ АВТОМАТИЧЕСКОГО ИЗВЛЕЧЕНИЯ КЛЮЧЕВЫХ СЛОВ	http://cyberleninka.ru	08.10.2015	Модуль поиска Интернет
1.03%	[5] Д.С. Новикова. Автоматическое выделение терминов из текстов предметных областей и установление связей между ними	http://masters.donntu.edu.ua	раньше 2011 года	Модуль поиска Интернет
0.86%	[6] Разрешение лексической многозначности	http://ru.wikipedia.org	раньше 2011 года	Модуль поиска Интернет
0.67%	[7] Скачать - 1,8 МБ	http://nauchkor.ru	12.04.2017	Модуль поиска Интернет
0.57%	[8] МОДЕЛИ И МЕТОДЫ АВТОМАТИЧЕСКОГО АНАЛИЗА ТЕКСТОВ НА ОСНОВЕ ЛИНГВИСТИЧЕСКИХ ОНТОЛОГИЙ В ИНФОРМАЦИОННО-АНАЛИТИЧЕСКИХ СИСТЕМАХ	http://nntu.ru	16.12.2016	Модуль поиска Интернет
0.47%	[9] Использование справочно - поисковых систем. Справочно-поисковые системы. Филиал ИПИ в Волгограде. Официальный сайт.	http://distanz.ru	10.01.2016	Модуль поиска Интернет
0.45%	[10] Adaptation of Statistical Machine Translation Model for Cross-Lingual Information Retrieval in a Service Context	http://aclweb.org	23.05.2016	Модуль поиска Интернет
0.31%	[11] Скачать - 0 байт	http://nauchkor.ru	13.04.2017	Модуль поиска Интернет
0.3%	[12] Стемминг	http://ru.wikipedia.org	30.11.2014	Модуль поиска Интернет
0.26%	[13] Поисковая оптимизация - SEO - Лекция	http://works.doklad.ru	раньше 2011 года	Модуль поиска Интернет
0.21%	[14] МЕТОДЫ ПОИСКА ДУБЛИКАТОВ СКОМПОНОВАННЫХ ТЕКСТОВ НАУЧНОЙ СТИЛИСТИКИ - тема научной статьи по автоматике и вычислительной технике, читайте бесплатно текст научно-исследовательской работы в электронной библиотеке КиберЛенинка	http://cyberleninka.ru	01.12.2014	Модуль поиска Интернет
0.19%	[15] http://iiorao.ru/iio/pages/konf_ob/archive_nauch_conferences/nauch_conf_2014/minsk_2014/?download=true&file=%F1%E1%EE%F0%ED%E8%EA_OSTIS-2014.zip	http://iiorao.ru	16.11.2016	Модуль поиска Интернет
0.19%	[16] не указано	http://hse.ru	раньше 2011 года	Модуль поиска Интернет
0.15%	[17] C.: CLEF 2009 Ad Hoc Track Overview: TEL and Persian Tasks	http://dei.unipd.it	17.04.2017	Модуль поиска

				Интернет
0.13%	[18] http://www.iiorao.ru/iio/pages/konf_ob/archive_nauch_conferences/nauch_conf_2013/minsk_2013/?download=true&file=%F1%E1%EE%F0%ED%E8%EA_OSTIS-2013.rar	http://iiorao.ru	18.11.2016	Модуль поиска Интернет
0.12%	[19] Полный текст диссертации	https://istina.msu.ru	26.11.2016	Модуль поиска Интернет

Текст отчета

Министерство образования республики беларусь

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет прикладной математики и информатики

Кафедра информационных систем управления

CROSS-LANGUAGE ФУНКЦИОНАЛЬНОСТЬ АВТОМАТИЧЕСКОГО ПОИСКА В СЕТИ INTERNET РЕЛЕВАНТНЫХ ДОКУМЕНТОВ

Отчёт по преддипломной практике

Исаченко Дмитрия Александровича

студента 5 курса,

специальность «информатика»

Научный руководитель:

доктор технических наук,

профессор И.В. Совпель

Минск 2017

РЕФЕРАТ

Отчёт по преддипломной практике, 24 стр., 8 рис., 9 источников.

Объектом исследования являются системы, позволяющие для произвольного документа обнаружить релевантные ему документы в сети интернет, в том числе на отличном от исходного языка.

Цель работы: исследовать решения поиска текстовых документов релевантных данному, разработать соответствующий алгоритм и, в соответствии с ним, реализовать приложение.

Результатом работы является приложение под платформу Android, взаимодействующее с облачным хранилищем и позволяющие для произвольного документа обнаружить на заданных пользователем языках релевантные ему документы в сети интернет.

Область применения результатов: классификация и анализ текстов, информационный поиск.

ОГЛАВЛЕНИЕ

ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ И СИМВОЛОВ

ЕЯ - естественный язык;

ПОД - поисковой образ документа;

Стоп-слова - слова, не несущие в себе смысловой и содержательной нагрузки, такие как междометия, предлоги и прочие.

Стемминг- это процесс нахожденияосновы словадля заданного исходного слова. Основа слова необязательно совпадает с морфологическимкорнем слова;

TF (term frequency) – частота термина в контексте определённого документа;

IDF (inverse document frequency) - обратная частота термина в корпусе документов;

TF-IDF - статистическая[1] мера, используемая для оценки важности слова в контексте документа, являющегося частью коллекции документов или корпуса;[1]

CLIR - разновидность информационного поиска, при которой язык извлечённой информации может отличаться от языка запроса.

ВВЕДЕНИЕ

В наше время огромное количество информации, в том числе текстовые документы, доступны в электронном виде. Информационные системы, оперирующие большими объемами текстовых документов произвольной предметной области и успешно решающие различные прикладные задачи, становятся все более востребованными как предприятиями и организациями, так и отдельными пользователями. При этом обработка информации, представленной в документах на различных языках, не является тривиальной. В связи с этим актуальна задача автоматизации поиска в сети интернет документов релевантных данному, в том числе на отлихных от иходного языках, что позволяет получить максимальное количество различной информации по теме исходного документа. Эту задачу можно свести к формированию поискового запроса, максимально описывающего тему данного документа, что в свою очередь сводится к задаче определения ключевых слов в тексте. Существуют следующие категории методов выделения ключевых слов: статистические, лингвистические и гибридные, которые являются их комбинацией. Популярные статистические методы для измерения важности слов используют статистическую меры TF-IDF, которая учитывает, как часто данное слово встречается в документе и в то же время как редко – в корпусе документов. Такой способ нахождения ключевых слов обладает более высокой точностью по сравнению с другими, в которых не задействуется предварительное обучение системы на корпусе документов.

В данной работе описывается подход по формированию поискового запроса для автоматизации поиска релевантных данному документов, в том числе на отлихных от иходного языках. В связи с тем, что большинство людей сейчас проводят больше времени в своих мобильных телефонах/планшетах, чем в десктопах, приложение будет разрабатываться под платформу Android на языке программирования Java.

ГЛАВА 1. Поиск релевантных документов в одноязычной информационной среде

Задача автоматизации поиска в сети интернет документов релевантных данному относится к классическим задачам информационного поиска и её можно решать одним из двух следующих способов:

реализовать собственную поисковую систему при разработке приложения;

воспользоваться уже существующей поисковой системой при разработке приложения.

1.1 Реализация собственной поисковой системы

Работу поисковой системы можно представить следующим образом:

рисунок 1.1 – схема работы поисковой системы

Основными её составляющими являются: поисковый робот, индексатор, поисковик.

