МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
**«Южно-Уральский государственный университет
(национальный исследовательский университет)»**
Высшая школа электроники и компьютерных наук
Кафедра системного программирования

РАБОТА ПРОВЕРЕНА

Рецензент
Зам. директора по инф.
технологиям и безопасности
ГБУЗ «ЧОМИАЦ»

_____ А.С. Староверов

“___”_____ 2020 г.

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой,
д.ф.-м.н., профессор

_____ Л.Б. Соколинский

“___”_____ 2020 г.

Разработка приложения для интеллектуального анализа отзывов пользователей магазина приложений Google Play

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
ЮУрГУ – 02.03.02.2020.308-035.ВКР

Научный руководитель,
к.ф.-м.н., доцент
_____М.Л. Цымблер

Автор работы,
студент группы КЭ-401
_____П.И. Шумилин

Ученый секретарь
(нормоконтролер)
_____ И.Д. Володченко
“___”_____ 2020 г.

Челябинск-2020

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра системного программирования

УТВЕРЖДАЮ
Зав. кафедрой СП
_____ Л.Б. Соколинский
09.02.2020

ЗАДАНИЕ
на выполнение выпускной квалификационной работы бакалавра
студенту группы КЭ-401
Шумилину Павлу Игоревичу,
обучающемуся по направлению
02.03.02 «Фундаментальная информатика и информационные технологии»

1. Тема работы (утверждена приказом ректора от 24.04.2020 № 627)
Разработка приложения для интеллектуального анализа отзывов пользователей магазина приложений Google Play.

2. Срок сдачи студентом законченной работы: 05.06.2020.

3. Исходные данные к работе

3.1. Dodds P.S. et al. Temporal patterns of happiness and information in a global social network: Hedonometrics and Twitter. // PLoS one, 2011. Vol.6, №12. P. e26752.

3.2. Hu M., Liu B. Mining and summarizing customer reviews. // Proceedings of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining, 2004. P. 168-177.

3.3. Peng Q., Zhong M. Detecting Spam Review through Sentiment Analysis // JSW, 2014. Vol.9, №8. P. 2065-2072.

3.4. Zhang Z., Varadarajan B. Utility scoring of product reviews // Proceedings of the 15th ACM international conference on Information and knowledge management, 2006. P. 51-57.

4. Перечень подлежащих разработке вопросов

4.1. Провести обзор работ на тему интеллектуального анализа отзывов потребителей в интернете.

4.2. Разработать подход к анализу отзывов.

4.3. Выполнить проектирование архитектуры системы.

4.4. Реализовать систему.

4.5. Провести эксперименты, исследующие эффективность предложенного подхода к анализу отзывов.

5. Дата выдачи задания: 09.02.2020.

Научный руководитель,
к.ф.-м.н., доцент М.Л. Цымблер

Задание принял к исполнению П.И. Шумилин

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Актуальность темы

В связи с активным развитием социальных сетей, форумов, блогов и других сетевых ресурсов, которые предоставили людям возможность дискутировать, общаться, выражать свое мнение и массово взаимодействовать друг с другом, образовалось большое количество публично доступных неструктурированных текстовых данных. Возможность получить ценную информацию о субъективных суждениях и оценках людей стала причиной большого интереса к исследованиям в области *анализа мнений (opinion mining)* и *анализа тональности текстов (sentiment analysis)*. Одной из форм выражения мнения является пользовательский *отзыв*.

Одной из популярных площадок для публикации мобильных приложений является *магазин приложений Google Play*. Эта площадка предоставляет разработчикам доступ к потенциальной аудитории и снабжена механизмом публикации отзывов, что позволяет пользователям публично делиться своим мнением о *приложении*. Такая информация является полезной не только для разработчиков, но и для самих пользователей. Существует большое количество приложений со схожей функциональностью, и при выборе пользователи часто руководствуются *оценками* и отзывами других пользователей. Однако изучение отзывов с целью формирования объективного представления о качестве приложения является непростой задачей, которая может потребовать значительных временных затрат. Причиной этого является многочисленность отзывов, их противоречивость и недостоверность. Все это затрудняет потенциальному пользователю оценку качества приложения и принятие решения о его покупке, установке и использовании.

Данная работа посвящена созданию программной системы, которая осуществляет *группировку* отзывов по упоминаемым в них *аспектам* приложения и *тональности* по отношению ним, с использованием технологий интеллектуального анализа данных. Это позволит потенциальному пользо-

вателю выборочно ознакомиться с общим мнением относительно конкретной интересующей его характеристики приложения и сделать более осознанный выбор среди аналогов.

Цель и задачи

Целью данной работы является разработка приложения для интеллектуального анализа отзывов пользователей магазина приложений Google Play.

Для достижения цели работы необходимо решить следующие задачи.

1. Провести обзор работ на тему интеллектуального анализа отзывов потребителей в интернете.

2. Разработать подход к анализу отзывов.

3. Выполнить проектирование архитектуры системы.

4. Реализовать систему.

5. Провести эксперименты, исследующие эффективность предложенного подхода к анализу отзывов.

Структура и объем работы

Работа состоит из введения, 5 разделов, заключения и списка литературы. Объем работы составляет 48 страниц. Объем списка литературы составляет 20 наименований.

Содержание работы

В первом разделе, «Анализ предметной области», содержится обзор работ по тематике исследования и описание особенностей магазина приложений Google Play.

Во втором разделе, «Проектирование», даны формальные определения предметной области, выполнен анализ требований, приведена диаграмма вариантов использования системы, построен макет графического интерфейса и диаграмма сущность-связь.

В третьем разделе, «Подход к анализу отзывов», приведена диаграмма деятельности разработанного подхода, рассмотрены этапы извлечения

аспектов, извлечения тональной лексики, формирования резюме, а также составляющие их шаги.

В четвертом разделе, «Реализация», приведена диаграмма компонентов системы, описана реализация каждого компонента, используемые технологии и инструменты.

В пятом разделе, «Эксперименты», описано исследование эффективности предложенного подхода к анализу отзывов и процесс формирования набора данных для экспериментов, построены графики.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

В разделе 1.1. выполняется обзор работ по тематике выявления спам-отзывов, определения полезности отзывов и обобщения отзывов. В разделе 1.2. рассмотрены основные особенности магазина приложений Google Play в контексте обработки пользовательских отзывов.

1.1. Обзор работ по тематике исследования

С ростом числа отзывов воспринимать их становится труднее. В связи с этим рейтинговой системе необходимо упростить задачу для потребителя, обеспечив ему показ в первую очередь качественных отзывов и скрытие злонамеренных или предвзятых отзывов. Поиск способов осуществления этого привел к ряду работ в области анализа мнений, направленных на выявление *спам-отзывов (opinion spamming)* и на прогнозирование *полезности отзывов (opinion helpfulness)*. Еще одним способом упростить потребителю задачу анализа отзывов является *обобщение отзывов (opinion summarization)*.

Выявление спам-отзывов

Спам-отзыв – это отзыв, который пытается ввести читателя в заблуждение путем содержания ложного мнения о каком-либо продукте или услуге. Целью таких отзывов является воздействие на систему ранжирования для изменения рейтинга и позиции продукта.

Исторически борьба со спамом началась в области рассылок электронной почты, а в позже в силу специфики задачи появилось направление, связанное со спамом в отзывах. Для решения задачи выявления спам-отзывов распространены текстовые подходы, методы определения тональности, обработки естественного языка, машинного обучения с учителем, конструирования признаков.

В работе [14] был предложен подход, комбинирующий пять критериев для расценки отзыва как спама. Среди них наличие вопросов в тексте отзыва, формулировок, которые являются сравнениями между продуктами

или компаниями, сообщений, набранных только заглавными буквами, ссылок на внешние источники, а также несоответствие между тональностью текста отзыва и оценкой пользователя. Авторы работы установили, что между оценкой и тональностью отзыва существуют положительная корреляция, что позволяет использовать критерий их согласованности для обнаружения спама. Преимуществом своего подхода авторы называют возможность оценки отзывов без необходимости знаний о предметной области.

Несмотря на то, что подходы, основанные на анализе текста отзыва, пользуются успехом, они подвержены существенным недостаткам [2]. Обнаружение повторяющегося текста требует дорогостоящих сравнений, и без предварительного сужения числа кандидатов число необходимых сравнений может быстро стать неосуществимым. Часто требуются новые обучающие данные для различных предметных областей. Ручная идентификация спам-отзывов для использования в обучении может быть дорогостоящей и трудоемкой.

Кроме работ, направленных на автоматическое выявление очевидных человеку спам-отзывов, существуют работы, целью которых было определить сфабрикованные отзывы, которые стремятся выглядеть аутентично и неотличимо от настоящих [11]. Такие отзывы могут принести гораздо больше вреда пользователю, потому что их сложно идентифицировать. В работе приводится подход, который решают задачу с точки зрения обнаружения психолингвистического обмана, где рассматриваются психологические эффекты лжи, такие как усиление негативных эмоций и психологическое дистанцирование.

В отличие от многих работ, базирующихся на анализе текста отзыва, в работе [13] предлагается метод поиска спам-отзывов для рейтинговых систем, которые содержат только оценку или текст в которых опционален. Определяются пользователи с аномальной долей отзывов, которые отличаются от большинства других. Авторы отмечают, что при сохранении

точности, сравнимой с другими методами, данный подход требует меньше вычислительных ресурсов, так как исключается обучение модели.

Определение полезности отзывов

Полезность отзывов – некоторая численная характеристика, с помощью которой оценивается ценность отзыва и целесообразность его показа потенциальному потребителю. Способы расчета полезности разнятся в зависимости от работ.

Проблему определения полезности отзывов исследовали в работе [7]. В отличие от использования для упорядочивания отзывов статистики ручного голосования потребителей «полезен» или «бесполезен», авторами был предложен алгоритм автоматической оценки полезности отзывов. Алгоритм использует метод опорных векторов для решения задачи регрессии. Были рассмотрены признаки, связанные с рейтинговыми, структурными, синтаксическими и семантическими свойствами отзывов. В ходе исследования были определены наиболее существенные признаки для оценки полезности отзывов, такие как длина отзыва, составляющие его униграммы и рейтинг.

В работе [16] ставится задача прогнозирования полезности отзывов. Эксперименты показали, что воспринимаемая полезность отзыва сильно зависит от лингвистического стиля его текста.

Обобщение отзывов

Под обобщением отзывов понимают два основных подхода [9]. Традиционный подход рассматривает задачу обобщения отзывов как классическую задачу обработки естественного языка, называемой *автоматическим реферированием (text summarization).* Реферирование на основе большого множества документов формирует один документ меньшего объема, содержащий ключевые предложения из всего множества. Однако данный подход, успешно применяемый для статей и новостных текстов, плохо подходит для обобщения отзывов, в которых выражаются различные и возможно противоречащие друг другу мнения.

