Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа: Информационных технологий и робототехники
Направление подготовки: 09.04.01 Информатика и вычислительная техника
Отделение школы (НОЦ): Информационных технологий

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Классификатор эмоционального тона сообщений пользователей социальной сети Twitter

УДК: 004.773.6:316.472:159.942

Студент

Группа	ФИО	Подпись	Дата
8ВМ6Г	Байкадир Ж.Б.		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент	Цапко С.Г.	к.т.н		

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент	Рыжакина Т.Г.	к.э.н		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
ассистент	Авдеева И.И.	–		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
доцент	Кочегурова Е.Е.	к.т.н		

Томск – 2018 г.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Код результатов	Результат обучения (выпускник должен быть готов)
	Общепрофессиональные компетенции
Р1	Воспринимать и самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте.
Р2	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях.
Р3	Демонстрировать культуру мышления, способность выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных, анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.
Р4	Анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего образования и профессиональной мобильности. Владеть, по крайней мере, одним из иностранных языков на уровне социального и профессионального общения, применять специальную лексику и профессиональную терминологию языка.
	Профессиональные компетенции
Р5	Выполнять инновационные инженерные проекты по разработке аппаратных и программных средств автоматизированных систем различного назначения с использованием современных методов проектирования, систем автоматизированного проектирования, передового опыта разработки конкурентно способных изделий.
Р6	Планировать и проводить теоретические и экспериментальные исследования в области проектирования аппаратных и программных средств автоматизированных систем с использованием новейших достижений науки и техники, передового отечественного и зарубежного опыта. Критически оценивать полученные данные и делать выводы.
Р7	Осуществлять авторское сопровождение процессов проектирования, внедрения и эксплуатации аппаратных и программных средств автоматизированных систем различного назначения.
	Общекультурные компетенции
Р8	Использовать на практике умения и навыки в организации исследовательских, проектных работ и профессиональной эксплуатации современного оборудования и приборов, в управлении коллективом.
Р9	Осуществлять коммуникации в профессиональной среде и в обществе в целом, активно владеть иностранным языком, разрабатывать документацию, презентовать и защищать результаты инновационной инженерной деятельности, в том числе на иностранном языке.
Р10	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.
Р11	Демонстрировать способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности, способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, способность к педагогической деятельности.

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа: Информационных технологий и робототехники
Направление подготовки: 09.04.01 Информатика и вычислительная техника
Отделение школы (НОЦ): Информационных технологий

УТВЕРЖДАЮ:
Руководитель ООП

_____ _____ _____
(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

Магистерской диссертации

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	**ФИО**
8ВМ6Г	Байкадиру Жансерику Багдатулы

Тема работы:

Классификатор эмоционального тона сообщений пользователей социальной сети Twitter	
Утверждена приказом директора (дата, номер)	

Срок сдачи студентом выполненной работы:	

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	1. Существующие готовые решения задач анализа естественного языка. 2. Алгоритмы машинного обучения. 3. Методы кодирования слов. 4. Требование организовать сбор тренировочного набора данных. 5. Требование реализовать программный сервис классификации эмоционального тона сообщений социальной сети Twitter.

Перечень подлежащих исследованию, проектированию и разработке вопросов	1. Рассмотреть существующие алгоритмы анализа естественного языка. 2. Определить функционал программного сервиса. 3. Спроектировать архитектуру программного сервиса. 4. Разработать способ формирования данных с участием пользователей. 5. Реализовать планировщика обучения. 6. Реализовать классификатор на основе выбранной архитектуры машинного обучения.
Перечень графического материала	Презентация в формате .pptx на 18 слайдов.

