

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

на тему:

**Автоматический анализ тональности на материале сообщений о
политических партиях в социальных сетях**

основная образовательная программа магистратуры по направлению
подготовки 45.04.02 «Лингвистика»

Исполнитель:

Обучающийся 2 курса
Образовательной программы
«Прикладная и экспериментальная
лингвистика»
очной формы обучения
Миронюк Елизавета Евгеньевна

Научный руководитель:
к.ф.н., доц. Захаров В.П.

Рецензент:
к.ф.н. Карнуп Е.В.

Санкт-Петербург
2018

Оглавление

Введение	3
Глава 1. Автоматический анализ тональности как задача прикладной лингвистики	7
1.1. Основные понятия анализа тональности	7
1.2. Задачи автоматического анализа тональности	9
1.3. Проблемы автоматического анализа тональности.....	11
1.4. Методы автоматического анализа тональности	15
1.5. Выводы к главе 1	20
Глава 2. Разработка алгоритма автоматического анализа тональности	22
2.1. Создание и характеристика корпуса текстов как материала исследования	22
2.2. Анализ особенностей материала.....	23
2.3. Создание и структура базы данных для хранения корпуса текстов	32
2.4. Предварительная обработка текстов на основе их особенностей	36
2.5. Разработка алгоритма и описание используемого метода	38
2.6. Выводы к главе 2	46
Глава 3. Создание системы автоматического анализа тональности и анализ результатов	47
3.1. Проектирование архитектуры системы и реализация алгоритма	47
3.2. Оценка эффективности системы.....	51
3.3. Разработка веб-интерфейса для дальнейшего взаимодействия эксперта с системой	54
3.4. Анализ результатов и дальнейшее направление исследований	60
3.5. Выводы к главе 3	63
Заключение.....	64
Список использованной литературы.....	67

Введение

С появлением и стремительным распространением таких социальных сервисов Web 2.0, как блоги, социальные сети, вики-проекты, интернет-пользователи получили возможность формировать собственный Web-контент и обмениваться мнениями касательно любых процессов и явлений. Взаимодействие и обмен информацией, доступ к которой можно получить, находясь практически в любой точке мира, происходит почти в режиме реального времени. Так, например, спортивные обозреватели ведут текстовые репортажи матчей прямо с места событий, туристы-блогеры делятся впечатлениями о посещаемых ими странах с аудиторией, представители которой, в свою очередь, также находятся в самых разных местах земного шара и не имеют возможности увидеть всё своими глазами.

Профиль в социальных сетях имеют все компании, организации, знаменитости, политические и общественные деятели, заинтересованные в создании, формировании и поддержании имиджа и репутации. Всё это приводит к образованию огромных массивов текстовых данных, которые увеличиваются в объёмах с каждой минутой, что делает затруднительным какой-либо ручной экспертный анализ и сбор статистики для определения отношения пользователей к тому или иному лицу, событию, продукту и так далее. Для решения подобных задач существует набор автоматизированных методов, объединённых названием «Анализ тональности текста» (англ. «Sentiment Analysis»).

Цель данной работы состоит в разработке и реализации системы автоматического анализа тональности на материале сообщений о политических партиях в социальных сетях.

Сформулированная подобным образом цель определяет ряд стоящих перед нами **задач**:

1. Обзор существующих исследований в области автоматического анализа тональности и определение методов, соответствующих с данным исследованием.

2. Анализ проблем и трудностей, которые существуют в области анализа тональности и могут негативно сказаться на результатах исследования, и поиск возможных решений данных проблем.

3. Создание корпуса, состоящего из сообщений о политических партиях и представляющего материал исследования, и базы данных для хранения корпуса и операций с материалом.

3. Анализ особенностей материала и соответствующая обозначенным особенностям предварительная обработка.

4. Разработка системы автоматического анализа тональности на основе особенностей материала и её реализация в виде программного средства.

5. Проведение экспериментального исследования для определения эффективности работы системы и анализ результатов.

6. Определение дальнейшего направления исследований.

Объектом исследования являются сообщения о политических партиях в социальных сетях, представленные в виде неструктурированных текстов на естественном языке, тогда как **предметом** выступает тональность данных сообщений.

Настоящее исследование проводится на материале сообщений из социальных сетей. Социальная сеть — онлайн-сервис, созданный на платформе Web 2.0, структура которого представляет собой граф: в узлах (вершинах) графа находятся пользователи, каждый из которых имеет профиль с определённой информацией, а рёбрами являются связи между пользователями. Примерами наиболее популярных социальных сетей служат: Facebook¹, Instagram², Twitter³, ВКонтакте⁴. **Материалом** для исследования

¹ <https://www.facebook.com/>

² <https://www.instagram.com/>

³ <https://www.twitter.com/>

послужил корпус сообщений (постов), отобранный по определённым параметрам из базы постов социальной сети Твиттер (англ. *Twitter*). Пост в Твиттере также называют твит. Данная социальная сеть была выбрана на основании того, что Twitter предоставляет официальный доступ к базе сообщений через API⁵ (Application Programming Interface — интерфейс прикладного программирования).

Данное исследование проводится на материале сообщений о политических партиях, преодолевших пятипроцентный порог на выборах депутатов Государственной думы Федерального собрания Российской Федерации, которые состоялись 18 сентября 2016 года (Единая Россия, ЛДПР, КПРФ, Справедливая Россия). Политическая партия — «общественное объединение, созданное в целях участия граждан Российской Федерации в политической жизни общества посредством формирования и выражения их политической воли, участия в общественных и политических акциях, в выборах и референдумах, а также в целях представления интересов граждан в органах государственной власти и органах местного самоуправления»⁶.

Работа состоит из введения, трёх глав, заключения и списка используемой литературы. В **первой главе** разрабатывается понятийно-терминологический аппарат автоматического анализа тональности, исследуются задачи, проблемы и методы анализа тональности.

Во **второй главе** содержится описание процесса отбора материала и создания базы данных, а также характеристика особенностей материала, на основе которых производится предобработка и разрабатывается алгоритм автоматического анализа тональности.

⁴ <https://www.vk.com/>

⁵ <https://developer.twitter.com/>

⁶ Федеральный закон от 11.07.2001 N 95-ФЗ (ред. от 05.12.2017) «О политических партиях»

В третьей главе рассматривается реализация системы автоматического анализа тональности в виде программного средства, производится оценка эффективности системы, анализируются результаты и определяется дальнейшее направление исследований. Список литературы содержит 54 наименования.

Актуальность исследования определяется стремительным развитием сети Интернет, ростом популярности социальных сетей и увеличением генерируемого пользователями контента, вследствие чего появляется необходимость и возможность исследования общественного мнения и настроений на основе автоматического анализа и обработки Больших Данных.

Научная новизна работы заключается в разработке оригинального метода, основанного на особенностях текстов малоизученной в контексте анализа тональности предметной области политики.

Практическая значимость состоит в том, что полученные результаты могут применяться при проведении социологических и политических исследований, а также для решения задач интеллектуального анализа текстов.

Глава 1. Автоматический анализ тональности как задача прикладной лингвистики

1.1. Основные понятия анализа тональности

Анализ тональности представляет одну из важных задач прикладной лингвистики, входящих в группу подзадач интеллектуального анализа текстов. **Интеллектуальный анализ текстов (ИАТ, англ. text mining)** — «класс лингвистических и статистических методов, направленных на решение задач извлечения, моделирования, структурирования и анализа информационного содержания текстовых данных для бизнес-анализа, разведочного анализа данных, научных исследований и др.» [Feldman, Sanger 2006: 2].

Термин «анализ тональности» является переводом английского словосочетания «**sentiment analysis**» на русский язык. В литературе можно встретить и другой вариант перевода, который представляет собой частичную транслитерацию англоязычного термина и звучит как «сентимент-анализ». Понятие «sentiment analysis» впервые появляется в 2003 году в работе [Nasukawa, Yi 2003: 71], где определяется как «задача извлечения эмоций автора, выраженных в положительных или отрицательных комментариях, вопросах, просьбах или требованиях, с помощью анализа большого массива данных». В исследовании [Pang, Lee 2008: 10] звучит следующее определение: sentiment analysis — «это набор методов, техник и инструментов, направленных на выявление в тексте и извлечение субъективной информации, например, мнения или отношения». В отношении равнозначности терминов «анализ тональности» и «**извлечение (анализ) мнений**» нет единогласного решения. Некоторые исследователи, в частности Bing Liu [Liu 2017:12], считают, что два вышеприведённых термина не являются синонимами, но оба относятся к подзадачам ИАТ. В таком случае анализ мнений связывается с извлечением суждения, точки зрения или заявления, основанного на субъективной интерпретации автором

объективных фактов, тогда как анализ тональности работает только с определением полярности слова, фразы, предложения, текста или документа. В исследовании [Wiegand et al. 2010: 334] sentiment analysis определяется как «задача автоматического извлечения и классификации мнений, выраженных в тексте на естественном языке».

В рамках настоящего исследования было выработано следующее рабочее определение: **анализ тональности (сентимент-анализ, анализ мнений)** — группа задач, методов и инструментов интеллектуального анализа текстов, направленных на определение, извлечение, измерение аффективных состояний (в значении психических состояний, характеризующихся эмоциональной окрашенностью: эмоциональные состояния, состояние аффекта, настроение и т.д.) и эмоциональной оценки автора, выраженных в текстах на естественном языке, и классификацию текстов на основе выявленной оценки.

Объектом тональной оценки является предмет, явление, событие или персона, на которые направлено данное оценочное суждение. **Субъектом** тональности является автор данного суждения. Например, в тексте сообщения: «ЛДПР — единственная партия, за которую стоило проголосовать» — объектом выступает ЛДПР («Либерально-демократическая партия России»), а субъектом — пользователь, который написал данное сообщение.

Тональность текста может быть **положительной** (указывает на состояние счастья, радости, удовлетворённости и т.д. со стороны автора текста), **отрицательной** (указывает на состояние печали, недовольства, сожаления, отвращения и т.д. со стороны автора текста) или **нейтральной**, если в тексте не содержится никаких эмоционально окрашенных слов. Положительная и отрицательная тональность также называется **полярностью**. Тональной оценке, как правило, присваивается значение $[-1;1]$, основанное на степени положительности или отрицательности. Некоторые исследователи, в частности Бо Панг [Pang, Lee 2005] и

Бенджамин Снайдер [Snyder, Barzilay 2007], предлагают расширить бинарную шкалу до шкалы, принимающей более широкий диапазон, например, от -10 до 10, или оценивать тональность по n-балльной шкале (3-х, 4-х или 5-ти).

Автоматический анализ тональности часто представляет собой задачу классификации, где некоторую совокупность изучаемых объектов (текстов, предложений, слов) необходимо разделить по соответствующим классам (в зависимости от выбранной шкалы, например, класс положительных и класс отрицательных). Алгоритм, выполняющий классификацию на основе автоматического анализа тональности, называется **классификатором**.

1.2. Задачи автоматического анализа тональности

Целью анализа тональности является выявление эмоциональной оценки и аффективных состояний, выраженных в тексте. Какие свойства или элементы оценки интересуют исследователя или заказчика, зависит от решаемой задачи.

Задачи анализа тональности представляется возможным разделить на следующие группы:

1.2.1. Распознавание эмоций и определение полярности

Двумя основными задачами анализа тональности являются **распознавание эмоций** (emotion recognition) и **определение полярности** (polarity detection). Тогда как первая фокусируется на приписывании единице анализа (слову, предложению, тексту) эмоционального тега (радость, раздражённость, грусть и так далее), последняя обычно представляет собой задачу бинарной классификации, где выходными данными будут являться два класса: «положительный» или «отрицательный». Две вышеуказанные задачи являются до такой степени взаимосвязанными, что в некоторых моделях [Cambria et al. 2012], в которых предсказания тональности делаются на основании эмоциональной окрашенности текста, их принято считать единой задачей. Решение данных задач разрабатывается в работах [Picard

1997], [Calvo, D’Mello 2010], [Zeng et al. 2009], [Schuller et al. 2011], [Pang, Lee, 2008], [Wilson et al. 2005], [Cambria, 2016] и др.

1.2.2. Установление степени полярности и интенсивности эмоции

Дополнительно с распознаванием эмоций и определением полярности может решаться задача **установления степени полярности** (polarity degree classification), а также **интенсивности эмоции** (emotional strength measuring) [Pang, Lee 2005], [Snyder 2007]. Например, тексты могут быть классифицированы по следующей шкале:

- 2 — сильно отрицательный;
- 1 — слабо отрицательный;
- 0 — нейтральный;
- 1 — слабо положительный;
- 2 — сильно положительный.

1.2.3. Выявление субъективности

Следующей задачей является **выявление субъективности** (subjectivity detection). Определить субъективность значит установить, выражает ли данный текст мнение (субъективный текст) или сообщает факты (объективный текст) [Chaturvedi, Cambria, Vilares 2016: 4474]. Иногда субъективность выявляется не на уровне текста, а на уровне предложений, так как текст в целом может состоять из комбинации субъективных и объективных предложений (например, новостная статья, где цитируются мнения других людей).

В работе [Lin et al. 2011] данная задача решается с помощью модели латентного размещения Дирихле (Latent Dirichlet allocation), на основе которой происходит автоматическое распознавание субъективности предложения.

1.2.4. Выявление аспекта оценки

В исследованиях последних лет распространённой задачей стало **выявление аспекта оценки** (aspect detection). Аспект — это атрибут или компонент объекта оценки, например, *экран* (аспект) *телефона* (объект), *сервис* (аспект) *в ресторане* (объект), *качество изображения* (аспект) *камеры* (объект). Когда автор текста выражает своё мнение, часто позитивная/негативная оценка направлена на часть объекта (его аспект), а не на весь объект полностью.

В работе [Poria et al. 2016] используется глубинная свёрточная нейронная сеть⁷, состоящая из семи скрытых слоёв и позволяющая приписать каждому слову в субъективном предложении один из тегов: «аспект» или «не-аспект».

1.3. Проблемы автоматического анализа тональности

В одном из источников 2010 года звучит фраза: «Автоматический анализ тональности показывает не большую точность, чем подбрасывание монетки» [Turning conversations... 2010: 9]. С 2010 года многое изменилось — новые технологии и исследования в данной области, а также значительные финансовые вложения способствовали появлению новых, более современных и сложно устроенных методов, которые дают более высокие результаты, но исследователям всё ещё приходится сталкиваться с множеством проблем и трудностей, некоторые из которых так и не имеют однозначного решения.