Под *обобщением отзывов на основе аспектов (aspect-based opinion summarization)* понимают составление структурированного резюме, итоговой сводки по всем отзывам. Резюме формируется на основе информации, полученной с помощью методов извлечения аспектов продукта и тональностей по отношению к этим аспектам. Такое резюме позволяет пользователю быстро ознакомиться с мнением пользователей.

В работе [5] авторами описан подход, лежащий в основе разработанной ими системы FBS (Feature-Based Summarization). Данная система на основе большого количества отзывов генерирует структурированные сводки, каждая из которых представляют собой характеристику продукта и списки позитивных и негативных высказываний по этой характеристике. Характеристики продукта определялись с помощью разметки частей речи слов текста отзывов и применения алгоритма поиска частых наборов Apriori [1]. Тональность, с которой пользователь высказывался о характеристике, определялась с помощью ручного определения тональности 30 ключевых прилагательных, словарь которых расширялся с помощью поиска синонимов и антонимов из лексической базы данных WordNet [10]. Среди особенностей работы авторы отмечают отличие от классической задачи обобщения текста наличием структуры сводок и определение тональности отзывов на уровне характеристик, а не отзывов целиком.

На основе работы [5] был предложен метод бинарной классификации отзывов на рекомендующий продукт или нет [4].

В работе [12] была предложена система OPINE, которая превосходит FBS в точности для задачи извлечения характеристик на 22% при снижении метрики полноты на 3% по сравнению с FBS. OPINE использует подход релаксационного определения семантической ориентации слов в контексте, что приводит к высокой производительности для задачи поиска фраз, выражающих мнение, и определения их тональности.

1.2. Особенности магазина приложений Google Play

Компания Google прикладывает множество усилий для борьбы со спамом и заказными отзывами на своей площадке. В 2018 году была внедрена система, в которой сочетается использование как ручной модерации, так и алгоритмов искусственного интеллекта [19]. Среди прочих мер, усложняющих появление недостоверных отзывов, можно назвать следующие. Написание отзыва пользователем возможно только после установки приложения. Пользователь может оставить для одного приложения только один отзыв, однако он может его редактировать, при этом история изменений сохраняется. Пользователи и разработчики могут отмечать отзывы как спам или бесполезные, тем самым давая обратную связь системе.

При просмотре отзывов к популярным приложениям откровенных спам-отзывов обнаружено не было. Однако отзывы, нарушающие политику публикации [20], все равно встречаются и публикуются. Среди новых отзывов присутствует множество тех, которые оставлены реальными пользователями, однако являющиеся неинформативными и не представляющими какой-либо ценности для потенциального пользователя, так как не помогают ему понять причины выставленной оценки.

Для разработчиков существует множество статистических и аналитических инструментов, расположенных в Google Play Console. Однако данные инструменты недоступны для пользователей. Пользователи при анализе и выборе приложений ограничены списком новых или актуальных отзывов с возможностью фильтрации по выставленной оценке. Отзывы, размещаемые в категорию актуальных, определяются алгоритмами Google.

Вывод

Большое количество научных работ свидетельствует о интересе к области анализа отзывов. Подходы к анализу сильно разнятся и используют множество методов интеллектуального анализа данных, естественной обработки языка, лингвистики, машинного обучения, статистики.

Компания Google прикладывает усилия для борьбы со спам-отзывами и продвижения полезных отзывов. Однако в настоящее время у пользователей нет возможности просматривать отзывы по конкретному аспекту приложения. Таким образом, анализ отзывов в данной работе направлен на решение задачи обобщения отзывов в смысле составления резюме.

2. ПРОЕКТИРОВАНИЕ

В разделе 2.1. приведены формальные определения предметной области. В разделе 2.2. проводится анализ требований, а в разделе 2.3. описаны варианты использования разрабатываемой системы. В разделе 2.4. описано проектирование графического интерфейса системы, приведен ее макет, а в разделе 2.5. приводится описание модели предметной области и диаграмма сущность-связь.

2.1. Формальные определения

Введем определения используемой терминологии в контексте предметной области.

Приложение – прикладная программа, предназначенная для выполнения определенных задач. *Магазин приложений* – интернет-платформа для публикации приложений разработчиками, а также покупкой и установкой приложений пользователями. *Google Play* – магазин приложений для мобильных устройств на операционной системе Android от компании Google.

Отзывом назовем кортеж, представленный в формуле (1).

$$review = (h_r, t_r, r, u, c), \quad (1)$$

где h_r – имя автора;

t_r – дата публикации;

$r \in [1..5]$ – оценка от низшей до высшей;

$u \in \mathbb{Z}^+$ – полезность;

c – текстовый комментарий, который может содержать в себе несколько мнений.

Полезность – число одобрений отзыва другими пользователями.

Мнением назовем кортеж, представленный в формуле (2).

$$opinion = (e, a, s, h_{op}, t_{op}), \quad (2)$$

где e – приложение, к которому оставлен отзыв;

a – аспект приложения;

s – тональность по отношению к аспекту;

h_{op} – автор;

t_{op} – дата высказывания.

Заметим, что $\forall k, i \in \mathbb{N}: c_k$ содержит $\{opinion_i | h_{r_k} = h_{op_i}, t_{r_k} = t_{op_i}\}$, то есть для любого отзыва k его текстовый комментарий c_k содержит множество мнений, таких что автор данного отзыва h_{r_k} является автором мнения h_{op} и дата публикации отзыва t_r совпадает с датой высказывания мнения t_{op}. Таким образом, для того, чтобы однозначно определить мнение, необходимо установить его аспект a и тональность s, так как остальные значения известны и могут быть получены из отзыва, содержащего это мнение.

Аспект – уникальная черта, свойство, характеристика приложения, о которой пользователи высказываются в отзывах.

Тональность – выраженное в отзыве эмоциональное отношение пользователя к некоторому аспекту. В данной работе *полярность* тональности может быть положительной или отрицательной.

Группой отзывов назовем множество отзывов, каждый из которых содержит мнение о данном аспекте. Определение представлено в формуле (3).

$$\forall k, i \in \mathbb{N}: group_k = \{review_i | \exists\ opinion_{i_j}, a_{i_j} = a_k\}, \quad (3)$$

Резюме – обобщенное представление пользователей о приложении, состоящее из групп отзывов, каждая из которых поделена на подгруппы в зависимости от полярности тональности. Определение представлено в формуле (4).

$$summary = [group_1, \dots group_n], \quad (4)$$

где $group_i = (\{review_i \mid s_i = positive\}, \{review_j \mid s_j = negative\})$

Разрабатываемая система, представляет собой приложение, формирующее резюме с помощью определения частых аспектов приложения и

характеризующей их тональности на основе технологий интеллектуального анализа данных.

2.2. Требования к системе

Функциональные требования

Были определены следующие функциональные требования к разрабатываемой системе.

1. Система должна предоставлять пользователю возможность выбора приложения для анализа и просмотра отзывов.

2. Система должна предоставлять пользователю возможность выполнить анализ отзывов.

3. Система должна предоставлять пользователю возможность настроить параметры анализа отзывов.

4. Система должна группировать отзывы по аспектам приложения и полярности тональности.

5. Система должна предоставлять пользователю возможность просмотреть отзывы, отнесенные в данную группу.

6. Система должна предоставлять пользователю возможность менять порядок отзывов внутри группы.

Нефункциональные требования

Нефункциональные требования перечислены ниже.

1. Отзывы пользователей должны быть получены из Google Play.

2. Должен производится анализ русскоязычных отзывов.

3. Реализация графического интерфейса, обработки и анализа отзывов должна быть выполнена на языке программирования Python.

2.3. Варианты использования системы

Для описания способов взаимодействия с системой была разработана диаграмма вариантов использования, представленная на рисунке 1.

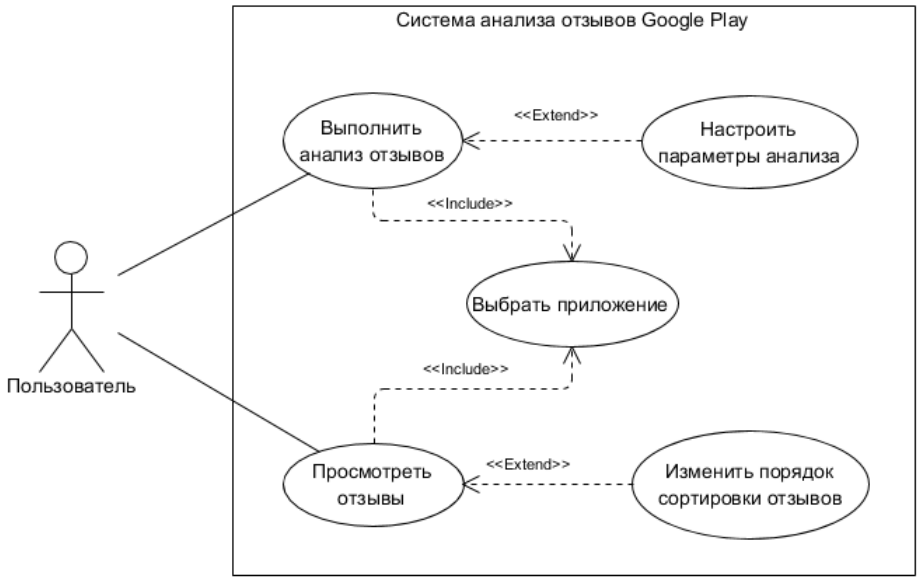

Рис. 1. Диаграмма вариантов использования

Единственным актером, взаимодействующим с системой, является *пользователь*. Для него определены следующие варианты использования.

1. *Выполнить анализ отзывов.* Пользователь инициирует повторное выполнение анализа отзывов с текущими значениями параметров анализа. Анализ заключается в извлечении частых аспектов и тональной лексики по отношению к этим аспектам, а также определении ее полярности.

2. *Выбрать приложение.* Пользователь должен выбрать приложение, отзывы для которого он хочет проанализировать или просмотреть.

3. *Настроить параметры анализа.* Пользователь может задать значение минимальной поддержки, которое влияет на количество найденных частых аспектов, и значение радиуса окна поиска тональной лексики вокруг аспекта.

4. *Просмотреть отзывы.* Выбирая желаемый аспект и полярность тональности, пользователь может просматривать отзывы, которые содержат мнения о данном аспекте с данной тональностью. Пользователю доступна информация о имени автора отзыва, дате его публикации, оценке приложения, полезности отзыва и текстовом комментарии.

5. *Изменить порядок сортировки отзывов.* Пользователю доступно изменение порядка сортировки отзывов по каждому из полей, перечисленных в предыдущем пункте.

2.4. Графический интерфейс

Макет основного окна спроектированного графического интерфейса представлен на рисунке 2. На макете отражены основные структурные и функциональные элементы интерфейса.