Консультанты по разделам выпускной квалификационной работы

Раздел	Консультант
Финансовый менеджмент, русурсоэффективность и ресурсосбережение	Рыжакина Татьяна Гавриловна
Социальная ответственность	Авдеева Ирина Ивановна

Названия разделов, которые должны быть написаны на русском и иностранном языках:

Аналитический обзор

Проектирование и реализация программного сервиса

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Цапко Сергей Геннадьевич	Кандидат технических наук		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ВМ6Г	Байкадир Жансерик Багдатулы		

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа: Информационных технологий и робототехники
Направление подготовки: 09.04.01 Информатика и вычислительная техника
Уровень образования: магистратура
Отделение школы (НОЦ): Информационных технологий
Период выполнения осенний/весенний семестр 2017/2018 учебного года

Форма представления работы:

Магистерская диссертация

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
	Основная часть	75
	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
	Социальная ответственность	10

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Цапко Сергей Геннадьевич	Кандидат технических наук		

СОГЛАСОВАНО:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Кочегурова Елена Алексеевна	Кандидат технических наук		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
8ВМ6Г	Байкадир Жансерик Багдатулы

Школа	ИШИТР	Отделение	Отделение информационных технологий
Уровень образования	Магистратура	Направление/специальность	09.04.01 Информатика и вычислительная техник

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:	
1. *Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих*	*Работа с информацией, представленной в российских и иностранных научных публикациях, аналитических материалах, статических бюллетенях и изданиях, нормативно-правовых документах; анкетирование; опрос.*
2. *Нормы и нормативы расходования ресурсов*	
3. *Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования*	
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. *Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения*	*Проведение предпроектного анализа. Определение целевого рынка и проведение его сегментирования. Выполнение SWOT-анализа проекта*
2. *Определение возможных альтернатив проведения научных исследований*	*Определение целей и ожиданий, требований проекта. Определение заинтересованных сторон и их ожиданий.*
3. *Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок*	*Составление календарного плана проекта. Определение бюджета НТИ*
4. *Определение ресурсной, финансовой, экономической эффективности*	*Проведение оценки экономической эффективности разработки программного комплекса, который позволяет производить автоматизированную классификацию сообщений пользователей социальной сети Twitter по эмоциональному окрасу.*
Перечень графического материала *(с точным указанием обязательных чертежей):*	
1. *Оценка конкурентоспособности технических решений* 2. *Матрица SWOT* 3. *График проведения и бюджет НТИ* 4. *Расчёт денежного потока* 5. *Оценка ресурсной, финансовой и экономической эффективности НТИ*	

Дата выдачи задания для раздела по линейному графику	

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Рыжакина Татьяна Гавриловна	Кандидат экономических наук		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ВМ6Г	Байкадир Жансерик Багдатулы		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8ВМ6Г	Байкадир Жансерик Багдатулы

Школа	ИШИТР	**Отделение**	Информационных технологии
Уровень образования	Магистратура	**Направление/специальность**	09.04.01 Информатика и вычислительная техника

Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объектом исследования является разрабатываемый программный комплекс, который позволяет производить автоматизированную классификацию сообщений пользователей социальной сети Twitter по эмоциональному окрасу. Разработка системы происходит в помещениях и требует работы с компьютерами и другими электронными устройствами, которые являются источниками вредных излучений и могут оказывать негативное влияние на здоровье и жизнедеятельность человека.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Производственная безопасность	Возможные вредные факторы в офисном помещении: • Недостаточная вентиляция; • Недостаточное или неправильное освещение; • микроклимат; • Повышенный уровень электромагнитного излучения; • Нервно-психические перегрузки • Шум Возможные опасные факторы в офисном помещении: • Короткое замыкание; • Электрический ток. • Статическое электричество
2. Экологическая безопасность:	В процессе разработки и эксплуатации искусственной нейронной сети возможно образование следующих видов отходов: • образование твердых отходов, относящихся к IV классу опасности (системный блок компьютера, принтеры, сканеры, клавиатура, манипулятор "мышь") и жидких отходов; • Жидкие отходы: сточные воды; • Люминесцентные лампы.

