МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра Прикладная математика и информатика
(наименование)

09.04.03 Прикладная информатика
(код и наименование направления подготовки)

Управление корпоративными информационными процессами
(направленность (профиль))

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ)

на тему «Модели и алгоритмы системы управления поиском информации в торговой организации»

Обучающийся В.С. Котлов
(Инициалы Фамилия) (личная подпись)

Научный руководитель к.п.н., доцент, О.В. Оськина
(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2023

Оглавление

Введение

На современном ИТ-рынке широко представлены промышленные системы управления поиском информации на предприятии.

По своим функциональным и архитектурным особенностям эти системы относятся к корпоративным поисковым системам.

В торговой организации система управления поиском информации (СУПИ) позволит значительно повысить релевантность результатов поиска для клиентов организации. Кроме того, СУПИ повышает эффективность внутренних бизнес-процессов и высвобождает время на реализацию более приоритетных задач сотрудниками организации. Совершенно очевидно, что в основу СУПИ торговой организации должны быть положены модели и алгоритмы, позволяющие объединять возможности технологий построения поисковых систем и интеллектуального анализа текста.

Исследование и разработка таких моделей и алгоритмов актуальны и представляют научно-практический интерес.

Объектом исследования магистерской диссертации является поиск информации в торговой организации.

Предметом исследования система управления поиском информации в торговой организации.

Цель работы – исследование и разработка моделей и алгоритмов эффективной системы управления поиском информации в торговой организации.

Для достижения поставленной цели необходимо решать следующие задачи:

- проанализировать современное состояние исследований в области построения систем управления поиском информации в торговой организации;
- проанализировать методы и технологии управления поиском

информации в торговой организации;

- разработать модели и алгоритмы эффективной системы управления поиском информации в торговой организации;
- выполнить апробацию предлагаемых проектных решений и оценить их эффективность.

Гипотеза исследования: применение разработанных в рамках диссертационного исследования моделей и алгоритмов обеспечит повышение эффективности поиска информации в торговой организации.

Методы исследования. В процессе исследования будут использованы следующие положения и методы: системный анализ, методы и технологии управления поиском информации, методы интеллектуального анализа текста, объектно-ориентированный подход.

Новизна исследования заключается в разработке моделей и алгоритмов, которые обеспечат повышение эффективности управления поиском информации в торговой организации.

Практическая значимость исследования заключается в возможности применения предлагаемых моделей и алгоритмов при проектировании эффективной системы управления поиском информации в торговой организации.

Теоретической основой диссертационного исследования являются научные труды российских и зарубежных ученых, занимающихся проблемами поиска и анализа корпоративной информации.

Основные этапы исследования: исследование проводилось с 2020 по 2023 год в несколько этапов.

На первом (констатирующем) этапе формулировалась тема исследования, выполнялся сбор информации по теме исследования из различных источников, проводилась формулировка гипотезы, определялись постановка цели, задач, предмета исследования, объекта исследования и выполнялось определение проблематики данного исследования.

Второй этап – поисковый. В ходе проведения данного этапа осуществлялся анализ методов и технологий управления поиском информации в торговой организации, опубликована научная статья по теме исследования в научном сборнике.

На третьем этапе осуществлялась апробация предлагаемых проектных решений, произведена оценка их эффективности, сформулированы выводы о полученных результатах по проведенному исследованию.

На защиту выносятся:

- модели и алгоритмы эффективной системы управления поиском информации в торговой организации;
- результаты апробации и оценки эффективности предлагаемых проектных решений.

По теме исследования опубликована 1 статья:

Котлов В.С. Классификатор текстовых документов // Вестник научных конференций (принята к публикации).

Диссертация состоит из введения, четырех глав, заключения и списка литературы.

В первой главе дан анализ современного состояния исследований в области построения систем управления поиском информации в торговой организации.

Во второй главе дан анализ методологических подходов к построению систем управления поиском информации в торговой организации.

Третья глава посвящена разработке моделей и алгоритмов эффективной системы управления поиском информации в торговой организации.

В четвертой главе выполнены апробация предлагаемых проектных решений и оценка их эффективности.

В заключении приводятся результаты исследования.

Работа изложена на 70 страницах и включает 22 рисунка, 8 таблиц и 38 источников.

Глава 1 Анализ современного состояния исследований в области построения систем управления поиском информации в торговой организации

1.1 Анализ проблемы применения корпоративных поисковых систем в торговой сфере

Информационный поиск — это область, связанная со структурой, анализом, организацией, хранением, поиском и извлечением информации. Это деятельность по получению информационных ресурсов, соответствующих информационной потребности, из набора информационных ресурсов.

Корпоративный поиск — это прежде всего поиск документов, причем не только в специализированных базах данных и архивированных каталогах, но и в массивах с неструктурированной информацией, таких как Интернет, почтовые сообщения, рабочие документы, тексты, голосовые и видеоданные, презентации и т.д.

Информационно-поисковая система (Information Retrieval System, IRS) предназначена для поиска документов или информации, необходимой сообществу пользователей. Это должно сделать реальную информацию доступной для реального пользователя.

Таким образом, информационно-поисковая система (ИПС) нацелена на сбор и систематизацию информации в одной или нескольких предметных областях с целью предоставления ее пользователю в кратчайшие сроки. Другими словами, она служит мостом между генераторами информации и ее пользователями.

Структурно-функциональная схема типовой ИПС представлена на рисунке 1.

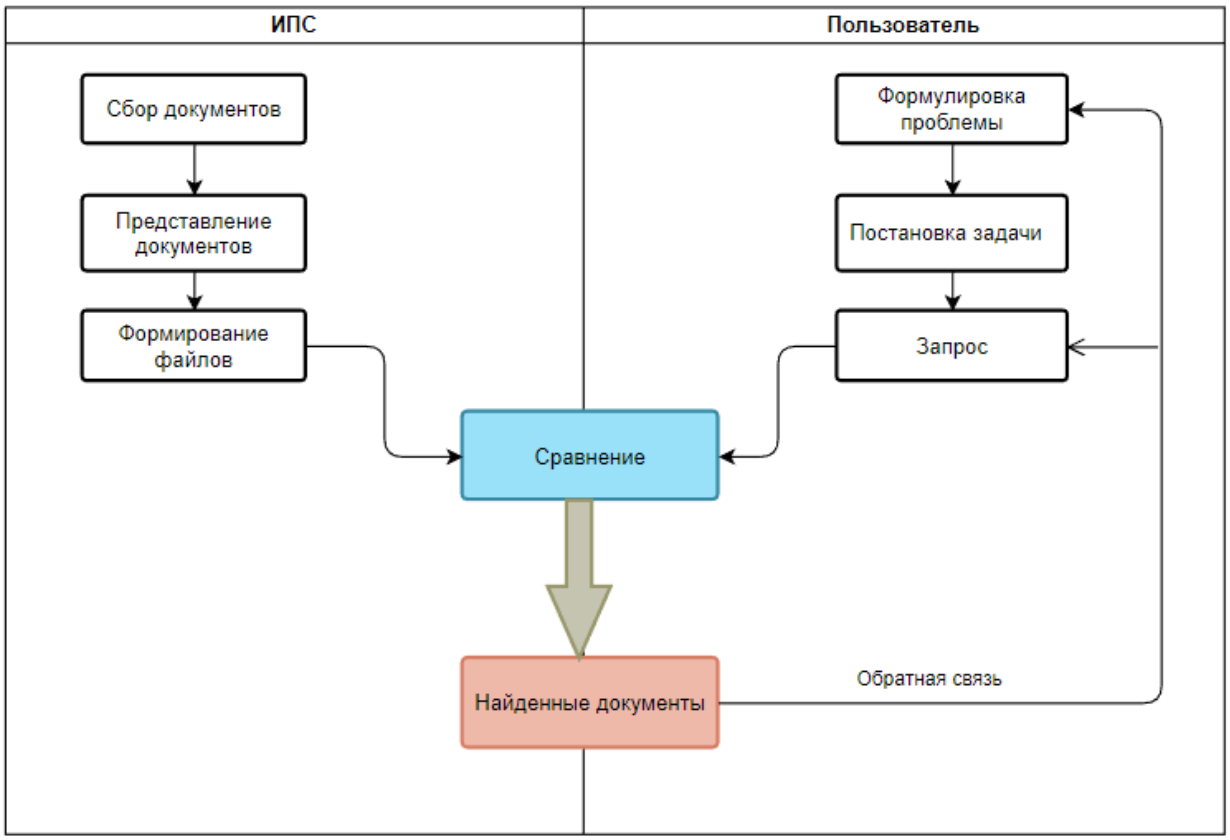


Рисунок 1 – Структурно-функциональная схема типовой ИПС

Следует отметить, что современные ИПС являются системами управления поиском информации [27].

Такие методы, как предложение запроса, расширение запроса и обратная связь по релевантности, используют взаимодействие и контекст для уточнения исходного запроса, чтобы получить более качественные результаты.

«Для комплексного решения проблемы корпоративного поиска разрабатываются различные программы, иногда значительно отличающиеся от классических информационно-поисковых систем.

Отличия заключаются в первую очередь в том, что корпоративные поисковые системы являются узкоспециальными, поэтому могут быть использованы лишь на ограниченном количестве предприятий» [14].

К категории таких ИПС относятся СУПИ в торговых организациях.

Следует отметить, что интерес к СУПИ для торговой сферы возрос

благодаря бурному развитию сектора e-commerce (электронной коммерции).

Как показывает практика, необходимость использования корпоративных поисковых систем в торговых организациях для решения задач рекламы и маркетинга не вызывает сомнения. Основными проблемами электронной коммерции, которые позволяют решить СУПИ, являются извлечение актуальной информации о товарах и предложение клиентам товаров, отвечающих их потребностям.

Наиболее востребованным является поиск информации об отзывах клиентов торговой организации о качестве предлагаемых товаров.

В этой связи одним из требований к функционалу современной СУПИ является возможность анализа тональности отзывов клиентов.

Обозреватели и аналитики признают, что направления развития корпоративных поисковых систем в торговой сфере со временем могут стать сверхприбыльными, поскольку спрос на этот вид поиска активно растет [15].

Таким образом, проблема применения СУПИ в торговых организациях представляет практический интерес.

1.2 Анализ современных подходов к построению систем управления поиском информации в торговой организации

Рассмотрим современные подходы к построению СУПИ в торговой организации, представленные в российских и зарубежных источниках.

В работе [4] предлагается поэтапная технология разработки ИПС предприятия торговли (рисунок 2).

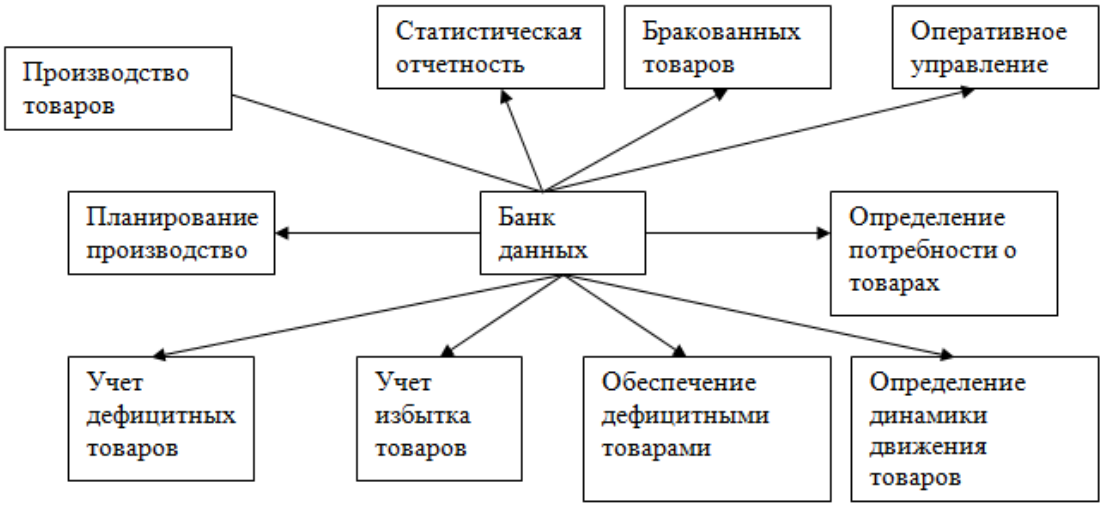


Рисунок 2 – Структурная схема ИПС предприятия торговли

«На первом этапе сформируется банк данных всех предприятий торговли принадлежащих по отраслевому, так и по территориальному признаку.

На втором этапе следует создать банк данных всех работающих производителей. Этот этап можно разделить на два подэтапа:

- формирование банков данных для всех промышленных предприятий;
- дополнение первоначального банка данных файлами информации других предприятий и организаций.

На третьем этапе, когда будет накоплен определенный опыт работы с системой, и когда будут функционировать автономные банки данных по выпускаемым товарам и всем работающим предприятий, сложатся реальные условия для разработки и внедрения ИПС по торговли.

По мнению авторов, поэтапная разработка ИПС создает условия для приобретения навыков по проектированию к работе с системой. При этом будет решаться ряд проблем организационного характера. К ним относятся: подготовка специалистов, сбор и обработка первичной информации о производственных товарах, создание постоянно действующей системы внесения изменений в данные, хранящиеся в сервере и др.» [4].

Примером промышленного решения корпоративной ИПС является система Naumen Enterprise Search — поисковая система для решения задач в области корпоративного поиска информации [12].

Интеллектуальная система поиска с применением машинного обучения и средств обработки естественного языка (Natural Language Processing, NLP) способна обрабатывать большие массивы информации и делать накопленную информацию доступной для сотрудников организации.

Окно ввода поискового запроса системы показано на рисунке 3.

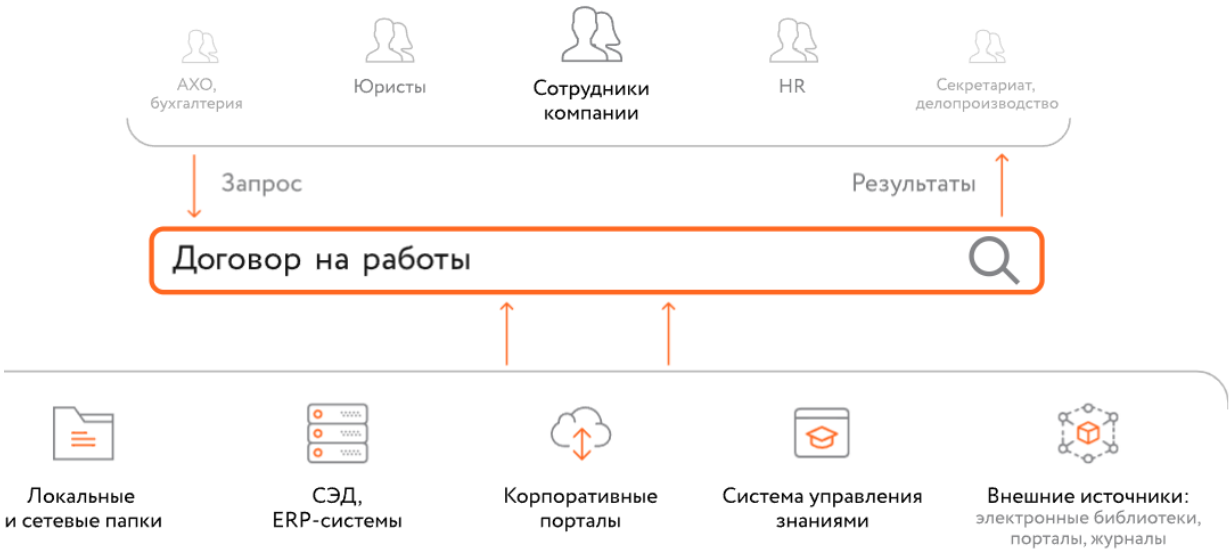


Рисунок 3 – Окно ввода поискового запроса

Системой Naumen Enterprise Search смогут пользоваться все сотрудники компании. В первую очередь, система поиска поможет снизить трудозатраты у специалистов, ежедневно работающих с документами в производственных подразделениях, финансовых, юридических и кадровых службах, секретарей и делопроизводителей.

В работе [22] представлен автоматический поиск общедоступной информации о продуктах в Интернете, анализ и агрегирование этой

информации и отправка соответствующих уведомлений пользователям.

Разработан Telegram-бот, который периодически проверяет информацию по определенным товарам и анализирует полученные данные.

Это избавляет покупателей от периодических посещений интернет-магазинов. Приведены примеры использования разработанного приложения для получения информации о наличии товара, повышения или понижения цены, отзывов о товаре, размещения новых объявлений на сайтах продаж. Из-за отсутствия единого стандарта представления информации о товарах в электронной коммерции разработаны адаптеры для каждого сайта.

Приложение основано на методах извлечения веб-данных, микросервисах и заданиях cron.

На рисунке 4 показана схема сервиса веб-скрейпинга информации.