Поисковый робот — составная часть поисковой системы, основной функцией которой является перебора страниц Интернета. Данный перебор осуществляется с целью занесения информации о[13] страницах в базу данных поисковика.[13] Поисковой робот исследует содержимое страницы и затем сохраняет поисковой образ на сервере поисковой машины, которой принадлежит, после этого исследуются следующие страницы, которые доступны по ссылкам с текущей. Большие сайты зачастую проиндексированы поисковой машиной не целиком, так как обычно для поисковой машины глубина проникновения внутрь сайта и максимальный размер сканируемого текста ограничены. Переходы между страницами реализуются с помощью ссылок, которые содержаться на исходных страницах. Порядок обхода страниц, частота визитов, защита от зацикливания, а также критерии выделения значимой информации[13] определяются. В зависимости от алгоритмов информационного поиска определяются порядок обхода страниц и частота визитов, так же предотвращается возможность зацикливания.

Современны поисковые системы дают пользователю возможность ручного добавления сайта в очередь для индексирования, с целью ускорения процесса индексирования сайта. Если на сайт невозможно попасть по внешним ссылкам, то это вообще оказывается единственной возможностью уведомить поисковую систему о существовании сайта.

Для сбора информации о сайте выполняется процесс индексирования, в ходе которого робот поисковой системы помещает в базу данных сведения о сайте (ключевые для сайта слова, ссылки, изображения, аудио…), которые затем используются при поиске. Индексирование страницы осуществляется непосредственно с помощью индексатора, который анализирует страницу, предварительно разбив её на части,[9] при этом каждый элемент веб-страницы анализируется отдельно. Полученные индексатором данные о веб-страницах хранятся в индексной базе данных для[9] возможности использования их в последующих запросах. Это существенно ускоряет поиск информации по пользователя.

Поисковый запрос — последовательность символов, которую пользователь вводит в поисковую строку, для обнаружения релевантной информации. Формат поискового запроса зависит от 2-х вещей: от типа информации для поиска и от устройства поисковой системы. Обычно поисковый запрос представляет собой набор слов или фразы.

Работу поисковой системы можно разбить на следующие шаги: сначала исходный контент принимается поисковым роботом, затем

согласно контенту, в ходе процесса индексирования определяется доступный для поиска индекс, после чего можно обнаруживать с помощью поисковой системы исходные данные. Данные шаги выполняются каждый раз при обновлении поисковой системы.

В большинстве случаев для поисковых систем основным источником для анализа и получения информации о веб-странице является HTML страница, соответствующая ей. Основное внимание при извлечении информации уделяется заголовкам и метатегам.

Поисковые гиганты, такие как Google, имеют возможность полностью сохранять контент исходной страницы целиком или только часть её(кэш). Последнее позволяется значительно увеличить скорость поиска информации на ранее посещённых страницах(кэшированные). Текст запроса пользователя обычно сохраняется вместе с кэшированной страницей, чтобы сохранить актуальность в случае обновления исходной. Пользователь формирует запросы для поисковика, который затем обрабатывает их, анализируя данные полученные в ходе процесса индексации, и затем возвращает результаты поиска. Запросы пользователя зачастую представляют собой набор ключевых слов. В тот момент, когда пользователь вводит запрос, поисковая система уже начинает анализировать имеющиеся индексы, после чего пользователь получает наиболее релевантные веб-страниц, также поисковая системы может возвращать их вместе с краткой аннотацией, которая представляет собой заголовок документа и возможно некоторый отрывок из текста. Поисковая система характеризуется следующими двумя оценками: оценка точности найденных релевантных страниц и оценка полноты найденных релевантных страниц. Для того, чтобы в начале списка результатов были наиболее актуальные для пользователя, многие поисковые систем испол

ьзуют методы ранжирования, которые в свою очередь определяют, какие страницы более релевантны, а также очередь отображения результатов. На данный момент выделяют два основных типа поисковых систем: системы[9] предопределённых и иерархически упорядоченных ключевых слов, и системы, в которых генерируется инвертированный индекс на основе анализа текста.

В[9] связи с огромной трудоёмкостью реализации собственной поисковой системы при разработке приложения будет использоваться поисковая система Google. Таким образом задача автоматизации поиска в сети интернет документов релевантных данному сведётся к формированию поискового запроса. Поисковым запросом для Google будет являться поисковой образ документа, который формируется из ключевых для исходного текста слов. Количество слов в запросе можем варьироваться в зависимости от размера документа. Согласно рекомендациям Google, поисковой запрос должен состоять из ключевых слов, оптимальное количество ключевых слов должно находиться в диапазоне 6-9.

1.2 Составление поискового образа документа

Поисковый образ документа - текст, выражающий на информационно поисковом языке основное содержание документа и в последующем используемый для информационного поиска. Для формирования ПОД необходимо выделить из документа ключевую информацию.

Любой алгоритм извлечения ключевых слов/словосочетаний реализует одну или несколько систем распознавания образов, разбивающих входное множество слов на два класса (ключевые и прочие). По наличию элементов обучения выделяют необучаемые, обучаемые и самообучаемые методы извлечения ключевых слов. Более простые необучаемые методы подразумевают контекстно-независимое выделение ключевых слов/словосочетаний из отдельного текста на основе априорно составленных моделей и правил. Они подходят для гомогенных по функциональному стилю корпусов текстов, увеличивающихся со временем в объемах, например научных работ или нормативных актов. Обучаемые методы предполагают использование разнообразных лингвистических ресурсов для настройки критериев принятия решений при распознавании ключевых слов. Здесь большое значение имеет корректное выделение ключевых слов в выборке, используемой для обучения. Среди методов с обучением можно выделить подкласс самообучаемых, если обучение ведется без учителя или с подкреплением (на основе пассивной адаптации). По второму признаку классификации, прежде всего, следует выделить статистические и структурные методы извлечения ключевых слов. Статистические методы учитывают относительные частоты встречаемости морфологических, лексических, синтаксических единиц и их комбинаций. Это делает создаваемые на их основе алгоритмы довольно простыми, но недостаточно точными, т.к. признак частотности ключевых слов не является превалирующим.

Для выделения ключевых[1] словосочетаний используется анализ коллокаций,[1] которые выявляются в ходе лексического анализа текста.

Коллокация – словосочетание, состоящее из двух или более слов, имеющее признаки синтаксически и семантически целостной единицы, в котором выбор одного из компонентов осуществляется по смыслу, а выбор второго зависит от выбора первого.[1] Для обнаружения коллокаций используются различные меры ассоциативной связи, которые оценивают, является ли взаимное появление лексических единиц случайным, или оно статически значимо.

В[1] нашем случае ПОД, будет состоять из ключевых слов исходного документа, и являться запросом для поисковой системы Google.

1.2.1 Предварительная обработка документа перед составлением ПОД

Одним из способов повышения эффективности работы систем информационного поиска является предоставление поисковым системам способа обнаружения различных форм одного и того же слова. Для реализации процесса обнаружения различных форм можно воспользоваться стеммерами. Стемминг также используется в информационном поиске для уменьшения размера индексных файлов. Таким образом сначала для исходного текста выделяются границы. Затем для обнаружения различных форм одного и того же слова необходимо выполнить процесс нахождения основы слова для всех исходных слов текста – стемминг. Стемминг выполняет морфологический разбор слова, находит общую для всех его грамматических форм основу,

отсекая суффиксы и окончания.[14]

Существует несколько критериев оценки стеммеров: корректность, эффективность поиска и производительность сжатия.