3. Безопасность в чрезвычайных ситуациях:	Наиболее типичная чрезвычайная ситуация при работе в офисе – пожар. Превентивные меры включают инструктаж по пожарной безопасности, контроль состояния проводки и электрических приборов, своевременное профилактическое обслуживание.
4. Правовые и организационные вопросы обеспечения безопасности:	Параметры рабочего места офисного работника регулируются ГОСТ 12.2.032–78 ССБТ, СанПиН 2.2.2/2.4.1340–03, «Трудовой кодекс Российской Федерации» от 30.12.2001 №197-ФЗ.

Дата выдачи задания для раздела по линейному графику	**– 14.03.2018**

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Авдеева Ирина Ивановна	—		**14.03.2018**

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ВМ6Г	Байкадир Жансерик Багдатулы		**14.03.2018**

Реферат

Выпускная квалификационная работа содержит 131 страниц, 37 рисунков, 33 таблиц, 3 приложения, и 38 использованных литературных источников.

Ключевые слова: нейронная сеть, классификатор, свёрточные нейронные сети, анализ тональности, социальная сеть Twitter.

Объектом исследования являются различные методы и архитектуры машинного обучения для решения задач анализа естественного языка.

Цель работы: разработка классификатора эмоционального тона русскоязычных сообщений пользователей социальной сети Twitter.

Работа представлена введением, 6 разделами (главами) и заключением, приведен список использованных литературных источников.

В результате проделанной работы было спроектировано и реализовано программное обеспечение для классификации эмоционального тона сообщений пользователей социальной сети Twitter с возможностью формирования новых тренировочных данных с участием пользователя.

Реализованный в ходе работы классификатор обладает точностью классификации равной 76,37% – тестирование проводилось на тестовых данных, не входящих в тренировочный набор.

В будущем планируется провести ряд изменений в реализованном классификаторе: улучшить точность классификатора путем увеличения тренировочного набора данных.

Сокращения и обозначения

– Deep learning — глубокое обучение;

– MaxPooling — операция объединения максимальных значений матриц в свёрточных нейронных сетях;

– Softmax — функция мягкого максимума;

– Dropout — метод регуляризации для предотвращения переобучения сети;

– Batches — группы примеров, используемые для обучения сети;

– CNN (Convolutional Neural Network)— свёрточная нейронная сеть;

– SVM (Support Vector Machine) — метод опорных векторов;

– API (Application Programming Interface) — программный интерфейс приложения.

Оглавление

Введение

За последнее десятилетие значительно возросло использование различных онлайн-ресурсов, в частности, социальных сетей, таких как Twitter. Многие компании и организации определяют эти ресурсы как значимые для маркетинговых исследований [1]. Обычно, чтобы получить обратную связь и понимание того, как покупатели относятся к их продукции, компании проводят интервью, анкетирования и опросы. Эти стандартные методы часто требуют больших затрат времени и денег; более того, они не всегда приносят желаемый результат.

Для решения задачи автоматического определения эмоциональной окраски текста используются алгоритмы обработки естественных языков. Среди которых на данный момент наиболее популярными являются алгоритмы глубокого обучения. Существует большое количество работ, посвящённых обработке естественного языка и, в частности, анализу тональности с использованием нейронных сетей. Но большая часть из них адаптирована для применения к английскому языку [2].

На данный момент существуют такие веб-сервисы Tone Analyzer [3] и для определения тональности текста, но большинство из них работают только с английским языком. Существуют также сервис Repustate [4], который поддерживает русскоязычные тексты, но данный сервис является доступным пользователям по платной подписке. В данной работе затронуты основные моменты связанные с реализацией задачи анализа тональности текстов на русском языке. Актуальность работы обусловлена тем, что на текущий момент существует малое количество систем, способных анализировать тональность текста на русском языке.

Также, по причине того, что разговорный язык, использующийся в социальных сетях постоянно развивается, чтобы обеспечить наилучшую точность классификатора, необходимо периодически обновлять тренировочный набор данных.

Целями данной работы являются:

– разработка классификатора эмоционального тона русскоязычных сообщений пользователей социальной сети Twitter;

– разработать способ формирования тренировочных данных с участием самих пользователей.