⁷ <https://habr.com/post/309508/>

1.3.1. Большие объёмы данных

Согласно статистике 330 миллионов пользователей хотя бы раз в месяц отправляют сообщение в Twitter⁸. Facebook, в свою очередь, активно пользуются больше 2 миллиардов людей⁹. Более 97 миллионов человек ежемесячно пользуются ВКонтакте¹⁰. Количество пользователей социальных сетей измеряется миллиардами, а количество операций, совершаемых одновременно, огромно и постоянно растёт. Само понятие «Большие Данные» или «Big Data» предполагает, что входные данные, которые получают алгоритмы обработки, не всегда являются структурированными, что существенно затрудняет их автоматический анализ (см., например, [Моррисон 2010]). Ручной же анализ, в свою очередь, невозможен из-за огромных объёмов.

1.3.2. Зависимость от тематической области и контекстуальная обусловленность

Существует множество лексических единиц, тональность которых может изменяться в зависимости от тематической области. Например, слово формализм согласно Большому толковому словарю под ред. Кузнецова имеет два значения¹¹:

- 1) Соблюдение внешней формы в ущерб существу дела.

Бюрократический ф.

⁸ Twitter Announces Third Quarter 2017 Results, 2017. Режим доступа: <https://www.prnewswire.com/news-releases/twitter-announces-third-quarter-2017-results-300543850.html>. Дата обращения: 28.10.2017.

⁹ Number of monthly active Facebook users worldwide as of 1st quarter 2017 (in millions), 2017. Режим доступа: <https://www.statista.com/statistics/264810/number-of-monthly-active-facebook-users-worldwide/>. Дата обращения: 28.10.2017

¹⁰ Аудитория ВКонтакте. Режим доступа: https://vk.com/page-47200925_44240810. Дата обращения: 28.10.2017

¹¹ Большой толковый словарь русского языка / Ред. Кузнецов С.А. — СПб.: 2000. — С. 535.

2) Направление в искусстве, эстетике и других гуманитарных науках, отдающее первенствующее значение форме, внешнему выражению. *Ф. в литературоведении.*

Таким образом, первое значение лексемы «формализм» при автоматическом анализе тональности должно быть отрицательным, тогда как второе значение не подразумевает эмоциональной окрашенности.

При анализе гетерогенных данных изначально требуется установить, к какой теме относится каждый документ. С этой задачей помогают справиться методы тематического моделирования. Тематическое моделирование — «способ построения модели коллекции текстовых документов, которая определяет, к каким темам относится каждый из документов» [Коршунов, Гомзин 2012].

Некоторые слова в зависимости от контекста могут обладать двойственной полярностью. Например, имеется два предложения: «Кандидат *X* непредсказуемый» и «У фильма *Y* непредсказуемая концовка». В первом случае прилагательное «непредсказуемый» несёт в себе негативную оценку, тогда как «непредсказуемая концовка», как правило, является положительной характеристикой для кинематографического произведения.

1.3.3. Эффект обманутого ожидания

Иной раз автор текста специально задаёт контекст только для того, чтобы опровергнуть его в конце. Например, в нескольких предложениях автор ведёт речь о том, что предвыборная кампания определённого кандидата была многообещающей, а завершает сообщение словами «*Но всё это похоже на большой обман*». Несмотря на превалирующее большинство положительно окрашенных лексических единиц, общая оценка данного сообщения должна быть отрицательной только вследствие наличия имеющего решающее значение последнего предложения.

1.3.4. Неоднозначность отрицания

Много проблем для автоматического анализа представляет отрицание. Принято считать, что отрицание меняет полярность на противоположную, но не всегда это так: «Новый смартфон X не красивый, а прекрасный!». Отрицание может быть выражено имплицитно, без эксплицитного использования отрицательной частицы. Например, глагол «отказаться» несёт в себе отрицательное значение, и его семантика равна «не согласиться». Отрицание может быть компонентом значения целого предложения, не имея при этом отдельного выражения: «Много ты понимаешь» в значении «Ты ничего не понимаешь».

1.3.5. Распознавание сарказма

Задача распознавания сарказма является одной из самых сложных не только для автоматических систем, но иногда даже для человека. Исследователи Иванко и Пэкман [Ivanko, Pexman 2003: 242] определяют сарказм как кортеж, состоящий из 6 элементов: (S, H, C, u, p, p'), где:

S = говорящий;

H = слушающий;

C = контекст;

u = высказывание;

p = буквальная пропозиция;

p' = интенциональная пропозиция.

Кортеж может быть прочитан как «Говорящий S генерирует высказывание u в контексте C, которое буквально означает p, но намерение S состоит в том, чтобы слушающий H понял p'». Таким образом, сарказм меняет значение высказывание на противоположное, например, «Какая чудесная служба поддержки — через три дня перезвонили». Автоматическое распознавание сарказма на данный момент представляет относительно малоизученный объект исследования и трудноразрешимую проблему [Pang, Lee 2008: 90]. В исследовании [Poria et al. 2016] предлагается один из

методов решения данной проблемы на материале сообщений из Твиттера с использованием свёрточных нейронных сетей.

1.3.6. Мультиmodalность текстов

В сообщениях в социальных сетях пользователи имеют возможность размещать не только текстовую информацию, но и аудиальный, визуальный и аудиовизуальный контент в виде изображений, аудио- и видеозаписей, которые дополняют вербальную составляющую, а иной раз даже её замещают. Такой текст ещё называют поликодовым или креолизированным, то есть сообщение в нём закодировано разнородными средствами (вербальными и невербальными) [Сорокин, Тарасов 1990, с.180]. В работе [Poria et al. 2016] представлен подход к анализу новостных видеороликов на основе аудиальных, лингвистических и визуальных признаков, извлекаемых из анализируемого контента. В исследовании [Poria, Chaturvedi 2016] мультиmodalный анализ выполняется при помощи свёрточных нейронных сетей.

1.4. Методы автоматического анализа тональности

В связи с ростом спроса на проведение анализа тональности растёт и количество исследований, а вместе с тем и разнообразие техник, алгоритмов и подходов. В данном параграфе будут рассмотрены некоторые из существующих методов sentiment-анализа, которые можно условно разделить на три группы: лингвистические методы, методы машинного обучения, а также гибридные методы.

1.4.1. Лингвистические методы (knowledge-based)

Подобные методы основаны на поиске эмотивной лексики и базируются на предположении, что общая тональность предложения, текста или документа есть сумма тональностей отдельных лексических элементов. Для определения тональности отдельных слов или фраз заранее создаются тональные словари и прописываются правила, в основе которых лежат

принципы лингвистического анализа. Лингвистические методы используются в работах [Stevenson et al. 2007], [Somasundaran et al. 2008], [Rao, Ravichandran 2009].

Недостатком подобных методов является то, что они представляются трудоёмкими и занимают большое количество времени, однако зачастую методы, основанные на правилах и словарях, показывают более высокую точность.

1.4.2. Методы машинного обучения

1.4.2.1. Методы машинного обучения с учителем

При помощи стратегий машинного обучения с учителем система сначала обучается на конечной совокупности пар «вход-эталонный выход», называемой обучающей выборкой. На основе данных, полученных в ходе обучения, система строит алгоритм, устанавливающий зависимость между входными и выходными данными, который затем может быть применён на наборе новых неизвестных данных, называемых тестовой (или контрольной) выборкой, по которой оценивается качество работы алгоритма.

Краткий алгоритм может быть описан следующим образом:

1. сбор коллекции документов;
2. представление каждого документа в виде вектора признаков;
3. тональная разметка документов;
4. выбор алгоритма классификации и метода для обучения классификатора;
5. использование модели для вычисления тональности документов из новой коллекции, которая не содержит разметки.

Ниже перечислены и описаны некоторые из наиболее известных методов машинного обучения с учителем.

А) Метод опорных векторов (Support Vector Machines, SVM)

Основным принципом работы SVM-классификатора является поиск разделяющей гиперплоскости в n -мерном пространстве, которая

максимизирует расстояние до двух параллельных гиперплоскостей. Алгоритм основывается на предположении, что чем больше расстояние от одной гиперплоскости до другой, тем больше вероятность того, что классифицируемые объекты относятся к разным классам.

В работе [Chien, Tseng 2011] для классификации отзывов о цифровых камерах и MP3-плеерах используются два многоклассовых подхода, существующих внутри данного метода: One-versus-All SVM («один-против-всех») и Single-Machine Multiclass SVM («одномашинный алгоритм для многоклассовой классификации»).

Б) Наивный байесовский классификатор (Naïve Bayes classifier, NBC)

Наивный байесовский классификатор — один из самых простых и наиболее часто используемых классификаторов. Модель использует теорему Байеса и (наивное) предположение о статистической независимости случайных величин:

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)},$$

где $P(c|d)$ — апостериорная вероятность того, что документ d принадлежит классу c ;

$P(d|c)$ — правдоподобие, априорная вероятность встретить документ d среди документов класса c ;

$P(c)$ — априорная вероятность класса c ;

$P(d)$ — априорная вероятность документа d .

Для определения тональности предложения данное уравнение можно преобразовать следующим образом [Tamilselvi, ParveenTaj 2013]:

$$P(\textit{Sentiment} | \textit{Sentence}) = \frac{P(\textit{Sentiment})P(\textit{Sentence} | \textit{Sentiment})}{P(\textit{Sentiment})}$$

При этом $P(\textit{Sentiment} | \textit{Sentence})$ вычисляется как произведение всех $P(\textit{Sentiment} | \textit{Word})$.

В) Метод максимальной энтропии (maxent)

В основе работы классификатора maxent лежит принцип максимальной энтропии, согласно которому наилучшим образом отражает текущее состояние неопределённой среды то распределение вероятностей, которое максимизирует уровень энтропии при заданной информации о «поведении» среды. Из этого следует, что из всех распределений, соответствующих эмпирическим данным, следует выбирать то, которое обладает наибольшей равномерностью и, как следствие, наибольшим уровнем энтропии. Метод предполагает использование размеченной обучающей выборки для извлечения набора ограничений, которые характеризуют специфичные для данного класса ожидания распределения. Таким образом, распределение должно быть не только максимально равномерным, но и подчиняться наложенным ограничениям. На основе данного метода проводятся следующие исследования: [Nigam et al. 1999], [Raychaudhuri et al. 2002]

Г) Метод k-ближайших соседей (классификатор kNN)

Метод ближайших соседей основан на оценивании сходства объектов. Согласно данному методу, объект принадлежит тому классу, который является наиболее распространённым среди его соседей — k ближайших объектов из обучающей выборки, классы которых уже известны. Далее была разработана модификация данного метода, которая называется «метод взвешенных ближайших соседей» [Srivastava, Singh 2014]. Для реализации данного метода каждому элементу обучающей выборки присваивается вес, который в дальнейшем используется для подсчёта тональности текста.

1.4.2.2. Методы обучения с частичным привлечением учителя (semi-supervised learning) и без учителя (unsupervised learning)

Главная задача классификации текстов — распределение документов по заданному числу заранее определённых категорий. Для выполнения данной задачи при использовании методов обучения с учителем необходимо наличие большого количества размеченных документов, составляющих

обучающую выборку. При выполнении классификации иногда сложно составить такой корпус из предварительно размеченных документов, так как это представляет собой трудоёмкую задачу, отнимающую много времени. Методы обучения без учителя (unsupervised learning) помогают преодолевать подобные трудности, так как их применение не требует наличия заранее размеченной обучающей выборки. Таким образом, обучение без учителя — это вид машинного обучения, при котором известно только множество объектов, а задачей является обнаружение внутренних взаимосвязей, закономерностей, зависимостей между объектами, тогда как при обучении с учителем для каждого обучающего объекта задаётся «правильный ответ» и требуется найти зависимость между объектом и ответом.

Обучение с частичным привлечением учителя (semi-supervised learning) — вид машинного обучения, при котором используется небольшое количество размеченных данных, но большая часть данных не содержит разметки.

Во многих исследованиях используются подобные методы (см., например, [Ko, Seo 2000]). Авторы данной работы предлагают алгоритм, который разбивает документ на предложения, определяет категорию каждого предложения, используя списки ключевых слов для каждой категории и коэффициент сходства предложений (sentence similarity measure).

Обучение с частичным привлечением учителя реализуется в работе [Yulan, Deyu 2011]. В данном исследовании классификатор предварительно обучается на размеченных данных, извлечённых из словаря тональной лексики. Подобные первоначальные данные используются в качестве признаков, чтобы наложить ограничения на предсказания модели, полученные при работе с неразмеченными данными с использованием обобщённых критериев математического ожидания. В данной работе исследователям удалось идентифицировать тональные слова, специфичные для определённой тематической области и доказать тем самым идею того, что полярность слов может меняться при изменении области их применения.

Существуют и другие методы машинного обучения без учителя или с частичным привлечением учителя, которые зависят от семантической ориентации [Turney 2002] или лексических ассоциаций и используют меру ассоциации PMI (англ. Pointwise Mutual Information — «поточечная взаимная информация»), связаны с анализом семантических пространств, дистрибутивного сходства [Read, Carroll 2009] и так далее.

1.4.3. Гибридные методы

Гибридные методы сочетают в себе техники, предлагаемые методами, основанными на правилах и словарях, и методами машинного обучения. Например, в [Cambria, Hussain 2015] для определения полярности текста используется комбинация знаний о лингвистических моделях и статистических методов. Исследование [Dragoni, Tettamanzi, da Costa Pereira 2014] основано на использовании данных из таких семантических сетей, как WordNet, ConceptNet и SenticNet, которые требуются для извлечения ключевых концептов предложения. iFeel [Araújo et al. 2014] — это система, которая позволяет пользователям создавать свою собственную структуру, предназначенную для анализа тональности, комбинируя данные из SenticNet, SentiWordNet и другие методы sentiment-анализа. В других исследованиях также предлагается совместное использование методов, основанных на знаниях, и машинного обучения для анализа тональности твитов [Bravo-Marquez, Mendoza, Poblete 2014], классификации коротких текстовых сообщений [Gezici et al. 2013], а также извлечения мнений на основе фреймов [Recupero et al. 2014].

1.5. Выводы к главе 1

В первой главе определены основные термины и понятия, которые используются в данном исследовании; рассмотрены задачи, которые могут стоять перед исследователем при реализации алгоритмов тонального анализа; проанализированы проблемы, возникающие при автоматическом анализе; приведена краткая характеристика методов автоматического

определения тональности, которые делятся на следующие группы: методы, основанные на знаниях, методы машинного обучения (с учителем, с частичным привлечением учителя и без учителя) и гибридные методы.