При реализации стемминга нужно найти баланс между следующими двумя проблемами: чрезмерный стемминг, что приводит к объединению несвязанных терминов и соответственно это понижает точность поиска, так как извлекаются нерелевантные документы; основа слова выделяется слишком слабо, в связи с чем будет понижаться полнота поиска.

В зависимости от необходимой точности/полноты поиска, а также скорости работы можно выбрать один из следующих стеммеров.

Для английского языка на данный момент одним из самых распространённых стеммеров является стеммер Портера в силу его быстрой скорости работы, отсутствия необходимости в предварительной обработке корпуса документов и использования каких-либо баз основ. В основе данного стеммера лежит алгоритм усечения окончаний, использующий для своей работы небольшой набор правил, например, если слово оканчивается на "ет", то удалить "ет" и так далее. Алгоритмы усечения окончаний достаточно эффективны на практике, но в то же время обладают некоторыми недостатками. Алгоритмы усечения окончаний неэффективны в случае изменения корня слова, например, изменения или выпадения гласной. Данные алгоритмы эффективны для тех частей речи, которые имеют хорошо известные окончания и суффиксы. Стеммер Портера основывается на том, что количество словообразующих суффиксов в языках ограниченно. Благодаря этому алгоритм может выполняться с помощью установленных вручную определённых правил. Алгоритм выделения основы слова стеммера Портера для английского языка состоит из пяти шагов. На каждом шаге у[12] слова отсекается словообразующий суффикс,[12] затем оставшаяся часть проверяется на соответствие[12] заданным правилам. В случае, если правила удовлетворены осуществляется переход на следующий шаг алгоритма, иначе выбирается другой суффикс для отсечения. Из описания хода работы алгоритма видно, что у стеммера Портера существует недостаток: он может обрезать слово больше необходимого, что в свою очередь затруднит получение правильной основы слова и соответственно уменьшит точность извлечение релевантной информации. Ещё одним недостатком стеммера Портера является отсутствие возможности работать при изменении корня слова, например, в случае выпадающих беглых гласных.

Стеммер, использующий таблицы поиска флективных норм. Трудностью при реализации данного стеммера является необходимость перечислять все флективные формы в таблице, если какая-то из форм будет отсутствовать, то она обрабатываться не будет. В связи с этим получается, что таблица поиска может иметь большой размер. В качестве плюсов можно выделить простоту подхода, скорость работы и простоту обработки исключений. Таблицы поиска, которые используются в стеммерах, обычно генерируются в полуавтоматическом режиме. Чтобы избежать проблемы, когда разные слова относятся к одной лемме (ошибка лемматизации), при реализации алгоритма поиска можно использовать предварительную частеречную разметку.

Можно улучшить проход к выделению основы слова посредством определения части речи слова и затем в зависимости от результата применения соответствующих для каждой части речи правил нормализации.

Основной недостаток классических стеммеров – они не различают слова, имеющие схожий синтаксис, но абсолютно разные значения, например, в английском языке "news" и "new" для данных стеммеров будут различными формами одного и того же слова. С целью разрешения данных проблем было реализованы стеммеры на основе корпусов текстов. Ключевой идеей их является создании классов эквивалентности для слов классических стеммеров, которые после разделят некоторые слова, объединенные на основе их встречаемости в корпусе.[12] Такие алгоритмы работают с базой данных основ, которые не обязательно соответствуют обычным словам и зачастую представляют собой. Для определения основы слова алгоритм сопоставляет его с основами из базы данных, используя различные ограничения, такие как длина искомой основы в слове относительно длины самого слова и т.п.

1.3 Методы извлечения ключевых слов

Существуют следующие категории методов выделения ключевых слов: статистические, лингвистические, и гибридные, которые являются их комбинацией.

1.3.1 Лингвистические методы извлечения ключевых слов.

В основе лингвистических методов лежат значения слов, семантические данные о слове, а также используются онтологии, которые формализуют знания из некоторой области с помощью концептуальной схемы. При использовании данных подходов возникает трудность, связанная с реализацией онтологий, что само по себе очень трудоёмкий процесс. Часть операций, которая при лингвистическом анализе текстов выполняется вручную, усложняют процесс анализа документов из-за дополнительной возможности возникновения ошибок и неточностей.

Наиболее популярными лингвистическими методами при обработке естественного языка являются лингвистические методы в основе которых лежат графы. Главная задача данных методов представляет собой построение семантического графа. Семантический граф является взвешенным графом. Термины исходного документа будут вершинами в графе. Между вершинами графа есть ребро в том и только в том случае, если присутствует семантическая связь между терминами. Вес в семантическом графе равен значению семантической близости связанных ребром терминов. Поиск ключевых слов осуществляется с помощью алгоритмов обработки графа. Определяющими характеристиками лингвистических методов, основанных на графах, являются способ отбора множества терминов, а также алгоритм определения весов рёбер (семантической близости терминов

1.3.2 Статистические методы извлечения ключевых слов.

Статистические методы базируются на численных данных о встречаемости слова в тексте. Их преимуществами являются универсальность алгоритмов извлечения ключевых слов, отсутствие необходимости в трудоемких процедурах построения лингвистических баз знаний, а[1] также относительная простота реализации. Максимальную точность и полноту имеют алгоритмы, в основе которых лежат статистические исследования корпусов документов. Алгоритмы, которые предварительно не обрабатывают никаких документов, кроме того, ключевые слова которого необходимо извлечь, обладают сравнительно более низкой точностью. Классическими подходами в области статистической обработки естественного языка можно считать использование метрики TF-IDF и ее модификаций (для выделения ключевых слов), а[1] также анализ коллокаций (для выделения словосочетаний).[1] Одним из элементарных статистических методов извлечения ключевых слов является построение множества кандидатов[3] путем ранжирования всех словоформ или лексем документа по частоте. Фильтрация в[4] данном случае осуществляется через отбор в качестве ключевых наиболее частотных словоформ/лексем.

Если в качестве параметра для автоматического обнаружения ключевых слов использовать только частоты слова в документе, то в данном случае вычисление частоты словоформ реализуется следующим образом: полученная в результате частота ключевых слов вычисляется посредством сравнения словоформ, приведённых к одной форме, как правило, к основе или лемме. Выделение основы у словоформы представляет собой разновидность задачи морфологического анализа, которая является достаточно трудоёмкой. При реализации статистических подходов для поиска ключевых слов задействованы различные эвристические алгоритмы, обычно приводящие словоформу к ее квази-основе, что достигается посредством выделения у словоформы некоторого количества букв. Данные алгоритмы(стемминг-алгоритмы) обсуждались выше при описании предварительной обработке документа. В ходе алгоритмов стемминга выделялись основы слов, которые затем ранжировались по частоте. Словоформы с наибольшей частотой считаются ключевыми. Статистические методы, обученные для повышения точности поиска ключевых слов на корпусе текстов, достаточно популярны. Но в тоже время необходимо наличие таких корпусов для каждой определённой предметной области, что значительно затрудняет возможность данных методов. С целью повышения точности описания контента документа разрабатываются методики, у которых мерой релевантности является вес лексемы, полученный посредством определённой комбинации значений различных параметров лексем, таких как, расположения в тексте, статистика совместной принадлежности слов одному и тому же документу и т.п.

Положительными сторонами использования статистических методов является универсальность и относительная простота реализации алгоритмов извлечения ключевых слов, которая связана с тем что не нужно выполнять трудоемкие и занимающие огромное количество времени операции построения лингвистических баз знаний.[3] Однако методы извлечения ключевых слов, а[3] основе которых лежит только статистический подход иногда не обеспечивают желаемого качества результатов, особенно невысокие результаты для языков с богатой морфологией,[4] например, для русского языка, в[4] котором лексемы характеризуются огромным количеством словоформ с невысокой частотностью в отдельно рассматриваемом тексте.