Для достижения целей, требуется выполнить следующие задачи:

– Проанализировать существующие методы решения задачи анализа тональности текста;

– Провести анализ существующих алгоритмов машинного обучения;

– Выбор архитектуры машинного обучения;

– Реализовать кодирование входных данных;

– Разработать классификатор тональности сообщений социальной сети Twitter;

– Обеспечить точность классификации не менее 75%;

– Протестировать и сравнить полученную модель с существующими методами решения;

– Разработать веб-приложение для взаимодействия пользователя с системой для определения эмоционального тона сообщений пользователей социальной сети;

– Разработать способ формирования тренировочных данных с участием пользователей.

1 Обзор технологии и решений

1.1 Существующие исследования

В публикациях были рассмотрены решения задачи классификации текста методом обучения с учителем [5]. В этих статьях рассматривались тексты обзоров на туристические компании на английском языке: задача заключалась в том, чтобы выяснить, рекомендует ли рецензент компанию, которому посвящен обзор.

В работе [6], с помощью алгоритмов машинного обучения была достигнута точность классификации, которая составляет 89,6% на разных англоязычных выборках данных.

В работе [7] авторы классифицируют тексты на уровне символов. Был использован набор из 70 символов для того чтобы представить каждый символ как one-hot вектор. Также был установлена фиксированная длина текста в 1014 символа. Таким образом, текст представлен бинарной матрицей размером 70x1014. Данный метод не имеет представления о словах и видит их как комбинации символов, и информация о семантической близости слов не предоставлена сети, как в случаях, заранее натренированных векторов слов.

1.2 Предварительная обработка и способы кодирования данных

Предобработка текста положительно влияет на качество классификации с помощью алгоритмов машинного обучения с учителем, т.к. позволяет убрать из тренировочного набора слова или окончания слов, не влияющие на результат классификации. Например, слова имеющие разное склонение несут один и тот же смысл. На данный момент используется несколько способов предобработки:

1. Стэмминг — удаление окончаний, приведение слова к основе.
2. Лемматизация — приведение слова к начальной форме.

3. Удаление стоп-слов из списка

В частности, предобработка текста реализованы в библиотеке nltk [8]. Не все методы, предложенные библиотекой nltk применимы к предварительной обработке текстов на русском языке, в основном только методы, которые указаны в возможностях библиотеки pymorphy [9].

Способы кодирования данных

Входные данные для алгоритмов машинного обучения данные должны быть числовыми

Данные которые подаются на вход нейронной сети должны быть числовыми. В рамках исследования были рассмотрены следующие методы преобразования текста в вектор:

1. Векторы слов;
2. Корзина слов (Bag of Words);
3. One-hot кодирование.

Одним из основных реализаций векторов слов является метод преобразования слов в векторные представления Word2vec [10].

Word2vec — программный инструмент анализа семантики естественных языков, который принимает большой текстовый корпус в качестве входных данных и сопоставляет каждому слову вектор, выдавая координаты слов на выходе. Сначала он создает словарь, «обучаясь» на входных текстовых данных, а затем вычисляет векторное представление слов. Векторное представление основывается на контекстной близости: слова, встречающиеся в тексте рядом с одинаковыми словами (следовательно, имеющие схожий смысл), в векторном представлении будут иметь близкие координаты векторов-слов. Полученные векторы-слова могут быть использованы для обработки естественного языка и машинного обучения [10]. Недостатками word2vec являются: переобучение модели под каждый естественный язык и

необходимость в больших корпусах текстов. Также word2vec не достигает высокой точности при анализе словосочетаний и комбинации слов.

Корзина слов – это модель текстов на естественном языке, в которой каждый документ или текст выглядит как неупорядоченный набор слов без сведений о связях между ними. Его можно представить в виде матрицы, каждая строка в которой соответствует отдельному документу или тексту, а каждый столбец — определенному слову. Ячейка на пересечении строки и столбца содержит количество вхождений слова в соответствующий документ. Основным недостатком данного метода является то, что для получения упорядоченной матрицы. Игнорирование семантических связей между словами – главный недостаток модели Bag-of-words. Другой ее важный недостаток в том, что тексты как наборы слов проецируются в пространство высокой размерности и высокой разреженности, что обусловлено объемом используемого словаря [11].