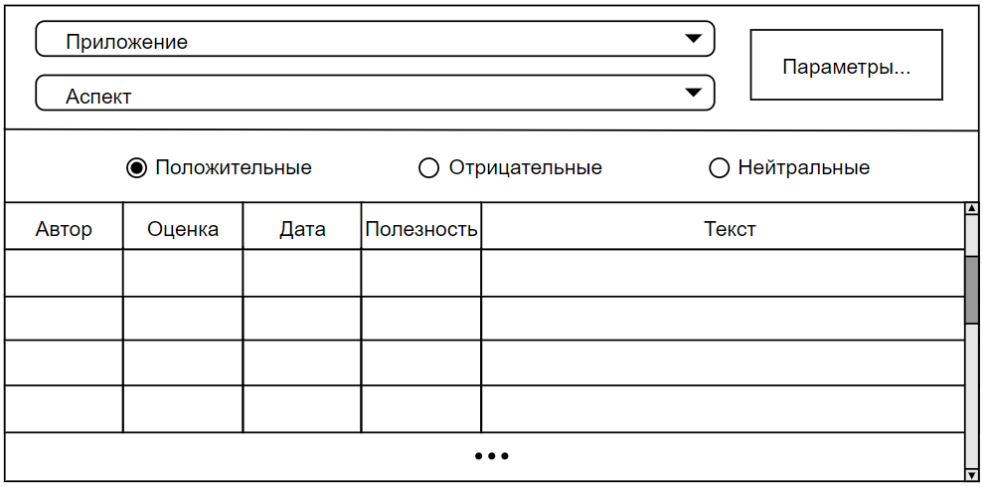

Рис. 2. Макет основного окна графического интерфейса

Блок управления отображением желаемых отзывов расположен в верхней части окна. Он включает в себя выпадающий список выбора приложения, отзывы для которого необходимо просмотреть или проанализировать. Ниже находится выпадающий список найденных аспектов для выбранного приложения. Еще ниже расположена панель выбора полярности тональности отзывов для данного аспекта данного приложения.

Подходящие под заданные критерии отзывы отображаются в таблице в нижней части окна. Таблица содержит информацию об авторе отзыва, выставленной им оценки приложению, дате публикации отзыва, полезности отзыва и тексте отзыва.

Нажав на кнопку параметров, открывается второе окно, макет которого представлен на рисунке 3. В окне можно задать желаемые значения

параметров анализа отзывов и запустить повторный анализ с введенными значениями.

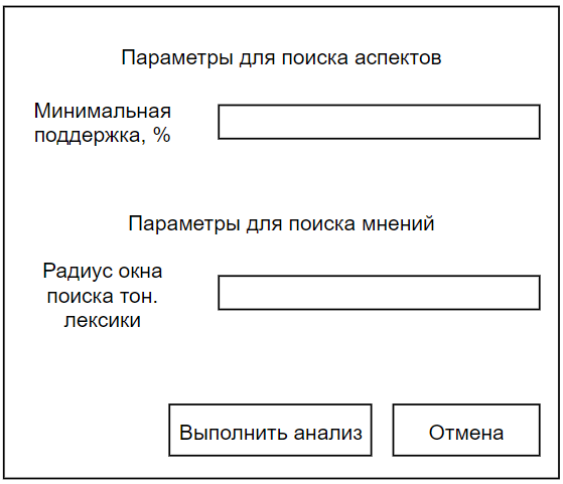

Рис. 3. Макет окна настроек параметров анализа отзывов

2.5. Описание модели предметной области

На рисунке 4 представлена диаграмма сущность-связь, которая содержит описание модели предметной области.

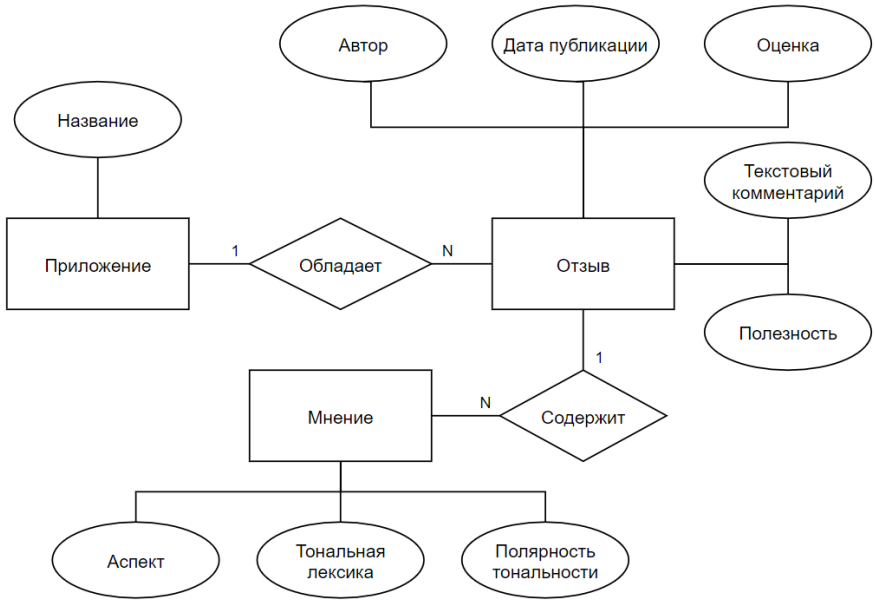

Рис. 4. Диаграмма сущность-связь

Основными сущностями являются приложение, отзыв и мнение. Приложение обладает множеством отзывов, в каждом из которых содержится множество мнений. Важным атрибутом отзыва является его текстовый комментарий, описывающий опыт пользователя и его впечатления от

приложения. Мнение характеризуется аспектом приложения, тональной лексикой и ее полярностью.

Описание атрибутов всех сущностей приведено в таблице 1.

Табл. 1. Описание атрибутов сущностей

	Атрибут	**Тип**	**Семантика**	**Пример**	**Примечание**
1	Название	CHAR(255)	Название приложения	Яндекс.Почта	Используется как идентификатор приложения
2	Автор	CHAR(255)	Имя пользователя Google Play	Иван Иванов	
3	Дата публикации	CHAR(255)	Дата публикации отзыва	2020-05-15	Строка формата YYYY-MM-DD
4	Оценка	INTEGER	Оценка приложения пользователем	4	Оценка от 1 до 5
5	Текстовый комментарий	TEXT	Содержание отзыва	Приложение понравилось. Отличный дизайн и функциональность. Приятно пользоваться, спасибо разработчикам.	Содержит множество мнений
6	Полезность	INTEGER	Количество человек, посчитавших отзыв полезным	3	Может быть нулем
7	Аспект	CHAR(255)	Характеристика приложения	дизайн	
8	Тональная лексика	CHAR(255)	Оценочное слово по отношению к аспекту	отличный	
9	Полярность тональности	CHAR(255)	Полярность сентимента	positive	Принимает значения positive, или negative

Вывод

В данном разделе на основе проведенного анализа требований было выполнено проектирование системы, которое включает в себя построение диаграммы вариантов использования, создание макета графического интерфейса, описание модели предметной области.

3. ПОДХОД К АНАЛИЗУ ОТЗЫВОВ

Введем необходимые определения из области лингвистики и обработки естественного языка.

Корпус – подобранная и обработанная по определённым правилам совокупность текстов, используемых в качестве базы для исследования языка. В контексте данной работы корпусом будем называть множество всех загруженных для данного приложения отзывов, для которых выполняется анализ и формирование резюме.

Частеречная разметка (POS-tagging) – процесс определения части речи каждого слова и его маркировки.

Именная группа (noun phrase) – словосочетание, где главным словом является имя существительное. Имя существительное, не содержащее зависимых слов, также относят к именной группе, состоящей из одного слова.

На основе рассмотренных научных работ по теме основанного на аспектах обобщения отзывов был сформирован подход к их анализу. Используемый в данной работе подход основан на методах, представленных в работах [5, 6], и адаптирован для русскоязычных отзывов. На рисунке 5 изображена диаграмма деятельности, отражающая содержание подхода к анализу отзывов.

Подход к анализу отзывов состоит из трех этапов: извлечения аспектов, извлечения тональной лексики и формирования резюме. Данный подход применяется для предварительно загруженных из Google Play отзывов на приложение. В начале производится очистка данных отзывов и их *частеречная разметка*. Данная разметка необходима для *формирования транзакций* – преобразования данных к специальному виду необходимому для *поиска частых аспектов.* На этом шаге осуществляется поиск частых аспектов с помощью алгоритма Apriori. Далее на основе найденных аспектов выполняется *поиск тональной лексики* по отношению к аспектам. Под тональной лексикой понимаются слова, которые выражают отношение

субъекта высказывания мнения к объекту высказывания. На следующем шаге происходит *определение полярности тональной лексики* на основе тонального словаря. Полярность может быть положительной или отрицательной. На заключительном шаге, на основе найденных мнений, производится *формирование резюме*, которое представляет собой структурированное отображение отзывов в соответствии с содержащимися в них мнениями.

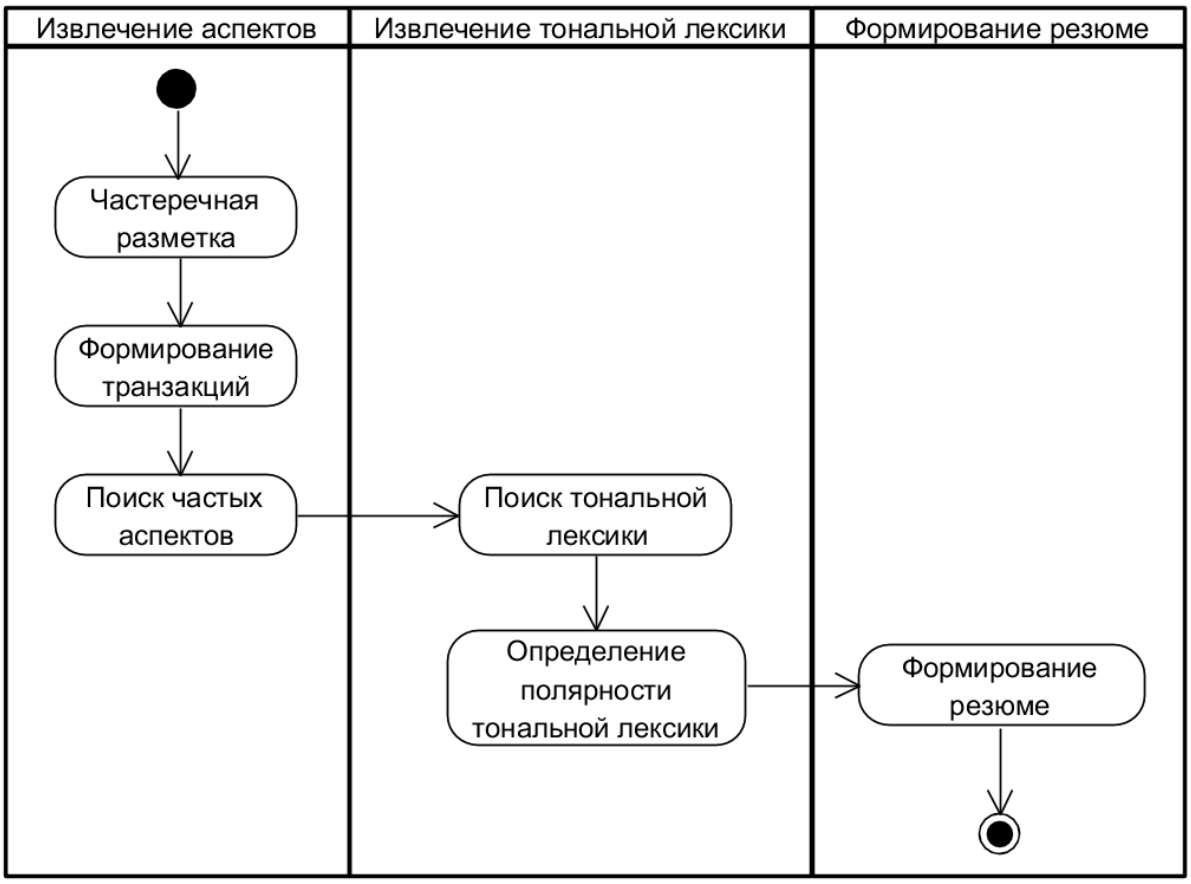

Рис. 5. Подход к анализу отзывов

Более подробно шаги, представленные на диаграмме деятельности, рассмотрены далее.