Глава 2. Разработка алгоритма автоматического анализа тональности

2.1. Создание и характеристика корпуса текстов как материала исследования

Материалом для данного исследования послужил корпус твитов, отобранный с использованием Twitter API¹² по следующим параметрам:

1. Наличие в тексте твита ключевого слова:

А) названия партий, преодолевших пятипроцентный барьер на выборах в Государственную Думу Федерального Собрания Российской Федерации VII созыва, состоявшихся 18 сентября 2016 года: «Единая Россия», «КПРФ», «ЛДПР», «Справедливая Россия», а также все падежные формы («Единой России», «Единую Россию» и так далее);

Б) фамилии лидеров вышеперечисленных партий: «Медведев», «Зюганов», «Жириновский», «Миронов», а также все падежные формы («Медведева», «Медведеву» и так далее);

2. Временной промежуток: 13.09.2016 — 23.09.2016. Данный диапазон выбран, исходя из предположения, что наиболее активно пользователи сети Twitter будут выражать своё мнение относительно политических партий за несколько дней до официальной даты парламентских выборов, в день выборов (18.09.2016), а также несколько дней после официальной даты выборов;

3. Географическое положение пользователя: Российская Федерация. Twitter API не имеет функции фильтрации твитов по языку, поэтому данный параметр был выбран для того, чтобы в корпус попало меньше текстов на языках, отличных от русского.

В итоге объём корпуса составил 74 817 твитов или 1 031 321 словоупотреблений.

¹² <https://developer.twitter.com/>

В Таблице 1 приводится распределение сообщений из корпуса в соответствии с ключевыми словами, по которым производился поиск.

Таблица 1 — Распределение твитов по ключевым словам

Ключевое слово / фраза	Количество твитов	% от объёма корпуса
Единая Россия	26 469	35.38
Медведев	26 238	35.07
КПРФ	10 163	13.58
Зюганов	2 210	2.95
ЛПДР	8 493	11.35
Жириновский	6 300	8.42
Справедливая Россия	1 715	2.29
Миронов	2 382	3.18

Общее количество сообщений из таблицы 1, составляющее 83 943 твита, превышает объём корпуса, так как разные ключевые слова могут встречаться в тексте одного твита:

но ЛДПР спойлер Путина, а Жириновский только иногда говорит правду под действием наркоты

Необходимо указать, что в корпус попали «нежелательные» твиты. Например, по ключевому слову «Миронов» в результат выдачи попадает не только лидер партии «Справедливая Россия», но и актёры Евгений и Андрей Мироновы. Также в результат выдачи попали твиты на отличных от русского языках. Для решения данной проблемы в дальнейшем предполагается разработка алгоритма фильтрации «нежелательных» твитов, но на данном этапе наличие таких твитов несущественно влияет на результаты.

2.2. Анализ особенностей материала

В ходе данного исследования были выделены следующие особенности сообщений в социальной сети Твиттер, на которые следует обратить внимание при дальнейшей работе с материалом: специфические особенности текстов в связи с их отнесением к предметной области политики, строго

ограниченная длина сообщений, отсутствие контекста, наличие в тексте эмодзи и стикеров, особые виды взаимодействия пользователей и твитов через упоминания, хэштеги, ретвиты, а также сокращения, намеренное искажение написания, авторская пунктуация.

2.2.1. Специфика текстов предметной области

Специфика политической предметной области определяет частотное использование некоторой группы лексических единиц, характерных для данной предметной области (*агитация, выборы, партия, фальсификация, кандидат, кампания* и т.д.). Более того, подобные лексические единицы могут приобретать особое значение тональности, обусловленное их использованием в текстах данной области. Например, на выборах существует такой метод фальсификации, как «*карусель*». Слово «карусель» в данном значении имеет негативную коннотацию:

и что ? У единой России ещё более жёсткие приемы. Подтасовка результатов выборов ,карусели," и так далее

Недавно начальница (член партии ЕдРа) сманивала жену к участию в карусели. Получила отказ. Так она и куётся победа Единой России.

судя по фото и видео в раше у "единой россии" реально почти нулевая поддержка, раз такие жестокие карусели...

При этом слово «карусель» в своём первоначальном значении «*вращающейся площадки с сиденьями в виде лошадок, лодок и т.п. для катанья*»¹³ не предполагает никакой негативной эмоциональной составляющей.

Слово «*спойлер*» употребляется в текстах в значении партии, которая не претендует на победу, а специально создаётся для того, чтобы отнять часть голосов другой партии. Так отзывались о партии «Коммунисты России», участвовавшей в выборах в Государственную думу 2016 года —

¹³ Большой толковый словарь русского языка / Ред. Кузнецов С.А. — СПб.: 2000. — С. 411.

из-за схожести названия с КПРФ и того факта, что эти две партии оказались рядом в избирательных бюллетенях, избиратели могли по ошибке выбрать не ту партию, за которую хотели проголосовать изначально:

Мне непонятно, кто проплатил спойлер КПРФ - партию Коммунисты России? опять грязные технологии, оттянуть голоса у КПРФ

Глянула в бюллетень - перепутать КПРФ с их спойлерами можно элементарно. Должно быть, многие так и сделают.

Согласно словарю И. Мостицкого¹⁴ слово «спойлер» имеет также следующие значения:

- авто: устройство, которое превращает ламинарный поток воздуха в турбулентный поток (не путать с антикрылом, которое предназначено для создания прижимной силы);
- авиа: тормозной щиток на крыльях;
- преждевременно раскрытая важная информация, которая портит впечатление от произведения и разрушает интригу; иногда — лицо, которое эту информацию преждевременно раскрыло.

Из этого следует, что тональность данного слова является обусловленной употреблением в текстах определённой предметной области: если речь идёт об автомобилях или авиации, то никакой эмоциональной оценки не предполагается.

2.2.2. Длина сообщений

Начиная с момента создания в 2006 году, в Твиттере было установлено строгое ограничение на длину одного сообщения — 140 символов, связанное с лимитом в 160 символов для SMS-сообщений: по замыслу создателей, твит должен быть короче, чем SMS, не переставая при этом быть ёмким и информативным. В 2016 году было решено не включать некоторые элементы веб-коммуникации (например, ссылки, длина которых может достигать 2 083

¹⁴ Универсальный дополнительный практический толковый словарь. И. Мостицкий. 2005-2012 [Электронный ресурс] — <https://mostitsky.universal.academic.ru/5107/спойлер>

символов) в вышеуказанное ограничение. В сентябре 2017 года первоначальный лимит был увеличен до 280 знаков.

В данной работе используется материал, полученный до увеличения лимита, поэтому актуальным остаётся ограничение в 140 знаков с возможностью прикрепления к сообщению ссылок, фотографий, видеороликов и т.д., которые не включаются в 140 знаков. Редко длина твита превышает одно предложение.

2.2.3. *Отсутствие контекста*

Часто пользователи пишут о том, что волнует общественность на данный момент (выборы, футбольный матч, митинги) или о том, что волнует лично самих авторов сообщения (серия сообщений о своём путешествии, отзыв о только что посещённом ресторане). Таким образом, контекст задаётся событиями окружающего мира, на который могут опираться живые люди, благодаря владению информацией и наличию языкового и внеязыкового опыта, но не могут машины. Например, при анализе твита: *«При строительстве дачи Медведева ни одна уточка не пострадала. Держитесь там, всего хорошего»* — исключительно машинными методами, без опоры на некоторые фоновые знания его тональность, скорее всего, будет определена как положительная. Контекст данного сообщения следующий: в 2016 году Алексей Навальный¹⁵ озвучил результаты громкого расследования некоммерческой организации «Фонд борьбы с коррупцией», в ходе которого выяснилось, что Дмитрий Анатольевич Медведев обладает огромной резиденцией, наличие которой держится в тайне. На территории данной резиденции премьер-министр построил отдельный домик для утки, что стало поводом для многочисленных шуток в интернете (например, подобные сообщения: *«Даже уточка имеет свой дом на даче у Медведева, а ты*

¹⁵ Алексей Навальный — общественный деятель, прославившийся благодаря своим расследованиям о феномене коррупции в России и позиционирующий себя в качестве главного оппонента правительству России.

нет!»). Вторая часть твита: «Держитесь там, всего хорошего» является отсылкой к фразе, сказанной Медведевым во время визита в Крым в ответ на жалобу о размере пенсии. В [Белоедова 2017] автор пишет, что данная фраза обросла «шлейфом эмоциональных коннотаций, стала использоваться в СМИ как прецедентный текст, как «мем» для оценки сложных ситуаций, из которых власть не может найти решение». Подобная контекстуальная обусловленность, знания о которой необходимы, например, при декодировании иронии и сарказма, может значительно снизить качество работы классификатора.

2.2.4. Эмотиконы и эмоджи

Пользователи социальных сетей активно общаются с помощью эмотиконов и языка эмоджи. Использование подобных символов заметно сокращает длину сообщения, повышая при этом информативность в отношении оценки и эмоциональности.

Эмотикон (от англ. *emotion* — «эмоция» и *icon* — «символ») — «короткая алфавитно-цифровая последовательность символов, составленная из типографских знаков и условно иллюстрирующая выражение лица или эмоцию автора» [Baum, Egelhof 2017: 1]. В повседневной речи эмотиконы также часто называют смайлами или смайликами, хотя изначальное значение последних было связано только с положительными эмоциями (от англ. *smiley* — «улыбающийся»). Эмоджи (от яп. *e* — «картинка» и *moji* — «знак, символ») — модернизированный вариант эмотикона: графический символ, изображающий не только мимику и жесты, но и объекты, концепты, идеи, действия и так далее [Ibid: 1]. Актуальная на момент сбора материала версия кодировки Unicode 9.0 содержала 2 666 эмоджи, включая различные вариации (пол, цвет кожи) одних и тех же эмоджи [How many emoji... 2017].

2.2.5. Взаимодействие пользователей и твитов через хэштеги, упоминания, ретвиты

При общении в социальных сетях пользователи активно пользуются хэштегами, популярность которых выросла вместе с ростом популярности сети Твиттер. Хэштег (от англ. hash — знак «решётка» и tag — «метка») представляет собой слово или объединение слов без пробела, которому предшествует октокорп (знак «решётка», #), например, #политика, #выборы2016, #каннскийкинофестиваль. Изначальная задача хэштега — облегчения поиска сообщений по интересующей теме. Если один и тот же хэштег используют большое количество пользователей Твиттера, он появляется в «Актуальных темах». Подобным образом хэштеги могут влиять на формирование и популяризацию новых трендов.

Однако не все пользователи ставят хэштеги с целью классификации сообщений для последующего поиска. Хэштеги могут использоваться для создания контекста вокруг данного сообщения, для выражения эмоций и мнения, обозначения иронии или сарказма, тогда такие хэштеги можно использовать при определении тональности текстов. Авторы работы [Ashequl, Riloff 2013] предлагают классифицировать хэштеги в соответствии с той эмоцией, которую в них вкладывают пользователи. Например, к эмоциональному классу «AFFECTION» относятся такие хэштеги, как #loveyou, #sweetheart, #romantic, #missyoutoo и так далее.

Иногда только с помощью хэштега можно определить неявно выраженное отношение автора текста к сообщаемой информации:

*...небольшая дачка #Медведев'а, ну совсем небольшая... #выборы
#Путин #коррупция #единаяроссия #парнас #Мальцев*

*Путин и Медведев заявили о победе «Единой России» на выборах
в Госдуму ... #146% #мафия #тоталитарнаясекта*

В вышеприведённых примерах именно хэштеги (#коррупция #мафия #тоталитарнаясекта) показывают отношение автора к тому, о чём говорится в тексте.

Некоторые пользователи используют хэштеги для графического выделения тех слов, которые кажутся им важными, или пишут весь текст исключительно с использованием хэштегов:

#Подольские #коммунисты #провели #митинг #в #поддержку #кандидатов-#коммунистов #КПРФ #МККПРФ

Хэштег может быть включён в текст и выполнять те же функции, что и обычное слово, предшествовать основному тексту или следовать за ним:

Сергей Миронов принял участие в форуме по проблемам #трудоустройства людей с #инвалидностью

В Феодосии бюджетников сгоняют на митинг «Единой России» #праздник #туризм

#навстречувыборам #единаяроссия Медведев: очереди в детсады необходимо ликвидировать к 2016 году

Для взаимодействия с другими пользователями автор может использовать функцию упоминания. Для этого необходимо вставить в сообщение знак «коммерческое at» (@), за которым следует имя пользователя без пробела. В таком случае @имяпользователя получит уведомление и увидит данное сообщение. Знак @ используется для привлечения внимания конкретного пользователя или для ответа на уже существующее сообщение данного пользователя.

Если пользователь хочет процитировать без изменений какой-либо твит с упоминанием авторства, он делает ретвит или ретвитит нужное сообщение. В таком случае твит имеют возможность увидеть читатели не только автора сообщения, но и того, кто сделал ретвит. Ретвиты не являются самостоятельными публикациями — при удалении исходного твита исчезнут и все ретвиты.

2.2.6. Сокращения, намеренное искажение написания и авторская пунктуация

В интернет-коммуникации пользователи активно используют сокращения, что позволяет ускорить процесс написания, а также вместить необходимую информацию при наличии лимитов на длину сообщения. Классификация сокращений, используемых в онлайн-коммуникации, разрабатывается в работе [Баринова 2007]. Кроме общепринятых сокращений, которые также называются кодифицированными, например, СССР — Союз Советских Социалистических Республик, *т.е.* — то есть, *с.м.* — смотри, в сети представлено огромное множество полукодифицированных и окказиональных сокращений: консонантограммы (пжлст — «пожалуйста»), транслитерация иноязычных аббревиатур (имхо — «ИМНО» в значении «по моему скромному мнению»), сокращения в функции эвфемизмов (ПЖиВ — Партия Жуликов и Воров) и многие другие.

Тексты в социальных сетях также характеризуются большим количеством ошибок и опечаток:

Майцев нацист, Жириновский просто старыйполитик. Один доджен на нарах сидеть, второму просто нужно мягкое кресло

Нарушаются пунктуационные нормы, правила постановки пробелов:

Партии"Единой России" плевать на проблемы россиян,а партии оппозиционеров плевать на россиян.. Госдума-место,где это будет сделано законно..