Данные методы обладают следующими недостатками при обработке текстов на русском языке:

– низкая точность при работе с определенными конструкциями языка (отрицание, знаки препинания, словосочетания);

– нехватка количества словарей на русском языке;

– необходимость в переобучении модели с появлением новых данных.

Наряду с этим, сообщения из социальных сетей могут иметь орфографические ошибки или новые слова, по этой причине, векторные представления пропустят много информации, поскольку векторные представления игнорируют слова которых нет в словаре [10].

One-hot-encoding является представлением символов или в качестве вектора с двоичными значениями. Данный метод позволяет получить значительное увеличение точности в задаче классификации текста в языках где присутствуют сложные конструкции слов (окончания, суффиксы,

предлоги и т.д). Пример представления слова на уровне символов с помощью посимвольного кодирования изображена на рисунке 1.1

Рисунок 1.1 – Пример посимвольного кодирования

Основные преимущества использования посимвольного кодирования:

– модель будет намного меньше (около 50-100 мб для всей модели по сравнению с более чем 3 ГБ для классического Word2Vec) [12];

– модель учитывает эмоциональный тон повторяющихся букв, например, слово урааа! [22];

– модель имеет низкий уровень восприимчивости к опечаткам [12];

– подходит для обучения модели на текстах любых языков [12].

1.3 Обзор методов классификации текста основанные на методе обучения с учителем

Методы обучения с учителем позволяют классифицировать данные на основе предварительно обученного набора данных, называемого тренировочного набором. Такие методы должны реализовывать две функции: обучение на тренировочных данных и классификацию на новых данных.

Обучение — это процесс построения или аппроксимация функции, которая позволит обобщить классификацию на все данные так, чтобы эта классификация была ближе всего к фактической [13].

Методы обучения с учителем подвержены проблеме переобучения. Переобучение – это явление, в котором построенная модель с большой точностью классифицирует примеры из тренировочного набора, но при работе с примерами, отличающихся от тренировочных данных, показывает низкую точность. Данное явление происходит, по причине того, что во время процесса обучения модель выявляет некоторые закономерности в тренировочном наборе данных, отсутствующие в общей совокупности. Методы борьбы с переподготовкой зависят от конкретных методов и моделей [13].

К методам обучения с учителем относятся следующие алгоритмы:

- Наивный байесовский классификатор
- Метод опорных векторов
- На основе нейронных сетей
- Алгоритм случайный лес (Random forest)

1.3.1 Наивный байесовский классификатор

Наивный байесовский классификатор простой вероятностный классификатор, основанный на применении теоремы Байеса со строгими (наивными) предположениями о независимости [14].

Согласно статье [2] на основе наивного байесовского классификатора была достигнута точность равная 78,4 % на заданном тестовом наборе данных который состоял из обзоров на фильмы и товары, предложенные сайтом http://imhonet.ru (таблица 1.1). В текстах социальных сетей сложно обеспечить такую точность используя наивный байесовский классификатор, т.к. тексты социальных сетей содержат слова о разных тематиках, не только о товарах, фильмах и услугах. Исследование проводилось на корпусах текста на русском языке, но на данных.

Таблица 1.1 – Точность классификации текста с использованием наивного байесовского классификатора [2].

Описание признаков	**Точность [%]**
Униграммы без стемминга	76.79
Униграммы со стеммингом	75.66
Обработка отрицаний без стемминга	78.26
Удаление излишних признаков без стемминга	78.39

Основные недостатки наивного байесовского классификатора:

- низкая точность при работе со сложными данными [7];
- не учитывает порядок слов;
- не учитывает отрицание в тексте.