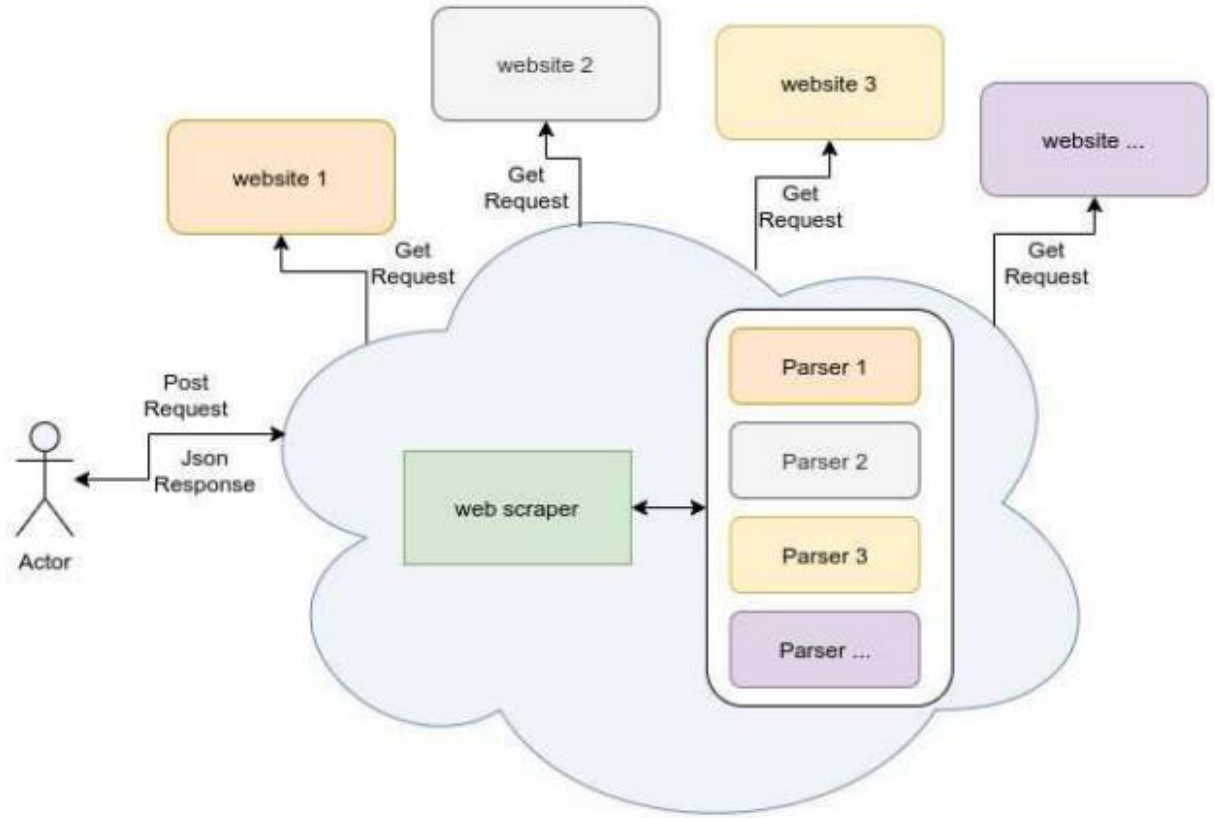


Рисунок 4 – Схема сервиса веб-скрейпинга информации

В работе [16] представлена информационно-поисковая система, основанная на онтологии электронной коммерции, представляющая различные свойства продуктов в Интернете, которая будет использоваться для построения модели индексирования (рисунок 5).

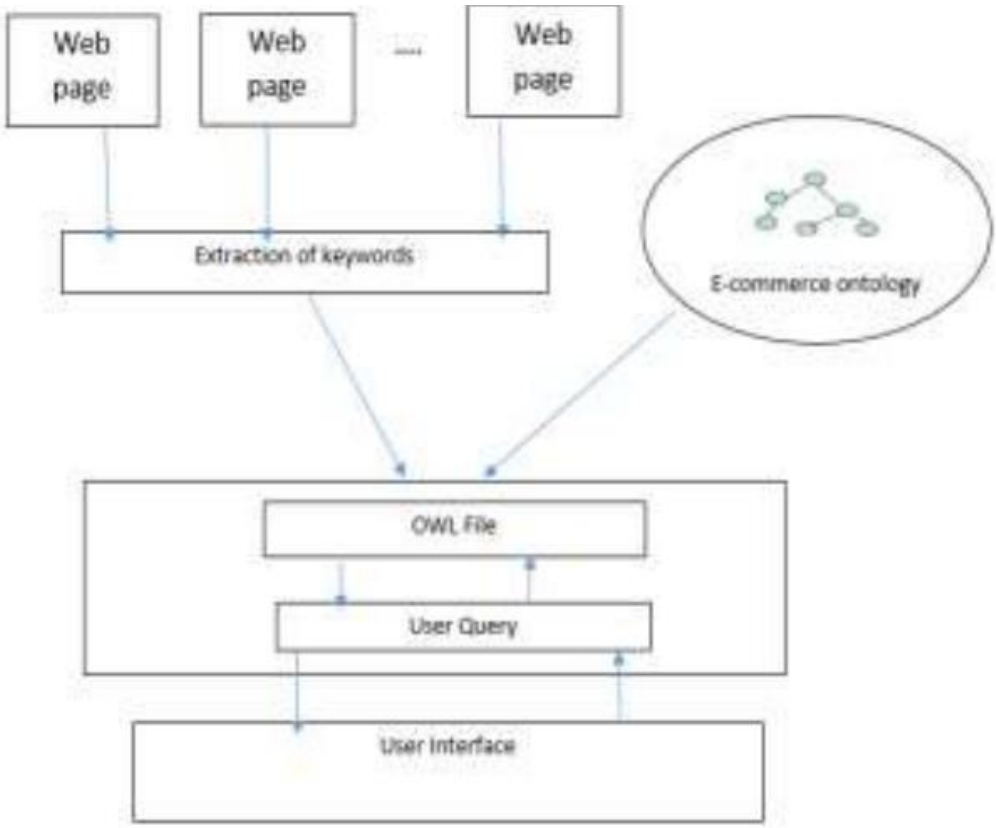


Рисунок 5 – Структурная схема ИПС на основе онтологии электронной коммерции

Пользователь формирует запрос, вводя ключевые слова. Далее этот запрос будет обработан и преобразован в запрос SPARQL, который будет запущен на данных репозитория и вернет список экземпляров, которые соответствуют этому запросу, а также ссылку к соответствующим документам.

Авторы утверждают, что предлагаемая ИПС поможет пользователям получить желаемую информацию о продуктах, которые отвечают их потребностям.

В работе [30] представлена архитектура ИПС, основанной на онтологии системы поиска информации о продуктах электронной коммерции и предлагается основанная на онтологии адаптация классической модели векторного пространства с учетом веса атрибута продукта (рисунок 6).

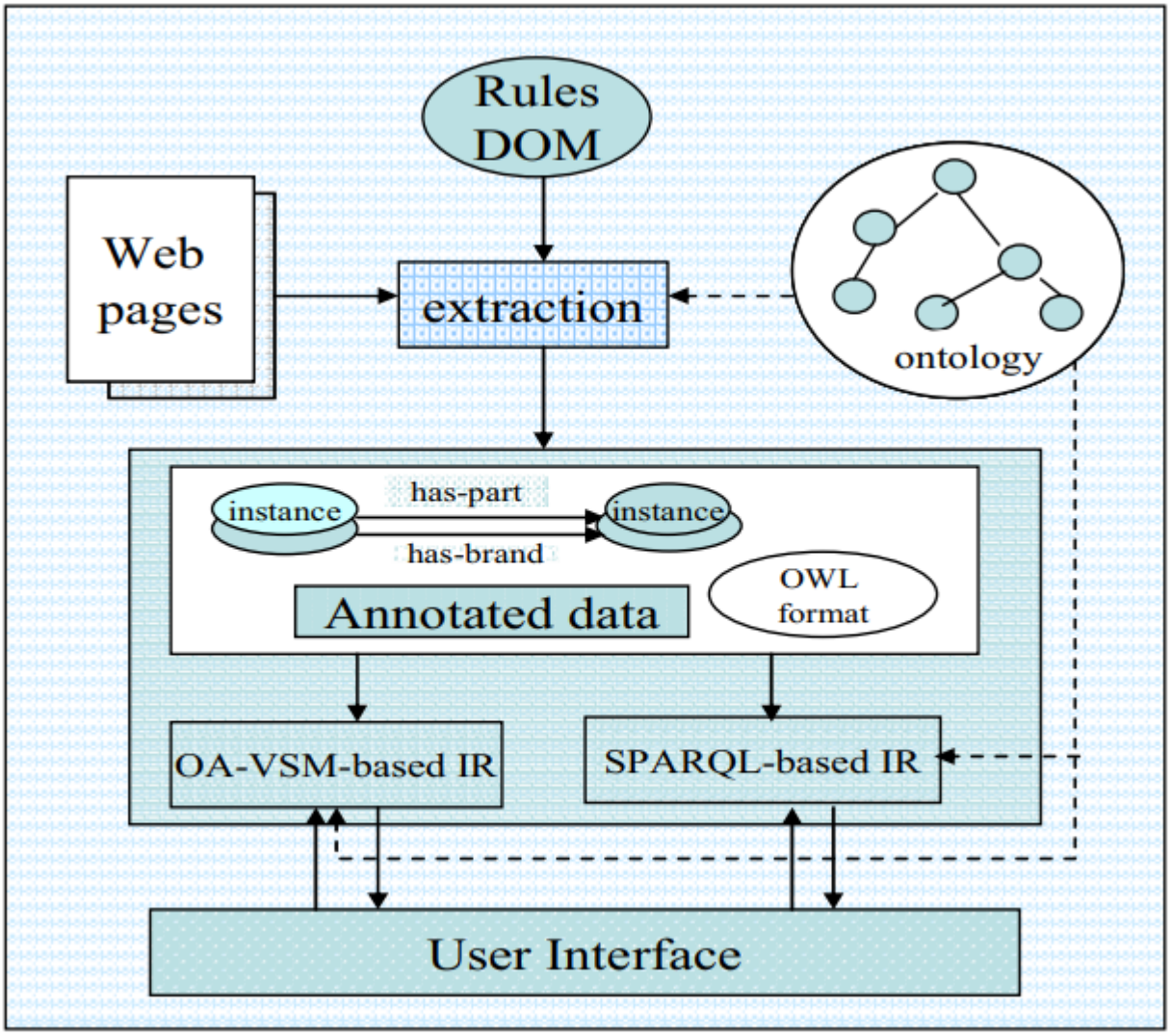


Рисунок 6 – Архитектура ИПС на основе онтологии системы поиска информации о продуктах электронной коммерции

Построена онтология, связанная с компьютером и компонентами, которая используется для аннотирования html-документов и построения концептуальных векторов документов.

В настоящее время активно развиваются решения, в которых

используются методы интеллектуального анализа текста (Text mining).

Так, в работе [19] представлена архитектура ИПС, использующую рекуррентную сверточную нейронную сеть (RCNN), которая эффективно извлекает текстовые документы и информацию для пользовательского запроса (рисунок 7).

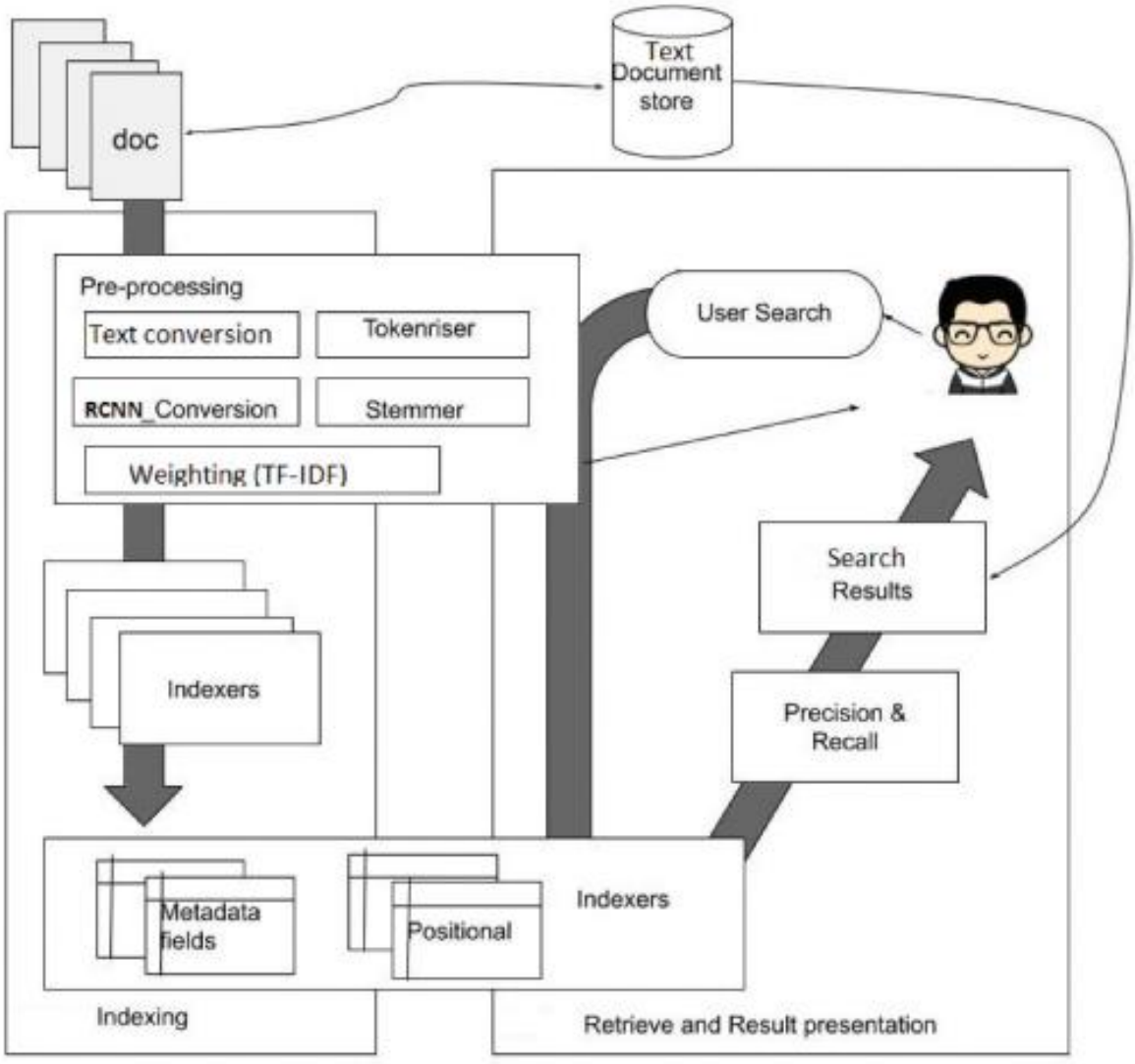


Рисунок 7 – Архитектура системы поиска документов

Как было отмечено выше, одним из требований к функционалу современной СУПИ является возможность анализа тональности отзывов клиентов.

Анализ тональности позволяет синтезировать тональность клиентов и определять бизнес-аспекты как возможности для улучшения. Эта функция поможет менеджменту торговой организации понять, что работает хорошо, а что нужно исправить. Это может помочь управлять бизнес-действиями, имеющими опыт, который дает высокую степень удовлетворенности и лояльности клиентов.

Клиенты интернет-магазинов часто пишут отзывы, делятся своими впечатлениями и опытом использования купленных товаров.

Поскольку такие отзывы доступны большому кругу интернет-пользователей по всему миру, они играют важную роль в формировании общественного мнения о товаре, способствуют росту или сокращению объема продаж того или иного товара.

Отзывы, опубликованные в сети, представляют собой уникальный ресурс объективных данных о продукции: покупатели делятся своим мнением о товарах с другими потенциальными покупателями без участия производителя [8].

В последнее время для обеспечения эффективного поиска в интернет-магазинах используются внутренние поисковые системы с элементами искусственного интеллекта [17].

«Умный» поиск по сайту электронной коммерции может повысить качество обслуживания посетителей, повысить лояльность клиентов и повысить коэффициент конверсии на сайте.

По некоторым данным, посетители, использующие поиск, приносят от 30 до 60% всех доходов сайта электронной коммерции.

На рисунке 8 представлена диаграмма сравнения коэффициентов конверсий крупнейших платформ электронной коммерции.