Для оценки важности слова в контексте документа будем использовать статистическую меру TF-IDF, которая является произведением двух статистик: частоты термина в данном документе и обратной частоты термина в корпусе документов. Существуют различные способы определения данных статистик.

Введём следующие обозначения:

D - корпус документов;

N - размер корпуса документов;

t - термин, важность которого хотим определить в документе d.

Тогда:

n(t) = 1 + количество документов, в которых встречается термин t;

f(t,d) = количество раз, которое термин t встречается в документе d.

Способы определения статистики TF:

по частоте встречаемости (raw frequency): ;

логический (boolean frequency): ;

логарифмически нормализованный (logarithmically scaled frequency): ;

нормализованный по максимальной частоте слова (augmented frequency): .

Способы определения статистики IDF:

;

;

;

.

Различные варианты схемы взвешивания TF-IDF часто используются поисковыми системами в качестве основного инструмента при ранжировании по релевантности документов для данного поискового запроса. Так же TF-IDF может быть успешно использован при фильтрации стоп-слов в различных предметных областях.

1.3.3 Гибридные методы извлечения ключевых слов.

Для повышения точности автоматического обнаружения ключевых слов в тексте используются гибридные методы, представляющие собой комбинацию статистических методов обработки документов, дополненных несколькими лингвистическими процедурами, такими как морфологический, синтаксическим, и семантический анализ, а также различными лингвистическими базами знаний. В основе гибридных методов поиска ключевых слов в документе, может лежать обучение на корпусе текстов. Например, метод Кена Баркера, осуществляет поиск в исходном тексте базовых именных групп посредством морфосинтаксического анализа на основе словарей и вычисление релевантности БИГ.[4] Именные группы, обладающие показателем релевантности выше заданного порога, относятся к ключевым. Одной из разновидностей гибридных методов поиска ключевых слов являются методы на основе машинного обучения, в[1] которых задача извлечения ключевых слов[1] является задачей классификации. Как известно, для построения обучающей выборки, по которой будет обучен классификатор, необходимы корпуса документов, в которых выделены ключевые слова. Выделенные ключевые слова играют роль положительного примера, остальные слова – отрицательного примера. После этого для каждого слова тренировочного текста путем сопоставления ему вектора значений различных параметров[4] вычисляется его релевантность. Запоминается разница между значениями векторов данных параметров для ключевых и не являющихся таковыми слов. Затем происходит обучение модели посредством расчёта вероятности принадлежности каждого слова к группе ключевых и задания соответствующего порога. Поиск ключевых слов во входном документе осуществляется с помощью классификатора, путем расчёта актуальности слов в соответствии с построенной моделью.

Проанализировав вышеописанные методы, было замечено, что общая схема извлечения ключевых слов из текста практически одинакова для всех используемых методов и состоит из следующих[1] этапов:

Предварительная обработка текста,[1] призванная представить текст в формате, удобном для последующего распознавания. Она включает в себя: удаление стоп-слов, не несущих смысловой нагрузки (предлоги, союзы, частицы, местоимения, междометия и т. д.),[11] выделение основы слова;

Отбор кандидатов: выделяются все возможные слова, фразы, термины или понятия (в зависимости от поставленной задачи), которые потенциально могут быть ключевыми;

Анализ свойств: для каждого кандидата нужно вычислить свойства, которые указывают, что он может быть ключевым. Например, кандидат, появляющийся в названии книги, скорей всего является ключевым;

Отбор ключевых слов из числа кандидатов, посредством вычисления весов важности ключевых слов/словосочетаний в контексте документа.

рисунок 1.2 – Типовая последовательность этапов извлечения ключевых слов

В связи с трудоёмкостью реализации собственного лингвистического процессора в данной работе для выделения ключевых слов при формировании поискового запроса будет рассмотрен статистический метод, использующий статистическую меру TF-IDF.

Глава 2. Поиск релевантных документов в многоязычной информационной среде

При разработке собственной поисковой системы, поддерживающей обнаружение информации на языке отличной от языка запроса, можно было бы перевести все имеющиеся документы на все возможные языки запросов.

рисунок 1.2 – CLIR с переводом документов

Так же можно было бы ввести промежуточный язык, на который бы переводились все документы и поисковой запрос.

рисунок 1.3 – CLIR с переводом документов и запроса на промежуточный язык

У каждого из данных подходов имеются свои плюсы и минусы, но так как мы решили не разрабатывать собственную поисковую систмему, а воспользоваться уже существующей, то рассмотрим третий подход, основывающийся на переводе запроса.

рисунок 1.4 – CLIR с переводом запроса

В данном случае запросом является исходный текст, процесс индексации – процесс построения поискового образа документа, а затем с учётом того, что в поисковой системе Google язык результатов запроса тот же, что и язык исходного запроса, необходимо выполнить машинный перевод на целевой для результатов язык.

Алгоритм составления ПОД для документов на различных языках может отличаться, в силу особенностей языков. Например, в Китайском языке нету пробельных разделителей и в связи с этим для использования статистических методов необходимо предварительно выделить границы слов в исходном тексте.

Таким образом для исходного текста сначала будем составлять ПОД, а затем выполнять его перевод. Для повышения точности перевода будем использовать тезаурус синсетов.

2.1 Электронные тезаурусы. Лексическая база данных Wordnet.

Тезаурус – словарь, охватывающие понятия, определения и термины специальной области знаний. Слова в тезаурусах упорядочены по смысловой близости, не по алфавиту.

Наиболее распространёнными типами смысловых отношений между словами в тезаурусах являются:

синонимия, базирующаяся на критерии, что два выражения являются синонимичными, если замена одного из них на другое в предложении не меняет значения истинности этого высказывания,[8] например, быстрый – шустрый, бортпроводница – стюардесса;

антонимия, основанная на смысловом противопоставлении, например, тёплый – холодный, светло – темно;

гипо-гиперонимия, представляющая собой отношение общего и частного, например, машина – самосвал;

меронимия, т.е. отношение часть-целое, например, компьютер – процессор, тетрадь – страница.

Синсетом называется множество слов, связанных отношением синонимии. Синсеты разбивают множество всех лексических единиц на классы эквивалентности. Если для некоторого слова не существует синонимов, то соответствующий ему синсет будет состоять только из одного слова.При работе со словом учитываются все его значения, особенно те, в которых это слово является синонимом к другим словам. Многозначные слова, рассматриваемые в разных значениях, входят и в разные синсеты:золотая(монета) – сделанная из золотаизолотой(работник) –хороший.

WordNet - это огромная лексическая база знаний для английского языка. WordNet является семантической сетью, узлы которойпредставляют собой синсеты, связанные различными отношениями, такими как гипонимия, гиперонимия, голонимия, меронимия и т.п. WordNet приобрёл популярность благодаря его содержательным и структурными характеристиками. Принстонский WordNet и все последующие варианты для других языков направлены на отображение состава и структуры лексической системы языка в целом, а не отдельных тематических областей. Для каждого синсета имеется описание на естественном языке, а так же примеры использования входящих в него слов. В состав тезауруса входят лексемы, относящиеся к четырем частям речи: прилагательное, существительное,[8] наречие и глагол. Лексемы различных частей речи хранятся отдельно, и описания, соответствующие каждой части речи, имеют различную структуру.[8]

Существительные, прилагательные, глаголы, наречия сгруппированы в наборы когнитивных синонимов (синсеты), каждый из которых выражает отдельное значение. Синсеты взаимосвязаны между собой посредством концептуально-семантических и лексических отношений. WordNet является свободно распространяющейся и соответственно общедоступной для загрузки базой знаний. Тем самым структура WordNet делает его полезным инструментом для вычислительной лингвистики и обработки естественного языка.