а_полковник,_у_которого_при_обыске_нашли_почти_9_миллиардов,_о_н_в_партии_единая_россия_состоял?

Знаки пунктуации могут играть важную роль при определении тональности. Например, несколько следующих друг за другом восклицательных знаков уже говорят о том, что сообщение не является нейтральным, а также показывают интенсивность эмоции. То же самое можно сказать и про использование пользователем клавиши «Caps Lock»:

текст, набранный прописными буквами, воспринимается как повышение тона или крик:

ЕДИНАЯ РОССИЯ-ПОЗОР РОССИИ!!!

*ПРИДУРОК ! ТЫ ПОСМОТРИ НА РОЖИ
ПУТИНА,МЕДВЕДА,ЛАВРУХИ,ЖИРИНОВСКОГО КАДЫРОВА,ШОЙГУ !!!*

Служить маркером оценочности может и намеренное искажение написания слова: эрративная лексика часто используются в пейоративной функции. Например, неправильное написание фамилии «Путин» в виде «Путен» говорит о пренебрежительном отношении автора текста к данной персоне:

Сходил проголосовал за Единую Россию, верю Путену!!

В приведённом выше примере сарказм выявляется именно благодаря подобному искажению фамилии.

Из всего вышесказанного следует, что не только вербальная составляющая сообщения определяет его тональность. Для того, чтобы классификатор учитывал невербальный компонент текста, эмоджи, хэштеги, пунктуация и другие особенности должны быть каким-то образом преобразованы в понятный классификатору формат данных, что не всегда представляется возможным.

2.3. Создание и структура базы данных для хранения корпуса текстов

Для последующих операций с твитами в рамках данной работы и дальнейших исследований, в Microsoft SQL Server Express Edition создаётся база данных SentDB, структура которой изображена на Рисунке 1.

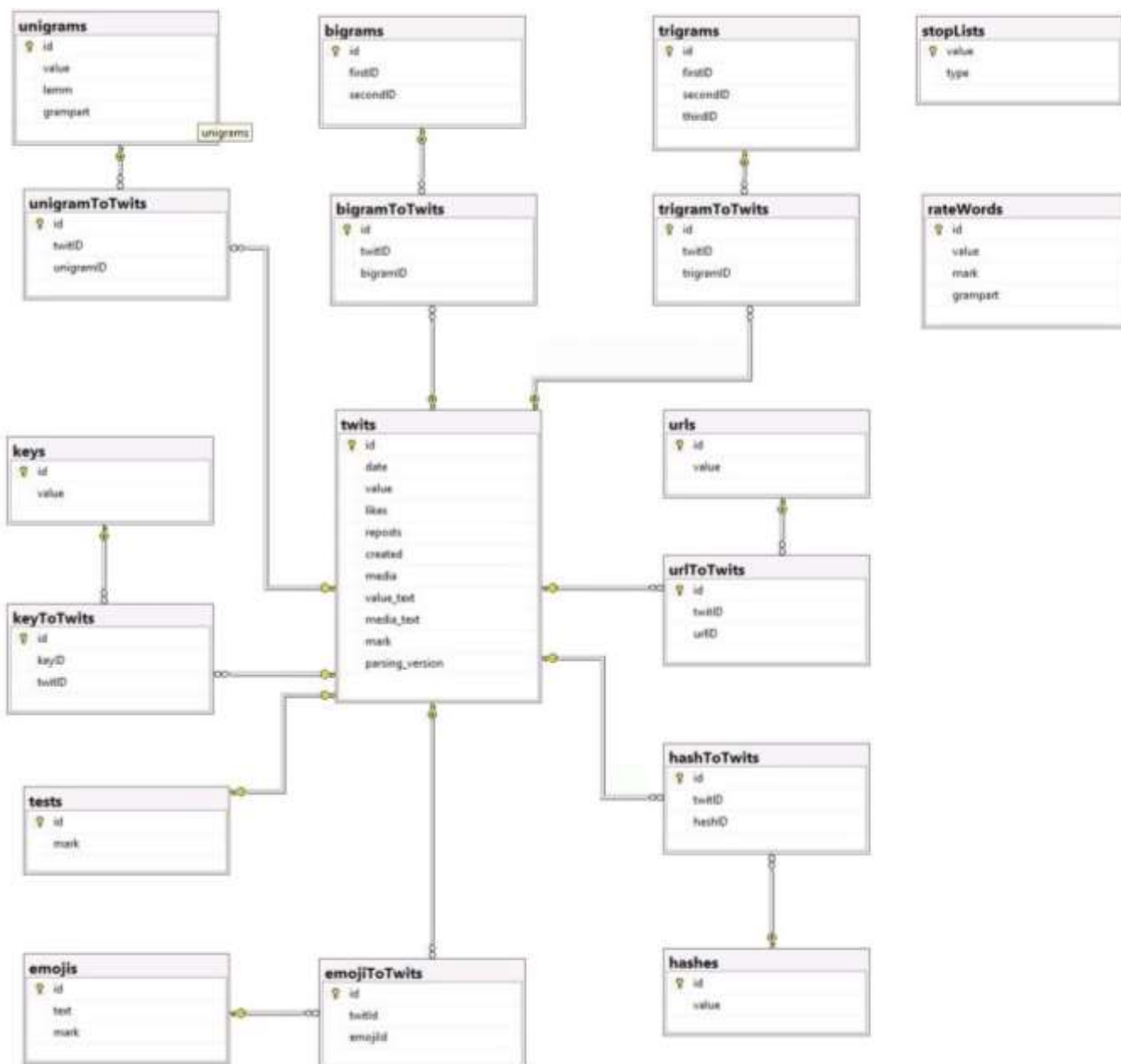


Рисунок 1 — Диаграмма базы данных SentDB

2.3.1. Таблица *Twits*

Главная таблица, в которой содержатся сообщения, отобранные из Твиттера по параметрам, описанным в 2.1 настоящего исследования. Содержит столбцы:

- *id* — уникальный идентификатор твита;
- *date* — дата публикации твита;
- *value* — текст твита в формате HTML;
- *likes* — количество лайков;
- *reposts* — количество репостов;
- *created* — дата добавления твита в базу данных;
- *media* — невербальная составляющая твита (картинки, видео и т.д.);
- *value_text* — вербальная составляющая твита (текст);
- *media_text* — текст, сопровождающий *media*;
- *mark* — тональность твита;
- *parsing_version* — проверочное поле, используемое для контроля версии обработки твита.

2.3.2. Таблицы, связанные с *twits* через *nToTwits* (*unigramToTwits*, *bigramToTwits*, *keyToTwits* и т.д.)

Для того, чтобы связать определённую таблицу с таблицей *twits*, создаются промежуточные таблицы *nToTwits*, где *n* — название исходной таблицы (*unigrams*, *bigrams*, *keys* и т.д.) в единственном числе (*unigram*, *bigram*, *key* соответственно). Например, *unigramToTwits* связывает *unigrams* с *twits* через уникальные идентификаторы униграммы и твита, показывая, что униграмма *id* содержится в твите *id*.

2.3.2.1 Таблицы *n*-грамм

Признаками для автоматического анализа тональности машинными методами могут выступать *n*-граммы (униграммы, биграммы, триграммы и их сочетания), поэтому база данных содержит соответствующие таблицы.

A) Таблица unigrams:

- *id* — уникальный идентификатор униграммы;
- *value* — текст униграммы (лексема);
- *lemm* — основная форма лексемы;
- *grampart* — часть речи униграммы.

Б) Таблица bigrams:

- *id* — уникальный идентификатор биграммы;
- *firstID* — уникальный идентификатор первой лексемы (из таблицы unigrams);
- *secondID* — уникальный идентификатор второй лексемы (из таблицы unigrams).

В) Таблица trigrams:

- *id* — уникальный идентификатор триграммы;
- *firstID* — уникальный идентификатор первой лексемы (из таблицы unigrams);
- *secondID* — уникальный идентификатор второй лексемы (из таблицы unigrams);
- *thirdID* — уникальный идентификатор третьей лексемы (из таблицы unigrams).

2.3.2.2. Таблица keys

Таблица ключевых слов, которые используются для поиска в Твиттере:

- *id* — уникальный идентификатор ключевого слова;
- *value* — значение (например, «Единая Россия»).

2.3.2.3. Таблица tests

Таблица обучающей выборки с размеченными вручную твитами:

- *id* — уникальный идентификатор твита;
- *mark* — тональная оценка, которая будет приписана экспертами.

2.3.2.4. Таблица *emojis*

Таблица с эмоджи, которые встречаются в текстах твитов, представленных в корпусе:

- *id* — уникальный идентификатор эмоджи;
- *text* — текст-описание эмоджи (например, «лицо, кричащее от страха», «улыбающееся лицо со счастливыми глазами», «плачущая кошачья мордочка») согласно рекомендациям CLDR (Common Locale Data Repository — Общий Репозиторий Языковых Данных);
- *mark* — тональная оценка эмоджи.

2.3.2.5. Таблица *hashes*

Таблица хэштегов, которые содержатся в текстах твитов, представленных в корпусе:

- *id* — уникальный идентификатор хэштега;
- *value* — значение хэштега без октоторпа (например, путинизм, социальная деградация, ВВП России).

2.3.2.6. Таблица *urls*

Таблица ссылок, которые содержатся в текстах твитов, представленных в корпусе:

- *id* — уникальный идентификатор ссылки;
- *value* — значение ссылки (в формате URL, например: <http://politicalnews12345.ru/news/putin-i-medvedev>)

2.3.3. Таблица *rateWords*

Таблица, предназначенная для хранения словарей тональной лексики:

- *id* — уникальный идентификатор единицы словаря;
- *value* — значение (например, *абсурдный*);
- *mark* — тональная оценка;
- *grampart* — часть речи;

- *subj* — предметная область, к которой относится данное лексическая единица (например, *политика* или NULL, если тональности слова не меняется в зависимости от предметной области).

2.3.4. Таблица *stopLists*

Таблица для хранения стоп-слов, которые могут создавать «шум» при автоматическом анализе тональности (включая ключевые слова, используемые для отбора твитов в корпус):

- *value* — значение стоп-слова (например, *я, это, чтобы*);
- *type* — тип стоп-слова (1 — «шумовые слова», 2 — ключевые слова).

2.4. Предварительная обработка текстов на основе их особенностей

Использование всех перечисленных в подразделе 2.2 настоящей работы особенностей текстов при разработке системы тонального анализа, представляется трудноразрешимой задачей: различные пользователи по-разному используют эмоджи, прописные буквы не всегда служат маркером эмоциональности, а автокоррекция ошибок и опечаток сама по себе представляет достаточно трудоёмкую отдельную задачу прикладной лингвистики. Тем не менее представляется возможным и целесообразным проведение некоторых этапов предварительной обработки:

1. В тексте графические эмоджи заменяются на текст «*emojN*», где N — уникальный идентификатор соответствующей эмоджи в базе данных.

2. Из текста твита удаляются все ссылки. В базе данных ссылки хранятся в таблице *urls*, связанной с основной таблицей *twits*, чтобы при необходимости была возможность восстановить исходный текст сообщения;

3. Удаляется некириллический текст;

4. Включённые в текст хэштеги преобразуются в обычные слова (изначальный твит: «*Даже с поддержкой #Путин'а: в России упал рейтинг*

#Медведев'а и партии власти», преобразованный твит: «Даже с поддержкой Путина: в России упал рейтинг Медведева и партии власти»);

5. Хэштеги, которые не включаются в текст, удаляются из твита, но хранятся в базе данных в таблице *hashes*, которая связана с основной таблицей *twits* через промежуточную таблицу *hashesToTwits*, так как некоторые хэштеги могут содержать эмоциональную оценку: *#коррупция, #несправедливаяроссия, #позор, #беспредел;*

6. Количество повторяющихся гласных («ну неееееет») сокращается до двух («ну неет»). Предполагается, что интенсивность тональности слов с пролонгированными гласными выше, чем тех же слов в привычном написании, поэтому не производится преобразования к словарной форме;

7. Количество множественных восклицательных знаков («Единая Россия — позор России!!!!!!!!!!») сокращается до двух («Единая Россия — позор России!!») на основе того же предположения, что описано в пункте 6.

8. Создаётся отдельный словарь стоп-слов, который содержится в базе данных в таблице *stopLists*. Стоп-слова или шумовые слова — это слова, которые не являются значимыми (в основном служебные слова: предлоги, частицы, личные местоимения и так далее), а значит, создают «шум» при классификации, поэтому целесообразным считается их из текста удалять. Например, в список стоп-слов включаются следующие лексические единицы: *а, бы, в, для, это, я*. В список стоп-слов попадают также ключевые слова и фразы, по которым производился поиск и отбор твитов. Делается это, исходя из следующего: если в обучающей выборке большинство твитов, содержащих фразу «Единая Россия», будут отрицательными, то и незамеченный твит, содержащий упоминание Единой России, скорее всего, будет классифицирован как отрицательный.

9. Выполняется лемматизация и морфологическая разметка с помощью морфологического анализатора *MyStem*¹⁶ компании «Яндекс»¹⁷.

¹⁶ <https://tech.yandex.ru/mystem/>

2.5. Разработка алгоритма и описание используемого метода

Для того, чтобы успешно добиться поставленной цели, заключающейся в разработке и реализации системы классификации текстов на основе автоматического анализа тональности, был выбран **гибридный метод**. Исследования [Karkaletsis et al. 2010], [Prabowo, Thelwall 2009], [Худякова, Давыдов, Васильев 2011] доказывают, что именно комбинация статистических и лингвистических методов показывает более высокие результаты классификации. В данном исследовании обучается Наивный байесовский классификатор (NBC), а затем вероятности соотнесения твита с данным классом пересчитываются на основе вхождения элементов твита в словари.