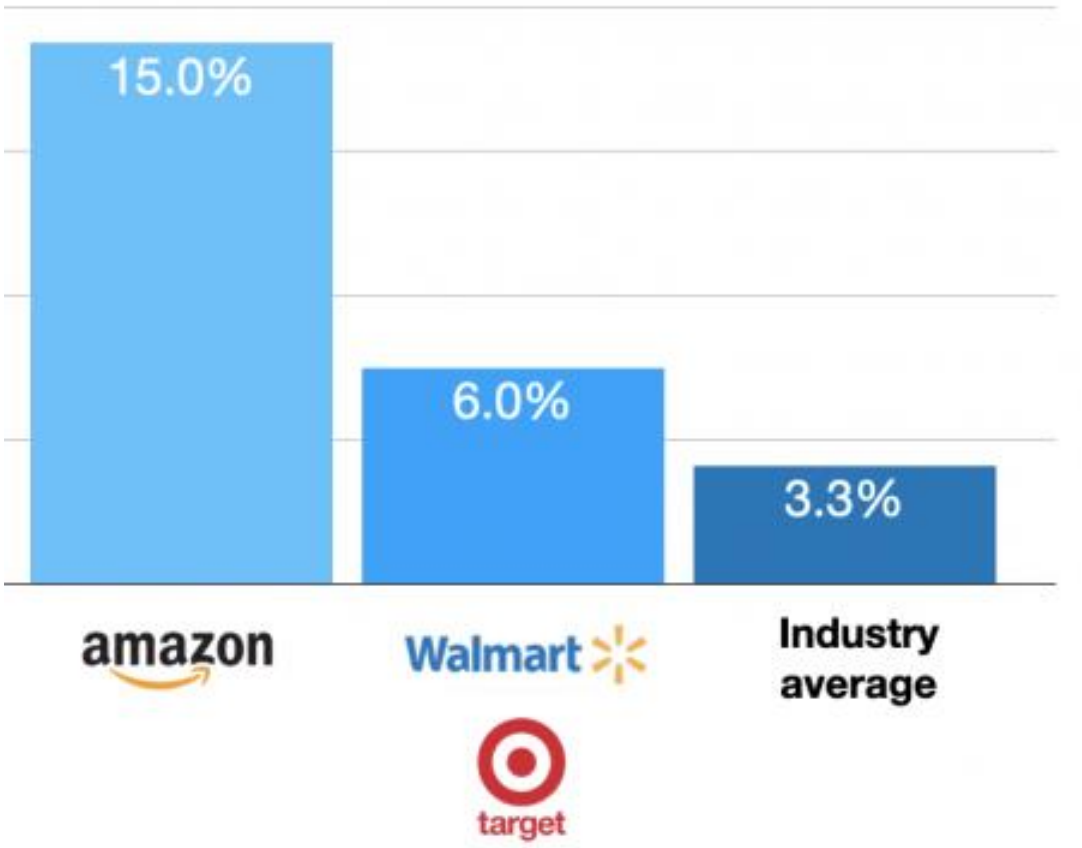


Рисунок 8 – Диаграмма сравнения коэффициентов конверсий крупнейших платформ электронной коммерции

В Amazon.com коэффициент конверсии в пять раз превышает средний показатель по отрасли (он даже выше для членов Prime). Поиск жизненно важен на крупнейшем в мире рынке для поиска чего угодно, поэтому, естественно, Amazon вложила значительные средства в поисковую инженерию в течение 20 лет; сегодня над поиском по сайту Amazon работают более 1500 человек [20].

Одной из задач, которую позволяет решить внутренний поисковик – это персонализированный мерчандайзинг.

В [9] представлены характеристики продвинутых поисковых движков для электронной коммерции.

На рисунке 9 представлена структурно-функциональная схема поисковика Algolia.

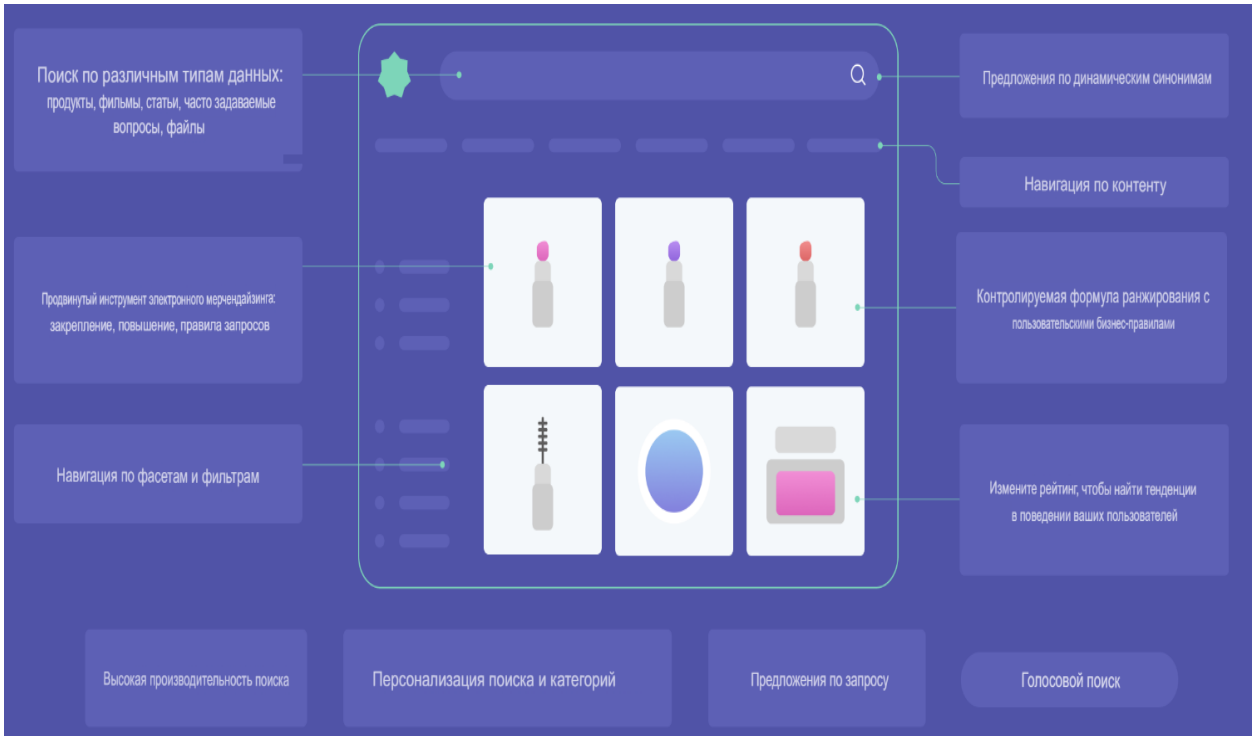


Рисунок 9 – Структурно-функциональная схема поисковика Algolia

Среди недостатков внутренних поисковик для электронной коммерции можно выделить следующие:

- затраты времени на поиск;
- сложность конкурентных ключевых слов;
- изменение алгоритмов.

Вместе с тем проведенный анализ позволил констатировать недостаточность работ, посвященных проблеме построения систем управления поиском корпоративной информации в торговых организациях, что подтверждает актуальность темы настоящего исследования.

Выводы по главе 1

Результаты проделанной работы позволили сделать следующие выводы:

- по своим архитектурным и функциональным особенностям СУПИ в торговой сфере относятся к корпоративным поисковым системам;

- в торговых организациях корпоративные поисковые системы используются для решения задач рекламы и маркетинга;
- одним из требований к функционалу современной СУПИ является возможность анализа тональности отзывов клиентов о качестве приобретаемых товаров;
- в настоящее время активно развиваются решения, в которых используются методы интеллектуального анализа текста;
- для обеспечения эффективного поиска в интернет-магазинах используются внутренние поисковые системы с элементами искусственного интеллекта;
- среди недостатков внутренних поисковиков для электронной коммерции можно выделить следующие: затраты времени на поиск, сложность конкурентных ключевых слов, изменение алгоритмов.

Вместе с тем проведенный анализ позволил констатировать недостаточность работ, посвященных проблеме построения систем управления поиском корпоративной информации в торговых организациях, что подтверждает актуальность темы настоящего исследования.

Глава 2 Анализ методов и технологий управления поиском информации в торговой организации

2.1 Методы интеллектуального анализа текста

«Интеллектуальный анализ текста (text mining, также называемый текстовой аналитикой) – это технология искусственного интеллекта (ИИ), которая использует обработку естественного языка (NLP) для преобразования свободного (неструктурированного) текста в документах и базах данных в нормализованные структурированные данные, подходящие для поиска, анализа или управления машинным обучением (ML)» [38].

В то время как традиционные поисковые системы, такие как Google, теперь предлагают уточнения, такие как синонимы, автозаполнение и семантический поиск (история и контекст), подавляющее большинство результатов поиска указывают только на местоположение документов, оставляя поисковиков с проблемой необходимости тратить часы вручную извлечение необходимых данных путем чтения отдельных документов.

Ограничения традиционного поиска усугубляются ростом больших данных за последнее десятилетие, что помогло увеличить количество результатов, возвращаемых по одному запросу поисковой системой, такой как Google, с десятков тысяч до сотен миллионов.

Исследование, проведенное в декабре 2018 года Международной корпорацией данных (IDC), показало, что объем больших данных, по прогнозам, будет расти быстрее в здравоохранении, чем в производстве, финансовых услугах или СМИ в течение следующих семи лет: совокупный годовой темп роста (CAGR) составляет 36%.

В розничной торговле – это таргетированная реклама для выделения конкретных продуктов на основе условий в запросах клиентов; обучение чат-ботов автоматическим ответам на распространенные запросы клиентов о продуктах.

Интеллектуальный анализ текста анализирует массу неструктурированного текста, классифицируя каждый документ по его основной теме, намерению и настроению (положительному, отрицательному или нейтральному). Он использует методы NLP для анализа неструктурированного текста, а затем методы ИИ, такие как машинное обучение, для классификации документов. Этот процесс раскрывает закономерности и отношения, которые в противном случае были бы скрыты в тексте. Алгоритмы машинного обучения также могут создавать модели, которые предсказывают новые модели и модели поведения.

Программное обеспечение для интеллектуального анализа текста использует NLP вместе с системами на основе правил и машинным обучением, чтобы обнаруживать скрытые взаимосвязи, шаблоны и настроения в текстовых документах.

Сначала неструктурированный текст предварительно обрабатывается с помощью NLP.

Эта предварительная обработка может включать любой из следующих этапов [28]:

- очистка – удаление маленьких слов (a, an, the) и исправление орфографических ошибок;
- стемминг – сокращение слова до его основы путем удаления префиксов и суффиксов (например, «нанять» — это основа как для «найма», так и для «нанят»);
- токенизация – разделение текста на отдельные слова и словосочетания.

«Метод токенизации – процесс сегментации текста на слова или предложения. Электронный текст представляет собой линейную последовательность символов (символов, слов или фраз). Естественно, прежде чем приступить к реальной обработке текста, текст должен быть разделен на лингвистические единицы, такие как слова, знаки препинания, числа, цифры

и т. д.

Этот процесс называется токенизация. В английском языке слова часто отделены друг от друга пробелами (пробелами), но не все пробелы равны.

Токенизация – это своего рода предварительная обработка; идентификация базовых единиц, подлежащих обработке. Без этих четко разделенных базовых единиц, невозможно провести какой-либо анализ или генерацию. Идентификация единиц, которые не нуждаются в дальнейшей декомпозиции для последующей обработки, является чрезвычайно важной.

Ошибки, сделанные на этом этапе, вызовут больше ошибок на более поздних этапах обработки текста» [11].

Блок схема алгоритма токенизации показана на рисунке 10.

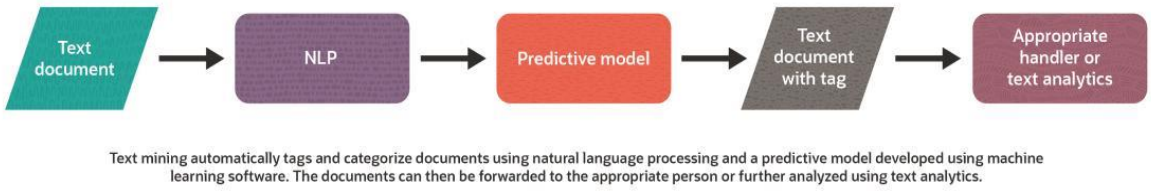
Рисунок 10 – Блок-схема алгоритма токенизации

На этапе обозначения частей речи определяются части речи в тексте, такие как существительные, глаголы и прилагательные.

На этапе синтаксиса разбора проводится анализ структуры предложений и словосочетаний для определения роли разных слов. Это идентифицирует, например, подлежащее, глагол и объект предложения.

В результате, получим данные, готовые для моделей машинного обучения, которые определяют шаблоны и отношения в документах. Каждую модель машинного обучения необходимо сначала обучить, передав ей документы, которые были вручную помечены как принадлежащие к определенной категории или содержащие определенное настроение. На основе входных данных обучения система машинного обучения создает прогностическую модель. Затем новые документы передаются в прогностическую модель, которая присваивает соответствующую классификацию или определяет тональность документа.

Диаграмма на рисунке 11 представляет процесс интеллектуального анализа текста.



Text mining automatically tags and categorize documents using natural language processing and a predictive model developed using machine learning software. The documents can then be forwarded to the appropriate person or further analyzed using text analytics.

Рисунок 11 – Процесс интеллектуального анализа текста

Программное обеспечение для анализа текста сначала предварительно обрабатывает текстовый документ с использованием NLP, а затем передает его в прогностическую модель, построенную с использованием обучающих данных.

Модель присваивает документу тег категории. В зависимости от категории документ может быть направлен соответствующему обработчику или программному обеспечению для анализа текста.

Например, если система анализирует электронные письма, которые клиенты отправляют на адрес электронной почты службы поддержки клиентов компании, и модель присваивает тег «дефектный продукт», электронное письмо может быть перенаправлено в отдел обеспечения качества для изучения проблемы.

Программное обеспечение для текстовой аналитики также может проверять помеченные электронные письма, чтобы определить процент дефектных доставленных продуктов.

Интеллектуальный анализ текста фокусируется на извлечении ценности из неструктурированного текста, который сегодня составляет большую часть бизнес-данных. Инструменты анализа текста используют несколько основных и расширенных методов для извлечения смысла из текста.

Основные методы анализа текста исследуют отдельные слова или фразы в каждом документе.

К ним относятся:

- частота слов — это метод, который находит слова, наиболее часто используемые в документе, распознавая синонимы. Он используется, чтобы помочь определить тему документа;
- словосочетание находит слова, которые обычно встречаются вместе в последовательности или в одном предложении, например, «пользовательский интерфейс». Это помогает определить их значение. Последовательности могут состоять из двух слов (например, «обслуживание клиентов»), которые называются биграммами, или из трех слов («сарафанное радио», «предполагаемая дата доставки»), которые называются триграммами;

- согласие определяет значение слова на основе его контекста. Ведь многие слова в английском языке имеют несколько значений. Например, означает ли слово «бассейн» место для купания, игру с участием кия и бильярдных шаров или кучу внесенных денег?

Расширенные методы анализа текста учитывают контекст всего документа или ищут темы в нескольких документах.

К ним относятся нижеследующие методы.

Классификация текста – это процесс категоризации текста по одному или нескольким различным классам для организации, структурирования и фильтрации по любому параметру.

Классификация текста – метод, который определяет тему, намерение и настроение документа:

- тематический анализ определяет основной предмет или тему документа и, возможно, также второстепенные темы;
- анализ настроений определяет эмоции, настроение или чувства, выраженные в документе, независимо от того, являются ли они положительными, отрицательными или нейтральными;
- идентификация языка классифицирует документ на основе его естественного языка, например, английского или испанского;
- идентификация намерения определяет цель документа. Пытается ли клиент купить продукт или получить дополнительную информацию перед покупкой?

Извлечение текста, также называемое извлечением информации, выделяет важные слова или другие данные в документе:

- извлечение ключевых слов находит наиболее распространенные или наиболее важные слова в тексте, особенно слова, которые встречаются в этом документе чаще, чем в аналогичных документах;
- распознавание именованных объектов извлекает имена организаций, лиц, продуктов или мест. Это может быть полезно для отслеживания

разговоров в социальных сетях о продуктах компании и конкурентов;

- распознавание характеристик извлекает фразы, описывающие характеристики продукта, такие как цвет и размер, или информацию о клиенте, такую как номера телефонов и адреса.

Анализ нескольких документов выявляет тенденции и закономерности в нескольких документах:

- кластеризация собирает документы в группы на основе общих характеристик, которых нет в других группах. Например, кластеризация может помочь идентифицировать спам-сообщения, в которых используются одни и те же фразы;
- совместное появление определяет появление одних и тех же терминов в разных документах. Например, обнаружение одного и того же названия продукта вместе со словом «ошибки» в нескольких документах может означать наличие проблем с продуктом;
- анализ тренда находит различия в темах или в том, как эти темы рассматриваются в разные периоды времени. Появляются ли новые темы, в то время как некоторые темы исчезают?

Как было отмечено выше одним из требований к функционалу современной СУПИ является возможность анализа тональности отзывов клиентов о качестве приобретаемых товаров.

На основе результатов анализа методов интеллектуального анализа текста для анализа отзывов клиентов торговой организации выбираем методы классификации текста на основе машинного обучения.