Существительные, прилагательные, глаголы, наречия сгруппированы в наборы когнитивных синонимов (синсеты), каждый из которых выражает отдельное значение. Синсеты взаимосвязаны между собой посредством концептуально-семантических и лексических отношений. WordNet является свободно распространяющейся и соответственно общедоступной для загрузки базой знаний. Тем самым структура WordNet делает его полезным инструментом для вычислительной лингвистики и обработки естественного языка.

Слова в WordNet группируются вместе на основе их значений. WordNet внешне напоминает тезаурус, однако есть некоторые важные различия. Во-первых, WordNet связывает не только словоформы, но и слова со схожим смыслом. Во-вторых, WordNet отмечает семантические отношения между словами, тогда как группировки слов в тезаурусе не следуют какой-либо явной схеме, кроме сходства.

Основным отношением между словами в WordNet является синонимия. Синонимы - слова, которые обозначают одну и ту же концепцию и являются взаимозаменяемыми во многих контекстах. Они группируются в неупорядоченные наборы (синсеты). Каждый из 117 000 синсетатов WordNet связан с другими синсетами с помощью небольшого числа смысловых отношений. Кроме того, синсет содержит краткое определение и, в большинстве случаев,

одно или несколько коротких предложений, иллюстрирующих использование слов из данного синсета. Формы слов с несколькими различными значениями представлены в виде множества различных синсетов.

Синонимы обязаны быть взаимозаменяемы хотя бы в некотором непустом множестве контекстов. Для отношения синонимии не требуется заменимость всех синонимов во всех контекстах, иначе в естественном языке было бы слишком мало синонимов.[2] Для существительных в WordNet установлены следующие семантические отношения: синонимия, антонимия, гипонимия/гиперонимия,[2] меронимия.

Наиболее часто встречающимся отношением между синсетами является гиперонимия и гипонимия. Гиперонимия связывает более общие синсеты, такие как мебель, с более специфическими, такими как кровать. Таким образом, согласно WordNet в категорию мебели входит кровать, которая, в свою очередь, включает в себя двухъярусную кровать. Наоборот, понятия типа кровати и двухъярусной кровати составляют категорию мебели. Все иерархии существительных в конечном счете поднимаются на корневой узел. Отношение гипонимии является переходным: если кресло является своего рода стулом, а стул есть мебель, то кресло является своего рода мебелью. СинсетА – гипонимсинсетаВ, в том случае, когда существуют предложения типа А есть (является разновидностью) В. И соответственно гаоборот СинсетА – гиперонимсинсетаВ, в том случае, когда существуют предложения типа А имеет разновитность В.

Меронимия или другими словами отношение «часть-целое» имеет место между синсетами, такими как, например, стул и спинка, стул и ножки. В WordNet выделяются три подвида отношения часть-целое:[2] быть частью, быть элементом, быть сделанным из. Части у различных сущностей могут иметь одинаковое название, например, острие может быть у иголки, карандаша, стрелы, ножа, булавки и т.д. Таким образом А является меронимомВв том случае, если предложения вида А содержит В и А является частью В являются естественными для А и В, интерпретируемых как родовые понятия.

Так же в WordNet выделяют 2 категории глаголов согласно их смысловому значению: глаголы, обозначающие действия (действия и события), и глаголы состояния. Среди глаголов действий и событий выделяют следующие 14 групп: контакта, движения, коммуникации, восприятия, изменения, соревнования, познания, создания, эмоций, потребления, обладания, ухода за телом и глаголы, относящиеся к социальному поведению.[2] Однако, в связи с тем, что нельзя однозначно отнести многие глаголы к той или иной группе, границы между группами точно не установлены. Отношение логического следования устанавливается междусинсетамиглаголовАиВ, если из того что выполняется А, следует, что выполняется В. Например, из того, что девушка говорит, следует, что девушка издаёт звуки.

Для установления иерархических отношений между глаголами было введено отношение тропонимии. То есть делатьАозначает делатьВв определённой форме. Например, "Шептать – это тихо разговаривать". Отношениетропонимии– особый вид отношения следования. Отношение причины связывает два[2] глагольныхсинсета, один из которых[2] называетсярезультатив, а второй каузатив. Отношение причины также может быть рассмотрено как специальный случай следования. Если[2] Авлечёт за собойВ, то изВтакже логически следуетА.

Большинство отношений WordNet связывают слова, являющиеся одной частью речи. Таким образом можно сказать, что WordNet действительно состоит из четырех подсетей, по одному для существительных, глаголов, прилагательных и наречий, с несколькими перекрестными POS-указателями.

2.2 Разрешение лексической многозначности слов.

Под неоднозначностью/многозначностью языкового выражения понимают наличие у него одновременно нескольких различных смыслов.[6] Многозначность подразделяется на следующие типы: лексическую, синтаксическую и речевую, однако термин «WSD» включает в себя разрешение именно лексической (смысловой).[6] Например, слово "ключ" может употребляться в одном из следующих значений: ключ как инструмент для открывания и ключ как источник воды.[6]

Процесс разрешения требует нескольких вещей:системы словарных[6] знанийдля определения множества значений слов[6] икорпустекстов для разрешения. Знания являются одними из ключевых моментов разрешения многозначности: они предоставляют данные, на которые опирается сам процесс разрешения. Эти данные могут быть как корпусы текстов, так и словари,[6] тезаурусы, глоссарии, онтологии и т. д.

Среди основных методов разрешения лексической многозначности выделяют: методы, использующие внешние источники информации, и методы, базирующиеся на машинном обучении, работающие на размеченных корпусах текстов. Также применяются комбинации этих методов По другой классификации, методы разрешения лексической многозначности различают по типу используемых внешних источников информации: структурированные источники данных (машиночитаемые словари, тезаурусы, онтологии), неструктурированные источники данных в виде корпусов.

Далее будут представлены примеры методов и алгоритмов разрешения лексической многозначности, разбитые на группы:

методы, основанные на использовании тезаурусов, словарей;

методы, использующие нейронные сети;

бустинг;

лексические цепочки – построение последовательности семантически связанных слов;

метод ансамбля байесовских классификаторов и сочетаемостные ограничения на основе байесовских сетей;

контекстная кластеризация – кластеризация контекстных векторов, где разные кластеры соответствуют разным значениям слова.

2.2.1 Методы, основанные на использовании тезаурусных знаний.

В качестве примера одного из данных методов рассмотрим метод Леска, который основан на поиске значения словав списке словарных определений с учетомконтекста, в котором используется данное слово. Основным критерием при выборе значения является следующее правило: заложенный в этом определениисмыслдолжен был частично совпадать со смысломзначений соседних слов в контексте.

Метод леска можно разбить на следующие шаги:

Для исходного слова выделяется контекста, размер которого не более 10 ближайших по расположениюслов;

Для исходного слова осуществляется поиск всех определений всловаре;

Сопоставление слов из контекста с каждым найденным определением. В случае если какое-либоиз контекста слово присутствует в определении, то этому определению дается балл;

Наиболее вероятным значением является то, определение которого набрало наибольшее количество баллов.