3.1. Извлечение аспектов

Обозначим характеристики отзывов, для которых выполняется анализ, и принимаемые допущения при извлечении аспектов. В работе [5] отзывы делятся на два типа в зависимости от формы выражения мнения. Для

неявно выраженных мнений аспект не присутствует в тексте отзыва дословно, но содержится его описание. Это значительно усложняет его обнаружение и требует анализа на уровне семантики текста. Для явно выраженных мнений слово, выражающее аспект, содержится в тексте отзыва. В данной работе выполняется анализ отзывов, в которых аспект выражен явно и аспекты представлены именными группами. Согласно работе [5] 60-70% аспектов выражены в тексте как именные группы. В данной работе предполагается, что для русского языка доля именных групп среди аспектов также значительна.

Для извлечения аспектов из мнений используют статистические, лингвистические подходы, а также машинное обучение [18]. Представленный в данной работе подход к анализу отзывов рассматривает задачу извлечения аспектов как задачу поиска частых наборов. Поскольку люди, как правило, используют одни и те же слова для ссылки на характеристики сущности, то частоту их упоминания можно использовать для извлечения аспектов приложения. Приведем формальную постановку задачи поиска частых наборов применительно к теме данной работы [1].

Пусть $\mathcal{I} = \{i_1, i_2, \dots i_m\}$ – множество именных групп, любое непустое его подмножество назовем *набором.* Наборы состоящие из k именных групп назовем *k-наборами.* Назовем *транзакцией* пару (TID, I), где TID – уникальный идентификатор транзакции, $I \subseteq \mathcal{I}$ – набор. Множество транзакций, полученных на корпусе отзывов, обозначим D. *Поддержкой* набора $I \subseteq \mathcal{I}$ является доля транзакций D, содержащих данный набор. Определение приведено в формуле (5).

$$support(I) = \frac{|\{T \in D \mid I \subseteq T.I\}|}{|D|} \quad (5)$$

Параметром задачи является наперед задаваемый *порог поддержки* $minsup$. Набор, поддержка которого не ниже $minsup$, назовем *частым.*

Классическим алгоритмом поиска ассоциативных правил, который может быть применен и для поиска частых наборов, является алгоритм

Apriori, предложенный в работе [1]. Данный алгоритм использует свойство антимонотонности, которое заключается в том, что поддержка любого набора не может превышать минимальной поддержки любого из его подмножеств. Благодаря этому свойству удается значительно повысить эффективность перебора, заведомо отбрасывая неподходящие наборы.

На первом шаге этапа извлечения аспектов выполняется частеречная разметка отзыва. Частеречная разметка необходима, чтобы обнаружить в тексте именные группы.

На основе размеченных отзывов производится формирование транзакций, наборы которых состоят из лемматизированных слов, входящих в именные группы. *Лемматизация* – это процесс приведения словоформы к ее словарной форме. Это преобразование позволяет рассматривать разные формы слова как одно слово.

Таким образом, на основе исходных текстов отзывов формируются транзакции, каждая из которых содержит набор именных групп в словарной форме, которые были использованы в отзыве.

На множестве полученных транзакций T с помощью алгоритма Apriori выполняется поиск частых аспектов. Значение $minsup$ является переменным параметром, который доступен пользователю для изменения. Аспекты чаще всего выражаются одним существительным, поэтому максимальная длина набора была ограничена 1 словом.

3.2. Извлечение тональной лексики

Так как в одном отзыве пользователи выражают мнения сразу по нескольким аспектам приложения, то важно определять тональность на уровне каждого аспекта отдельно, а не на уровне предложения или всего отзыва целиком. Для того чтобы установить тональность s по отношению к аспекту a мнения $(e, a, s, h_{op}, t_{op})$ необходимо выполнить поиск тональной лексики. В данной работе тональной лексикой считаются прилагатель-

ные, в силу того, что оценочные значения чаще всего выражаются в языках при помощи признаковых слов.

Существует два глобальных подхода к автоматическому извлечению тональной лексики: на основе словарей и на основе текстовых коллекций [17]. В данной работе определение тональности прилагательного устанавливается по тональному словарю, в котором содержатся соответствия между словарной формой прилагательного и его тональностью. Если данного прилагательного нет в тональном словаре, или для аспекта в отзыве не было найдено зависимого прилагательного, то тональность по отношению к данному аспекту устанавливается нейтральной.

Прилагательные часто используются для выражения эмоций и отношения автора мнения к определенному аспекту и выступают в роли зависимых слов для именных групп. В наиболее простых случаях такая зависимость локализована. В связи с этим, в данной работе используется эвристика, утверждающая, что ближайшее к аспекту прилагательное отражает его тональность. Таким образом, для определения тональности автора мнения по отношению к аспекту выполняется поиск ближайшего к данному аспекту прилагательного, которое грамматически согласуется с ним, то есть $\forall a \in (e, a, s, h_{op}, t_{op})$ обнаруживается (a, ADJ), так что $s = ADJ_{pol}$, где ADJ – ближайшее грамматически согласующееся с a прилагательное, ADJ_{pol} – полярность прилагательного по тональному словарю.

Тональный словарь был сформирован на основе объединения слов из переведенного на русский язык тезауруса WordNet-Affect [3] и прилагательных из словаря оценочной лексики РуСентиЛекс [8].

WordNet-Affect [15] является лексическим ресурсом, который содержит слова, описывающие эмоции. Он был создан на базе семантического лексикона английского языка WordNet [10] путем выбора и разметки синсетов (наборов синонимов) эмоциональными концепциями. Слова WordNet-Affect разбиты на 6 файлов, соответствующие категориям: «ра-

дость», «страх», «гнев», «печаль», «отвращение», «удивление». Среди прочей информации файлы содержат синсет переведенных на русский язык слов оригинального английского синсета с тем же смыслом. Для слов из категорий «радость» и «удивление» была назначена положительная тональность, для остальных – отрицательная.

РуСентиЛекс представляет собой словарь оценочных слов и выражений русского языка. Актуальная версия 2017 года словаря содержит более 12 тысяч слов и выражений. В словаре представлены следующие типы русскоязычных слов, значения которых связаны с тональностью: слова литературного русского языка, для которых хотя бы одно значение имеет оценочный компонент, то есть отношение либо явно выражается словом, либо передается через выражаемую эмоцию; слова, не передающие оценочное отношения автора, но имеющие положительную или отрицательную коннотацию; сленговые и ругательные слова из социальной сети Твиттер, не являющиеся матом. Наряду с другой информацией словарь содержит часть речи, слово или словосочетание в лемматизированной форме и полярность тональности.

3.3. Формирование резюме

Для формирования резюме происходит разбиение группы отзывов на подгруппы в зависимости от полярности тональности. Система формирует три подгруппы: отзывы с положительной тональностью, отзывы с отрицательной тональностью, а также нейтральные отзывы. К ней относятся отзывы, тональность которых не была определена системой как положительная или отрицательная. Каждая подгруппа представлена в виде списка.

Для каждого аспекта производится подсчет отзывов, которые содержат мнения с положительной и отрицательной тональностью по отношению к этому аспекту. Данное число позволяет визуально определить, тональность скольких отзывов была распознана системой для данного аспекта.

Наряду с возможностью ознакомиться с каждым из отзывов, резюме предполагает вычисление общего результата по всем отзывам, содержащихся в группе. Каждому аспекту, соответствующему группе отзывов, ставится в соответствие итоговое значение, которое отражает обобщенное мнение пользователей по данному аспекту и вычисляется по формуле (6):

$$groupRating = \frac{pos}{pos + neg}, \quad (6)$$

где pos – количество отзывов с положительной тональностью;

neg – количество отзывов с отрицательной тональностью.

Для $groupRating > 0.5$ название аспекта подсвечивается зеленым цветом, для $groupRating < 0.5$ – красным, а для $groupRating = 0.5$ – коричневым. Аспекты, все отзывы которого относятся к группе нейтральных, имеют черный цвет.

Внутри каждой подгруппы возможно изменение поля, по которому сортируются отзывы, а также порядка сортировки.

Вывод

В данном разделе была построена диаграмма деятельности, иллюстрирующая подход к анализу отзывов. Выполнено описание этапов извлечения аспектов, извлечения тональной лексики, формирования резюме и каждого шага, входящего в данные этапы.

4. РЕАЛИЗАЦИЯ

Архитектура реализованной системы отражена на диаграмме компонентов, представленной на рисунке 6.