Для упрощения, отзывы по конкретному товару можно разделить на негативные и позитивные [6].

Метод бинарной классификация – это один из типов задач классификации в машинном обучении, когда необходимо классифицировать два взаимоисключающих класса [1].

Рассмотрим основные методы классификации текста на основе машинного обучения на предмет использования для анализа тональности отзывов клиентов [18].

Поточечный метод (Pointwise Methods) является самым простым в реализации, и он был первым, предложенным для обучения ранжированию задач.

Потеря напрямую измеряет расстояние между основным истинным результатом y_i и прогнозируемым s_i, поэтому мы решаем эту задачу, эффективно решая проблему регрессии.

В качестве примера, при ранжировании подмножеств используется потеря среднеквадратичной ошибки (MSE) (1):

$$L(s, y) = \sum_{i=1}^{n} (s_i - y_i)^2 \quad (1)$$

Попарные методы (Pairwise Methods) не работают с абсолютной релевантностью. Вместо этого они работают с относительными предпочтениями: учитывая два документа, мы хотим предсказать, является ли первый более релевантным, чем второй.

Таким образом мы решаем бинарные классификация задач , где нам нужна только правда земли y_{ij} (=1, Если $y_i > y_j$, 0-нет) и мы с модельными данными для вероятностей с помощью логистической функции (2):

$$s_{ij} = \sigma(s_i - s_j). \quad (2)$$

Этот подход был впервые использован в RankNet, который использовал потерю двоичной перекрестной энтропии(2):

$$L(s, y) = -\sum_{i,j=1}^{n} y_{ij} \log(s_{ij}) + (1 - y_{ij}) \log(1 - s_{ij}) \quad (3)$$

«Списочный (listwise) метод заключается в построении модели, на вход которой поступают сразу все документы, соответствующие запросу, а на выходе получается их перестановка. Подгонка параметров модели осуществляется для прямой максимизации одной из перечисленных выше метрик ранжирования. Но это часто затруднительно, так как метрики ранжирования обычно не непрерывны и недифференцируемы относительно параметров ранжирующей модели, поэтому прибегают к максимизации неких их приближений или нижних оценок» [23].

Для сравнения методов классификации текста разработана таблица 1.

Таблица 1 – Сравнение методов классификации текста на основе машинного обучения

Метод	Преимущества	Недостатки
Поточечный	Оценка для каждого документа не зависит от других документов, которые находятся в списке результатов для запроса	Относительно невысокая эффективность.
Попарный	Прогнозирование относительного порядка ближе к природе ранжирования	Оптимизируемый фунционал качества оценивает глобальный порядок, а не порядок для одной группы. Не учитываются зависимости между сравниваемыми парами в общей группе
Списочный	Решают проблему ранжирования более непосредственно, максимизируя оценочную метрику	Сложность реализации

По результатам сравнительно анализа для классификации текста на основе машинного обучения выбран попарный метод.

2.2 Анализ алгоритмов бинарной классификации текста

Как было отмечено выше, для анализа тональности отзывов используется метод бинарной классификации.

Для решения данной задачи используются программные средства-бинарные классификаторы, реализующие алгоритмы классификации текста.

Диаграмма деятельности бинарного классификатора текста показана на рисунке 12.

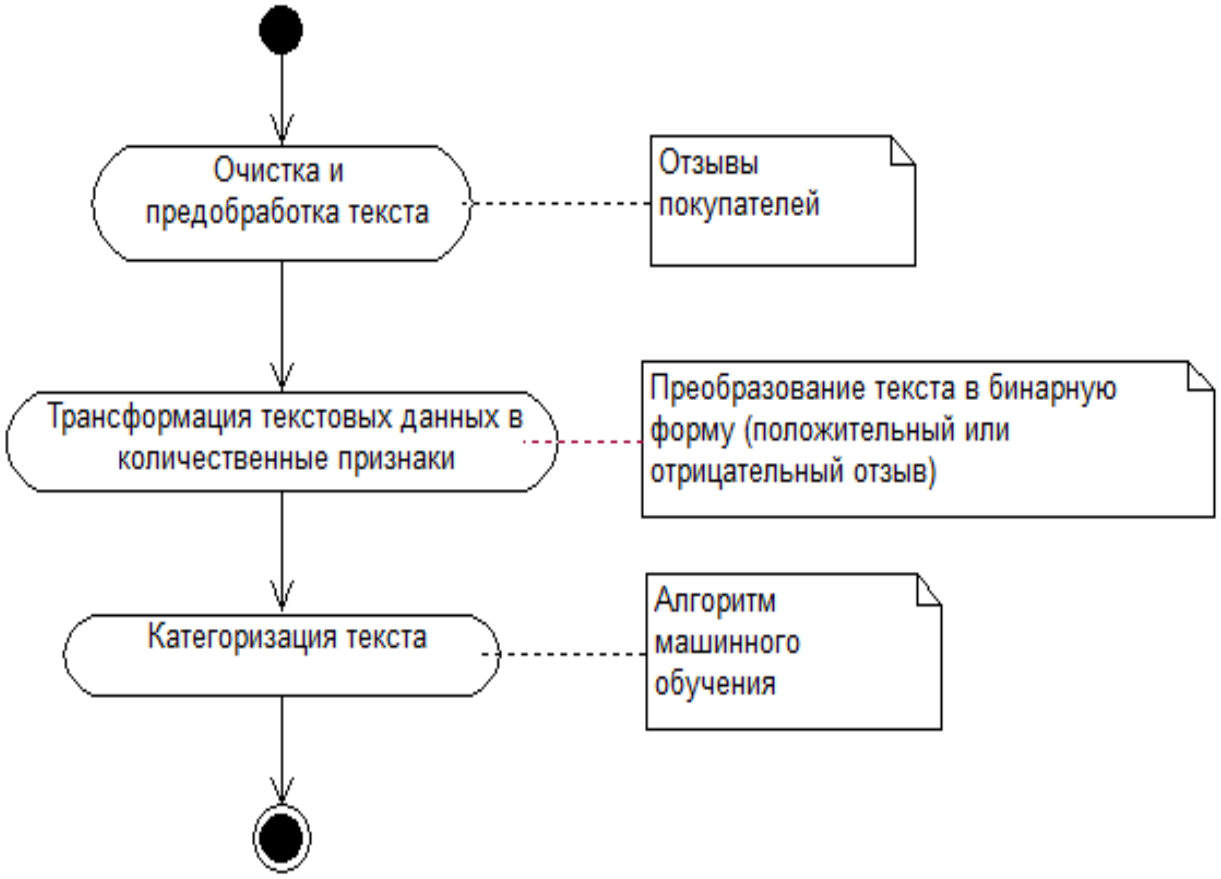


Рисунок 12 – Диаграмма деятельности бинарного классификатора текста

Рассмотрим известные алгоритмы бинарной классификации текста.

2.2.1 Наивный Байесовский алгоритм

Наивный байесовский классификатор – это обычный алгоритм статистической классификации, который классифицирует входные выборки на основе теоремы Байеса и независимости элементов собственного вектора.

Принцип его работы заключается в следующем: сначала текстовый вектор генерируется из текста, подлежащего классификации, то есть текстовый вектор документа x создается на основе заданного словаря, где n - количество элементов в наборе словарей W, и представляет собой количество вхождений в документ d (4):

$$W=\{w_1,w_2,\cdots,w_n\} \quad (4)$$
$$X=\{x_1,x_2,\cdots,x_n\}$$

Учитывая обучающие данные, вектор классификации указывает, что текст может быть разделен на m классов (5):

$$D=\{d_1,d_2,\cdots,d_m\} \quad (5)$$
$$Y=\{y_1,y_2,\cdots,y_m\}$$

D используется для обучения наивного байесовского алгоритма. Исходя из этого, классификация с максимальной вероятностью заключается в решении (6):

$$\text{argmaxP(yj|X)} \quad (6)$$

Согласно теореме Байеса (7):

$$P(y_j|X) = \frac{P(y_i)P(X|y_j)}{P(X)} = \frac{P(y_j)\prod_{i=1}^{n}P(x_i|y_j)}{\prod_{i=1}^{n}P(x_i)} \quad (7)$$

Из приведенных обучающих данных $D=\{d_1,d_2,\cdots,d_m\}$ мы можем вычислить $P(y_j)$ и каждый $P(x_i)$.

Итак, для новых входных данных X мы можем вычислить все $P(y_j|X)$, в котором наивысшим значением вероятности является результат классификации документа X.

Наивный алгоритм Байеса предполагает, что атрибуты набора данных независимы друг от друга, поэтому логика алгоритма очень проста. И алгоритм относительно стабилен.

Когда данные показывают разные характеристики, производительность классификации наивного Байеса не будет сильно отличаться.

Другими словами, алгоритм Байеса является относительно надежным и не покажет слишком большой разницы для разных типов наборов данных.

2.2.2 Алгоритм метода опорных векторов

Метод опорных векторов (Supporting Vector Machine, SVM) – это новый алгоритм статистической классификации, предложенный Вапником и его командой в Bell Labs в 1995 году, который в основном используется для решения задач бинарной классификации [29].

Основная идея состоит в том, чтобы найти гиперплоскость в пространстве объектов выборки, которая может хорошо разделить все данные выборки и сделать расстояние между точкой выборки и гиперплоскостью максимальным.

Принцип его работы заключается в следующем: текстовый вектор также генерируется из текста, подлежащего классификации, то есть текстовый вектор документа x создается на основе заданного словаря, где n - количество элементов в наборе словарей W, и x_i представляет собой количество вхождений x_i в документе d (8):

$$W=\{w_1,w_2,\cdots,w_n\} \qquad (8)$$
$$X=\{x_1,x_2,\cdots,x_n\}$$

Вектор классификации (9):

$$Y=\{y_1,y_2,\cdots,y_m\},\ y_i \in \{-1,1\}, \tag{9}$$

где $y_i \in \{-1,1\}$ представляет отрицательный класс и положительный класс в дихотомии, соответственно.

Предполагая, что в характеристическом пространстве входных выборочных данных существует граница принятия решения, все точки выборки могут быть разделены гиперплоскостью в соответствии с положительным классом и отрицательным классом, а расстояние между любой точкой выборки и плоскостью больше 1, тогда проблема классификации называется линейно разделяемой.

Граница принятия решения (10):

$$w^T X + b = 0 \tag{10}$$

Короткая точка на плоскости (11):

$$y_i(w^T X + b) \geq 1 \tag{11}$$

На рисунке 13 сплошная линия на рисунке является гиперплоскостью $(w^T X + b = 0)$, а w и b являются вектором нормали и пересечением гиперплоскости $(w^T X + b \geq 1)$, соответственно.

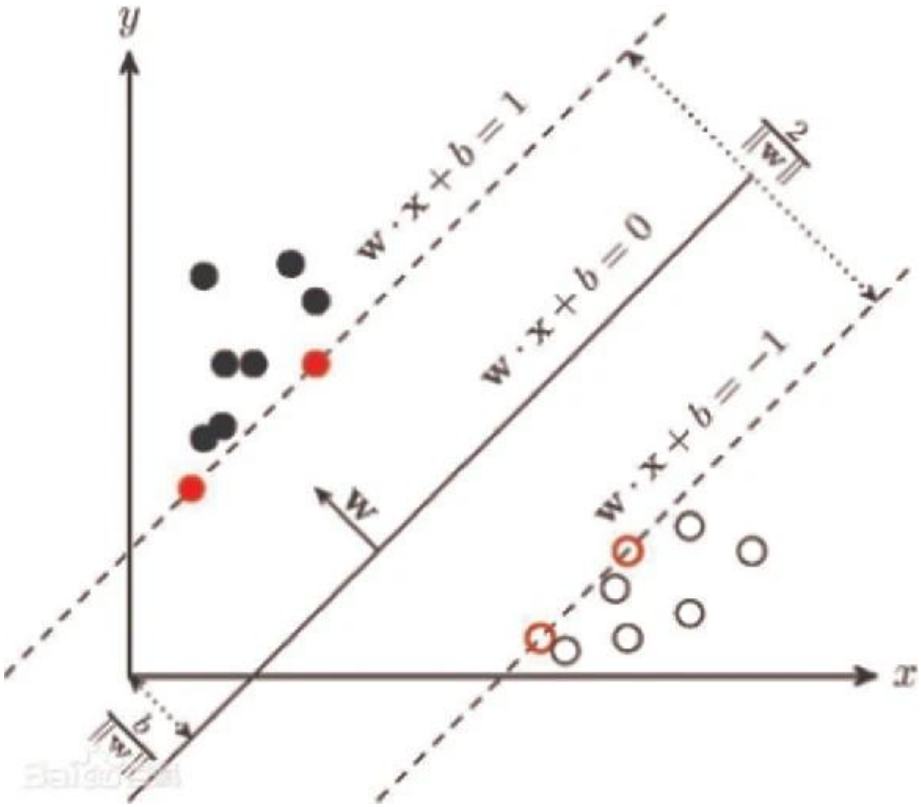


Рисунок 13 – Принцип работы алгоритма SVM

Сплошная красная точка обозначает верхнюю границу интервала нижняя граница интервала определяется полой красной точкой ($w^T X + b \leq -1$), с расстояние между двумя границами интервала d=2/‖w‖.

Алгоритм машинной классификации с опорными векторами обладает хорошей способностью к обобщению и обучению.

Он направлен на минимизацию структурного риска, и полученное решение является глобальным оптимальным решением. Этот алгоритм решает проблему «размерной катастрофы».

Он широко используется в автоматической классификации текста, распознавании лиц, экспрессии генов, распознавании рукописного ввода и других областях.

2.2.3 Дерево решений

Дерево решений – это метод классификации текстов путем построения древовидной системы принятия решений на основе известных вероятностей.

Поскольку ветвь системы принятия решений рисуется подобно ветвям дерева, она называется деревом решений.

Алгоритма дерева решений состоит из следующих шагов:

Шаг 1. Выбор объекта.

Выбор объекта относится к выбору некоторых объектов из текстовых векторов в качестве основы для принятия решения. Эти особенности повлияют на форму ветвей дерева решений и окажут важное влияние на результаты классификации.

Шаг 2. Генерация дерева решений.

Этот шаг является основным процессом принятия решений. Корневой узел представляет собой определенную последовательность слов, то есть существует только одно слово, у которого наилучший процент ошибок классификации среди всех слов и наибольшая вероятность для определенной категории. Последующие дочерние узлы разделяются на левое и правое поддеревья в соответствии с вышеупомянутым решением. Если коэффициент не равен нулю или последовательность слов не имеет дочерней последовательности, решение будет остановлено; если оно не равно нулю и не уникально, решение будет продолжено в возможной категории.

Шаг 3. Обрезка.

Дерево решений сформирует очень большую древовидную систему в соответствии со всеми обучающими выборками, которая обладает высокой точностью в обучающих выборках и низкой точностью в тестовых выборках, образуя явление чрезмерной подгонки. Решение проблемы чрезмерной подгонки требует ручного наблюдения и отладки, наблюдения и контроля размера дерева решений на каждом уровне, установки количества выборок наименьшего конечного узла, настройки минимального веса конечного узла и т.д.

2.2.4 Алгоритм KNN (K-ближайший сосед)

KNN (K-ближайшего соседа) один из простейших алгоритмов

классификации, но он также является одним из наиболее часто используемых алгоритмов классификации].

Алгоритм KNN – это контролируемый алгоритм классификации, который похож на другой алгоритм машинного обучения K-means.

Алгоритм KNN заключается в поиске K текстов в обучающем наборе, которые наиболее похожи на целевой текст в соответствии с известной категорией данных и текстом, подлежащим тестированию, а затем оценивают категорию кандидата в соответствии с K образцами.

Преимущество алгоритма KNN в том, что он прост и удобен в использовании. По сравнению с другими алгоритмами, KNN является относительно простым и понятным алгоритмом. Даже без высокой математической базы можно объяснить его принцип.

Модель имеет быстрое время обучения и хороший эффект прогнозирования.

Недостатком алгоритма KNN является то, что он требует большого объема памяти, поскольку в нем хранятся все обучающие данные, этап прогнозирования может быть очень медленным, и он чувствителен к нерелевантным функциям и масштабу данных.

2.2.5 Алгоритм «Случайный лес»

Случайный лес - это классификатор, который содержит несколько деревьев решений, и результаты его классификации определяются количеством голосов за все выходные результаты дерева.

Посредством интеграции случайный лес объединяет несколько деревьев вместе. Его основной единицей является дерево решений, а каждое дерево решений является классификатором.

Для входной выборки N деревьев будут иметь N результатов классификации. Случайный лес объединяет все результаты голосования по классификации и указывает категорию, набравшую наибольшее количество голосов, в качестве окончательного результата. Это самая простая идея

упаковки.

Принцип случайного леса заключается в:

Шаг 1. Пусть N используется для представления количества обучающих примеров (samples), а M используется для представления количества признаков.

Шаг 2. Количество входных признаков m используется для определения результата принятия решения узлом в дереве решений, где m должно быть намного меньше M.