NBC был выбран вследствие относительной простоты его реализации. Для обучения Наивного байесовского классификатора требуется наличие обучающей выборки. Как правило, методы машинного обучения с учителем, к которым относится и NBC, показывают более высокие результаты при наличии большой обучающей выборки. При этом тексты обучающей выборки должны удовлетворять нескольким критериям: совпадать по свойствам (язык, длина текстов, функциональный стиль и т.д.) с текстами тестовой выборки, иметь качественную разметку, удобный формат и прочее. На момент проведения исследования в открытом доступе не было представлено подходящей обучающей выборки. Составление своей обучающей выборки представляет трудоёмкую задачу и требует большого количества времени. Тогда было решено составить небольшую обучающую выборку объёмом 2000 твитов, а вместе с NBC применять также метод, основанный на словарях. Метод, основанный на словарях, в свою очередь, предполагает наличие словарей. В ходе анализа особенностей материала было принято решение составить 3 тональных словаря: универсальный словарь тональной лексики, пригодный для анализа текстов любой

¹⁷ <https://www.yandex.ru/>

предметной области, предметно-ориентированный словарь, в котором содержатся те слова, которые характерны для данной предметной области и тональность которых может меняться в зависимости от предметной области, а также тональный словарь эмоджи, которые, как и тональная лексика, могут являться маркерами тональности.

2.5.1. Наивный байесовский классификатор и формирование обучающей выборки

Для обучения Наивного байесовского классификатора требуется наличие обучающей выборки, в которой каждый документ (в данном случае текст твита) d представляется в виде вектора: $d = \{w_1, w_2, \dots, w_n\}$, где w_i — вес i -го термина (1 говорит о наличии термина w_i в документе d , 0 — о его отсутствии). Термином может выступать слово, последовательная комбинация слов, состоящая из n -элементов (n -грамма) и другие элементы документа. Термин является признаком для классификации.

Далее производится экспертная разметка выборки: каждому документу d приписывается класс c (положительный или отрицательный), после чего классификатор обучается на размеченных данных. Из генеральной совокупности объёмом в 74 817 твитов были отобраны 2000 наиболее представительных с точки зрения выраженной в них тональности сообщений (именуемый далее Золотой Стандарт). Разметка Золотого Стандарта производилась вручную тремя экспертами, двое из которых имеют лингвистическое образование. При наличии разногласий между экспертами твит удалялся из Золотого Стандарта, то есть в Золотой Стандарт попали только однозначно размеченные тремя экспертами твиты. Каждый твит получил метку 1, если его тональность была определена как положительная, и -1 при отрицательной тональности. Объём Золотого Стандарта составил 1446 отрицательных твитов и 650 положительных.

Для того, чтобы иметь машиночитаемый формат, текст должен быть преобразован в вектор признаков, которые необходимо определить. В рамках

данного исследования была выбрана модель n-грамм. В области компьютерной лингвистики n-граммой называют последовательность из идущих один за другим n элементов (звуков, слогов, букв, слов). Униграмма — n-грамма, состоящая из одного элемента, биграмма состоит из двух элементов, триграмма — из трёх.

В качестве примера рассмотрим предложение: «Я буду голосовать за ЛДПР». Данное предложение может быть представлено в виде вектора униграмм: [Я; буду; голосовать; за; ЛДПР]; биграмм: [Я буду; буду голосовать; голосовать за; за ЛДПР]; триграмм: [Я буду голосовать; буду голосовать за; голосовать за ЛДПР] и так далее. Также классификатор может работать с комбинацией униграмм и биграмм: [Я; буду; голосовать; за; ЛДПР; [Я буду]; [голосовать за]; [за ЛДПР]].

Далее корпус текстов представляется в виде матрицы слово-текст, где строки обозначают отдельные тексты, а колонки — словарь обучающей выборки (все слова, которые встречаются в корпусе текстов).

Например, даны два текста, представляющие собой простейшую обучающую выборку:

1. «Я буду голосовать за ЛДПР» (Текст 1);
2. «Я буду голосовать за Единую Россию» (Текст 2).

Тогда словарь данной выборки (wDict) будет выглядеть следующим образом:

wDict = {'я'; 'буду'; 'голосовать'; 'за'; 'лдпр'; 'единая'; 'россия'}

Таблица 2 представляет собой матрицу слово-текст для вышеуказанных примеров, в которой указано, встречается ли данный термин (1) в определённом тексте или не встречается (0).

Таблица 2 — Матрица термин-текст

	я	буду	голосовать	за	лдпр	единая	россия
Текст 1	1	1	1	1	1	0	0
Текст 2	1	1	1	1	0	1	1

Таким образом, Текст 1 и Текст 2 представляются в виде бинарных векторов: [1, 1, 1, 1, 1, 0, 0] и [1, 1, 1, 1, 0, 1, 1] соответственно.

В данном случае вес вектора рассчитывается на основе встречаемости слова в тексте: если слово встречается в тексте, его вес равен 1, иначе — 0. Бинарный вектор в таком случае представляется в виде последовательности нулей и единиц. Веса векторов могут рассчитываться и по-другому, например, на основе количества вхождений (частотности) слова в корпусе, но в исследовании [Pang 2002] бинарная функция взвешивания векторов признаётся более эффективной для классификации текстов на основании их тональности методами машинного обучения.

В данном исследовании признаками классификации было решено определить лемматизированные униграммы, эмоджи и хэштеги.

Теперь классификатор может работать с новыми, неразмеченными данными. Результатом такой классификации является вероятность того, что документ d принадлежит к классу c , принимающая значение $[0,1]$. Вероятность рассчитывается на основе теоремы (формулы) Байеса, которая в контексте сентимент-анализа и нашей задачи может быть записана следующим образом:

$$P(c | d) = \frac{P(d | c)P(c)}{P(d)},$$

где $P(c|d)$ — условная вероятность того, что документ d принадлежит к классу c .

Из того, что

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

следует:

$$P(c | d) = P(w_1, w_2, \dots, w_n | c)$$

Модель Наивного байесовского классификатора получила определение «наивная» из-за следующего допущения: предполагается, что все признаки документа d не зависят друг от друга, т.е. позиция термина в тексте не

является важной. Поэтому вероятность $P(w_1, w_2, \dots, w_n | c)$ представляется возможным рассчитать следующим образом:

$$P(w_1 | c_j) \times P(w_2 | c_j) \times \dots \times P(w_n | c_j) = \prod_i P(w_i | c_j),$$

где $P(w_i | c_j)$ — вероятность принадлежности термина w_i к классу c_j , которая вычисляется по следующей формуле:

$$P(w_i | c_j) = \frac{\text{count}(w_i | c_j)}{\sum_{w \in V} \text{count}(w, c_j)},$$

где $P(w_i | c_j)$ определяется как относительная частота термина w_i к общему количеству терминов в классе c_j . В данном случае V — словарь, в котором содержатся все термины всех классов, то есть все слова, эмоджи и хештеги обучающей выборки.

Здесь необходимо учесть, что в случае, если термин не встречается в обучающей выборке, то и вероятность его принадлежности к классу $P(w_i | c_j)$ будет равна нулю. Данный факт негативно скажется на классификации всего документа, так как вероятность принадлежности документа к классу вычисляется как произведение вероятностей принадлежности всех терминов документа, то есть $P(d | c_j)$ тоже будет равна нулю, а это неправильно. Самым простым и типичным решением представляется применение в таком случае аддитивного сглаживания (сглаживания Лапласа, *add-one smoothing*). Идея сглаживания Лапласа заключается в искусственном прибавлении единицы к частоте каждого термина. Таким образом, термины, отсутствовавшие в словаре обучающей выборки, получают не нулевое значение вероятности, что даст возможность причислить документ к какому-либо классу. Тогда формула вычисления вероятности для термина w_i примет следующий вид:

$$P(w_i | c_j) = \frac{\text{count}(w_i, c_j) + 1}{\sum_{w \in V} (\text{count}(w, c_j) + 1)}$$

2.5.2. Словарный метод и составление тональных словарей

Для реализации словарного метода необходимо предварительно составить словари. После их составления каждое слово из текста проверяется на наличие в словаре. Если термин представлен в словаре и его класс совпадает с классом c_j , то вероятности пересчитываются следующим образом:

$$P(w_i | c_j) = \frac{((\text{count}(w_i, c_j) + 1) + k_1 \times G_{c_j}) \times ((k_2 - 1) \times G_{c_j} + 1)}{\sum_{w \in V} (\text{count}(w, c_j) + 1)},$$

где k_1, k_2 — параметры, принимающие любое целочисленное значение, определяемое экспертом. Во избежание путаницы параметр k именуется далее add , параметр k_2 — k ;

G_{c_j} — вес класса, вычисляемый как отношение количества терминов в классе к количеству терминов всего словаря обучающей выборки:

$$G_{c_j} = \frac{\text{count}(w, c_j)}{\text{count}(w, V)}$$

2.5.2.1. Универсальный словарь тональной лексики

В качестве базы для составления словаря использовался имеющийся в открытом доступе «LINIS Crowd SENT — тональный словарь и коллекция текстов с тональной разметкой»¹⁸ объёмом 7546 лексических единиц, тональность которых может принимать значение $[-2; 2]$. Процесс создания данного тонального словаря подробно описан в [Koltsova, Alexeeva, Kolcov 2016] и [Алексеева, Кольцова, Кольцов 2015]. Затем на основе проведённого вручную анализа в словарь были внесены требуемые изменения:

1. В исходном словаре были просмотрены и в некоторых случаях заново размечены нейтральные слова: *содомский* (новое значение: -1), *регрессивный* (новое значение: -1), *драконовский* (новое значение: -1);

¹⁸ <http://linis-crowd.org/>

2. Из словаря были исключены 4764 нейтральных слова (слова, тональности которых было изначально приписано значение 0): *авиация, автор, одежда, репетиция*;

3. Если значение тональности было приписано словам, которые тональными не являются (выражают нейтральную тональность), то такие слова удалялись из общего списка: *аборт* (изначальное значение: -1), *авария* (изначальное значение: -1), *медалист* (изначальное значение: 1);

4. В исходном словаре диапазон тональности составляет [-2;2]. В рамках данного исследования работа идёт с диапазоном [-1;1], поэтому целесообразно привести значения тональности к виду -1 или 1, что и было выполнено;

5. Были исключены слова, тональность которых зависит от предметной области: *оскароносный, автократия, вброс, очередь, фальсификация*;

6. Тональный словарь был пополнен оценочными словами, которые встречаются в текстах твитов и тональность которых не зависит от предметной области.

Общий объём универсального словаря составил 3042 лексические единицы.

2.5.2.2. Предметно-ориентированный словарь тональной лексики

Универсальный и предметно-ориентированный словари хранятся в базе данных в одной таблице *rateWords*. При этом предметно-ориентированная лексика получает специальную помету в столбце *subj*. Делается это для того, чтобы при необходимости иметь возможность проверить работу системы на текстах другой предметной области, подключив только ту часть словаря, которая не имеет особых помет или дополнительно создав другой предметно-ориентированный список. Если лексическая единица относится к политической предметной области, в столбец *subj* вносится метка «П».

Для того, чтобы определить список лексических единиц, обладающих эмоциональным компонентом и отнесённых к политической предметной

области, на первом этапе был взят Золотой Стандарт с уже размеченными значениями тональности. Далее были составлены частотные списки слов в соответствии с их частью речи — учитывались только прилагательные, существительные, глаголы, наречия. Затем из частотных списков были исключены те лексические единицы, которые уже входят в универсальный словарь тональной лексики. Остаток был размечен вручную: каждое слово получило метку -1, если оно относится к классу отрицательных, и 1, если относится к классу положительных. Нейтральные лексические единицы были удалены. Общий объём списка предметно-ориентированной лексики составил 386 лексических единиц.

2.5.2.3. Тональный словарь эмоджи

Для того, чтобы классификатор учитывал эмоджи при определении тональности текста, создаётся словарь эмоджи, который в базе данных выглядит следующим образом: уникальный идентификатор — текстовое название-описание эмоджи¹⁹ — метка класса. На рисунке 2 показаны первые шесть строк словаря, который содержится в базе данных в таблице emoji.

	id	text	mark
1	1	Лицо, кричащее от страха	-1
2	2	Улыбающееся лицо со счастливыми глазами	1
3	3	Лицо со слезами радости	0
4	4	Ухмыляющееся лицо	-1
5	5	Стрелка «Вправо»	0
6	6	Улыбающееся лицо с открытым ртом и в холодном поту	0

Рисунок 2 — Первые шесть строк таблицы emoji

Общий объём словаря эмоджи составил 130 тональных единиц (с учётом нейтральных: 360).

¹⁹ Согласно рекомендациям CLDR (Common Locale Data Repository — Общий Репозиторий Языковых Данных)

2.6. Выводы к главе 2

В главе 2 описывается процесс разработки системы автоматического анализа тональности на основе особенностей материала. В первую очередь, из социальной сети Твиттер по определённым параметрам отбираются тексты, которые будут выступать материалом для проведения исследования, анализируются такие особенности материала, как специфика текстов данной предметной области, лимитированная длина сообщений, отсутствие контекста, наличие в тексте эмоджи и эмоджи, особые виды взаимодействия пользователей и твитов через упоминания, хэштеги, ретвиты, а также сокращения, намеренное искажение написания и авторская пунктуация, которые необходимо учитывать для разработки системы анализа тональности.

Для последующих операций с материалом исследований в Microsoft SQL Server Express Edition создаётся база данных со сложной табличной структурой, в которой хранится корпус текстов.

На основе особенностей текстов производится предобработка текстов.

Также в главе 2 содержится описание алгоритма, разрабатываемого на основе выделенных особенностей материала и гибридного метода, предполагающего обучение Наивного байесовского классификатора и реализацию метода, основанного на словарях. Для обучения Наивного байесовского классификатора создаётся обучающая выборка объёмом 2000 твитов. Для реализации словарного метода формируются 3 тональных словаря: универсальный словарь тональной лексики, предметно-ориентированный словарь тональной лексики, а также тональный словарь эмоджи.

Глава 3. Создание системы автоматического анализа тональности и анализ результатов

3.1. Проектирование архитектуры системы и реализация алгоритма

Языком разработки системы автоматического анализа тональности в данном исследовании выступает C#. Система разрабатывается в Microsoft Visual Studio 2017.

Разработанная нами система автоматического анализа тональности – Sentimentor - состоит из нескольких программных модулей:

3.1.1. Программный модуль *twitoGraber*

Данный модуль создан для агрегирования данных из социальной сети Твиттер с помощью Twitter API и формирования таблицы *twits* в базе данных SentDB.

3.1.2. Программный модуль *wordProcessor*

Модуль *wordProcessor* производит предобработку текстов твитов. В SentDB создаются таблицы: *emojis*, *hashes*, *unigrams*, *bigrams*, *trigrams*, *urls*, а также соответствующие промежуточные таблицы для связи с таблицей *twits* (*emojiToTwits*, *hashToTwits* и т.д.). Более подробно вышеназванные таблицы описаны в подразделе 2.3 настоящей работы.