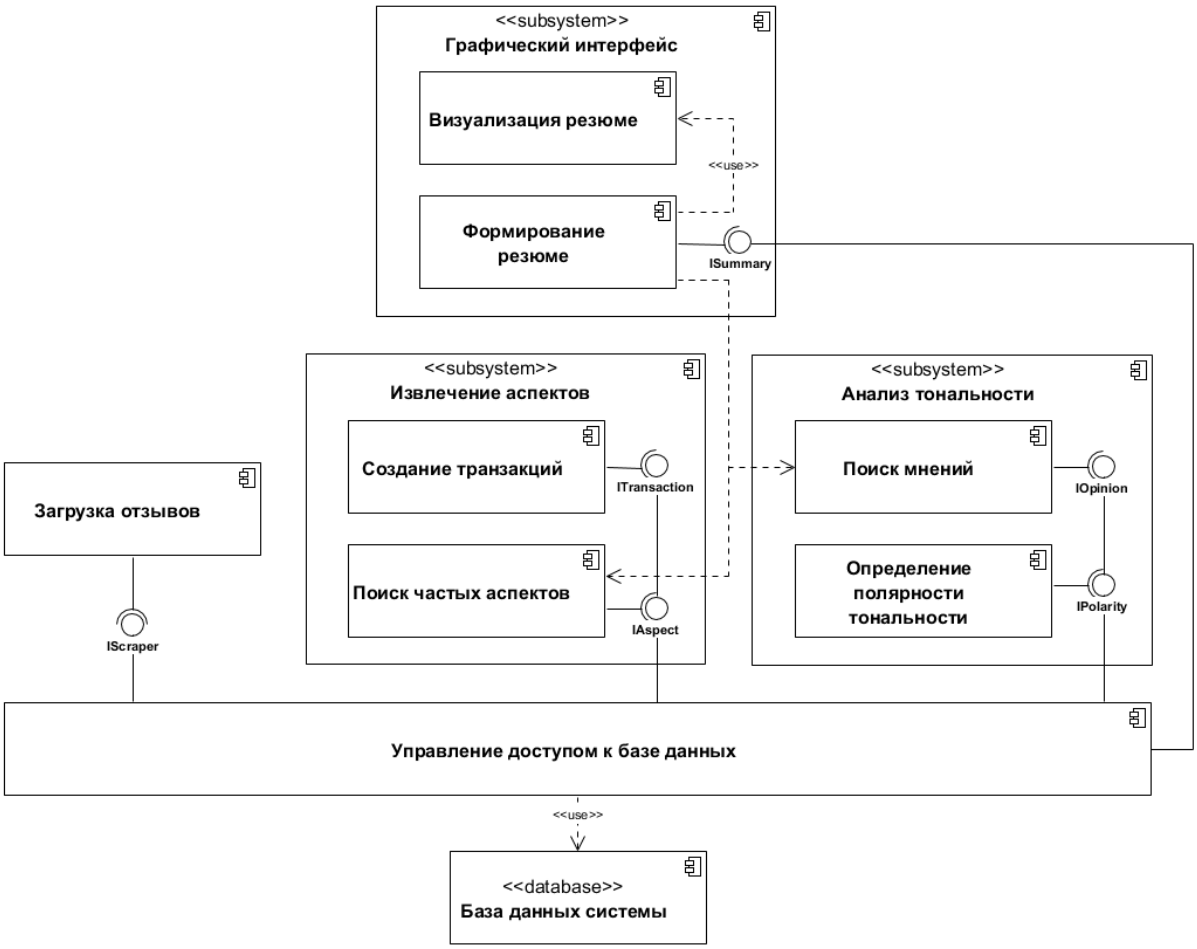

Рис. 6. Диаграмма компонентов

Компонент загрузки отзывов и компоненты подсистем извлечения аспектов и анализа тональности представляют собой последовательные этапы подхода к анализу отзывов. Компоненты обмениваются данными через базу данных посредством компонента управления доступом к базе данных, используя предоставляемые им интерфейсы. Подсистема графического интерфейса отвечает за формирование резюме и его визуализацию. Реализация всех компонентов выполнена на языке программирования Python.

4.1. Компонент базы данных

Сценарий использования разрабатываемой системы предполагает частое получение сведений о всех отзывах, содержащих мнение о данном аспекте с данной тональностью для отображения пользователю. При использовании реляционной базы данных в случае с нормализованными таблицами необходимо часто применять дорогостоящую операцию JOIN, а в случае отказа от нормализации и хранения всех данных в одной таблице получаем неоправданное множественное дублирование данных. В связи с этим хранение данных разрабатываемой системы будет осуществляться с применением не реляционной модели данных.

Для реализации компонента базы данных была выбрана относящаяся к классу NoSQL документоориентированная СУБД MongoDB. Особенностью такой организации данных является нефиксированная схема и свойство локальности данных, позволяющее получить данные, использующиеся совместно, одним запросом. Это достигается путем хранения данных в виде JSON-подобных документов, которые объединены в коллекции. Каждый документ по умолчанию имеет значение уникального идентификатора *_id*, по которому строится индекс. Далее приведено содержание документов коллекций реализованной системы.

Отзывы

Документы коллекции для работы с отзывами *reviews* содержат две группы атрибутов.

Первая группа – это атрибуты, значения которых присутствуют непосредственно в самом отзыве к приложению и получаются в момент синтаксического разбора загруженной веб-страницы. В нее входят: *name* – имя автора отзыва, *score* – оценка автором приложения, *date* – дата публикации отзыва, *utility* – полезность отзыва, *text* – текстовый комментарий отзыва.

Вторая группа – это атрибуты, получение значений которых осуществляется во время работы разработанной системы. В нее входят: *trans-*

action – транзакция, состоящая из содержащихся в тексте отзыва именных групп, приведенных к словарной форме, *opinions* – массив мнений, содержащихся в тексте отзыва. Каждое мнение содержит *aspect* – аспект приложения, найденный в тексте отзыва, *sentiment* – тональная лексика по отношению к аспекту. В случае обнаружения тональной лексики мнение также будет содержать *polarity* – полярность тональности.

Структура документа для коллекции отзывов *reviews* приведена на рисунке 7.

```
{
  $jsonSchema: {
     required: [ "name", "score", "date", "likes", "text", "transaction", "opinions" ],
     properties: {
        name: { bsonType: "string" },
        score: { bsonType: "int" },
        date: { bsonType: "string" },
        likes: { bsonType: "int" },
        text: { bsonType: "string" },
        transaction: { bsonType: "string" }
        opinions: [{
           bsonType: "object",
           required: [ "aspect", "sentiment" ],
           properties: {
               "aspect": { bsonType: "string" },
               "sentiment": { bsonType: "string" },
               "polarity": { bsonType: "string" }
           }
        }]
     }
  }
}
```

Рис. 7. JSON-схема коллекции reviews

Структура используется для валидации документов и отражает необходимость документа содержать конкретные атрибуты, а также их типы. Для других коллекций, рассмотренных далее, JSON-схема в силу схожести и аналогичности составления не приводится.

Аспекты

В данной коллекции хранятся документы, каждый из которых имеет единственный атрибут *aspect*, соответствующий словарной форме найденного частого аспекта. Коллекция используется, как хранилище всех найденных частых аспектов, на основе которых выполняется поиск мнений.

Параметры

В коллекции параметров *settings* хранятся документы, содержащие информацию о значениях, которые используют для своей работы компоненты поиска частых наборов и поиска мнений. Данные значения хранятся независимо для каждого приложения. Каждый документ содержит следующие атрибуты: *app_label* – название приложения, которое отображается в графическом интерфейсе, *collection_name_prefix* – префикс, с которого начинаются названия коллекций, относящихся к данному приложению, *minsup* – значение минимальной поддержки для поиска частых аспектов в процентах, *radius* – радиус окна для поиска тональной лексики вокруг аспекта.

Тональный словарь

В коллекции *sentiment_dictionary* хранятся документы, содержащие информацию о известных словах и полярности их тональности. Каждый документ содержит следующие атрибуты: *word* – слово, *polarity* – полярность. Данная коллекция используется как словарь, по которому определяется полярность тональной лексики, встретившейся в мнении.

4.2. Компонент управления базой данных

Диаграмма классов данного компонента приведена на рисунке 8. Компонент представлен классом *MongoDb*, который взаимодействует остальными компонентами системы через предоставляемые им интерфейсы *IScpaper, IAspect, ITransaction, IPolarity, IOpinion, ISummary*. Каждый из

этих интерфейсов содержит методы для доступа соответствующего компонента к базе данных.

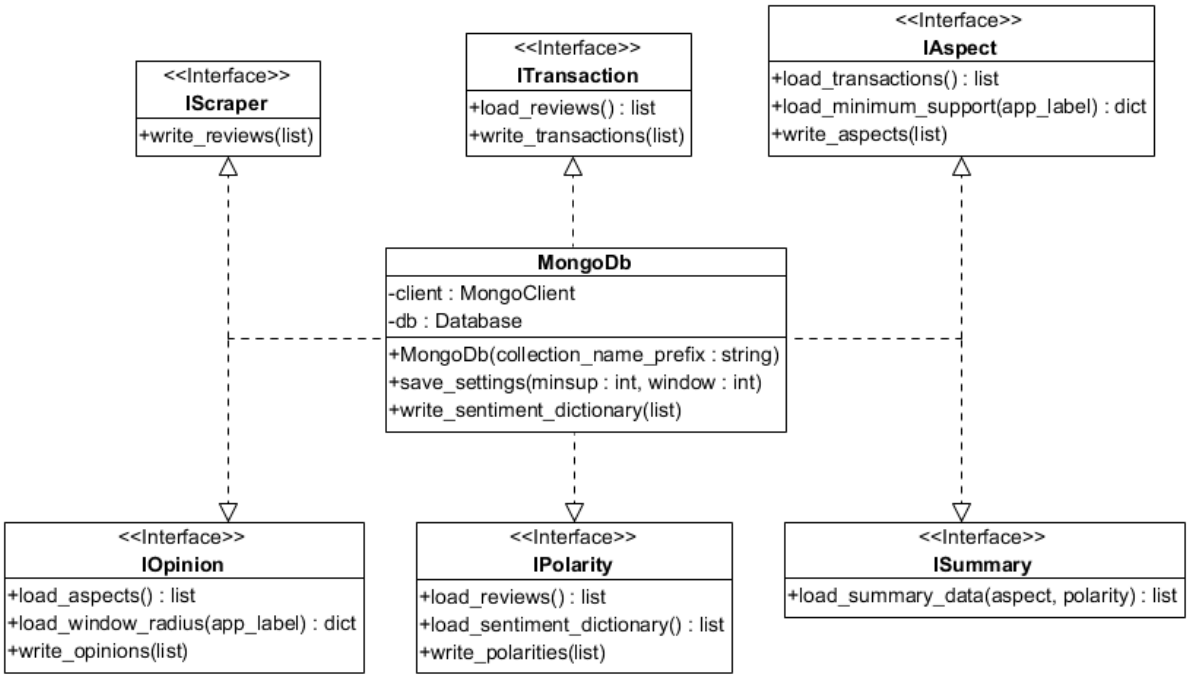

Рис. 8. Диаграмма классов компонента управления доступом к БД

Конструктор класса *MongoDb* принимает аргумент префикса имен коллекций *collection_name_prefix*. Он определяет какие коллекции относятся к данному приложению. Поле *client* содержит ссылку на объект *MongoClient* официального драйвера MongoDB для Python *pymongo*.

Множество методов, начинающихся с *load* и *write*, записывают данные в базу и получают данные из нее соответственно. Например, метод *load_transactions* интерфейса *IAspect* получает документы отзывов, которые содержат транзакции, добавленные после завершения работы компонента формирования транзакций, а метод *write_aspects* записывает найденные аспекты в базу данных. Важно отметить, что корректная работа системы предполагает последовательное синхронное исполнение компонентов в нужном порядке, так как каждый следующий компонент в своей работе опирается на данные, полученные и записанные в базу данных предыдущим компонентом.

Метод *write_sentiment_dictionary* используется однократно для загрузки в базу данных словаря тональной лексики. Метод *save_settings* используется для сохранения в базу обновленных значений минимальной поддержки и радиуса окна для поиска тональной лексики. Эти параметры влияют на работу компонентов поиска частых аспектов и поиска мнений соответственно.