Шаг 3. Обучающий набор (т.е. выборка начальной загрузки) был сформирован путем выборки N раз из N обучающих примеров (samples), и прогноз был сделан с использованием невыбранных вариантов использования (samples) для оценки ошибки.

Шаг 4. Для каждого узла случайным образом выбираются m признаков, и решение каждого узла в дереве решений определяется на основе этих признаков. В соответствии с этими m характеристиками рассчитывается оптимальный режим разделения.

Шаг 5. Каждое дерево будет расти неповрежденным без обрезки, которая может быть принята после построения обычного древовидного классификатора.

Для сравнения алгоритмов бинарной классификации текста разработана таблица 2.

Таблица 2 – Сравнение алгоритмов бинарной классификации текста

Алгоритм	Преимущества	Недостатки
Наивный байесовский алгоритм	Эффективен, когда связь между атрибутами набора данных относительно независима	Независимость атрибутов наборов данных во многих случаях трудно обеспечить, поскольку атрибуты наборов данных часто коррелируют друг с другом, эффект классификации будет значительно снижен

Продолжение таблицы 2

Алгоритм	Преимущества	Недостатки
SVM	Эффективен при работе с небольшими выборками, нелинейными данными и данными высокой размерности	Ограничен небольшой кластерной выборкой, может обрабатывать задачи классификации только второго класса
KNN	По сравнению с другими алгоритмами, является относительно простым и понятным алгоритмом	Требует больших вычислительных затрат, поскольку для классификации каждого текста необходимо вычислить расстояние от него до всех известных выборок, чтобы получить K ближайших соседних точек
Случайный лес	Поскольку используется интегрированный алгоритм, его точность выше, чем у большинства отдельных алгоритмов, поэтому он обладает высокой точностью, особенно в наборе тестов	Когда в случайном лесу много деревьев решений, пространство и время, необходимые для обучения, будут большими

По результатам сравнительно анализа в качестве алгоритма бинарной классификации текста выбран алгоритм метода опорных векторов SVM.

Главное преимущество данного алгоритма – высокая эффективность для относительно небольших объемов данных, к которым относится бинарный текст.

Кроме того, данный алгоритм рекомендуется во многих работах для анализа бинарного текста.

Выводы по главе 1

Результаты проделанной работы позволили сделать следующие выводы:

- интеллектуальный анализ текста анализирует массу неструктурированного текста, классифицируя каждый документ по

его основной теме, намерению и настроению (положительному, отрицательному или нейтральному);

- по результатам сравнительно анализа для классификации текста на основе машинного обучения выбран попарный метод;
- для анализа тональности отзывов клиентов используется метод бинарной классификации;
- для решения задач бинарной классификации используются программные средства - бинарные классификаторы, реализующие бинарные алгоритмы классификации текста;
- по результатам сравнительно анализа в качестве алгоритма бинарной классификации текста выбран алгоритм метода опорных векторов SVM.

Главное преимущество алгоритма SVM – высокая эффективность для относительно небольших объемов данных, к которым относится бинарный текст.

Кроме того, данный алгоритм рекомендуется во многих работах для анализа бинарного текста.

Глава 3 Разработка моделей и алгоритмов системы управления поиском информации в торговой организации

3.1 Алгоритма системы управления поиском информации в торговой организации

Блок-схема алгоритма бинарной классификации текста на основе SVM показан на рисунке 14 [24].

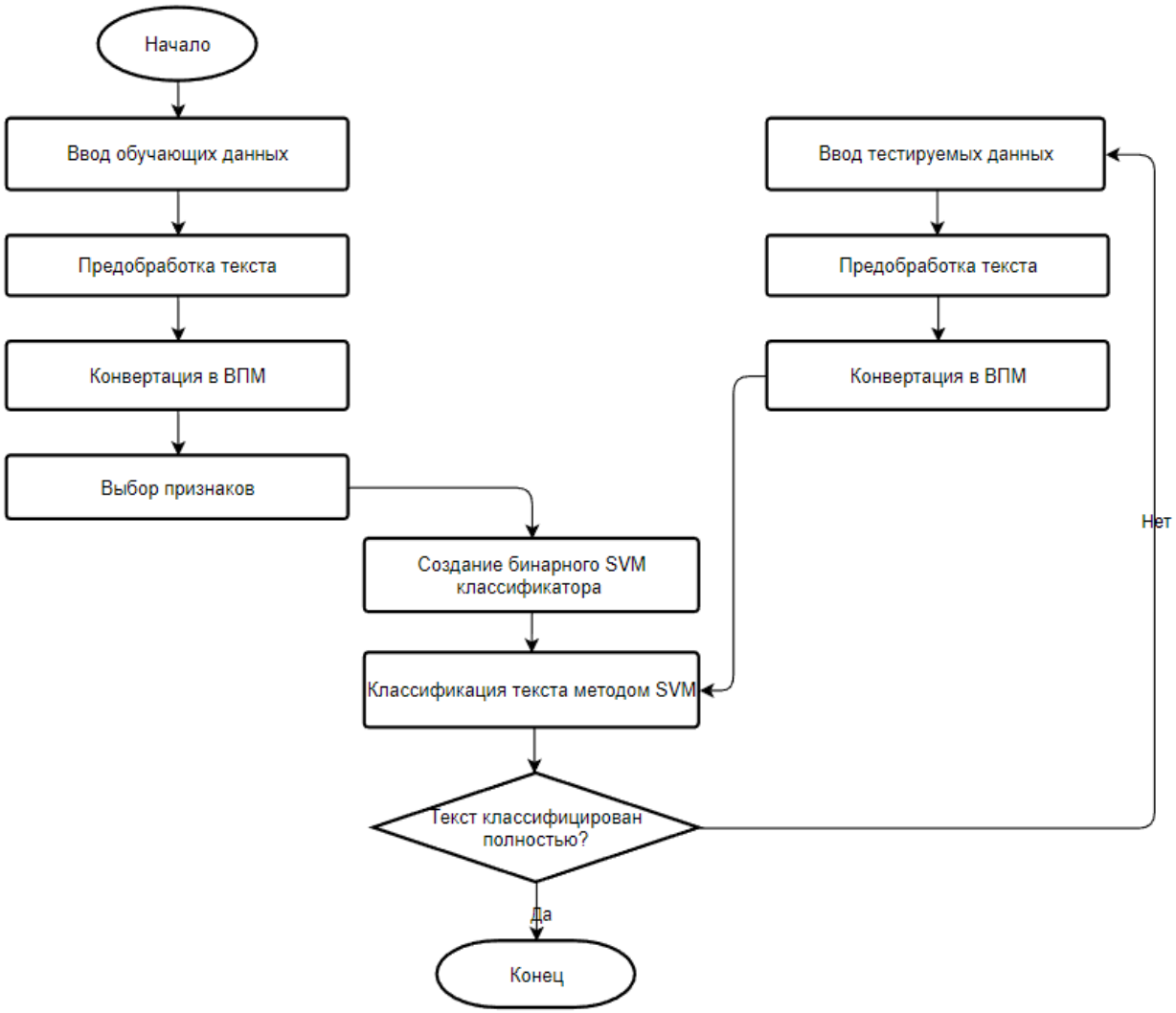


Рисунок 14 – Блок-схема алгоритма бинарной классификации текста на основе SVM

Процесс автоматической классификации включает в себя предварительную обработку текста, выбор функций, расчет веса, представление текста и обучение классификатора текста.

Далее текст преобразуется в векторную пространственную модель (ВПМ).

Векторное пространство содержит набор объектов, называемых векторами, которые являются числовыми представлениями слов, предложений и даже документов. В то время как простой вектор, такой как координаты карты, имеет только два измерения, те, которые используются в обработке естественного языка, могут иметь тысячи.

ВПМ (Vector Space Model, VSM) – это алгебраическая модель, которая представляет объекты (например, текст) в виде векторов. Это позволяет легко определить сходство между словами или релевантность между поисковым запросом и документом. Косинусное сходство часто используется для определения сходства между векторами [36].

ВПМ использует линейную алгебру с небинарными весами терминов. Это означает, что непрерывная степень сходства между двумя объектами, такими как запрос и документы, может быть рассчитана с учетом частичного совпадения.

Рассмотрим наиболее востребованные методы преобразования текстовых данных в числовую форму.

«Метод TF-IDF – это «частотность терминов-обратная частотность документов». В этом методе получаем математическую матрицу, описывающую частоту встречающихся терминов. В матрице столбцы соответствуют запросам, а строки терминам.

Показатель TF рассчитывается, как отношение количества вхождений слова из запроса к количеству все слов в запросе по формуле (12):

$$TF(t) = \frac{n_t}{\sum_k n_k}, \quad (12)$$

где n_k – число вхождений слова t в запрос.

Показатель IDF равен логарифму отношения количества запросов к количеству запросов, в которых встречается заданное слово и вычисляется по формуле (14):

$$IDF(t, D) = log \frac{|D|}{|d_i \in D, t \in d_i|}, \quad (13)$$

где в числителе – число запросов, а в знаменателе – число запросов, в которых встречается слово t» [10].

Общий показатель TF-IDF (14) является произведением формул (12) и (13):

$$\text{TF-IDF} = \text{TF}(t) \cdot \text{IDF}(t, D) \quad (15)$$

Метод прямого кодирования оставляется словарь, и каждое слово представляется в виде вектора, где одна координата равна 1, а остальные нулям.

«Одним из самых доступных, но при этом эффективных и распространенных методов обработки текста для представления в виде входного сигнала системы машинного обучения – это так называемый «мешок слов» (bag-of-words).

Используя это представление, можно удалить структуру исходного документа (разделы, подразделы, абзацы, предложения, настройки форматирования и т.д.) и посчитать частоту встречаемости каждого слова в каждой точке данных корпуса.

Создание «мешка слов» включает в себя следующие три этапа:

− токенизация (англ. tokenization). Каждый документ разбивается на слова, которые встречаются в нем (токены), например, с помощью

пробелов и знаков пунктуации;

- построение словаря (англ. vocabulary building). Выполняется занесение в словарь всех слов, которые появляются в каждом из документов, и их упорядочение (как правило, в алфавитном порядке).
- создание разреженной матрицы (англ. sparse matrix encoding). Для каждого точки данных корпуса выполняется подсчет, насколько часто каждое из слов, занесенных в словарь, встречается в данном документе» [10].

Выбор метода преобразования текста в числовой формат зависит от вида текста и решаемой задачи.

3.2 Моделирование системы управления поиском информации в торговой организации

В качестве методологии моделирования СУПИ используем объектно-ориентированный подход (ООП) к анализу и проектированию информационных систем.

Можно определить следующие преимущества объектно-ориентированного подхода [25]:

- расширяемость – это простота адаптации метода и реализации к более широкому кругу приложений или совершенно новым приложениям, которые включают расширение как словаря, так и правил. Поскольку знания постоянно развиваются, этот момент представляет особый интерес;
- возможность повторного использования – это возможность определить синтаксис для нескольких приложений и повторно использовать словарь и правила;
- совместимость – это возможность определять разные наборы словарей и правил, например, в разных областях (дизайн,

производство), которые могут быть взаимозаменяемы для разных приложений или поколений продуктов;

- эффективность означает, что формализация и инкапсуляция знаний поддерживаются таким образом, чтобы потребность в ресурсах была минимально возможной. Это требует вычислительных ресурсов, но более важно эффективно поддерживать трудоемкую человеческую задачу кодирования знаний;
- простота использования означает, что структура грамматики может быть понята так же интуитивно, как возможно и обеспечивает основу для фактора расширяемости.

Расширение методов синтеза дизайна таким образом, чтобы они включали концепции объектно-ориентированного программирования, дает широкий спектр преимуществ. Это многообещающий подход к реализации более эффективных, гибких и расширяемых систем.

Концепции объектно-ориентированного проектирования:

- ООП создает программное обеспечение в модулях (объектах). Каждый объект является независимым объектом системы, в которой он находится;
- у объекта есть методы, которые он может выполнять (методы), и инкапсулированные данные, к которым он может обращаться, поддерживать и использовать;
- объектно-ориентированный дизайн помогает создавать автономные структуры (объекты), представляющие объекты реального мира, способные взаимодействовать с другими объектами;
- методы ООП ведут к экономичной разработке программного обеспечения;
- концепция класса является важной особенностью ООП. Класс — это набор объектов с общей структурой и поведением;
- концепция наследования имеет особое значение в ООП.

Наследование используется для описания класса и его подклассов.

Для моделирования используется язык UML.

UML представляет собой набор элементов, разработанных для помощи аналитикам в раскрытии важных особенностей проектного проекта и, наконец, в получении набора моделей, которые будут использоваться для проектирования, документирования и реализации проекта (информационной системы) или программного обеспечения [7].

В качестве CASE-средства моделирования использован онлайн-сервис Visual Paradigm [37].

Для отражения статического аспекта бинарного классификатора текста разработана его диаграмма классов.

В программировании статическая часть фиксирует предположения, которые разработчик программного обеспечения сделал для решения данной задачи, и приводит к реализации классов и функций.

Аналогом этого подхода к синтезу является определение словаря (в метамодели) и правил (в наборе правил), т. е. определение формального синтаксиса, содержащего формализованные инженерные знания.

И словарный запас, и правила остаются неизменными во всем приложении и определяют объем исследования генеративного дизайна.

Диаграмма классов используется в моделировании как статическое описание элементов, которые могут быть созданы при выполнении программы.

В подходах к формальному моделированию определение пространства моделирования реализуется с помощью метамодели. Общепринятое понимание заключается в том, что метамодель – это модель, которая определяет модель, т. е. элементы моделирования, допустимые их комбинации и т. д.

Следовательно, во избежание путаницы, термины модель класса и метамодель эквивалентны и должны использоваться как синоним при

использовании объектно-ориентированных методов в синтезе компьютерного дизайна.

Диаграмма классов бинарного классификатора текста показана на рисунке 15.

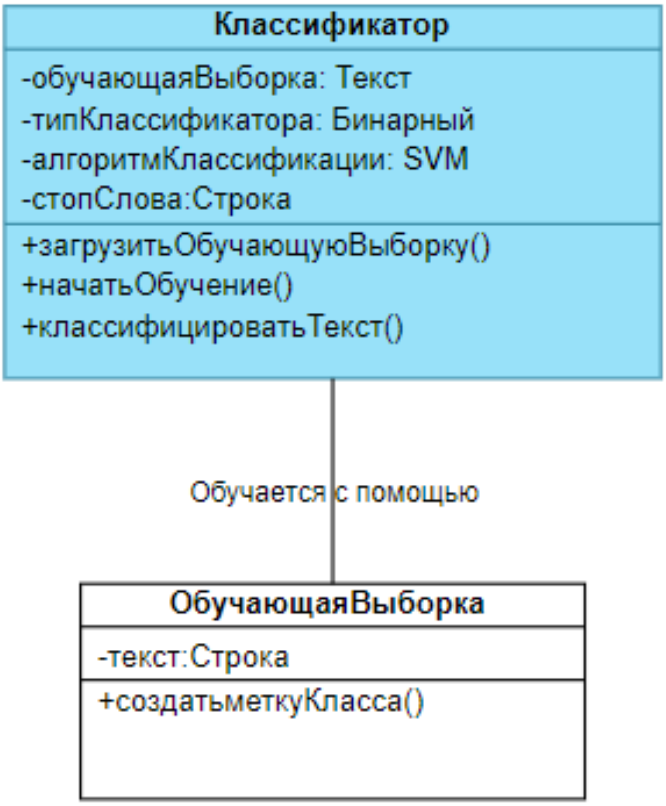


Рисунок 15 – Диаграмма классов бинарного классификатора текста

Спецификация диаграммы классов бинарного классификатора текста представлена в таблице 3.

Таблица 3 – Спецификация диаграммы классов бинарного классификатора текста

«Класс	Описание
Классификатор	Класс объектов, моделирующих на логическом уровне лиц классификаторы текста

Продолжение таблицы 3

Класс	Описание
Обучающая выборка	Класс объектов, моделирующих на логическом набор обучающих данных» [25]

С учетом вышеизложенного разработана диаграмма вариантов использования, которая представляет функциональную модель СУПИ.

Диаграмма вариантов использования – это описание того, как пользователи взаимодействуют с системой.

Чтобы разработать диаграмму вариантов использования, аналитик должен решить определить:

- границы системы и интерфейсы;
- акторов – действующих лиц, использующих систему;
- варианты использования – функции, вызываемые действующим лицом в системе для выполнения.

В результате анализа были выделены следующие акторы СУПИ: Аналитик продаж, Подсистема поиска, сбора и обработки информации (ППСОИ), Классификатор текста, Подсистема интеллектуального анализа текста (ПИАТ).

Варианты использования СУПИ описаны в таблицах 4-7.