2.2.2 Методы на основе нейронных сетей, построенных по данным машиночитаемых словарей

В типичной нейронной сети на вход подается слово, значение которого требуется установить, т. е. целевое слово, а также контекст, его содержащий. Узлы выхода соответствуют различным значениям слова. В процессе обучения, когда значение тренировочного целевого слова известно, веса связующих узлы соединений настраиваются таким образом, чтобы по окончании обучения выходной узел, соответствующий истинному значению целевого слова, имел наибольшую активность. Веса соединений могут быть положительными или отрицательными и настраиваются посредством рекуррентных алгоритмов (алгоритм обратного распространения ошибки, рекуррентный метод наименьших квадратов и т. д.). Сеть может содержать скрытые слои, состоящие из узлов

, соединенных как прямыми, так и обратными связями.

Целевое слово представлено узлом, соединенным активирующими связями со смысловыми узлами, представляющими все возможные значения слова, имеющиеся в словарных статьях. Каждый смысловой узел, в свою очередь, соединен активирующими связями с узлами, представляющими слова в словарной статье, соответствующей толкованию данного значения. Процесс соединения повторяется многократно, создавая сверхбольшую сеть взаимосвязанных узлов. В идеале сеть может содержать весь словарь.

При запуске сети первыми активируются узлы входного слова (согласно принятой кодировке). Затем каждый входной узел посылает активирующий сигнал своим смысловым узлам, с которыми он соединен. В результате сигналы распространяются по всей сети в течение определенного числа циклов. В каждом цикле узлы слова и его значений получают обратные сигналы от узлов, соединённых с ними. Узлы конкурирующих значений посылают взаимно подавляющие сигналы. Взаимодействие сигналов обратной связи и подавления, в соответствии со стратегией "победитель получает все", позволяет увеличить активацию узлов-слов и соответствующих им правильных узлов-значений, одновременно уменьшая активацию узлов, соответствующих неправильным значениям. После нескольких десятков циклов сеть стабилизируется в состоянии, в котором активированы только узлы-значения с наиболее активированными связями с узлами-словами. При обучении сети используется метод обратного распространения (back propagation).

2.2.3 Бустинг

Бустинг – это общий и доказуемо эффективный метод получения очень точного правила предсказания путем комбинирования грубых и умеренно неточных эмпирических правил.

Рассмотрим бустинг на примере алгоритма AdaBoost. AdaBoost является адаптивным алгоритмом, поскольку он может адаптироваться к уровням ошибок отдельных слабых гипотез. На вход алгоритма поступает обучающая выборка, где каждый элементпринадлежит некоторому домену или признаковому пространству и каждая метка принадлежит некоторому набору меток . Для каждого обучающего примера вес распределения для целых обозначается, где – это шаг алгоритма. За начальное распределение весов принимается. Пусть метки принимают значения из множества = {−1, 1}. Далее на каждом шаге , где = 1…, выполняется обучение с использованием текущего распределения, после чего строится слабая гипотеза : {−1; 1} с ошибкой первого рода , по которой выбирается уровень значимости и строится новое распределение для следующего шага: .

Конечная гипотеза () – это среднее из большинства решений слабых гипотез, где – вес, присвоенный гипотезе :

Идея алгоритма заключается в определении набора весов для обучающей выборки. Первоначально все веса примеров устанавливаются равными, но в каждом цикле веса неправильно классифицированных по гипотезе примеров увеличиваются. Таким образом получаются веса, которые относятся к

сложным примерам. Основное теоретическое свойство

AdaBoost – это способность алгоритма уменьшать ошибку обучения. AdaBoost обладает определенными преимуществами. Его быстро и просто запрограммировать. Он не имеет никаких параметров для настройки, за исключением количества циклов. Он не требует никаких предварительных знаний о слабом обучаемом и поэтому может быть скомбинирован с любым методом для нахождения слабых гипотез. Недостатки метода заключаются в следующем: фактическая производительность бустинга на конкретной задаче явно зависит от данных и слабо обучаемого алгоритма. Теоретически бустинг может выполниться плохо, если данных недостаточно, слабые гипотезы слишком сложные или, наоборот, слишком слабые. Также бустинг особенно восприимчив к шуму.

2.2.4 Использование лексических цепочек для разрешения многозначности.

Метод построения лексических цепочек включает шаги:

Выбирается набор[7] слов-кандидатов на включение в цепочки ([7] существительные и составные существительные).[7]

По словарю строится список всех значений[7] для каждого слова-кандидата.

Для каждого значения[19] каждого слова-кандидата[19] находится расстояние до каждого слова во всех уже построенных цепочках (слово в цепочке имеет строго[7] определенное значение, задаваемое другими словами в той же цепочке). Между двумя словами есть отношение, если мало расстояние между этими словами в тексте или между значениями этих слов существует путь в тезаурусе WordNet. Выделяют три вида отношений:

Extra-strong отношение существует для слов, повторяющихся в тексте. Повтор может быть на любом расстоянии от первого употребления слова.

Strong отношение определено между словами, связанными отношением в WordNet. Два таких слова должны находиться в окне не более семи предложений.

Medium-strong отношение указывается для слов, синсеты которых находятся на расстоянии больше одного вWordNet (но есть еще и дополнительные ограничения на путь между синсетами). Слова в тексте должны находиться в пределах трех предложений.

Слово-кандидат добавляется в цепочки, со словами которых найдена связь. Смысловая неоднозначность устраняется, в[7] цепочку добавляется не просто слово, а его конкретное значение ([7] благодаря выбору значения в словаре на шаге 2).

Для выбора приоритетной цепочки (для вставки слова-кандидата) отношения упорядочены так: extra-strong, strong, medium-strong. Цепочки можно выбирать жадным алгоритмом, при этом слово-кандидат попадает ровно в одну цепочку и после этого выбор уже не может быть изменен, даже если последующий текст покажет ошибочность первоначального решения. Так же приоритетную цепочку можно выбирать по следующей схеме, требующей рассмотрения всех возможных цепочек. Таким образом, будут сформированы цепочки с учетом всех возможных значений слов с последующим выбором наилучшей цепочки.

2.2.5 Разрешение лексической многозначности методом ансамбля байесовских классификаторов.

Наивный байесовский классификатор – это простой вероятностный классификатор на основе применения теоремы Байеса. Для различения значений учитывается совместная встречаемость слов в окне заданного размера в текстах корпуса. При разрешении лексической многозначности, представленном в виде задачи обучения с учителем, применяют статистические методы и методы машинного обучения к размеченному корпусу. В таких методах словам корпуса, для которых указано значение, соответствует набор языковых свойств.

Подход основан на объединении ряда простых классификаторов в ансамбль, который разрешает многозначность с помощью голосования простым большинством голосов. В проблеме разрешения лексической многозначности существует понятие контекста, в котором встречается многозначное слово. Этот контекст представляется в виде функции переменных, а значение многозначного слова представлено в виде классификационной переменной S. Все переменные бинарные. Переменная, соответствующая слову из контекста, принимает значение “ИСТИНА”, если это слово находится на расстоянии определенного количества слов слева или справа от целевого слова. Совместная вероятность наблюдения определенной комбинации переменных контекста с конкретным значением слова выражается следующим образом:, где и – параметры данной модели. Для оценки параметров достаточно знать частоты событий, описываемых взаимозависимыми переменными. Эти значения соответствуют числу предложений, где слово, представляемое, встречается в некотором контексте многозначного слова, упомянутого в значении . Если возникают нулевые значения параметров, то они сглаживаются путем присвоения им по умолчанию очень маленького значения. После оценки всех параметров модель считается обученной и может быть использована в качестве классификатора.