Далее при помощи морфологического анализатора MyStem, разработанного компанией «Яндекс», формируется таблица *lemms*, в которой содержатся леммы униграмм, а также промежуточная таблица *lemmToUnigrams*, связывающая таблицы *lemms* и *unigrams*.

В рамках данного модуля в таблицу *twits* добавляются следующие строки:

unigramed — вербальная составляющая твита, где каждая словоформа преобразуется в лемму;

tokens — текст твита, где каждый элемент (лемма, эмоджи, хэштег) заменяется на соответствующий ему идентификатор. Идентификаторы 1-

999999 получают леммы, 1000001-1999999 — эмоджи, 2000001-2999999 — хэштеги. Данное преобразование выполняется для увеличения производительности алгоритма;

vectors — текст твита в векторном представлении.

3.1.3. Программный модуль *Sentimentor*

Название модуля *Sentimentor* совпадает с названием системы, так как данный модуль является основным и производит непосредственно классификацию твитов методом, разрабатываемым в данном исследовании, а также подсчитывает эффективность классификатора.

В структуру программного модуля *Sentimentor* входят следующие библиотеки классов:

3.1.3.1. Библиотека классов *twitLIB*

Содержит классы, соответствующие структуре базы данных *SentDB*, для обращения к ней (класс *unigram* обращается к таблице *unigrams*, *hash* — к таблице *hashes* и т.д.). На рисунке 3 изображена диаграмма классов библиотеки *twitLIB*.

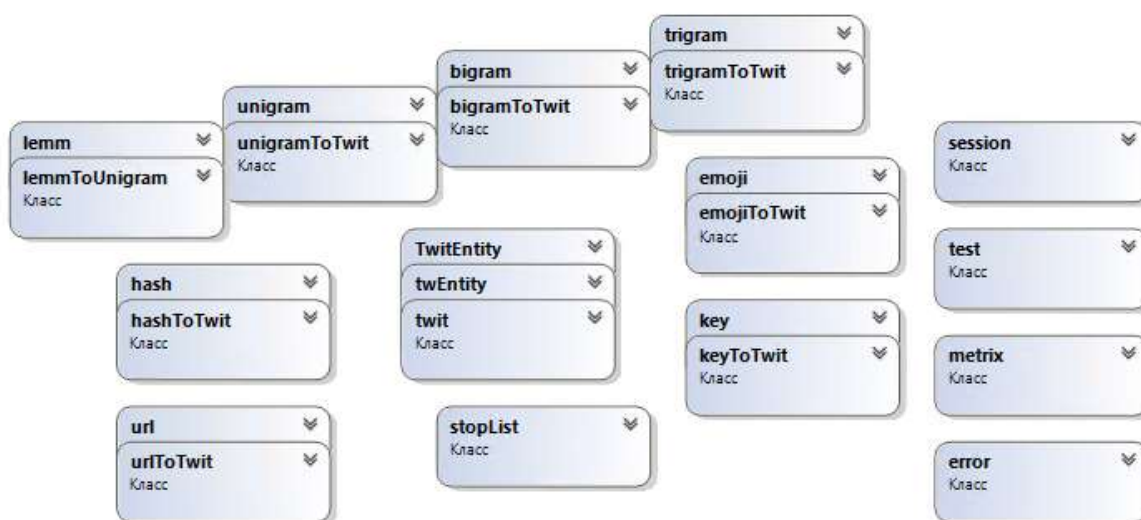


Рисунок 3 — Диаграмма классов библиотеки *twitLIB*

В базе данных SentDB создаются новые таблицы: sessions, metrix, errors, соответствующие классам session, metrix, error:

- *Session* — класс экспертной сессии (ручной разметки твитов). В соответствующую таблицу вносятся следующие данные: уникальный идентификатор сессии, дата, IP-адрес асессора, номер, присвоенный эксперту;
- *Metrix* — класс метрик оценки эффективности классификатора. В SentDB хранятся данные о уже рассчитанных метриках. Более подробно метрики оценки эффективности рассматриваются в подразделе 3.2 настоящего исследования;
- *Error* — класс ошибок (неверно размеченных классификатором твитов).

3.1.3.2. Библиотека классов cSifier

На рисунке 4 приводится диаграмма со структурой данной библиотеки.

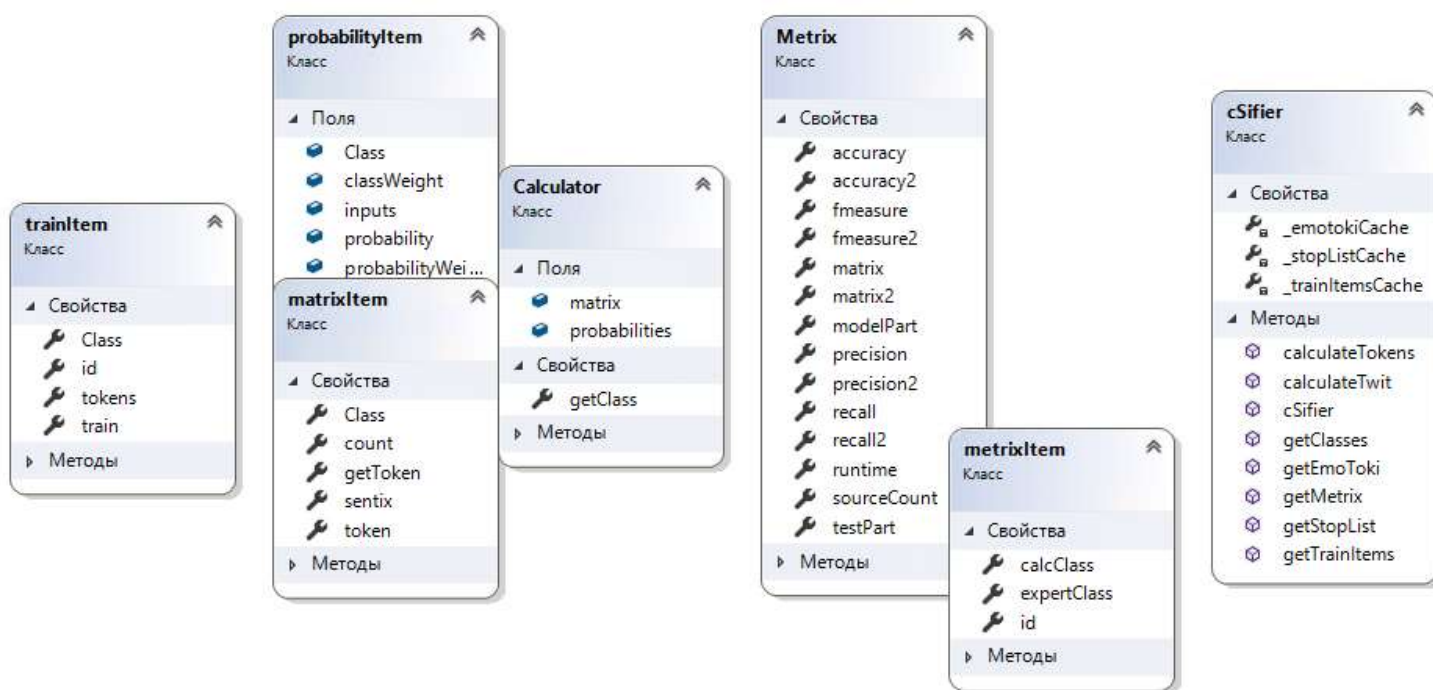


Рисунок 4 — Диаграмма библиотеки классов cSifier

Библиотека классов *cSifier* содержит следующие классы:

- *trainItem* — класс обучающей выборки: содержит уникальный идентификатор твита, класс твита, определённый экспертами, текст твита, представленный в виде последовательности из идентификаторов признаков классификации (униграмм, хэштегов, эмоджи), а также векторное представление твита;
- *probabilityItem* — класс, используемый для расчёта вероятностей. Содержит класс твита, поле вероятности, поле веса класса;
- *matrixItem* — класс матрицы классификации. Содержит идентификатор термина, количество вхождений термина в класс, класс термина из словаря;
- *Calculator* — класс, используемый для расчёта вероятности, который в качестве свойств имеет список *matrixItem* и список *probabilityItem*. Возвращает наиболее вероятный класс из *probabilityItem*, который считается результатом классификации.
- *metrixItem* — класс, используемый при расчёте метрик эффективности. В качестве свойств содержит идентификатор твита, класс твита, определённый экспертом, класс твита, приписанный системой;
- *Metrix* — класс расчёта метрик эффективности. Содержит обучающую и тестовую выборки, списки *metrixItem* для наивного байесовского метода и гибридного метода, поля метрик точности, полноты и F1-меры для каждого метода, время работы алгоритма в секундах;
- *cSifier* — основной класс, в котором производятся расчёты и классификация твита. Для каждого термина создаётся кортеж: (термин, класс, частота встречаемости термина в классе, значение тональности из словаря).

3.1.4. Программный модуль *CoefficientSelection*

Данный модуль производит подбор оптимальных параметров add и k . На первом шаге задаётся возможный диапазон значений параметров, которые модуль будет перебирать, например, $[0;20]$. Далее Золотой Стандарт случайным образом делится на 80% обучающей выборки и 20% тестовой. Классификатор обучается на 80% и затем производит разметку оставшихся 20% твитов, принимая разные значения add и k . На выход модуль выдаёт разность между F-1 мерами, где вычитаемым является F-1 мера для оценки эффективности наивного байесовского классификатора ($add = 0, k = 1$), а уменьшаемым — F-1 мера для классификации с изменёнными параметрами, значение которых меняется на каждой итерации. Чем выше значение разности, тем лучше результат классификации. Разность может принимать отрицательное значение, если вычитаемое больше уменьшаемого, то есть классификатор при данных значениях параметров сработал хуже байесовского. Работа модуля завершается, когда классификатор переберёт все возможные пары add и k .

3.2. Оценка эффективности системы

Традиционно для оценки алгоритмов sentiment-анализа используются такие метрики, как precision (точность), recall (полнота) и F1-мера. Они основываются на предположении, что правильные классы для некоторого числа документов известны заранее. Полнота алгоритма — доля найденных классификатором документов, принадлежащих классу, относительно всех документов данного класса в тестовой выборке. Точность алгоритма — доля документов, которые действительно принадлежат данному классу, относительно всех документов, которые классификатор причислил к данному классу. Данные метрики можно рассчитать на основе таблицы контингентности (Таблица 3).

Таблица 3 — Таблица контингентности для оценки эффективности системы

		Экспертная оценка	
		положительная	отрицательная
Оценка системы	положительная	TP	FP
	отрицательная	FN	TN

В таблице контингентности содержится информации о количестве правильных и неправильных решений системы по документам заданных классов, где:

- tp (true positives) — истинно-положительные решения;
- tn (true negatives) — истинно-отрицательные решения
- fp (false positives) — ложно-положительные решения;
- fn (false negatives) — ложно-отрицательные решения.

Формула для вычисления полноты (R, recall):

$$Recall = \frac{tp}{tp + fn}$$

Точность (P, precision) вычисляется по следующей формуле:

$$Precision = \frac{tp}{tp + fp}$$

F1-мера (F-мера, F) представляет среднее гармоническое между точностью и полнотой, которое стремится к 0, если точность или полнота стремится к 0. F1-мера рассчитывается по следующей формуле:

$$F = 2 \frac{Precision \times Recall}{Precision + Recall}$$

Для оценки качества работы классификатора Золотой Стандарт случайным образом подразделяется на 80% и 20% обучающей и тестовой выборки соответственно. Далее вычисляются точность, полнота и F-мера отдельно для класса положительных (1) и отрицательных (-1) твитов и среднее значение для двух классов (average). Случайное деление и подсчёт метрик выполняется в общей сложности 10 раз, затем на основе 10 итераций подсчитывается среднее значений для каждой метрики.

Сначала были рассчитаны метрики оценки эффективности системы, производящей классификацию только с помощью Наивного байесовского классификатора (NBC). Результаты, представленные в Таблице 4, выступают в качестве отправной точки (baseline) для оценки эффективности системы Sentimentor.

Таблица 4 — Эффективность NBC

Метка класса	Точность	Полнота	F-мера
-1	94,07%	52,22%	67,08%
1	52,56%	81,63%	63,84%
average	73,32%	66,92%	65,46%

С помощью программного модуля CoefficientSelection были рассчитаны оптимальные значения параметров add и k, которые дают максимальный прирост F1-меры. Наилучший результат показали следующие значения параметров: add = 13, k = 16. В таблице 5 приводятся результаты оценки эффективности системы, которая производит классификацию гибридным методом, разрабатываемым в данном исследовании, со значением параметров add = 13, k = 16.

Таблица 5 — Эффективность системы Sentimentor

Метка класса	Точность	Полнота	F-мера
-1	96,43%	82,21%	88,73%
1	76,03%	83,85%	79,69%
average	86,23%	83,03%	84,21%

В Таблице 6 приводится рассчитанный прирост значения метрик эффективности системы Sentimentor по сравнению с baseline.

Таблица 6 — Прирост эффективности Sentimentor по отношению к baseline

Метка класса	Точность	Полнота	F-мера
-1	+2,36%	+29,99%	+21,65%
1	+23,47%	+2,22%	+15,85%
average	+12,91%	+16,11%	+18,75%

Из приведённых выше результатов следует вывод, что гибридный метод, разработанный в данном исследовании, повышает эффективность системы по сравнению с классификацией на основе Наивного байесовского классификатора. При этом средняя точность для обоих классов увеличивается на 12,91%, полнота — на 16,11%, а значение F1-меры вырастает на 18,75%.

3.3. Разработка веб-интерфейса для дальнейшего взаимодействия эксперта с системой

Для облегчения дальнейшего взаимодействия эксперта с базой данных SentDB и повышения эффективности классификатора создаётся веб-ресурс Sentimentor (доступ по адресу: <https://sent.xxonii.com/>).

С помощью данного веб-интерфейса эксперт имеет возможность:

- предложить исправление класса тональности твита, вычисленного классификатором автоматически;
- пополнить Золотой Стандарт, на котором обучается классификатор, размеченными твитами;
- предложить слово-кандидат для пополнения словаря;
- подобрать оптимальные значения параметров α и k ;
- проанализировать значения, на основе которых классификатор производит расчёт вероятностей;
- рассчитать метрики оценки эффективности для разных параметров.