4.3. Компонент загрузки отзывов

Компонент загрузки отзывов представлен классом *GooglePlayScraper*. Его диаграмма представлена на рисунке 9. Согласно принятому соглашению двойное подчеркивание в именах методов служит для обозначения их приватности в языке программирования Python.

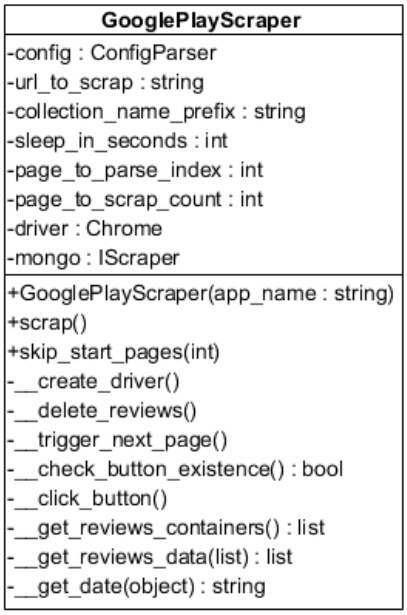

Рис. 9. Диаграмма класса GooglePlayScraper

Внутри конструктора объекта выполняется его инициализация параметрами, заданными в конфигурационном файле. Чтение параметров производится с помощью пакета *configparser*. Среди параметров *url_to_scrap* – URL страницы с отзывами, которые необходимо загрузить, *page_to_scrap_count* – количество страниц для загрузки, *sleep_in_seconds* –

таймаут между запросами, *collection_name_prefix* – префикс для имен коллекций для данного приложения. Так же в методе *__create_driver* выполняется создание объекта *chromerdriver,* хранящегося в *driver,* и создание объекта *MongoDb*, хранящегося в *mongo*.

Объект *chromerdriver* необходим для работы инструмента *Selenium*, с помощью которого выполняется загрузка отзывов путем синтаксического разбора веб-страниц в методе *scrap.* Наряду с тестированием веб-приложений данный инструмент также применяется для парсинга веб-страниц. Несмотря на более медленную работу, чем у других инструментов, используемых для получения данных, например, *Scrapy*, выбор был сделан в пользу *Selenium*, так как для загрузки следующей страницы с данными отзывов необходимо взаимодействовать с кнопкой, нажатие на которую вызывает обработку события с отправкой асинхронного запроса на сервер. Выполнить имитацию нажатия на кнопку и отправить запрос, не используя вызов обработчика события, не удалось, так как отправляемые данные намерено приведены к виду, затрудняющему их анализ.

Загруженные отзывы сохранялись в базу данных через экземпляр класса *MongoDb*, реализующего интерфейс *IScraper*. Метод *write_reviews* – записывает в базу данных загруженные отзывы.

4.4. Компонент извлечения аспектов

Данный компонент состоит из компонентов создания транзакций и поиска частых аспектов, представленных классами *TransactionsManager* и *AspectsExtractor* соответственно.

Диаграмма класса *TransactionsManager* приведена на рисунке 10. В конструкторе выполняется метод *__parse_config,* в котором происходит инициализация полей объекта значениями из конфигурационного файла *config_file_name.* Необходимые значения обнаруживаются по ключу *application_name.* Поле *mongo* содержит объект, реализующий интерфейс *ITransaction*, для работы с базой данных. Поле *review_docs* хранит доку-

менты из коллекции отзывов, а поле *transaction_docs* – документы транзакций.

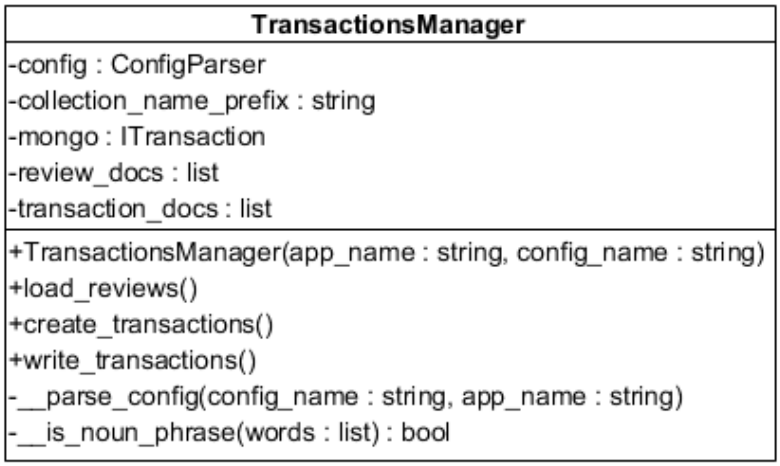

Рис. 10. Диаграмма класса TransactionsManager

Методы *load_reviews* и *write_transactions* используются для загрузки отзывов из базы данных и записи сформированных транзакций в базу данных соответственно. Эти методы переадресуют одноименные вызовы объекту *mongo*.

В методе *create_transactions* выполняется формирование транзакций. Для частеречной разметки был выбран морфологический анализатор *pymorphy2*. Данный инструмент использует словарь *OpenCorpora* и способен приводить словоформы к словарной форме, ставить слово в нужную форму и возвращать грамматическую информацию о слове. Для неизвестных и выдуманных слов строятся гипотезы о их грамматических характеристиках на основе лингвистических свойств русского языка. В транзакцию включались только нормальные формы именных групп. Метод *__is_noun_phrase* осуществляет проверку является ли словосочетание именной группой.

Диаграмма класса *AspectsExtractor* приведена на рисунке 11. При создании экземпляра класса в конструкторе производится инициализация полей из конфигурационного файла. Поле *mongo* содержит объект, реализующий интерфейс *IAspect*. В методе *load_transactions* происходит получе-

ние транзакций через обращение к *mongo* и их сохранение в поле *transaction_docs.*

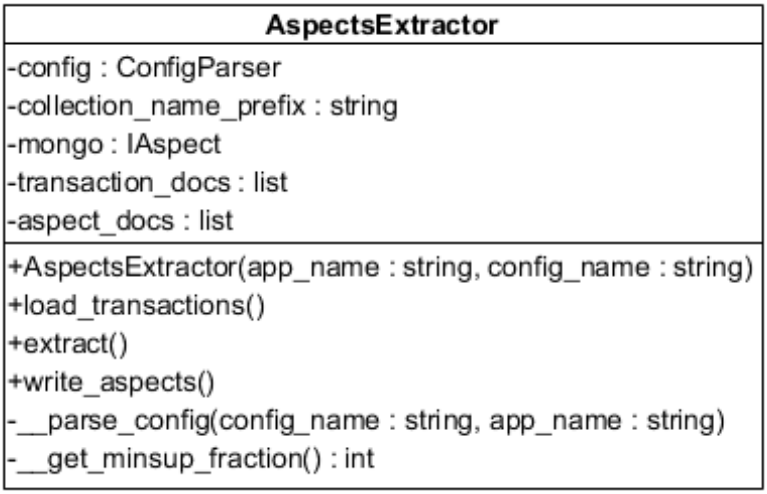

Рис. 11. Диаграмма класса AspectsExtractor

В методе *extract* производится извлечение частых аспектов множества отзывов на основе массива транзакций отзывов. Метод использует алгоритм поиска ассоциативных правил для поиска частых наборов Apriori, который содержится в пакете *mlxtend.frequent_patterns.apriori.* Для его работы данные были преобразованы к специальному виду с помощью пакета *pandas.*

Одним из параметров, настраиваемых работу алгоритма Apriori, является значение минимальной поддержки. По умолчанию значение используемой минимальной поддержки составляет 10% и хранится в базе данных. Однако пользователь может изменить это значение, тем самым влияя на работу алгоритма и количество аспектов, которые будут считаться частыми и фигурировать в итоговом резюме.

4.5. Компонент анализа тональности

Данный компонент состоит из компонентов поиска мнений и определения полярности тональности, представленных классами *OpinionsExtractor* и *SentimentAnalyzer* соответственно.

Диаграмма класса *OpinionsExtractor* приведена на рисунке 12. Класс *OpinionExtractor* на основе аспектов, полученных из базы данных с помощью класса *MongoDb,* реализующего интерфейс *IOpinion,* выполняет поиск мнений для данных аспектов в текстах отзывов.

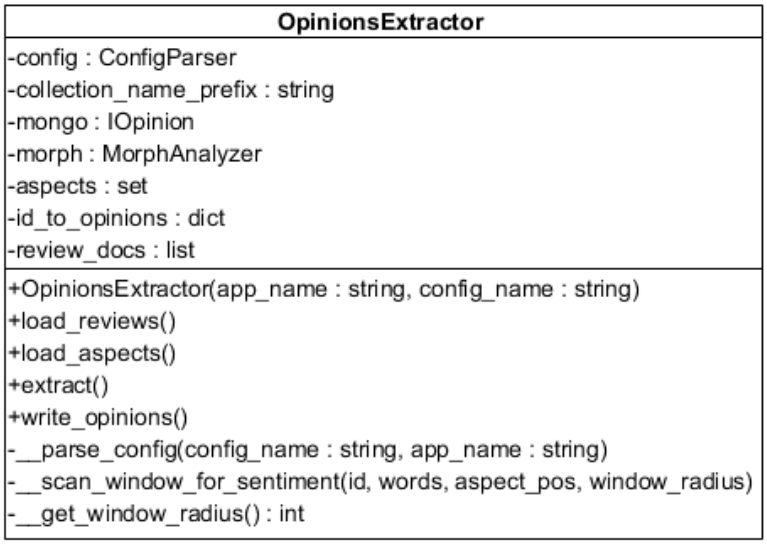

Рис. 12. Диаграмма класса OpinionsExtractor

В методе *extract* происходит последовательная оценка каждого отзыва на предмет наличия выявленных аспектов. Для поиска аспектов с помощью библиотеки *nltk* выполняется токенизация отзывов на уровне предложений и слов. Для найденных аспектов в методе *__scan_window_for_sentiment* происходит поиск ближайшего прилагательного, грамматически согласующегося с данным аспектом, на расстоянии от него не больше, чем значение *window_radius.* Морфологический анализатор *pymorphy2* используется для определения грамматических характеристик слов.

Поле *id_to_opinions* представляет собой словарь, где ключом является атрибут *_id* документа коллекции отзывов, а значением – список объектов мнений. Найденные мнения присоединяются к соответствующим документам коллекции отзывов в методе *write_opinons.*

Диаграмма класса *SentimentAnalyzer* представлена на рисунке 13.