Контекст представлен в виде bag-of-words (модель “мешка слов”). В этой модели выполняется следующая предобработка текста: удаляются знаки препинания, все слова переводятся в нижний регистр, все слова приводятся к их начальной форме (лемматизация). Контексты делятся на два окна: левое и правое. В первое попадают слова, встречающиеся слева от неоднозначного слова, и, соответственно, во второе – встречающиеся справа. Окна контекстов могут принимать 9 различных размеров: 0, 1, 2, 3, 4, 5, 10, 25 и 50 слов. Первым шагом в ансамблевом подходе является обучение отдельных наивных байесовских классификаторов для каждого из 81 возможных сочетаний левого и правого размеров окон. Наивный байесовский классификатор (,) включает в себя слов слева от неоднозначного слова и слов справа. Исключением является классификатор (0, 0), который не включает в себя слов ни слева, ни справа. В случае нулевого контекста классификатору присваивается априорная вероятность многозначного слова (равная вероятности встретить наиболее употребляемое значение). Следующий шаг при построении ансамбля – это выбор классификаторов, которые станут членами ансамбля. 81 классификатор группируется в три общие категории, по размеру окна контекста. Используются три таких диапазона: узкий (окна шириной в 0, 1 и 2 слова), средний (3, 4, 5 слов), широкий (10, 25, 50 слов). Всего есть 9 возможных комбинаций, поскольку левое и правое окна отделены друг от друга. Например, наивный байесовский классификатор (1, 3) относится к диапазону категории (узкий, средний), поскольку он основан на окне из одного слова слева и окне из трех слов справа. Наиболее точный классификатор в каждой из 9 категорий диапазонов выбирается для включения в ансамбль. Затем каждый из 9 членов классификаторов голосует за наиболее вероятное значение слова с учетом контекста. После этого ансамбль разрешает многозначность путем присвоения целевому слову значения, получившего наибольшее число голосов.

Для разрешения многозначности можно так же воспользоваться построением сочетаемостных ограничений на основе байесовских сетей. Сочетаемостные ограничения – это закономерности использования глагола относительно семантического класса его параметров (субъект, объект (прямое дополнение) и косвенное дополнение. Модели автоматического построения сочетаемостных ограничений важны сами по себе и имеют приложения в обработке естественного языка. Сочетаемостные ограничения глагола могут применяться для получения возможных значений неизвестного параметра пр

и известных глаголах. При построении предложения сочетаемостные ограничения позволяют отранжировать варианты и выбрать лучший среди них. Исследование сочетаемостных ограничений могло бы помочь в понимании структуры ментального лексикона. Системы обучения сочетаемостных ограничений без учителя обычно комбинируют статистические подходы и подходы, основанные на знаниях. Компонент базы знаний – это обычно база данных, в которой слова сгруппированы в классы.

Статистический компонент состоит из пар предикат-аргумент, извлеченных из неразмеченного корпуса. В тривиальном алгоритме можно было бы получить список слов (прямых дополнений глагола), и для тех слов, которые есть в WordNet, вывести их семантические классы. Семантическим классом называется синсет тезауруса WordNet, т.е. класс соответствует одному из значений слова. Таким образом, в тривиальном алгоритме на основе данных WordNet можно выбрать классы (значения слов), с которыми употребляются (встречаются в корпусе) глаголы.

Байесовские сети, или байесовские сети доверия (БСД), состоят из множества переменных (вершин) и множества ориентированных ребер, соединяющих эти переменные.

Такой сети соответствует ориентированный ациклический граф. Каждая переменная может принимать одно из конечного числа взаимоисключающих состояний. Пусть все переменные будут бинарного типа, т. е. принимают одно из двух значений: истина или ложь. Любой переменной A с родителями 1, ..., соответствует таблица условных вероятностей. Иерархия существительных в WordNet представлена в виде ориентированного ациклического графа. Синсет узла принимает значение “истина”, если глагол “выбирает” существительное из набора синонимов. Априорные вероятности задаются на основе двух предположений: во-первых, маловероятно, что глагол будет употребляться только со словами какого-то конкретного синсета, и вовторых, если глагол действительно употребляется только со словами из данного синсета(например, синсет ЕДА), тогда должно быть правомерным употребление этого глагола с гипонимами этого синсета (например, ФРУКТ).

Те же предположения, что для синсетов, верны и для употреблений слов с глаголами:

слово, вероятно, является аргументом глагола в том случае, если глагол употребляется с каким-либо из значений этого слова;

отсутствие связки глагол-синсет говорит о малой вероятности того, что слова этого синсета употребляются с глаголом.

Словам “вероятно” и “маловероятно” должны быть приписаны такие числа, сумма которых равна единице.

2.2.6 Контекстная кластеризация

Каждому вхождению анализируемого слова в корпус соответствует контекстный вектор. Выполняется кластеризация векторов, где разные кластеры соответствуют разным значениям слова. Алгоритмы кластеризации полагаются на дистрибутивную гипотезу, в соответствии с которой слова, употребляемые в схожих контекстах, считаются близкими по смыслу.

При решении задачи различения значений используются контекстные вектора: если целевое слово встречается в тестовых данных, то контекст этого слова представляется в виде вектора контекста. Вектор контекста - это средний вектор по векторам свойств каждого из слов контекста. Вектор свойств содержит информацию о совместной встречаемости данного слова с другими словами, этот вектор строится по данным корпуса текстов на этапе обучения.

Первоначально строится матрица совместной встречаемости слов по данным обучающего корпуса. Вектор свойств (строка матрицы) содержит информацию о совместной встречаемости данного слова с другими. После создания матрицы выполняется разделение тестовых данных, т. е.

группировка примеров употреблений (фраз) с целевым словом. Каждому слову в примере употребления в тестовых данных соответствует вектор свойств из матрицы встречаемости. Средний вектор свойств по всем словам соответствует вектору контекста. Таким образом, набор тестовых данных, включающих употребление исследуемого слова, преобразуется в набор контекстных векторов, каждый из которых соответствует одному из употреблений целевого слова.

Различение значений происходит путем кластеризации контекстных векторов с помощью разделяющего или иерархического “сверху вниз” алгоритма кластеризации. Получающиеся кластеры составлены из употреблений близких по значению фраз, и каждый кластер соответствует отдельному значению целевого слова. Векторы свойств, полученные по небольшому корпусу текстов, имеют очень малую размерность (несколько сотен), что не позволяет полностью описать закономерности совместной встречаемости слов. Для решения этой проблемы векторы свойств слов расширяются содержательными словами, извлеченными из словарных толкований разных значений данного слова.

Данный метод может быть полезен при различении значений слов без учителя при небольшом количестве обучающих данных.

1.3 Постановка задачи

Требуется разработать мобильное приложение под платформу Android, взаимодействующее с облачным хранилищем и позволяющие для произвольного документа обнаружить на заданных пользователем языках релевантные ему документы в сети интернет.

1.4 Выводы

В первой главе получены следующие результаты:

выполнен исследование предметной области;

поставлена задача разработки системы.

ГЛАВА 2. АЛГОРИТМИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ЗАДАЧИ

//FIXME

2.4 Выводы

Во второй главе получены следующие результаты:

выполнен анализ существующих алгоритмов и их улучшений для извлечения ключевых слов;

разработан алгоритм для автоматического поиска в сети интернет документов релевантных данному в том числе на отличных от исходных языков.