Таблица 4 – Формирование поискового запроса

«Прецедент: Формирование поискового запроса
ID: 1
Краткое описание: Аналитик продаж формирует поисковый запрос
Главный актор: Аналитик продаж
Второстепенный актор: нет
Предусловие: нет
Основной поток: – Аналитик продаж авторизуется в СУПИ; – Аналитик продаж вводит поисковый запрос и запускает поиск информации
Постусловие: нет
Альтернативные потоки: нет» [36]

Таблица 5 – Поиск отзывов по теме запроса

«Прецедент: Поиск отзывов по теме запроса
ID: 2
Краткое описание: ППСОИ осуществляет поиск отзывов по теме запроса Аналитика продаж
Главный актор: ППСОИ
Второстепенный актор: Аналитик продаж
Предусловие: нет
Основной поток: – ППСОИ принимает запрос Аналитика продаж; – ППСОИ осуществляет поиск отзывов по теме запроса.
Постусловие: нет
Альтернативные потоки: нет» [36]

Таблица 6 – Бинарная классификация текста

«Прецедент: Бинарная классификация текста
ID: 3
Краткое описание: Классификатор текста производит бинарную классификацию текста
Главный актор: Классификатор текста
Второстепенный актор: ППСОИ
Предусловие: нет
Основной поток: – ППСОИ производит предварительную обработку и преобразование полученного текста в числовую форму; – Классификатор текста производит бинарную классификацию текста с помощью алгоритма SVM
Постусловие: нет
Альтернативные потоки: нет» [36]

Таблица 7 – Анализ тональности отзывов клиентов

«Прецедент:
ID: 4
Краткое описание: Аналитик продаж проводит анализ тональности отзывов клиентов
Главный актор: Аналитик продаж
Второстепенный актор: ПИАТ
Предусловия: Бинарная классификация текста
Основной поток: – Аналитик продаж запускает процедуру анализа тональности текста; – ПИАТ проводит анализ тональности текста с помощью выбранной методики
Постусловие: нет
Альтернативные потоки: нет» [36]

Диаграмма вариантов использования СУПИ изображена на рисунке 16.

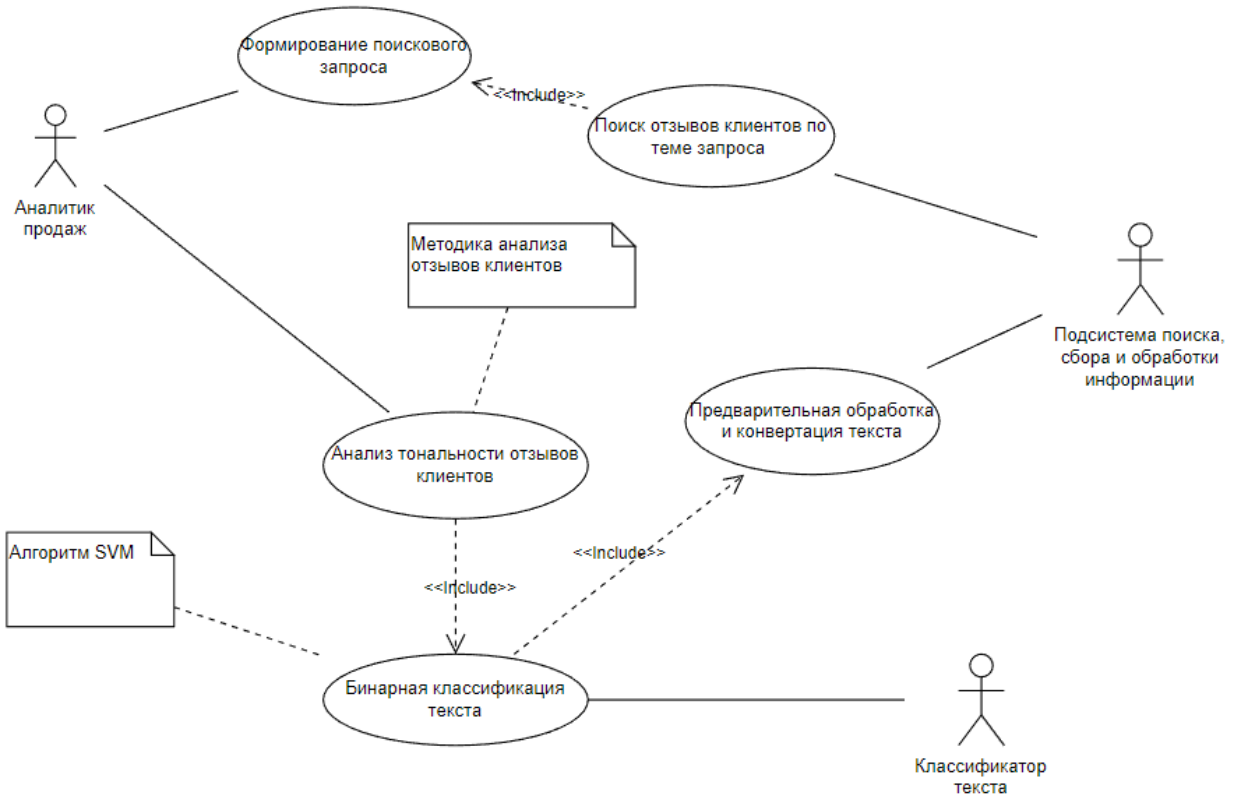


Рисунок 16 – Диаграмма вариантов использования СУПИ

Для представления программной архитектуры СУПИ использована диаграмма компонентов.

Диаграмма компонентов показывает компоненты, предоставляемые и требуемые интерфейсы, порты и отношения между ними. Этот тип диаграмм используется в компонентно-ориентированной разработке (CBD) для описания систем с сервис-ориентированной архитектурой (SOA).

Разработка на основе компонентов основана на предположениях, что ранее созданные компоненты могут быть повторно использованы и что компоненты могут быть заменены некоторыми другими «эквивалентными» или «совместимыми» компонентами, если это необходимо.

Артефакты, реализующие компонент, предназначены для независимого развертывания и повторного развертывания, например, для обновления

существующей системы.

Компоненты в UML могут представлять логические компоненты (например, бизнес-компоненты, компоненты процесса) и физические компоненты (например, компоненты CORBA, компоненты EJB, компоненты COM+ и .NET, компоненты WSDL и т. д.) [35].

Компоненты изображаются вместе с артефактами, которые их реализуют, и узлами, на которых они развертываются и выполняются.

Ожидается, что профили, основанные на компонентах, будут разработаны для конкретных компонентных технологий и соответствующих аппаратных и программных сред.

Программная архитектура СУПИ показана на рисунке 17.

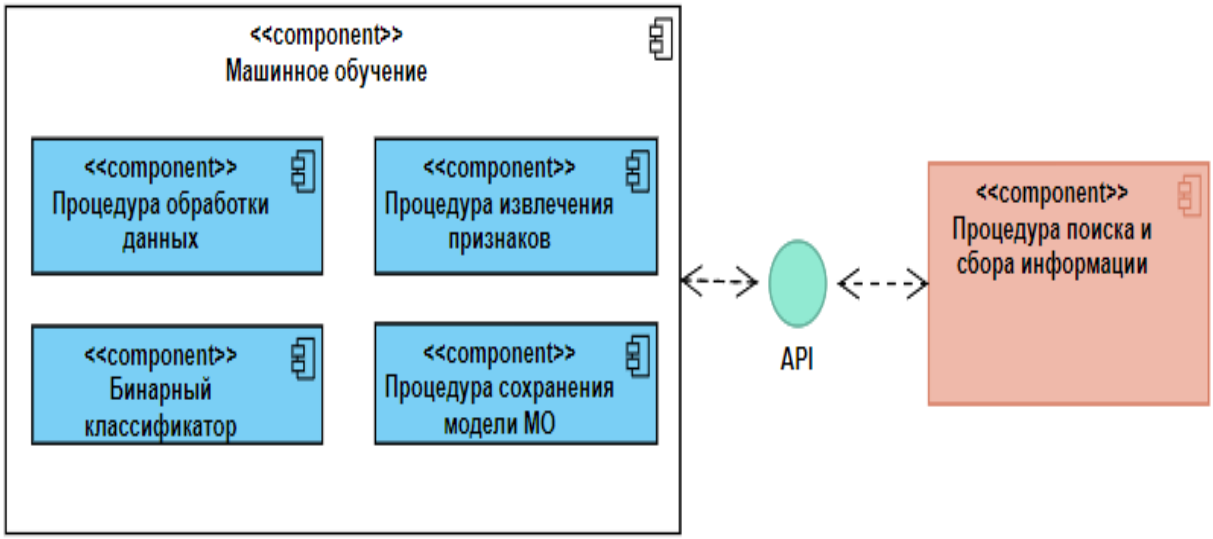


Рисунок 17 – Программная архитектура СУПИ

Как следует из диаграммы, программная архитектура СУПИ разделена на два основных компонента: машинное обучение процедуры поиска и сбора информации, взаимодействующие через API [31].

Выводы по главе 3

Результаты проделанной работы позволили сделать следующие выводы:

- процесс автоматической классификации включает в себя предварительную обработку текста, выбор функций, расчет веса, представление текста в числовой форме и обучение классификатора текста;
- выбор метода преобразования текста в числовой формат зависит от вида текста и решаемой задачи;
- в качестве методологии моделирования СУПИ выбран ООП к анализу и проектированию информационных систем;
- разработана диаграмма вариантов использования, которая представляет функциональную модель СУПИ.

Для представления программной архитектуры СУПИ использована диаграмма компонентов.

Глава 4 Апробация проектных решений и оценка их эффективности

4.1 Апробация проектных решений

Для реализации алгоритма машинного обучения использованы язык Python и среда Jupyter Notebook (anaconda 3) с библиотекой sklearn [32].

Код бинарного классификатора текста по методу SVM представлен в листинге 1 [21].

```
# Imports
«from sklearn.datasets import make_blobs
from sklearn.model_selection import train_test_split
import numpy as np
import matplotlib.pyplot as plt
from sklearn import svm
from sklearn.metrics import plot_confusion_matrix
# Configuration options
blobs_random_seed = 42
centers = [(0,0), (5,5)]
cluster_std = 1
frac_test_split = 0.33
num_features_for_samples = 2
num_samples_total = 1000
# Generate data
inputs, targets = make_blobs(n_samples = num_samples_total, centers = centers,
n_features = num_features_for_samples, cluster_std = cluster_std)
X_train, X_test, y_train, y_test = train_test_split(inputs, targets,
test_size=frac_test_split, random_state=blobs_random_seed)
```

```
# Save and load temporarily
# np.save('./data.npy', (X_train, X_test, y_train, y_test))
X_train, X_test, y_train, y_test = np.load('./data.npy', allow_pickle=True)
# Generate scatter plot for training data
plt.scatter(X_train[:,0], X_train[:,1])
plt.title('Linearly separable data')
plt.xlabel('X1')
plt.ylabel('X2')
plt.show()
# Initialize SVM classifier
clf = svm.SVC(kernel='linear')
# Fit data
clf = clf.fit(X_train, y_train)
# Predict the test set
predictions = clf.predict(X_test)
# Generate confusion matrix
matrix = plot_confusion_matrix(clf, X_test, y_test,
                               cmap=plt.cm.Blues,
                               normalize='true')
plt.title('Confusion matrix for our classifier')
plt.show(matrix)
plt.show()
# Get support vectors
support_vectors = clf.support_vectors_
# Visualize support vectors
plt.scatter(X_train[:,0], X_train[:,1])
plt.scatter(support_vectors[:,0], support_vectors[:,1], color='red')
plt.title('Linearly separable data with support vectors')
plt.xlabel('X1')» [21]
```

```
plt.ylabel('X2')
plt.show()
```

Рассмотрим практический пример анализа отзывов клиентов с помощью предлагаемого решения.

«Потребительские мнения, которые пользователи компаний оставляют в виде отзывов на страницах веб-сайтов, являются, как правило, неструктурированным текстом, получить общую тональность которого может быть сложной задачей для компьютерной программы.

Ввиду существенных достижений в области обработки больших объемов неструктурированных данных в последние годы, обработка естественного языка стала играть ключевую роль в принятии обоснованных решений о маркетинговых стратегиях – анализ тональности текстов способен предоставить полезную обратную связь о продуктах и услугах.

Для решения задачи был рассмотрен текстовый набор из 5000 отзывов на мобильные телефоны.

Было принято решение о создании системы классификации, работающей автономно от модуля создания, настройки и тестирования модели машинного обучения – предварительно обученная модель может быть сохранена в файл в виде настроек и затем загружена для решения задачи классификации в режиме реального времени.

Поскольку метод SVM является лучшим выбором для небольших объемов данных, был сделан вывод о том, что для использования в качестве модели машинного обучения лучше всего подходит линейный классификатор опорных векторов – метод опорных векторов, который использует линейное ядро для решения задачи классификации.

Текстовые данные (документы в наборе отзывов на мобильные телефоны) представлены в строковом формате, который не воспринимается моделью машинного обучения» [2].

Поэтому принято решение выполнить конвертацию строкового представления в числовую с помощью «мешка слов».

Для построения «мешка слов» в работе была использована Scikit-Learn – это библиотека для Python.

Несмотря на то, что в библиотеке scikit-learn не реализован ни один из способов нормализации, CountVectorizer позволяет задать собственный токенизатор, который преобразует каждый документ в список токенов с помощью параметра tokenizer.

«Для избавления от неинформативных слов в работе был использован список стоп-слов русского языка из пакета nltk – ведущей платформы для создания программ на Python для работы с данными на человеческом языке.

Следующий метод вместо исключения несущественных признаков пытается масштабировать признаки в зависимости от степени их информативности.

Одним из наиболее распространенных способов такого масштабирования является метод TF-IDF.

Идея этого метода заключается в том, чтобы присвоить большой вес термину, который часто встречается в конкретном документе, но при этом редко встречается в остальных документах набора.

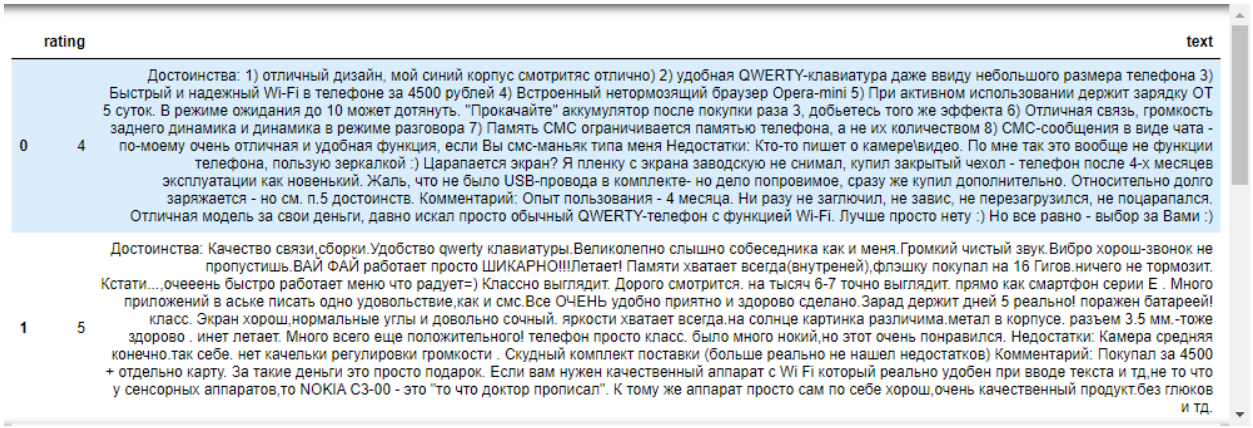
В библиотеке scikit-learn метод TF-IDF реализован в двух классах: TfidfTransformer, который принимает на вход разреженную матрицу, полученную с помощью CountVectorizer, и преобразует ее, и TfidfVectorizer, который принимает на вход текстовые данные и выполняет как выделение признаков «мешок слов», так и преобразование TF-IDF.

Набор отзывов на мобильные телефоны представляет собой набор текстов, сохраненных в формате *. json.

Данные считываются в табличную структуру данных DataFrame библиотеки Pandas для анализа данных» [2].