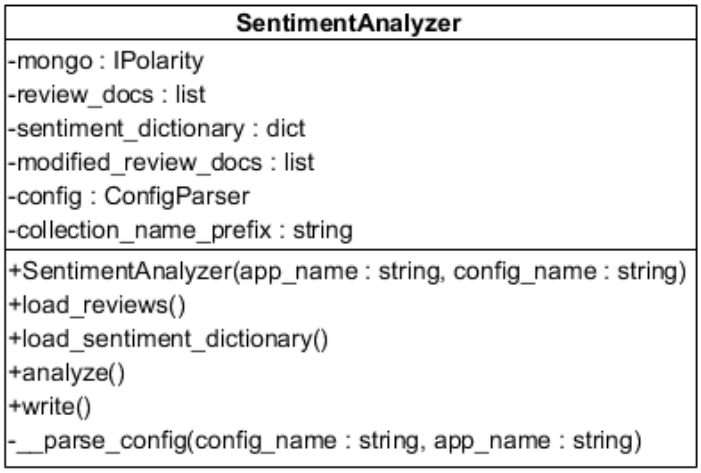

Рис. 13. Диаграмма класса SentimentAnalyzer

В конструкторе производится вызов метода *__parse_config* и инициализация поля *collection_name_prefix.*

В методе *analyze* производится поиск характеризующей аспекты тональной лексики в тональном словаре с целью определить ее полярность. Если искомое слово содержится в словаре, то в документ к соответствующему мнению дописывается полярность тональности.

4.6. Компонент графического интерфейса

Компонент состоит из двух классов – *MainWindow* и *SettingsWindow*. Графический интерфейс реализован с использованием *PyQt5*. Данный пакет представляет собой набор расширений графического фреймворка Qt для языка программирования Python. Он включает в себя большой набор готовых компонентов, из которых строится интерфейс. Приложение на PyQt5 использует событийно-ориентированный подход. В PyQt5 применяется механизм сигналов и слотов, где сигнал, передаваемый в ответ на событие, подключен к слоту, который является обработчиком этого события. Окна реализованного графического интерфейса представлены на рисунке 14 и рисунке 15.

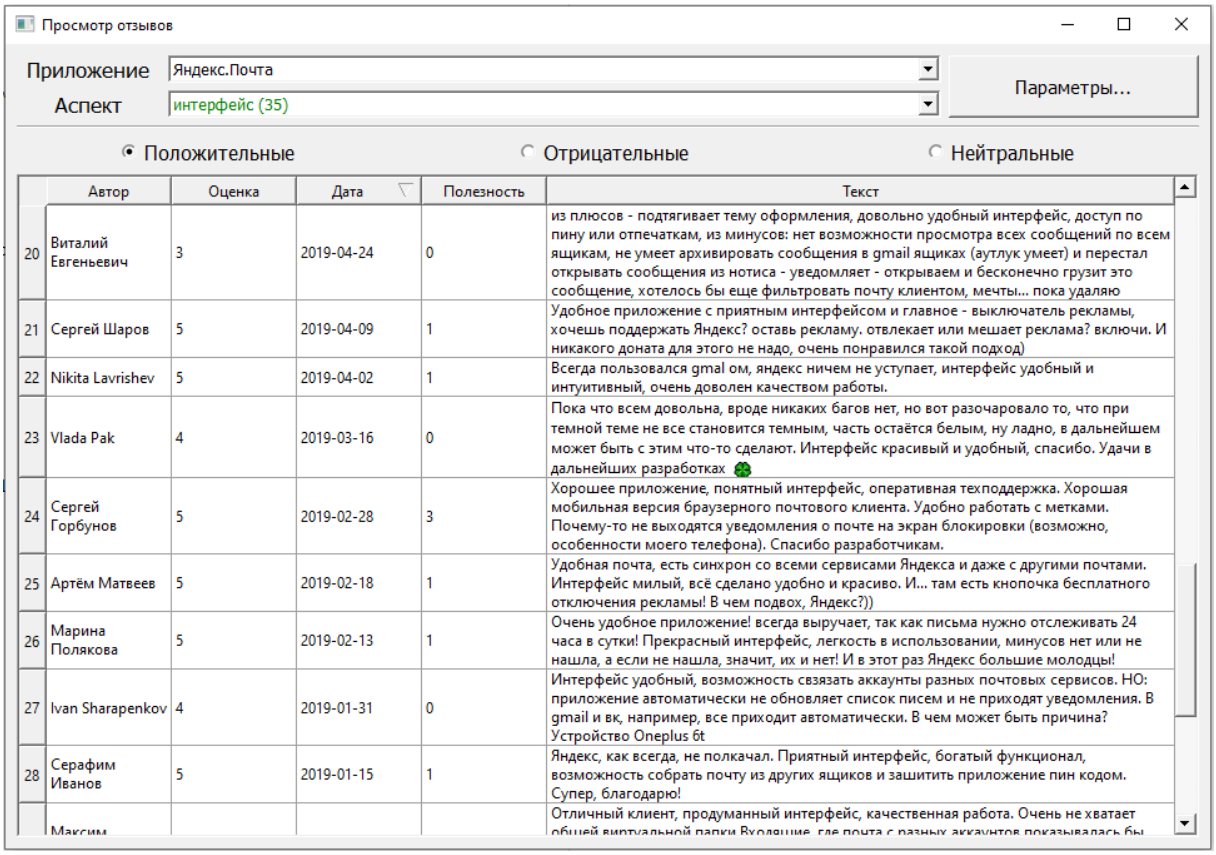

Рис. 14. Окно просмотра отзывов

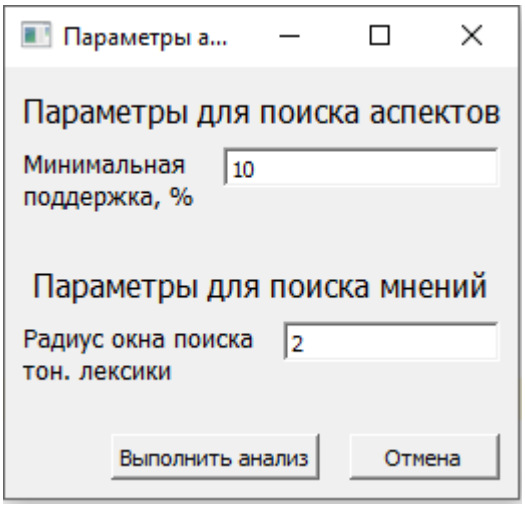

Рис. 15. Окно параметров системы

В верхней части окна просмотра отзывов расположены элементы выбора приложения, аспекта и полярности тональности отзывов. Подходящие под заданные критерии отзывы отображаются в таблице. В таблице содержится информация об авторе отзыва, его оценке приложению, дате публикации отзыва, полезности отзыва и тексте. При нажатии на имя ко-

лонки можно производить сортировку отзывов по возрастанию или убыванию значений в данной колонке. Рядом с названием аспекта содержится число, которые означает суммарное количество отзывов, содержащих положительные и отрицательные мнения для выбранного аспекта. Отзывы, для мнений которых тональность установить не удалось, относятся к группе нейтральных отзывов. Цвет аспекта устанавливается в соответствии с формулой (6), рассмотренной в разделе 3.3.

Окно параметров позволяет пользователю изменять их значения и влиять на поведение системы анализы отзыва. При нажатии на кнопку «Выполнить анализ» поиск частых аспектов, извлечение тональной лексики и определение ее полярности выполняется заново с учетом новых значений параметров. При этом окно параметров закрывается, а на окне просмотра отзывов появляется шкала прогресса, которая уведомляет пользователя о текущем этапе анализа.

Вывод

В данном разделе представлена диаграмма компонентов системы, отражающая ее архитектуру, описана организация хранения данных в базе данных, приведена реализация каждого компонента, разрабатываемой системы.

5. ЭКСПЕРИМЕНТЫ

В данном разделе представлены эксперименты по исследованию эффективности разработанного подхода к анализу отзывов, описанного в разделе 3.

Подготовка данных

Для проведения экспериментов было загружено 500 отзывов для приложения «Яндекс.Почта», опубликованного в Google Play, и вручную проведена разметка аспектов и выраженной по отношению к ним тональности. Размечены были только явно выраженные мнения, то есть те, которые содержат в тексте аспект и положительную или отрицательную оценку по отношению к нему. Если отзыв содержит упоминание аспекта без его оценки, то такие отзывы не размечались.

Данное число отзывов для экспериментов было использовано в работе [5]. Во время разметки данных отзывов было сделано следующее наблюдение. В отзывах люди в основном пишут о возникших у них проблемах, обращаются к разработчикам с просьбой их исправить, либо с предложениями желаемой функциональности для развития приложения. Относительно небольшое число людей детально оценивают аспекты приложения.

Серии экспериментов

Эксперименты проводились следующим образом. На сформированном наборе данных отзывов проводилась оценка подхода к анализу отзывов путем сравнения результатов его работы с размеченными данными.

Были проведены следующие серии экспериментов:

1) зависимость точности и полноты распознавания тональности по отношению к аспектам от значения минимальной поддержки;

2) зависимость точности и полноты распознавания тональности по отношению к аспектам от значения радиуса окна поиска тональной лексики;

3) зависимость количества найденных частых аспектов от значения минимальной поддержки.

Полнота рассчитывается как доля правильно определенной тональности мнений среди всех размеченных мнений по формуле (7).

$$recall = \frac{correctRecognised}{total}, \quad (7)$$

где $correctRecognised$ – количество мнений, тональность которых определена верно;

$total$ – количество размеченных мнений.

Точность рассчитывается как доля правильно определенной тональности мнений среди всех распознанных системой мнений по формуле (8).

$$precision = \frac{correctRecognised}{totalRecognised}, \quad (8)$$

где $correctRecognised$ – количество мнений, тональность которых определена верно;

$totalRecognised$ – количество распознанных системой мнений.

Точность и полнота системы определялись для трех групп аспектов, которые были получены при различных значениях $minsup$. Результаты приведены в таблице 2. Значение радиуса окна равнялось 2.

Табл. 2. Значения точности и полноты в зависимости от мин. поддержки

Минимальная поддержка, %	**Полнота (recall), %**	**Точность (precision), %**
30	50.69	79.61
10	49.30	85.17
3	44.50	83.77

При снижении минимальной поддержки увеличивается количество найденных частых аспектов, что ожидаемо приводит к снижению полноты определения тональности. Точность при этом составляет порядка 80%.

На рисунке 16 представлена зависимость точности и полноты определения тональности от радиуса окна при $minsup = 10\%$. Значение радиу-

са окна задает максимальную удаленность поиска тональной лексики от аспекта. Из графиков видно, что лучшей точности алгоритм достигает при значении радиуса окна равном 1, а лучшей полноты – при значении равном 2. Это можно объяснить тем, что прилагательные, входящие в состав именной группы и зависящие от существительного, расположены вблизи от него.