При переходе на главную страницу случайным образом выбирается твит, который имеется в базе данных, но не содержит экспертной разметки. Главная страница имеет блочную структуру. На главной странице имеется вкладка «Метрики», при нажатии на которую эксперт переходит на страницу «Metrix». Далее приводится описание структуры каждой страницы.

3.3.1. Структура главной страницы

3.3.1.1. Блок «Оригинальный твит»

Данный блок (Рисунок 5) содержит текст твита из столбца *value* таблицы *twits*, то есть текст (в формате HTML) в том виде, в котором он представлен в Твиттере. При этом сохраняются все графические элементы, хэштеги, эмоджи и так далее. Под текстом располагается ссылка на твит с уникальным номером и датой публикации.

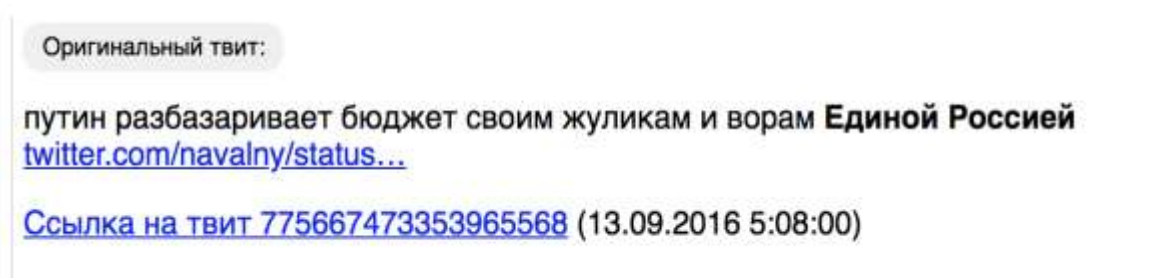


Рисунок 5 — Блок «Оригинальный твит»

3.3.1.2. Блок «Оценка»

Классификатор получает на вход текст твита, предобработанный программным модулем *wordProcessor*, и автоматически определяет его класс, представленный в блоке «Оценка» (Рисунок 6).



Рисунок 6 — Блок «Оценка»

На сайте в блоке оценки эксперт в поле «компьютер» видит результат автоматической классификации и может его исправить, нажав на кнопку соответствующего класса (*NEG* — отрицательный, *ZERO* — нейтральный, *POS* — положительный) в поле «особое мнение», если классификатор

сработал неправильно. В случае, когда результат классификации не противоречит мнению эксперта, эксперт подтверждает автоматическую классификацию путём нажатия кнопки класса в поле «компьютер». При условии, что оценка как минимум трёх экспертов совпадает, твит попадает в базу данных SentDB в таблицу tests, где содержатся размеченные экспертами твиты и становится частью Золотого Стандарта.

3.3.1.3. Блок «Текст твита»

Под блоком «Оценка» находится блок «Текст твита» (Рисунок 7), в котором представлена вербальная составляющая твита из столбца *value_text* таблицы *twits* базы данных SentDB.



Рисунок 7 — Блок «Текст твита»

3.3.1.4. Блок «Подбор параметров»

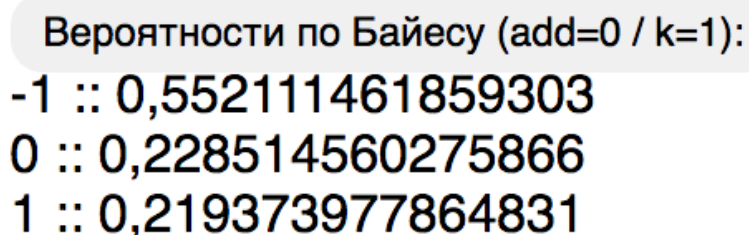
Как уже говорилось ранее при описании метода, для перерасчёта вероятностей с привлечением словарного метода используются два параметра *add* и *k*. Первоначально $add = 0$, $k = 1$, то есть класс твита определяется по формуле Байеса без изменений. Эксперт может задать другое целочисленное значение параметров *add* и *k* в блоке «Подбор параметров» (Рисунок 8).



Рисунок 8 — Блок «Подбор параметров»

3.3.1.5. Блок «Расчёт вероятностей»

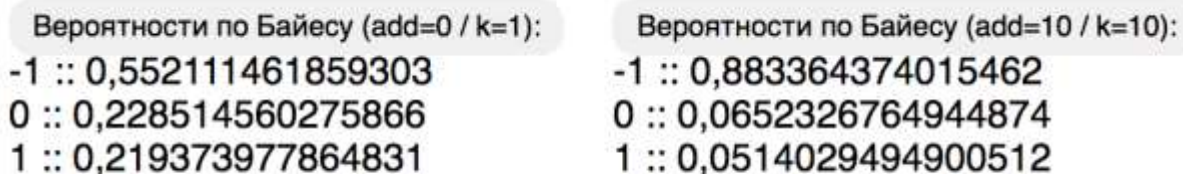
Изначально класс определяется по формуле Байеса без привлечения словарей. Результат байесовской классификации представлен в блоке «Расчёт вероятностей» (Рисунок 9).



Вероятности по Байесу (add=0 / k=1):
-1 :: 0,552111461859303
0 :: 0,228514560275866
1 :: 0,219373977864831

Рисунок 9 — Блок «Расчёт вероятностей»

Если эксперт изменяет значение параметров и нажимает кнопку «Обновить», блок расчёта вероятностей меняется: справа появляется результат при изменённых значениях параметров *add* и *k* (Рисунок 10).



Вероятности по Байесу (add=0 / k=1):	Вероятности по Байесу (add=10 / k=10):
-1 :: 0,552111461859303	-1 :: 0,883364374015462
0 :: 0,228514560275866	0 :: 0,0652326764944874
1 :: 0,219373977864831	1 :: 0,0514029494900512

Рисунок 10 — Блок «Расчёт вероятностей» при изменении параметров *add* и *k*

Ниже располагаются блоки, необходимые эксперту для того, чтобы понять, на что опирался классификатор при расчёте вероятностей.

3.3.1.6. Блок «Униграммы»

Блок «Униграммы» (Рисунок 11) представляет собой таблицу, в которой в первом столбце представлен текст твита, разбитый на униграммы, во втором столбце обозначается часть речи униграммы, в третьем — лемма, а в четвёртом — тональная оценка униграммы в соответствии со словарём. Строка перечёркивается и отмечается серым цветом, если униграмма

совпадает со словом из стоп-листа. При наличии в тексте хэштегов или эмоджи они также представляются в данном блоке.

Униграммы:			
путин	noun	путин	0
разбазаривает	Verb	разбазаривать	-1
бюджет	Noun	бюджет	0
своим	PronounAdjective	свой	0
жуликам	Noun	жулик	-1
и	conjunction	и	0
ворам	Noun	вор	-1
единой	adjective	единый	0
россией	noun	россия	0
...			0

Рисунок 11 — Блок «Униграммы»

3.3.1.7. Блок «Матрица»

В блоке «Матрица» (Рисунок 12) содержится 2 таблицы. Первая таблица представляет матрицу для расчёта вероятностей по формуле Байеса. Первый столбец соответствует леммам тех униграмм, которые не попали в список стоп-слов. Во втором столбце представлены все классы (-1 — отрицательный, 0 — нейтральный, 1 — положительный), по которым производится классификация. Третий столбец соответствует частоте униграммы в твитах определённого класса или *весу униграммы*. В четвёртом столбце представлена тональная оценка униграммы из словаря тональной лексики.

Вторая таблица — это матрица для расчёта вероятностей с изменёнными параметрами add и k . Вторая таблица, обозначенная « C_2 » содержит те же столбцы, что и первая, но при этом меняется значение третьего столбца: здесь показано, как алгоритм пересчитал вес униграммы на основе параметров add и k в соответствии с её встречаемостью в словаре.

Матрица			
разбазаривать	-1	1	-1
бюджет	-1	2	0
свой	-1	24	0
жулик	-1	16	-1
вор	-1	24	-1
разбазаривать	0	0	-1
бюджет	0	0	0
свой	0	1	0
жулик	0	0	-1
вор	0	0	-1
разбазаривать	1	0	-1
бюджет	1	0	0
свой	1	12	0
жулик	1	0	-1
вор	1	0	-1
--=C2=--			
разбазаривать	-1	54,4910144937625	-1
бюджет	-1	2	0
свой	-1	24	0
жулик	-1	160,288589120628	-1
вор	-1	216,713962254957	-1
разбазаривать	0	0	-1
бюджет	0	0	0
свой	0	1,21828358208955	0
жулик	0	0	-1
вор	0	0	-1
разбазаривать	1	0	-1
бюджет	1	0	0
свой	1	12	0
жулик	1	0	-1
вор	1	0	-1

Рисунок 12 — Блок «Матрица»

3.3.2. Структура страницы «Metrix»

На главной странице также представлена вкладка «Метрики», при нажатии на которую происходит переход на страницу «Metrix» (доступ по адресу: <https://sent.xxoniii.com/metrix>) с оценкой эффективности классификатора (Рисунок 13) по всем классам.

METRIX

add=0, k=1

Accuracy = 63,85%

Процент правильных решений системы по классам:
precision(-1) : 93,67%
precision(0) : 18,18%
precision(1) : 55,45%

Процент найденных по классам:
recall(-1) : 54,41%
recall(0) : 100,00%
recall(1) : 78,87%

F-меры по классам:
F(-1) : 68,84%
F(0) : 30,77%
F(1) : 65,12%

Рисунок 13 — Страница оценки эффективности

Алгоритм расчёта подробно описан в 3.2 настоящего исследования. На странице Metrix эксперт может задать значения параметров *add* и *k* и произвести перерасчёт метрик, нажав на кнопку «Пересчитать».

3.4. Анализ результатов и дальнейшее направление исследований

Тестирование системы показало достаточно высокий результат со средним значением F-меры для обоих классов в 84.21%.

В ходе анализа результатов были выявлены задачи, решение которых поможет в дальнейшем улучшить результат классификации. К ним относятся пополнение обучающей выборки и тональных словарей, извлечение именованных сущностей, выявление объекта оценки, распознавание сарказма и иронии, а также выполнение мультимодального анализа.

3.4.1. Пополнение обучающей выборки и тональных словарей

На момент проведения исследования обучающая выборка имеет небольшой объём в 2000 твитов, тогда как в базе данных их количество превышает 70 000. В тональных словарях представлена не вся оценочная лексика. Для дальнейшего пополнения обучающей выборки и тональных словарей планируется использовать результаты, полученные через веб-ресурс взаимодействия эксперта с системой, который рассматривается в подразделе 3.3 настоящего исследования.

На данный момент система работает только с двумя классами: положительный и отрицательный. Планируется обучить классификатор анализировать тексты, относящиеся к нейтральному классу, для чего необходимо пополнение обучающей выборки размеченными текстами соответствующего класса.

3.4.2. Извлечение именованных сущностей

Решение данной задачи поможет определить, что партия «Справедливая Россия» — это не то же самое, что словосочетание «справедливая Россия» (где прилагательное «справедливый» употребляется в значении «беспристрастный, объективный, законный») или что Сергей Миронов и Андрей Миронов — это разные именованные сущности, за которыми скрываются разные люди.

3.4.3. Выявление объекта оценки

Одно и то же сообщение может содержать несколько объектов оценки:

Единая Россия только и обещает, но ничего не делает. Нужно выбирать #ЛДПР7, они должны сдержат слово.

Надеюсь, что Единая Россия провалится на выборах и победит партия #ЛДПР7

Приведённые выше примеры содержат негативную оценку по отношению к партии «Единая Россия» и позитивную для партии ЛДПР, что делает невозможным определение тональности для текста в целом.

Решить данную задачу можно было бы, разбив твит на составные части, но для этого необходимо подключение модуля синтаксического анализа.

3.4.4. Распознавание сарказма и иронии

Значительная часть твитов из корпуса содержит сарказм или иронию:

Урааа "Единая Россия " заняла первое место в выборах!!! Кто бы мог подумать!

Дмитрию Анатольевичу Медведеву исполнилось 51 год. Хочется пожелать ему всего доброго, хорошего настроения, здоровья, но денег нет.

Данная задача является трудноразрешимой. Предлагаемое решение состоит в следующем: при экспертной разметке отмечать не только класс твита, но и приписывать специальную метку, если данный твит содержит сарказм, а затем обучить классификатор на получившейся обучающей выборке. Однако такое решение представляется весьма трудоёмким и требует много времени, так как необходима очень большая обучающая выборка, содержащая ручную разметку.

3.4.5. Выполнение мультимодального анализа

Твит представляет собой сочетание вербальной (непосредственно текст) и невербальной (картинки, видео, GIF) составляющих. Иной раз тональность сообщения можно определить только с помощью анализа именно невербальной составляющей. Например, твит «Сходил проголосовал за Единую Россию» дополняется фотографией испорченного бланка. Данная задача тесно связана с задачей распознавания сарказма и иронии.

В данном исследовании частично реализуется задача мультимодального анализа, так как система автоматического анализа тональности учитывает эмоджи, относящиеся к невербальному (графическому) компоненту твита.

3.5. Выводы к главе 3

В 3 главе был описан процесс реализации системы автоматического анализа тональности Sentimentor в виде программного средства. Классификатор, состоящий из нескольких программных модулей, был разработан на языке C# в Microsoft Visual Studio: задачей модуля twitoGraber является агрегация твитов из Твиттера; модуль wordProcessor выполняет предобработку текстов твитов; модуль Sentimentor производит классификацию твитов методом, разрабатываемым в данном исследовании, и подсчитывает метрики оценки эффективности; модуль CoefficientSelection подбирает оптимальные параметры для наиболее результативной классификации.

Далее в главе 3 производится оценка эффективности классификатора на основе значения метрик точности, полноты и F1-меры. В качестве baseline был выбран результат классификации на основе Наивного байесовского классификатора. Гибридный метод, разрабатываемый в данном исследовании, показал более высокие результаты: в среднем для обоих классов точность увеличилась на 12,91%, полнота — на 16,11%, прирост F1-меры составил 18,75%.

Для дальнейшего взаимодействия эксперта с системой создан веб-интерфейс, доступный по адресу <https://sent.xxoniii.com/>, с помощью которого эксперт имеет возможность исправить результат классификации, предложенный системой, пополнить обучающую выборку и словарь тональной лексики, подобрать оптимальные значения параметров, рассчитать метрики оценки эффективности.