Этот объект представляет собой аналог структуры данных «словарь», в

котором ключами выступают заголовки столбцов данных, а значениями – непосредственно столбцы. Пример данных, загруженных в таблицу, приведен на рисунке 18.



	rating	text
0	4	Достоинства: 1) отличный дизайн, мой синий корпус смотритяс отлично) 2) удобная QWERTY-клавиатура даже ввиду небольшого размера телефона 3) Быстрый и надежный Wi-Fi в телефоне за 4500 рублей 4) Встроенный неторомозящий браузер Opera-mini 5) При активном использовании держит зарядку ОТ 5 суток. В режиме ожидания до 10 может дотянуть. "Прокачайте" аккумулятор после покупки раза 3, добьетесь того же эффекта 6) Отличная связь, громкость заднего динамика и динамика в режиме разговора 7) Память СМС ограничивается памятью телефона, а не их количеством 8) СМС-сообщения в виде чата - по-моему очень отличная и удобная функция, если Вы смс-маньяк типа меня Недостатки: Кто-то пишет о камере\видео. По мне так это вообще не функции телефона, пользую зеркалкой :) Царапается экран? Я пленку с экрана заводскую не снимал, купил закрытый чехол - телефон после 4-х месяцев эксплуатации как новенький. Жаль, что не было USB-провода в комплекте- но дело попровимое, сразу же купил дополнительно. Относительно долго заряжается - но см. п.5 достоинств. Комментарий: Опыт пользования - 4 месяца. Ни разу не заглючил, не завис, не перезагрузился, не поцарапался. Отличная модель за свои деньги, давно искал просто обычный QWERTY-телефон с функцией Wi-Fi. Лучше просто нету :) Но все равно - выбор за Вами :)
1	5	Достоинства: Качество связи,сборки.Удобство qwerty клавиатуры.Великолепно слышно собеседника как и меня.Громкий чистый звук.Вибро хорош-звонок не пропустишь.ВАЙ ФАЙ работает просто ШИКАРНО!!!Летает! Памяти хватает всегда(внутреней),флэшку покупал на 16 Гигов.ничего не тормозит. Кстати...,очееень быстро работает меню что радует=) Классно выглядит. Дорого смотрится. на тысяч 6-7 точно выглядит. прямо как смартфон серии Е . Много приложений в аське писать одно удовольствие,как и смс.Все ОЧЕНЬ удобно приятно и здорово сделано.Зарад держит дней 5 реально! поражен батареей! класс. Экран хорош,нормальные углы и довольно сочный. яркости хватает всегда.на солнце картинка различима.метал в корпусе. разъем 3.5 мм.-тоже здорово . инет летает. Много всего еще положительного! телефон просто класс. было много нокий,но этот очень понравился. Недостатки: Камера средняя конечно.так себе. нет качельки регулировки громкости . Скудный комплект поставки (больше реально не нашел недостатков) Комментарий: Покупал за 4500 + отдельно карту. За такие деньги это просто подарок. Если вам нужен качественный аппарат с Wi Fi который реально удобен при вводе текста и тд,не то что у сенсорных аппаратов,то NOKIA C3-00 - это "то что доктор прописал". К тому же аппарат просто сам по себе хорош,очень качественный продукт.без глюков и тд.

Рисунок 18 – Пример точек данных исходного набора отзывов

«Оценки варьируются от 1 до 5, при этом каждый отзыв имеет метку класса – 0 (негативный) или 1 (позитивный), которую необходимо получить для отзыва, заданного пользователем.

Так как целевая переменная не сбалансирована, были выбраны только 1680 самых развернутых положительных отзывов в обучающую выборку.

Каждый отзыв состоит из трех частей – первая содержит описанные достоинства, вторая – недостатки, третья – комментарий.

В качестве данных для обучающей выборки из отзыва выделяется содержимое этих составляющих отзыва при помощи регулярных выражений (для этого реализована функция extract_info).

Предварительно отзывам назначаются соответствующие их оценкам метки – «1» для отзывов с рейтингом «5» и «0» для остальных отзывов.

В листинге 2 описывается функция создания конвейерной модели, принцип действия которой состоит в последовательном применении трех алгоритмов: векторизации, TF-IDF преобразования и классификации. В качестве классификатора в работе используется линейный классификатор

метода опорных векторов» [2].

Листинг 2 – Функция построения конвейерной модели

```
«def make_pipeline(vectorizer, transformer, classifier):
    return Pipeline([
        ('vectorizer', vectorizer),
        ('transformer', transformer),
        ('classifier', classifier)
    ])
```

Перед обучением модели на обучающей выборке был выполнен подбор оптимальных для задачи параметров при помощи метода RandomizedSearchCV.

Исходный код функции поиска параметров приведен в листинге 3

Листинг 3 – Функция поиска лучших параметров

```
from sklearn.model_selection import RandomizedSearchCV
def make_estimator(classifier, params_grid, scorer, data, labels):
    pipeline = make_pipeline(CountVectorizer(), TfidfTransformer(), classifier)
    grid_cv = RandomizedSearchCV(pipeline, params_grid, scoring=scorer, cv=5,
                                 random_state=777, n_iter=100, verbose=1, n_jobs=-1)
    grid_cv.fit(data, labels)
    return grid_cv
```

Для настройки параметров были использованы несколько константных значений, которые применялись к каждой из составных частей конвейерной модели» [2].

Например, для векторизации выбираются следующие параметры:

- минимальное значение количества документов, в которых должно появиться слово (1, 10, 20);
- доля от общего числа документов, в которых будут исключены часто встречающиеся слова (от 0,85 до 1.00);
- диапазон токенов, которые рассматриваются в качестве признаков (одиночные символы, а также идущие друг за другом пары, тройки и т.д. – юниграммы, биграммы, триграммы и т.д.);
- признак использования списка стоп-слов в выбранном языке (использовать этот список или нет).

Соответствующие параметры были выбраны также для объекта класса TfidfTransformer, выполняющего TF-IDF преобразование, и для линейного классификатора метода опорных векторов LinearSVC. В качестве метрики используется accuracy (точность классификации). Также используется кросс-валидация с параметром k = 5.

Результат подбора лучших параметров модели представлен на рисунке 19. Как следует из рисунка, точность классификации на тестовой выборке достигла значения 92%.

```
Fitting 5 folds for each of 100 candidates, totalling 500 fits
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers.
[Parallel(n_jobs=-1)]: Done  42 tasks      | elapsed:   32.2s
[Parallel(n_jobs=-1)]: Done 192 tasks      | elapsed:  2.3min
[Parallel(n_jobs=-1)]: Done 442 tasks      | elapsed:  4.6min
[Parallel(n_jobs=-1)]: Done 500 out of 500 | elapsed:  5.3min finished
LinearSVC:
Лучшее качество - 0.9279761904761905
Параметры - {'vectorizer__stop_words': None, 'vectorizer__ngram_range': (1, 2), 'vectorizer__min_df': 10, 'vectorizer__max_df':
0.85, 'transformer__use_idf': True, 'transformer__sublinear_tf': True, 'transformer__smooth_idf': True, 'transformer__norm': 'l
2', 'classifier__tol': 0.0001, 'classifier__max_iter': 200, 'classifier__loss': 'hinge', 'classifier__C': 0.9}
Wall time: 5min 17s
```

Рисунок 19 – Оптимальные параметры модели

Модель классификации с полученными параметрами также была проверена на обучающей выборке отзывов.

На рисунке 20 показано распределение длин текстов отзывов пользователей.

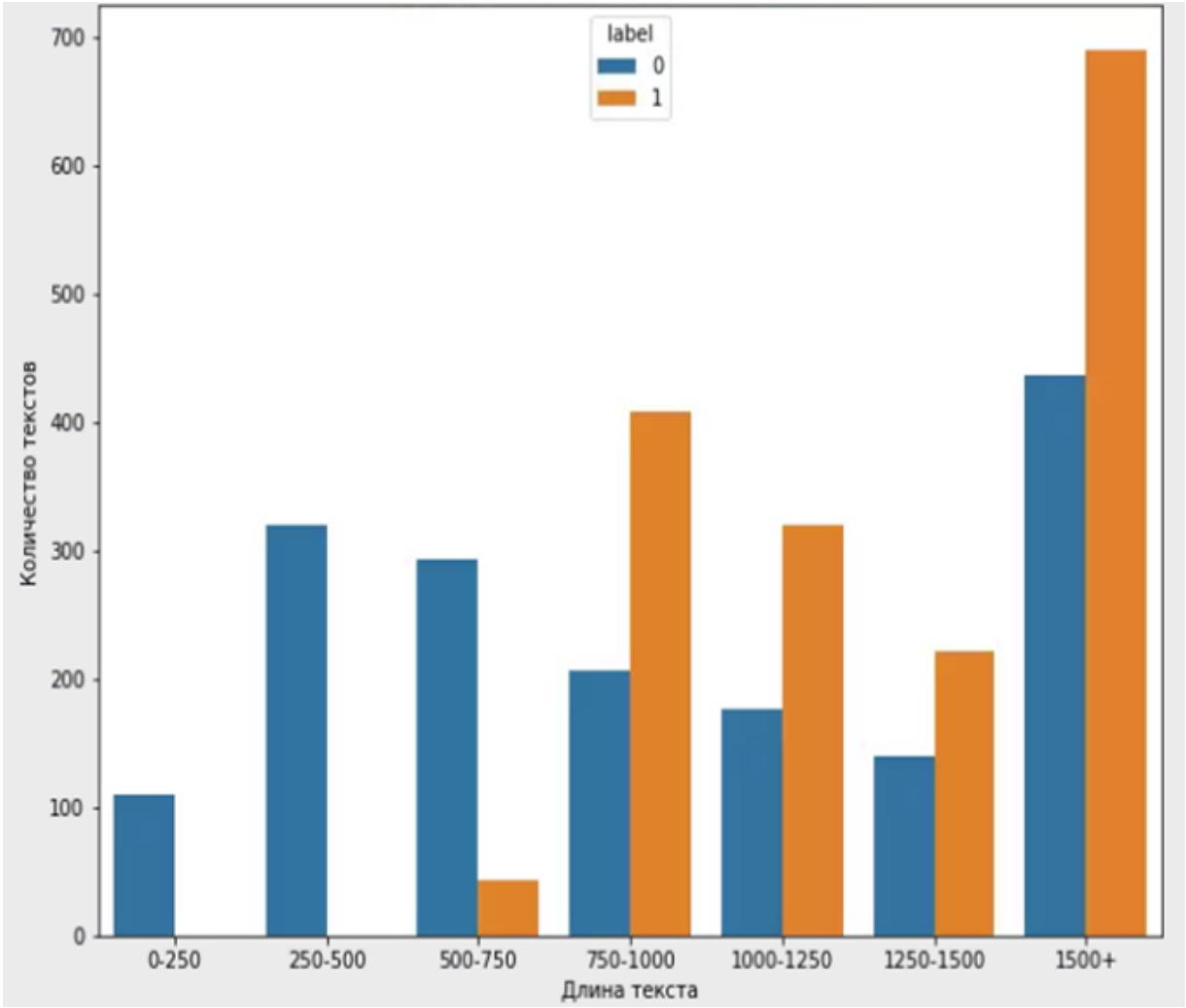


Рисунок 20 – Распределение длин текстов отзывов

На рисунке 21 показано распределение рейтинга мобильных телефонов в зависимости от отзывов пользователей.

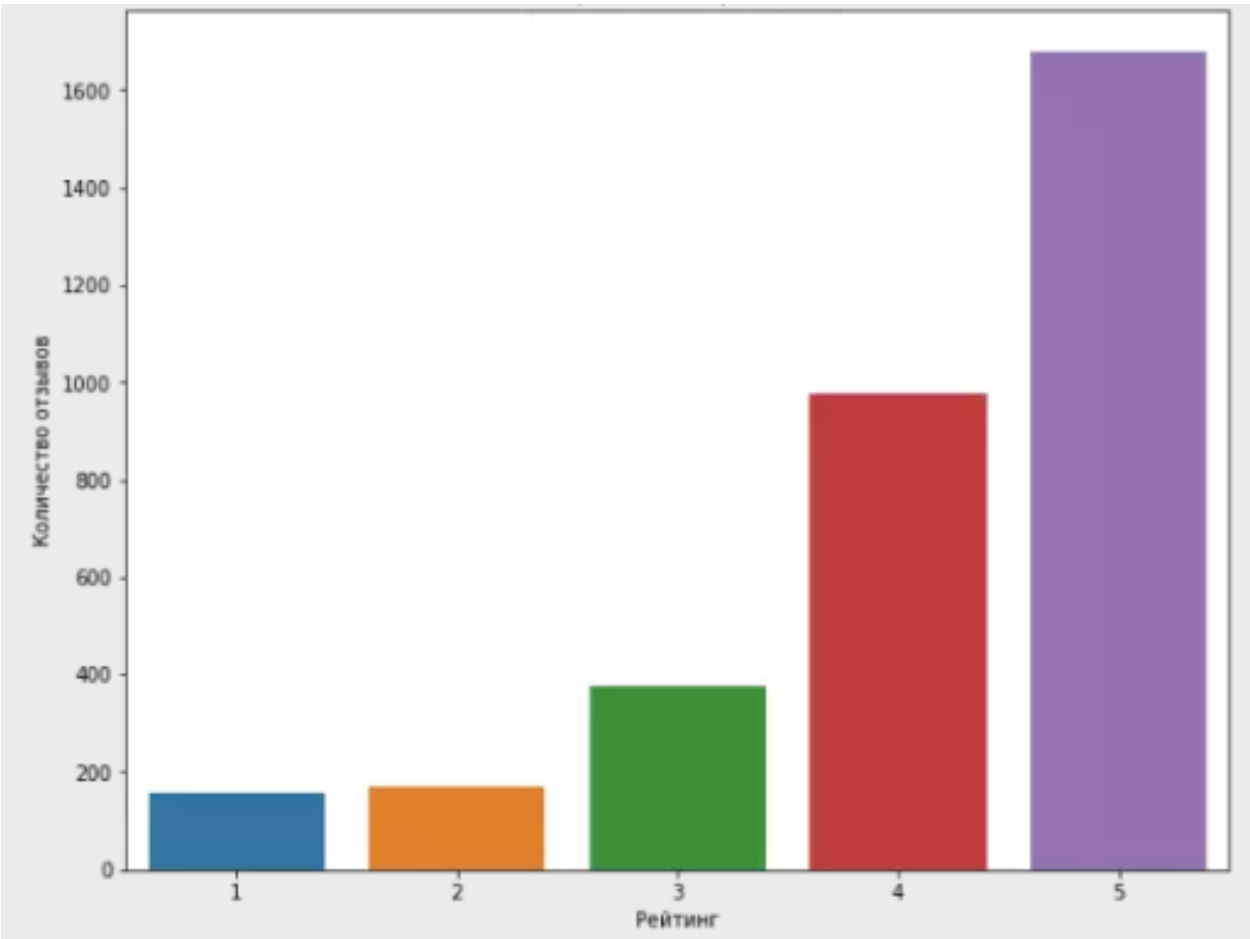


Рисунок 21 – Распределение рейтинга мобильных телефонов в зависимости от отзывов пользователей

Так как проверка тестовая, вместо конкретных марок мобильных телефон использована их условная нумерация.

Таким образом, апробация проектного решения подтвердила возможность его использования для анализа тональности отзывов клиентов о товарах.

4.2 Оценка экономической эффективности проектных решений

Для оценки экономической эффективности проекта используем методику сравнения затрат на разработку СУПИ внешним программистом по договору аутсорсинга (базовый вариант) и программистом торговой

организации (проектный вариант), соответственно [3].

«В калькуляцию себестоимости заказной разработки СППР включаются следующие статьи затрат:

- зарплата исполнителя проекта по трудовому договору ($ЗБ_1$);
- социальные страховые взносы ($ЗБ_2$);
- прочие прямые расходы ($ЗБ_3$);
- накладные расходы ($ЗБ_4$).

В заказной доработке задействован внешний программист» [3].

Средняя стоимость часа работы программиста Python по договору составляет 1200 руб [13].

Ориентировочное время разработки составляет 100 час.

Итого затраты базового варианта $С_{баз}$ составят (5):

$$С_{баз} = ЗБ_1 + ЗБ_2 + ЗБ_3 + ЗБ_4 = 1200*100 +0,271*1200*100+0+0 =152520 \text{ руб.} \quad (5)$$

В самостоятельной разработке СУПИ задействован программист торговой организации.

«В калькуляцию себестоимости собственной разработки ИС включаются следующие статьи затрат:

- зарплата исполнителей проекта с учетом затраченного времени 100 час ($ЗП_1$);
- социальные страховые взносы ($ЗП_2$);
- прочие прямые расходы ($ЗП_3$);
- накладные расходы ($ЗП_4$)» [3].

Итого затраты проектного варианта $С_{пр}$ составят (6):

$$С_{пр} = ЗП_1 + ЗП_2 + ЗП_3 + ЗП_4 = 80000 \text{ руб}+0,3*80000+0+0 = 104000 \text{ руб} \quad (6)$$

Сформируем таблицу и график показателей экономической эффективности (таблица 8, рисунок 22).