1.3.2 Метод опорных векторов

Основная идея метода — перевод исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с максимальным зазором в этом пространстве. Гиперплоскость — пространство, размерность которого на единицу меньше, чем размерность исходного пространства. Эффективность метода опорных векторов значительно снижается, если количество признаковых описаний очень велико [15].

В статье [7] с помощью метода опорных векторов на тестовом наборе была получена точность 81% на униграммах и комбинации между униграммами и биграммами (одно слово) и 76% на биграммах (два слова, словосочетание) (таблица 1.2). Исследование проводилось на корпусах текста на английском языке предоставленные РОМИП-2011, посвященные обзорам продуктов и услуг.

Таблица 1.2 – Точность классификатора на основе метода опорных векторов [7].

Описание признаков	**Точность [%]**
униграммы	81
биграммы	76
комбинация	81

1.3.3 Алгоритм случайный лес

Случайный лес (random forest) — это множество решающих деревьев. В задаче регрессии их ответы усредняются, в задаче классификации принимается решение голосованием по большинству. Все деревья строятся независимо по следующей схеме [16]:

– Выбирается подвыборка обучающей выборки размера samplesize (м.б. с возвращением) – по ней строится дерево (для каждого дерева — своя подвыборка).

– Для построения каждого расщепления в дереве просматривается max_features случайных признаков (для каждого нового расщепления — свои случайные признаки) [16].

– Затем выбирается наилучшие признаки расщепления по заранее заданному критерию. Дерево строится, как правило, до исчерпания выборки (пока в листьях не останутся представители только одного класса) [16].

Главным недостатком алгоритма случайный лес является склонность к переобучению на больших или зашумленных выборках данных [16].

1.3.4 Нейронные сети

Попытки воспроизвести способность биологических нервных систем обучаться и исправлять ошибки привели к созданию искусственных нейронных сетей. Искусственные нейронные сети представляют собой

семейство моделей, построенных по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма [17].

Понятие искусственной нейронной сети было предложено ещё в 1943 году У. Мак-Калоком и У. Питтсом в статье [17]. В частности, ими была предложена модель искусственного нейрона.

Чтобы отразить суть биологических нейронных систем, искусственный нейрон строится следующим образом. Он получает входные сигналы (исходные данные либо выходные сигналы других нейронов нейронной сети) через несколько входных каналов. Каждый входной сигнал проходит через соединение, имеющее определенный вес. С каждым нейроном связано определенное пороговое значение. Вычисляется взвешенная сумма входов, из нее вычитается пороговое значение и в результате получается величина активации нейрона. Сигнал активации преобразуется с помощью функции активации и в результате получается выходной сигнал нейрона.

На рисунке 1.2 приведен пример искусственного нейрона

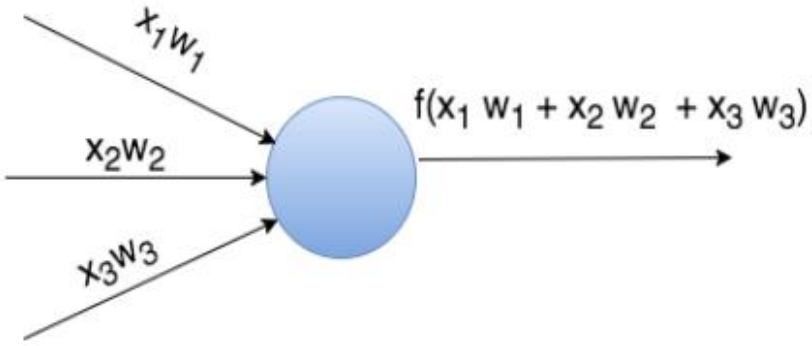


Рисунок 1.2 – Искусственный нейрон [17]

x_i – входной сигнал

w_i – вес входного сигнала

$f(s)$ – функция активации

1.3.5 Свёрточные нейронные сети

С появлением больших объемов данных и больших вычислительных возможностей стали активно использоваться нейронные сети. Особую популярность получили свёрточные нейронные сети, архитектура которых была предложена Яном Лекуном [18] и нацелена на эффективное распознавание изображений. Свое название архитектура сети получила из-за наличия операции свёртки, суть которой в том, что каждый фрагмент изображения умножается на матрицу (ядро) свёртки поэлементно, а результат суммируется и записывается в аналогичную позицию выходного изображения. В архитектуру сети заложены априорные знания из предметной области компьютерного зрения: пиксель изображения сильнее связан с соседним (локальная корреляция) и объект на изображении может встретиться в любой части изображения.