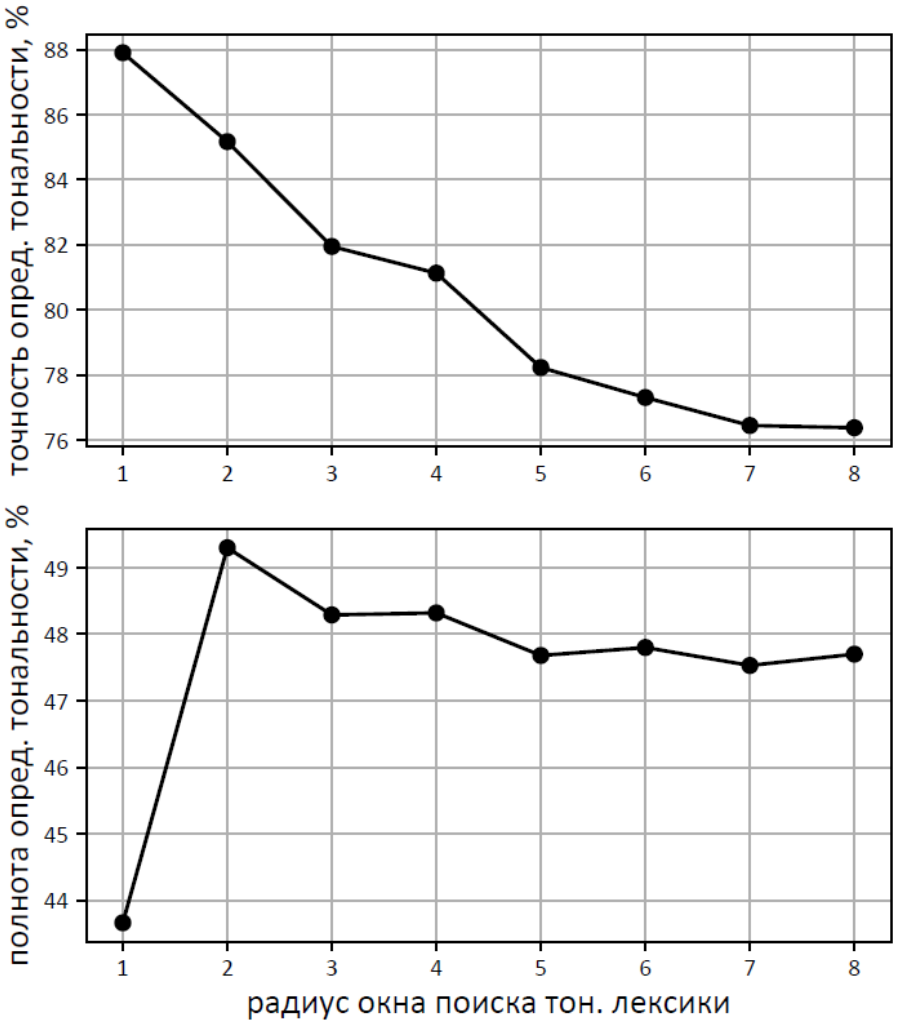

Рис. 16. Зависимость точности и полноты от радиуса окна

На рисунке 17 приведена зависимость количества аспектов от минимальной поддержки. Из графика видно, что для 500 отзывов количество обнаруженных аспектов резко возрастает при $minsup \leq 4\%$. Однако на низких значениях поддержки содержится большое количество часто употребляемых слов, которые аспектами не являются.

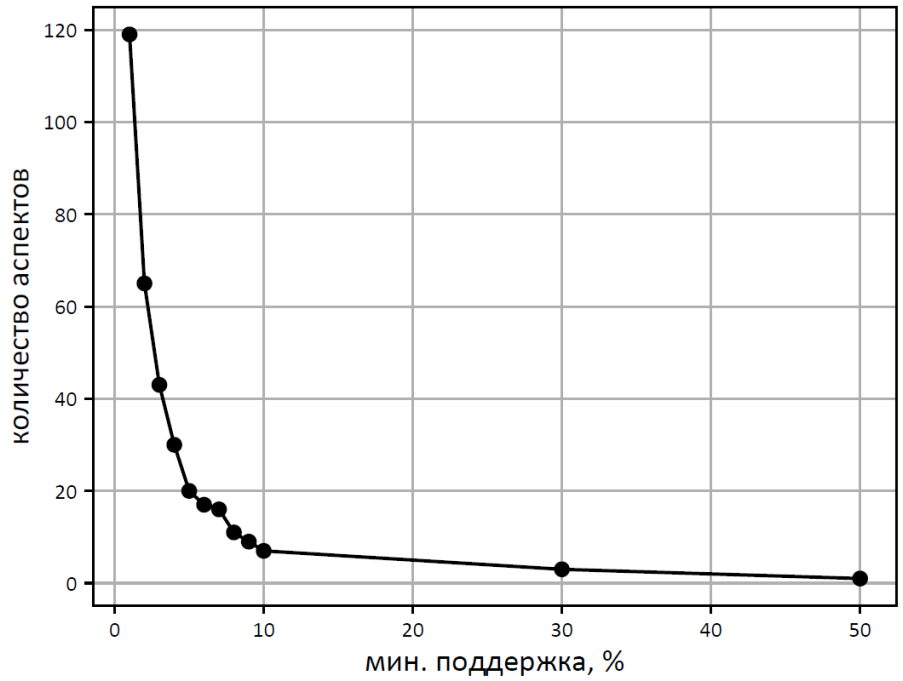

Рис. 17. Зависимость количества аспектов от мин. поддержки

Вывод

Проведенные эксперименты показали, что разработанный подход к анализу отзывов демонстрирует адекватные результаты точности и полноты определения тональности по отношению к аспектам. Видно, что при снижении $minsup$ полнота ожидаемо снижается. Точность при различных значениях $minsup$ составляет порядка 80%. Система способна определять полярность тональности в простых случаях, где отношение автора отзыва к аспекту выражается в виде словосочетания из прилагательного и существительного. Такая конструкция часто встречается в естественной речи.

Ошибки работы системы связаны со сложностями естественного языка, которые включат в себя определение границ групп слов и синтаксических зависимостей между словами, распознавание противопоставлений и сравнений, распознавание иронии, обработку опечаток, омонимию слов.

ЗАКЛЮЧЕНИЕ

В рамках данной выпускной квалификационной работы было разработано приложение для интеллектуального анализа отзывов пользователей магазина приложений Google Play, которое осуществляет поиск частых аспектов, определение тональности по отношению к ним и формирование резюме.

В ходе работы были выполнены следующие задачи.

1. Проведен обзор работ на тему интеллектуального анализа отзывов потребителей в интернете. Рассмотрены работы по направлениям выявления спам-отзывов, определения полезности отзывов и обобщения отзывов.

2. Разработан подход к анализу отзывов. Подход основан на поиске мнений, которые включают в себя аспект и тональную лексику. Аспекты, о которых пользователи часто высказывались в своих отзывах, извлекались с помощью алгоритма поиска частых наборов Apriori. Полярность тональной лексики определялась по тональному словарю, составленному из переведенного на русский язык лексического ресурса WordNet-Affect и РуСентиЛекс.

3. Выполнено проектирование архитектуры системы. Проектирование включает в себя описание модели предметной области, выбор модели организации данных, анализ требований и построение диаграммы вариантов использования системы, создание макета графического интерфейса, построение диаграммы компонентов системы.

4. Реализована программная система. Реализованы компоненты базы данных, управления доступом к базе данных, загрузки отзывов, извлечения аспектов, анализа тональности и графического интерфейса. Загрузка отзывов осуществляется с использованием технологии синтаксического разбора веб-страниц и инструмента для автоматизации действий браузера Selenium. Для обработки отзывов использовались пакеты nltk и pymorphy2. В качестве СУБД использовалась MongoDB. Графический интерфейс реализован на основе пакета PyQt5.

5. Проведены эксперименты, исследующие эффективность предложенного подхода к анализу отзывов. Был сформирован и вручную размечен набор данных из 500 отзывов. Проведена оценка точности и полноты определения тональности по отношению к аспектам при различных значениях минимальной поддержки.

Исходные коды компонентов системы, разработанной в рамках работы, свободно доступны в сети Интернет по адресу: https://github.com/ShumilinPavel/diploma.

ЛИТЕРАТУРА

1. Agrawal R., Srikant R. Fast Algorithms for Mining Association Rules // Proceedings of the 20th International Conference on Very Large Data Bases. 1994. С. 487–499.

2. Akoglu L., Chandy R., Faloutsos C. Opinion fraud detection in online reviews by network effects // Seventh international AAAI conference on weblogs and social media. 2013.

3. Bobicev V. [и др.]. Emotions in words: Developing a multilingual wordnet-affect // International Conference on Intelligent Text Processing and Computational Linguistics. 2010. С. 375–384.

4. Feng S. [и др.]. Recommended or not recommended? Review classification through opinion extraction // 12th International Asia-Pacific Web Conference. 2010. С. 350–352.

5. Hu M., Liu B. Mining and summarizing customer reviews // Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. 2004. С. 168–177.

6. Hu M., Liu B. Mining Opinion Features in Customer Reviews // AAAI. 2004. № 4 (4). С. 755–760.

7. Kim S.-M. [и др.]. Automatically assessing review helpfulness // Proceedings of the 2006 Conference on empirical methods in natural language processing. 2006. С. 423–430.

8. Loukachevitch N., Levchik A. Creating a general russian sentiment lexicon // Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). 2016. С. 1171–1176.

9. Maharani W., Widyantoro D.H., Khodra M.L. Aspect-based opinion summarization: a survey // Journal of Theoretical and Applied Information Technology. 2017. № 2 (31).

10. Miller G.A. WordNet: An electronic lexical database / G.A. Miller, MIT press, 1998.

11. Ott M. [и др.]. Finding deceptive opinion spam by any stretch of the

imagination // ACL-HLT 2011 - Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. 2011. (1). С. 309–319.

12. Popescu A.-M., Etzioni O. Extracting product features and opinions from reviews // Natural language processing and text mining. 2007. С. 9–28.

13. Savage D. [и др.]. Detection of opinion spam based on anomalous rating deviation // Expert Systems with Applications. 2015. № 22 (42). С. 8650–8657.

14. Sharma K., Lin K.-I. Review spam detector with rating consistency check // Proceedings of the 51st ACM southeast conference. 2013. С. 1–6.

15. Strapparava C., Valitutti A. WordNet-Affect: An affective extension of WordNet // Proceedings of the 4th International Conference on Language Resources and Evaluation, LREC 2004. 2004. № January 2004. С. 1083–1086.

16. Zhang Z., Varadarajan B. Utility scoring of product reviews // International Conference on Information and Knowledge Management, Proceedings. 2006. С. 51–57.

17. Лукашевич Н.В., Четвёркин И.И. Построение модели для извлечения оценочной лексики в различных предметных областях // Моделирование и анализ информационных систем. 2015. № 2 (20). С. 70–79.

18. Рой Д.А., Ефремова Н.Э. Методы извлечения аспектных терминов из мнений // Новые информационные технологии в автоматизированных системах. 2018. С. 5–7.

19. Android Developers Blog: In reviews we trust – Making Google Play ratings and reviews more trustworthy [Электронный ресурс]. URL: https://android-developers.googleblog.com/2018/12/in-reviews-we-trust-making-google-play.html (дата обращения: 20.03.2020).

20. Правила публикации отзывов [Электронный ресурс]. URL: https://play.google.com/about/comment-posting-policy/ (дата обращения: 20.03.2020).