Таблица 8 – Показатели эффективности проекта разработки СУПИ

«Затраты		Абсолютное изменение затрат	Коэффициент относительного снижения затрат	Индекс снижения затрат
Базовый вариант	Проектный вариант			
$С_{баз}$ (руб.)	$С_{пр}$ (руб.)	$\Delta С = С_{баз} - С_{пр}$ (руб.)	$К_С = \Delta С / С_{баз} \times 100\%$	$У_С = С_{баз} / С_{пр}$
152520	104000	48250	31,6	1,5» [3]

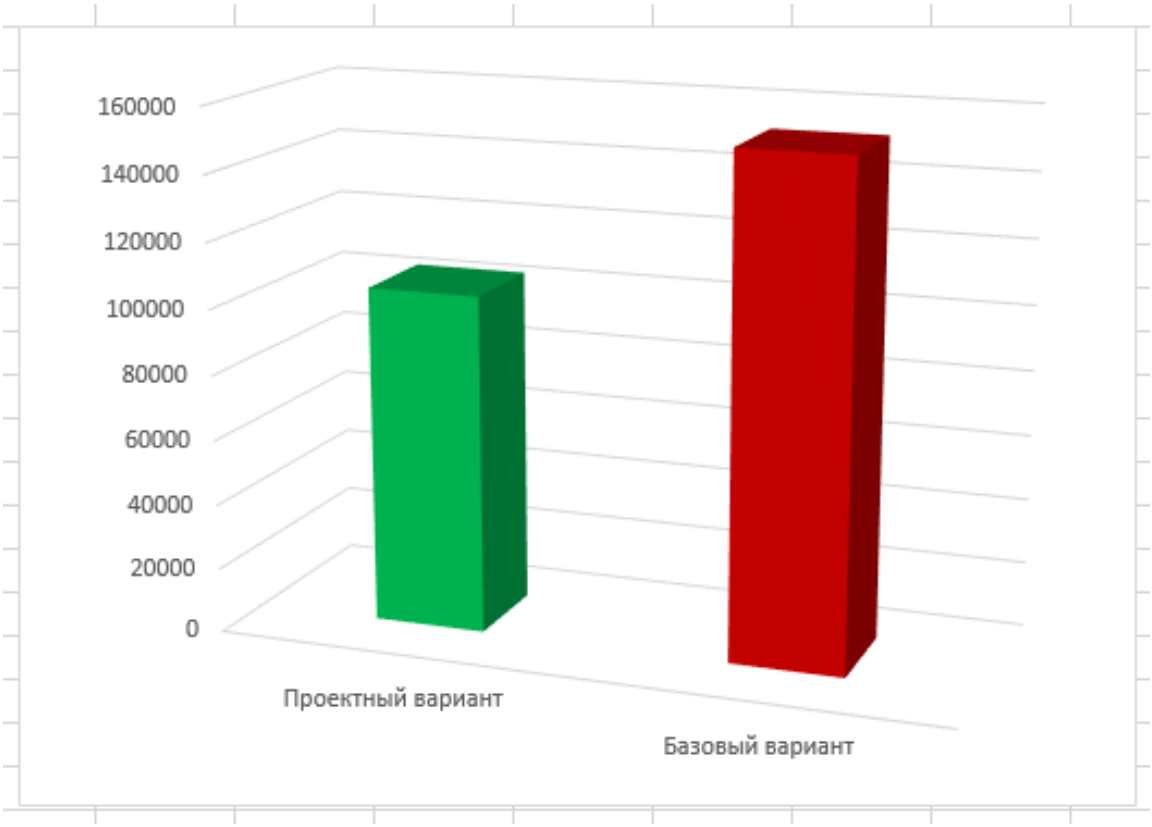


Рисунок 22 – Гистограмма сравнения затрат на разработку СУПИ

Таким образом, затраты при проектном варианте разработки СУПИ сократились в 1,5 раза.

«Срок окупаемости затрат на внедрение проектного решения ($Т_{ок}$) определяется по формуле (7):

$$Т_{ок} = К_{П} / \Delta С \text{ (мес.)}, \quad (16)$$

где $К_{П}$ – затраты на реализацию проектных решений (проектирование и внедрение СППР).

Следовательно, срок окупаемости адаптированного сайта равен (8):

$$Т_{ок} = 104000/48250 \approx 2{,}2 \text{ мес.} \quad (17)$$

Представленные расчеты подтвердили существенное снижение затрат на проектирование и эффективность проектного решения» [3].

Для оценки эффективности управления СУПИ торговой организации используем формулу [3] (18):

$$К_{эу} = \frac{\sum_{i=1}^{n} Р_{уi}}{n} \quad (18)$$

где:

n - количество функций управления, реализуемых СППР;

$Р_{уi}$ - вероятность выработки СУПИ торговой организации эффективного управляющего воздействия при реализации *i*-й функции управления.

Для управления поиском информации используются следующие функции:

- поиск и сбор отзывов клиентов;
- анализ тональности отзывов клиентов.

Как показывает практика, на выполнение функции «Поиск и сбор отзывов клиентов» может негативно повлиять человеческий фактор, например, при составлении поискового запроса.

«Пусть вероятность выработки эффективного управляющего воздействия для каждой функции равна 0.5.

В этом случае значение показателя функциональной эффективности управления СУПИ будет равно» [3]:

$$К_{эу} = 1,5/2 = 0,75$$

Таким образом, коэффициент эффективности управления предлагаемой СУПИ Кэу> 0,5, что свидетельствует о высокой функциональной эффективности управления поиском информации в торговой организации.

Выводы по главе 4

Результаты проделанной работы позволили сделать следующие выводы:

- для реализации алгоритмов машинного обучения использованы язык Python и среда Jupyter Notebook (anaconda 3) с библиотекой sklearn;
- апробация проектного решения подтвердила возможность его использования для анализа тональности отзывов клиентов о товарах;
- затраты при проектном варианте разработки СУПИ сократились в 1,5 раза.

Коэффициент эффективности управления предлагаемой СУПИ Кэу> 0,5, что свидетельствует о высокой функциональной эффективности управления поиском информации в торговой организации.

Заключение

В торговой организации система управления поиском информации (СУПИ) позволит значительно повысить релевантность результатов поиска для клиентов организации.

Кроме того, СУПИ повышает эффективность внутренних бизнес-процессов и высвобождает время на реализацию более приоритетных задач сотрудниками организации. Совершенно очевидно, что в основу СУПИ торговой организации должны быть положены модели и алгоритмы, позволяющие объединять возможности технологий построения поисковых систем и интеллектуального анализа текста.

Магистерская диссертация посвящена актуальной проблеме исследования и разработки моделей и алгоритмов эффективной системы управления поиском информации в торговой организации.

Выполненные в работе научные исследования представлены следующими основными результатами:

- проведен анализ современного состояния исследований в области построения систем управления поиском информации в торговой организации. Как показал анализ, по своим архитектурным и функциональным особенностям СУПИ в торговой сфере относятся к корпоративным поисковым системам. В торговых организациях корпоративные поисковые системы используются для решения задач рекламы и маркетинга. Одним из требований к функционалу современной СУПИ является возможность анализа тональности отзывов клиентов о качестве приобретаемых товаров. В настоящее время активно развиваются решения, в которых используются методы интеллектуального анализа текста. Для обеспечения эффективного поиска в интернет-магазинах используются внутренние поисковые системы с элементами искусственного интеллекта. Среди недостатков внутренних поисковиков для

электронной коммерции можно выделить следующие: затраты времени на поиск, сложность конкурентных ключевых слов, изменение алгоритмов. Вместе с тем проведенный анализ позволил констатировать недостаточность работ, посвященных проблеме построения систем управления поиском корпоративной информации в торговых организациях, что подтверждает актуальность темы настоящего исследования;

- проведен анализ методов и технологий управления поиском информации в торговой организации. Как показал анализ, интеллектуальный анализ текста анализирует массу неструктурированного текста, классифицируя каждый документ по его основной теме, намерению и настроению (положительному, отрицательному или нейтральному). По результатам сравнительно анализа для классификации текста на основе машинного обучения выбран попарный метод. Для анализа тональности отзывов клиентов используется метод бинарной классификации. Для решения задач бинарной классификации используются программные средства - бинарные классификаторы, реализующие бинарные алгоритмы классификации текста. По результатам сравнительно анализа в качестве алгоритма бинарной классификации текста выбран алгоритм метода опорных векторов SVM. Главное преимущество алгоритма SVM – высокая эффективность для относительно небольших объемов данных, к которым относится бинарный текст;
- разработаны модели и алгоритмы системы управления поиском информации в торговой организации. В качестве методологии моделирования СУПИ выбран ООП к анализу и проектированию информационных систем. Разработаны диаграмма классов бинарного классификатора и диаграмма вариантов использования, которая представляет функциональную модель СУПИ. Для

представления программной архитектуры СУПИ использована диаграмма компонентов.

- выполнены апробация проектных и решений и оценка их эффективности. Для реализации алгоритмов машинного обучения использованы язык Python и среда Jupyter Notebook (anaconda 3) с библиотекой sklearn. Апробация проектного решения подтвердила возможность его использования для анализа тональности отзывов клиентов о товарах. Затраты при проектном варианте разработки СУПИ сократились в 1,5 раза. Коэффициент эффективности управления предлагаемой СУПИ Кэу> 0,5, что свидетельствует о высокой функциональной эффективности управления поиском информации торговой организации

Таким образом, в работе решена актуальная научно-практическая проблема исследования и разработки моделей и алгоритмов эффективной системы управления поиском информации в торговой организации.

Гипотеза исследования подтверждена.

Список используемой литературы и используемых источников

1. Алгоритмы бинарной классификации в машинном обучении [Электронный ресурс]. URL: https://biconsult.ru/products/algoritmy-binarnoy-klassifikacii-v-mashinnom-obuchenii (дата обращения: 24.04.2023).

2. Бинарный классификатор отзывов на Python [Электронный ресурс]. URL: https://vc.ru/newtechaudit/319804-binarnyy-klassifikator-otzyvov-na-python (дата обращения: 27.04.2023).

3. Вдовин В.М., Суркова Л.Е., Шурупов А.А. М. : Дашков и К, 2016. 388 с.

4. Зольников В.К., Абдуллаев У.А. Технология разработки информационно-поисковой системы предприятия торговли // Современные проблемы науки и образования. 2014. № 5. [Электронный ресурс]. URL: https://science-education.ru/ru/article/view?id=15295 (дата обращения: 24.04.2023).

5. Классификация данных методом опорных векторов [Электронный ресурс]. URL: https://habr.com/ru/articles/105220/(дата обращения: 27.04.2023).

6. Классификация текстов и анализ тональности [Электронный ресурс]. URL: https://neerc.ifmo.ru/wiki/index.php?title=%D0%9A%D0%BB%D0%B0%D1%81%D1%81%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F_%D1%82%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2_%D0%B8_%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7_%D1%82%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D0%B8 (дата обращения: 24.04.2023).

7. Леоненков А. В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose : учебное пособие. М.: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. 317 с. [Электронный ресурс]. URL: https://www.iprbookshop.ru/97554.html (дата обращения: 24.03.2023).

8. Онлайн-отзывы о товарах как источник ценных данных [Электронный ресурс]. URL: https://www.megaputer.com/ru/leveraging-online-product-reviews/ (дата обращения: 24.04.2023).

9. Поисковые движки для интернет-магазинов [Электронный ресурс]. URL: https://www.shopolog.ru/services/section/search-engines/ (дата обращения: 24.04.2023).

10. Представление текстовых данных в векторном пространстве [Электронный ресурс]. URL: https://vc.ru/u/824164-yuliya-timonkina/503780-predstavlenie-tekstovyh-dannyh-v-vektornom-prostranstve (дата обращения: 24.03.2023).

11. Савенко А.Г., Шерстнев А.С. Модели и алгоритмы для адаптивного поиска в информационно-поисковых системах [Электронный ресурс]. URL: https://libeldoc.bsuir.by/bitstream/123456789/46769/1/Savenko_Modeli.pdf (дата обращения: 24.04.2023).

12. Система корпоративного поиска Naumen Enterprise Search [Электронный ресурс]. URL: https://www.naumen.ru/products/enterprise_search/(дата обращения: 27.04.2023).

13. Сколько стоят услуги программистов? [Электронный ресурс]. URL: https://www.kadrof.ru/articles/46641 (дата обращения: 17.02.2023).

14. Татарников О. Корпоративные поисковые системы [Электронный ресурс]. URL: https://ecm-journal.ru/material/Korporativnye-poiskovye-sistemy (дата обращения: 10.04.2023).

15. Умный поисковик для корпоративных систем и web [Электронный ресурс]. URL: https://edok-journal.ru/articles/biznes/umnyy_poiskovik_dlya_korporativnykh_sistem_i_web/ (дата обращения: 10.04.2023).

16. Abdelhadi Bahafid et al. Toward a new Information retrieval system based on an e-commerce ontology // International journal of advanced studies n computer science and engineering. 2015. Volume 4. Issue 10. P. 26-34.

17. Aziz H. Why the search engine in your web shop is important [Электронный ресурс]. URL: https://www.prestashop.com/en/blog/search-engine-on-your-site-important (дата обращения: 24.04.2023).

18. Casalegno F. Learning to Rank: A Complete Guide to Ranking using Machine Learning [Электронный ресурс]. URL: https://towardsdatascience.com/learning-to-rank-a-complete-guide-to-ranking-using-machine-learning-4c9688d370d4 (дата обращения: 24.04.2023).

19. Chiranjeevi H.S., Manjula K. S. Advanced text documents information retrieval system for search services // Cogent Engineering [Электронный ресурс]. URL: https://www.tandfonline.com/doi/full/10.1080/23311916.2020.1856467 (дата обращения: 24.04.2023).

20. Comparing the best e-commerce search solutions [Электронный ресурс]. URL: https://www.algolia.com/blog/ecommerce/comparing-the-best-ecommerce-search-solutions/ (дата обращения: 24.04.2023).

21. Creating a binary SVM classifier [Электронный ресурс]. URL: https://github.com/christianversloot/machine-learning-articles/blob/main/creating-a-simple-binary-svm-classifier-with-python-and-scikit-learn.md (дата обращения: 27.04.2023).

22. Dakov S., Malinova A. Automated product information retrieval in e-commerce // International Journal of Differential Equations and Applications. 2021. Volume 20. No. 2. P. 157-168.

23. Dandekar N. Pointwise vs. Pairwise vs. Listwise Learning to Rank [Электронный ресурс]. URL: https://medium.com/@nikhilbd/pointwise-vs-pairwise-vs-listwise-learning-to-rank-80a8fe8fadfd (дата обращения: 24.04.2023).

24. He BiShi et al. The Automatic Classification Research of Regional Medical Imaging // 3rd International Conference on Mechatronics and Industrial Informatics (ICMII 2015). 2015. P.362-369.

25. Helms B. and Shea K. Object-oriented concepts for computational design synthesis // International design conference. 2010. P. 1333-1342.

26. Holts A. et al. Automated Text Binary Classification using Machine Learning approach // XXIX International Conference of the Chilean Computer Science Society. P. 212-217.

27. Information Retrieval System [Электронный ресурс]. URL: https://www.stannescet.ac.in/cms/staff/qbank/CSE/Notes/CS6007-INFORMATION%20RETRIEVAL-1428610647-UNIT%20I%20IR%20Final.pdf (дата обращения: 10.04.2023).

28. Joey Dantoni. What Is Text Mining & How Does It Work? [Электронный ресурс]. URL: https://www.netsuite.com/portal/resource/articles/data-warehouse/text-mining.shtml (дата обращения: 24.04.2023).

29. Li R., Liu M., Xu D., Gao J., Wu F., Zhu L. A Review of Machine Learning Algorithms for Text Classification // Cyber Security. CNCERT 2021. Communications in Computer and Information Science, 2022, vol 1506. Springer, Singapore. https://doi.org/10.1007/978-981-16-9229-1_14.

30. Liyi Zhang et al. A Framework for an Ontology-based E-commerce Product Information Retrieval System // Journal of computers. 2009. Vol. 4/ No 6. P. 436-443.

31. Moraes L.P.M. A high-accuracy framework for binary text classification — Machine Learning [Электронный ресурс]. URL: https://medium.com/wearesinch/a-high-accuracy-framework-for-binary-text-classification-machine-learning-2e6128c6879c (дата обращения: 24.03.2023).

32. Project Jupyter [Электронный ресурс]. URL: https://jupyter.org/ (дата обращения: 27.04.2023).

33. Sharma R. Information Retrieval System Explained: Types, Comparison and Components [Электронный ресурс]. URL: https://www.upgrad.com/blog/information-retrieval-system-explained/#:~:text=An%20information%20retrieval%20(IR)%20system,the%20queries%20of%20a%20user / (дата обращения: 27.04.2023).

34. UML 2.ru – Сообщество Аналитиков [Электронный ресурс]. URL:

https://www.uml2.ru/ (дата обращения: 24.03.2023).

35. UML Component Diagrams [Электронный ресурс]. URL: https://www.uml-diagrams.org/component-diagrams.html (дата обращения: 24.03.2023).

36. Vector Space Model [Электронный ресурс]. URL: https://blog.marketmuse.com/glossary/vector-space-model-definition/ (дата обращения: 24.03.2023).

37. Visual Paradigm [Электронный ресурс]. URL: https://online.visual-paradigm.com/ (дата обращения: 24.03.2023).

38. What is Text Mining, Text Analytics and Natural Language Processing? [Электронный ресурс]. URL: https://www.linguamatics.com/what-text-mining-text-analytics-and-natural-language-processing (дата обращения: 24.04.2023).