ГЛАВА 3. РЕАЛИЗАЦИЯ СИСТЕМЫ

Разработанная система будет поддерживать следующие языки для исходного текста: English, French, German, Italian, Portuguese, Russian, Spanish, Swedish, язык же обнаруженных релевантных документов может быть один из 271 предоставленных здесь http://babelnet.org/stats#LanguagesandCoverage.

//FIXME

3.0 Описание алгоритма

Алгоритм:

Определить язык исходного текста;

Перевести исходный текст на заданные пользователем языки. Если система не обучалась для какого-то из заданных языков, то вместо данного языка перевести текст на английский.

Провести токенезацию переведённых текстов(возможно стемминг, в зависимости от языка);

Определить для токенов важность их в контексте соответствующего документа;

Исходя из полученных весов важности определить наборы ключевых слов;

Из составленных наборов ключевых слов сгенерировать поисковые запросы для Google;

Используя поисковые запросы, найти релевантные данному документы в сети интернет.

2.1 Описание алгоритма

Алгоритм:

Определить язык исходного текста;

Перевести исходный текст на заданные пользователем языки. Если система не обучалась для какого-то из заданных языков, то вместо данного языка перевести текст на английский.

Провести токенезацию переведённых текстов(возможно стемминг, в зависимости от языка);

Определить для токенов важность их в контексте соответствующего документа;

Исходя из полученных весов важности определить наборы ключевых слов;

Из составленных наборов ключевых слов сгенерировать поисковые запросы для Google;

Используя поисковые запросы, найти релевантные данному документы в сети интернет.

3.1 Разработка архитектуры системы

В качестве облачного хранилища была выбрана Firebase Realtime Database в связи с возможностью её бесплатного использования.

При разработке приложения был использован объектно-ориентированный подход. Приложение разбито на модули согласно функциональности. Имеются следующие агенты:

Агент взаимодействия с облачным хранилищем;

Агент пользовательского интерфейса, который преобразует команды пользователя и передаёт их соответствующим агентам;

Агент, инкапсулирующий логику формирования поискового запроса для исходного документа;

Агент, отвечающий за особенности работы с ОС Android.

3.2 Особенности реализации мобильного приложения

Так как реализованное мобильное приложение на текущий момент не взаимодействует с сервером и все полученные в результате вычислений данные хранятся на самом устройстве, то при обучении брался корпус, состоящий из 20 документов. Чтобы повысить точность и полноту результатов поисковых запросов перед проведением стемминга был добавлен этап фильтрации стоп-слов.

Список стоп слов для английского языка:

a, about, above, after, again, against, all, am, an, and, any, are, aren't, as, at, be, because, been, before, being, below ,between, both, but, by, can't, cannot, could, couldn't, did, didn't, do, does, doesn't, doing, don't, down, during, each, few, for, from, further, had, hadn't, has, hasn't, have, haven't, having, he, he'd, he'll, he's, her, here, here's, hers, herself, him, himself, his, how, how's, i, i'd, i'll, i'м, i've, if, in, into, is, isn't, it, it's, its, itself, let's, me, more, most, mustn't, my, myself, no, nor, not, of, off, on, once, only, or, other, ought, our, ours, ourselves, out, over, own, same, shan't, she, she'd, she'll, she's, should, shouldn't, so, some, such, than, that, that's, the, their, theirs, them, themselves, then, there, there's, , these, they, hey'd, they'll, they're, they've, this, those, through, to, too, under, until, up, very, was, wasn't, we, we'd, we'll, we're, we've, were, weren't, what, what's, when, when's, where, where's, which, while, who, who's, whom, why, why's, with, won't, would, wouldn't, you, you'd, you'll, you're, you've, your, yours, yourself, yourselves.

3.3 Методика применения разработанного приложения

Разработанная программа обладает интуитивно понятным интерфейсом.

Алгоритм использования приложения для поиска документов в сети интернет релевантных данному:

Запустить приложение;

Ввести текст документа/выбрать документ из локального хранилища и нажать “Продолжить”;

Выбрать актуальные для поиска информации языки и нажать “Продолжить”.

Так же можно просмотреть корпус документов, который использовался при обучении системы, и полученную в результате обучения системы статистику важности различных слов в различных языках.

3.4 Пример использования разработанной системы

Запуск программы и ввод текста для которого необходимо найти в сети интернет релевантные документы.

рисунок 3.1 – Главный экран приложения

По нажатию кнопки “Продолжить” появится экран выбора актуальных для пользователя языков. На данном экране отображены все поддерживаемые для поиска информации приложением языки.

рисунок 3.2 – Экран выбор актуальных пользователю языков

По нажатию кнопки “Продолжить” будет выдан список сформированных приложением запросов для поиска релевантных данному документов на

выбранных языках. В дальнейшем список языков можно будет поменять не вводя текст заново.

Экран результата запросов для поиска релевантных данному документов на выбранных языках. А так же экран результатов поиска для определённого запроса в поисковой системе Google приведены ниже.

рисунок 3.3 – Результаты поиска релевантных документов

Так же через навигационное меню можно просмотреть корпус документов, который использовался при обучении системы, и полученную в результате обучения системы статистику. Для этого в навигационном меню нужно выбрать пункт “Корпус” / “Словарь” соответственно.

рисунок 3.4 – Навигационное меню приложения

По клику на определённый документ на экране “Корпус” загрузится отдельный экран, позволяющий его прочесть и изучить детальную статистику по данному документу.

3.5 Выводы

В третьей главе получены следующие результаты:

разработана архитектура и описаны особенности реализации мобильного приложения;

описана методика применения разработанного приложения.

ЗАКЛЮЧЕНИЕ

В ходе проделанной работы:

Выполнено исследование предметной области;

Поставлена задача разработки системы;

Разработан алгоритм для автоматического поиска в сети интернет документов релевантных данному в том числе на отличных от исходного языках;

Разработана архитектура и описаны особенности реализации мобильного приложения;

Описана методика применения разработанного приложения.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Manning, C. D. Introduction to Information Retrieval. / C. D. Manning, P. Raghavan, H. Schütze. - Cambridge University Press, 2008. – 581 с.

Автоматическая обработка текстов на естественном языке и компьютерная лингвистика: учеб. пособие /Е.И. Большакова, Э.С.[15] Клышински, Д.В. Ландэ [и др.]. - М.: МИЭМ, 2011. - 272 с.

Matsuo, Y. Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information. / Y.[4] Matsuo – Tokyo, 2003. – 13 с.

Turney, P.D. Learning algorithms for keyphrase extraction. Information Retrieval / P.D.[4] Turney. - Ottawa, Ontario, Canada, 2000. – 477 с.

Porter, M.F. An algorithm for suffix stripping. / M.F. Porter – Cambridge, 1997. – 6 с.

ANDERKA, M., LIPKA, N., AND STEIN, B. 2009. Evaluating cross-language explicit semantic analysis and cross querying.[17] In Proceedings of the 10th Cross-language Evaluation Forum Conference on Multilingual Information Access Evaluation: Text Retrieval Experiments (CLEF’09).[10] Springer, 50–57 с.

Kraiij, W., Nie, J-Y., Simard, M.: Emebdding Web-based Statistical Translation Models in Cross-Language Information Retrieval. Computational[10] Linguistics (2003) – 39 с.

The Cross-Language Evaluation Forum ([10] CLEF). http://clef-campaign.org

Virqa, P., Khudanpur, S.: Transliteration of proper names in cross-lingual information retrieval. In: ACL Workshop on Multilingual and Mixed Language Named Entity Recogni-tion (2003) – 8 с.