Определено дальнейшее направление исследований. Для улучшения результатов классификации необходимо решение следующих задач: пополнение обучающей выборки и тональных словарей, извлечение именованных сущностей, выявление объекта оценки, распознавание сарказма и иронии, выполнение мультимодального анализа.

Заключение

За последние годы автоматический анализ тональности стал одной из самых популярных задач прикладной лингвистики, что объясняется не только научным, но и коммерческим интересом к исследованиям общественного мнения и настроений на основе обработки и анализа постоянно увеличивающегося потока Больших Данных.

В ходе данного исследования была разработана и реализована система автоматического анализа тональности на материале сообщений о политических партиях в социальных сетях, получившая название Sentimentor.

В процессе работы были поэтапно решены следующие задачи:

1. Изучена область автоматического анализа тональности: исследован понятийно-терминологический аппарат данной области, а также задачи, проблемы и существующие методы анализа тональности;

2. Создан корпус текстов, представляющий материал исследования и состоящий из сообщений о политических партиях в социальной сети Твиттер. Объём корпуса составил: 74 817 твитов или 1 031 321 словоупотреблений;

3. Для хранения корпуса текстов с помощью Microsoft SQL Server создана база данных SentDB;

4. На основе анализа корпуса выделены и подробно описаны особенности текстов данной сети и данной предметной области, которые были учтены при предварительной обработке материала и разработке метода.

5. С учётом особенностей материала разработан оригинальный метод, основанный на комбинации лингвистических методов и методов машинного обучения. Данный метод предполагает обучение Наивного байесовского классификатора на выборке небольшого объёма и дальнейший перерасчёт вероятностей принадлежности текста к определённому классу с помощью данных из тональных словарей;

6. Для реализации разрабатываемого метода была сформирована обучающая выборка, в которую вошли 2000 размеченных экспертами твитов, а также составлены словари тональной лексики: универсальный словарь тональной лексики объёмом 3042 лексические единицы и предметно-ориентированный словарь объёмом 386 лексических единиц, а также тональный словарь эмоджи объёмом 360 единиц;

7. На языке C# создана и протестирована система автоматического анализа тональности Sentimentor, показавшая достаточно высокую эффективность: точность системы в целых числах составляет 86%, полнота — 83%, а значение F-меры достигает 84%.

8. Для дальнейшего взаимодействия эксперта с системой и повышения её эффективности разработан веб-интерфейс, доступный по адресу: <https://sent.xxoniiiii.com/>. С его помощью эксперт может исправить результат классификации системы, пополнить обучающую выборку, предложить слово-кандидат для пополнения тональных словарей, подобрать оптимальные параметры расчёта вероятностей, рассчитать метрики оценки эффективности системы.

9. В ходе анализа результатов было определено последующее направление исследований и выявлены задачи, решение которых поможет в дальнейшем улучшить результат классификации. К ним относятся пополнение обучающей выборки и тональных словарей с помощью разработанного веб-интерфейса и создание модулей извлечения именованных сущностей, выявления объекта оценки, распознавания сарказма и иронии, а также выполнения мультимодального анализа.

Таким образом, цель исследования была достигнута, а его основные задачи решены.

В исследуемом материале длина твита редко превышает одно предложение, а потому задача определения тональности твита может быть приравнена к задаче определения тональности предложения. Из этого

следует, что предложенная методика может быть применена и к текстам из других социальных сетей.

Также разработанная система может быть настроена на анализ тональности текстов других предметных областей за счёт составления дополнительных предметно-ориентированных словарей.

Список использованной литературы

1. Алексеева, С.В. Linis-crowd.org: лексический ресурс для анализа тональности социально-политических текстов на русском языке / С.В. Алексеева, Е.Ю. Кольцова, С.Н. Кольцов // Компьютерная лингвистика и вычислительные онтологии: сборник научных статей. Труды XVIII объединенной конференции «Интернет и современное общество» (IMS-2015). — 2015. — С. 25-32.
2. Барина, С.О. Классификация сокращений в языке Интернета (на материале английского языка) / С.О. Барина // Известия РГПУ им. А.И. Герцена. 2007. — Т. 12. — №33. — С. 24-27.
3. Белоедова А. В. Типы источников информации в современном медиадискурсе и проблемы их достоверности // А.В. Белоедова // Научные ведомости БелГУ. Серия: Гуманитарные науки. — 2017. — №7 (256). — С. 87-94.
4. Большие Данные: как извлечь из них информацию / А. Моррисон [и др.] // Технологический прогноз. — 2010. — №3. — С. 22-29.
5. Гусейнов, Г. Ч. Берлога веблога. Введение в эрратическую семантику [Электронный ресурс] / Г.Ч. Гусейнов // «Говорим по-русски». — 2005. — Режим доступа: http://speakrus.ru/gg/microprosa_erratica-1.htm. — (Дата обращения: 05.05.2017).
6. Коршунов, А.В. Тематическое моделирование текстов на естественном языке / А.В. Коршунов, А.Г. Гомзин // Труды ИСП РАН. — 2012. — Т. 23. — С. 215-244.
7. Сорокин, Ю.А. Креолизованные тексты и их коммуникативная функция / Ю.А. Сорокин, Е.Ф. Тарасов // Оптимизация речевого воздействия. — 1990. — С. 180-186.
8. Худякова, М.В. Классификация отзывов пользователей с использованием фрагментных правил / М.В. Худякова, С. Давыдов, В.Г. Васильев // Компьютерная лингвистика и интеллектуальные технологии; по материалам ежегодной Международной конференции «Диалог». — 2012. — С. 66-78.

9. Araújo, M. iFeel: A system that compares and combines sentiment analysis methods / M. Araújo [et al.] // Proceedings of the companion publication of the 23rd international conference on World wide web companion. — 2014. — P. 75—78.
10. Baum, R. Die Verwendung von Emojis in der Konsumentenkommunikation. Eine stimmungsanalytische Betrachtung von Kurznachrichten im Social Web / R. Baum, T.Egelhof // Junior Management Science. — 2017. — №2. — P. 1-42.
11. Bravo-Marquez, F. Meta-level sentiment models for big social data analysis / F. Bravo-Marquez, M. Mendoza, B. Poblete // Knowledge-Based Systems. — 2014. — №69. — P. 86-99.
12. Calvo, R. Affect detection: An interdisciplinary review of models, methods, and their applications / R. Calvo R., S. D’Mello // IEEE Transactions on Affective Computing. — 2010. — №1. — P. 18-37.
13. Cambria, E. Affective computing and sentiment analysis / E. Cambria // IEEE Intelligent Systems. — 2016. — №2. — P. 102-107.
14. Cambria, E. Sentic computing: A common-sense-based framework for concept-level sentiment analysis / E. Cambria, A. Hussain. — Cham, 2015. — 196 p.
15. Cambria, E. The hourglass of Emotions / E. Cambria, A. Livingstone, A. Hussain // Cognitive Behavioural Systems. Lecture Notes in Computer Science; ed. A. M. Esposito, A. Vinciarelli, and R. Hoffmann, V. C. Muller. — Berlin, 2012. — P. 144-157.
16. Chaturvedi, I. Lyapunov filtering of objectivity for Spanish sentiment model / I. Chaturvedi, E. Cambria, D. Vilares // Proceedings of International Joint Conference on Neural Networks. — Vancouver, 2016. — P. 4474-4481.
17. Convolutional MKL based multimodal emotion recognition and sentiment analysis / S. Poria [et al.] // Proceedings of the IEEE International Conference on Data Mining series (ICDM). — 2016. — P. 439-448.
18. Dragoni, M. A fuzzy system for concept-level sentiment analysis / M. Dragoni, A.G. Tettamanzi, C. da Costa Pereira // Semantic web evaluation challenge. — Cham, 2014. — P. 21-27.

19. How many emoji characters are there? [Электронный ресурс] // Emojipedia, 2016. Режим доступа: <http://emojipedia.org/faq/>. Дата обращения: 23.12.2017
20. Feldman, R. The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data / R. Feldman, J. Sanger. — Cambridge, 2006. — 423 p.
21. Fusing audio, visual and textual clues for sentiment analysis from multimodal content / S. Poria [et al.] // Neurocomputing. — 2016. — №174. — P. 50-59.
22. Gezici, G. Su-sentilab: A classification system for sentiment analysis in twitter / G. Gezici [et al.] // International Workshop on Semantic Evaluation. — 2013. — P. 471-477.
23. Ivanko, S.L. Context incongruity and irony processing / S.L. Ivanko, P.M. Pexman // Discourse Processes. — 2003. — P. 241-278.
24. Karkaletsis, V. United we stand: improving sentiment analysis by joining machine learning and rule based methods / Karkaletsis [et al.] // 7th International Conference on Language Resources and Evaluation. — Malta, 2010. — P. 382-390.
25. Ko, Y. Automatic text categorization by unsupervised learning / Y. Ko, J. Seo // Proceedings of the 18th International Conference on Computational Linguistics (COLING). — 2000. — P. 453-459.
26. Koltsova, O.Y. An Opinion Word Lexicon and a Training Dataset for Russian Sentiment Analysis of Social Media / O.Y. Koltsova, S.V. Alexeeva, S.N. Kolcov // Компьютерная лингвистика и интеллектуальные технологии. — 2016. — P. 277-287
27. Liu, B. Many Facets of Sentiment Analysis / B. Liu // A Practical Guide to Sentiment Analysis. — 2017. — P. 11-39.
28. Ma, Y. Label embedding for zero-shot fine-grained named entity typing / Y. Ma, E. Cambria, S. Gao // Proceedings of the International Conference on Computational Linguistics (COLING). — 2016. — P. 171-180.

29. Mihalcea, R. What men say, what women hear: Finding gender-specific meaning shades / R. Mihalcea, A. Garimella // IEEE Intelligent Systems. — 2016. — №4. — P. 62-67.
30. Nasukawa, T. Sentiment analysis: Capturing favorability using natural language processing / T.Nasukawa, J. Yi. // Proceedings of the 2nd International Conference on Knowledge Capture (K-CAP). — 2003. — P. 70-77.
31. Nigam, K. Using maximum entropy for Text Classification / K. Nigam, J. Lafferty, A. McCallum // Proceedings of the International Joint Conference on Artificial Intelligence. — 1999. — P. 454-462.
32. Pang, B. Opinion mining and sentiment analysis / B. Pang, L. Lee // Foundations and Trends in Information Retrieval. — 2008. — №2. — P. 1-135.
33. Pang, B. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales / B. Pang, L. Lee // In Proceedings of the 43rd annual meeting of the Association for Computational Linguistics (ACL). — 2005. — №June 25-30. — P. 115-124.
34. Pang, B. Thumbs up? Sentiment Classification using Machine Learning Techniques / B. Pang, L. Lee // Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). — 2002. — P. 79-86.
35. Picard, R. Affective computing / R. Picard. — Boston: The MIT Press, 1997. — 306 P.
36. Poria, S. A deeper look into sarcastic tweets using deep convolutional neural networks / S. Poria [et al.] // Proceedings of the Conference on Computational Linguistics (COLING). — 2016. — P. 1601-1612.
37. Poria, S. Aspect extraction for opinion mining with a deep convolutional neural network / S. Poria, E. Cambria, A. Gelbukh // Knowledge-Based Systems. — 2016. — №108. — P. 42-49.
38. Poria, S. Common sense knowledge based personality recognition from text / S. Poria [et al.] // Advances in soft computing and its applications. — Berlin, 2013. — P. 484-496.

39. Prabowo, R. Sentiment analysis: A combined approach / R. Prabowo, M. Thelwall // *Journal of Informetrics*. — 2009. — №3(2). — P. 143-157.
40. Qadir, A. Bootstrapped Learning of Emotion Hashtags #hashtags4you / A. Qadir, E. Riloff // *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis (WASSA)*. — 2013. — P. 348-358.
41. Rao, D. Semi-supervised polarity lexicon induction / D. Rao, D. Ravichandran // *Proceedings of the European Chapter of the Association for Computational Linguistics*. — 2009. — P. 675-682.
42. Ratnaparkhi, A. A Simple Introduction to Maximum Entropy Models for Natural Language Processing / A. Ratnaparkhi // *IRCS Technical Reports Series*.
43. Read, J. Weakly supervised techniques for domain-independent sentiment classification / J. Read, J. Carroll // *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*. — 2009. — P. 45-52.
44. Recognising realistic emotions and affect in speech: State of the art and lessons learnt from the first challenge / B. Schuller [et al.] // *Speech Communication*. — 2011. — №53(9/10). — P. 1062-1087.
45. Sentilo: Framebased sentiment analysis / D.F. Recupero [et al.] // *Cognitive Computation*. — 2014. — №7(2). — P. 211-225.
46. Snyder, B. Multiple Aspect Ranking using the Good Grief Algorithm / B. Snyder B., R. Barzilay // *Proceedings of the Joint Human Language Technology/North American Chapter of the ACL Conference (HLT-NAACL)*. — 2007. — P. 300-307.
47. Somasundaran, S. Discourse level opinion interpretation / S. Somasundaran, J. Wiebe, J. Ruppenhofer // *Proceedings of the Conference on Computational Linguistics (COLING)*. — 2008. — P. 801-808.

48. Srivastava, A. Supervised SA of product reviews using Weighted k-NN Algorithm / A. Srivastava, Dr. M. P. Singh // Proceedings of 11th International Conference on Information Technology. — 2014. — P. 386-394.
49. Stevenson, R. Characterization of the affective norms for English words by discrete emotional categories / R. Stevenson, J. Mikels, T. James // Behavior Research Methods. — 2007. — №39. — P. 1020-1024.
50. Tamilselvi, A. Sentiment Analysis of Micro blogs using Opinion Mining Classification Algorithm / A. Tamilselvi, M. ParveenTaj // International Journal of Science and Research (IJSR). — 2013. — Vol. 2. — №2. — P. 102-108.
51. Turney, P.D. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews / P.D. Turney // Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02). — 2002. — P. 417-424.
52. Turning conversations into insights: A comparison of Social Media Monitoring Tools // A white paper from FreshMinds Research. — 2010. — URL: www.freshminds.co.uk.
53. Wilson, T. Recognizing contextual polarity in phrase-level sentiment analysis / T. Wilson, J. Wiebe, P. Hoffmann // Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP). — 2005. — P. 347-354.
54. Zeng, Z. A survey of affect recognition methods: Audio, visual, and spontaneous expressions / Z. Zeng, M. Pantic, G. Roisman, T. Huang // The IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2009. — №31(1). — P. 39-58.