Особое внимание светрочные нейронные сети получили после конкурса ImageNet, который состоялся в октябре 2012 года и был посвящен классификации объектов на фотографиях. В конкурсе требовалось распознавание образов в 1000 категорий. Победитель данного конкурса — Алекс Крижевский, используя свёрточную нейронную сеть, получилось снизить количество ошибок до 15% [19].

Успех применения свёрточных нейронных сетей к классификации изображений привел к множеству попыток использовать данный метод к другим задачам. В последнее время их стали активно использоваться для задачи классификации текстов.

1.3.6 Архитектура свёрточных нейронных сетей

Свёрточная нейронная сеть обычно представляет собой чередование свёрточных слоев (convolution layers), слоев подвыборки (subsampling layers) и

при наличии полносвязных слоев (fully-connected layer) на выходе. Все три вида слоев могут чередоваться в произвольном порядке [13].

В свёрточном слое нейроны, которые используют одни и те же веса, объединяются в карты признаков (feature maps), а каждый нейрон карты признаков связан с частью нейронов предыдущего слоя. При вычислении сети получается, что каждый нейрон 13 выполняет свёртку некоторой области предыдущего слоя (определяемой множеством нейронов, связанных с данным нейроном).

Пример архитектуры свёрточной нейронной сети представлен на рисунке 1.3

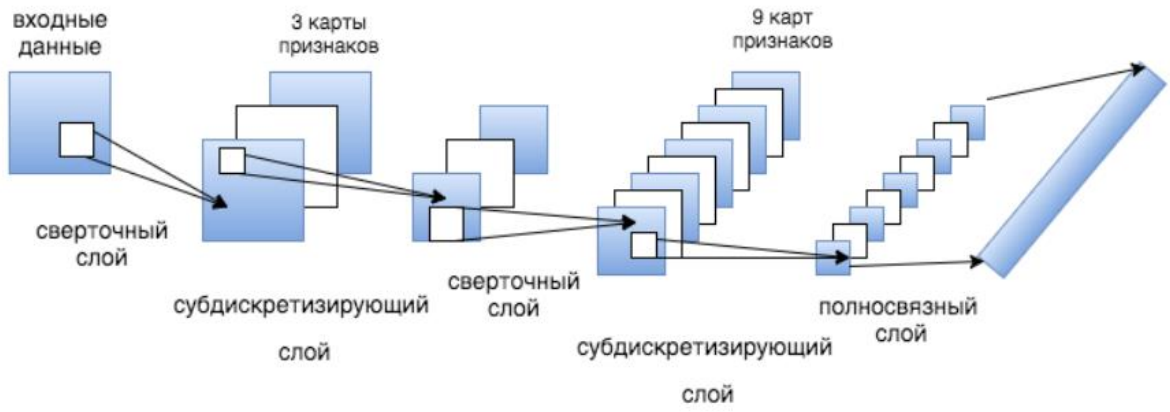


Рисунок 1.3 – Архитектура свёрточной нейронной сети [13]

Свёрточный слой

В отличие от полносвязного, в свёрточном слое нейрон соединен лишь с ограниченным количеством нейронов предыдущего уровня, т. е. свёрточный слой аналогичен применению операции свёртки, где используется лишь матрица весов небольшого размера (ядро свёртки), которую «двигают» по всему обрабатываемому слою.

Еще одна особенность свёрточного слоя в том, что он немного уменьшает изображение за счет краевых эффектов.

На рисунке 1.5 показан пример свёрточного слоя с ядром свёртки размера 3 × 3.

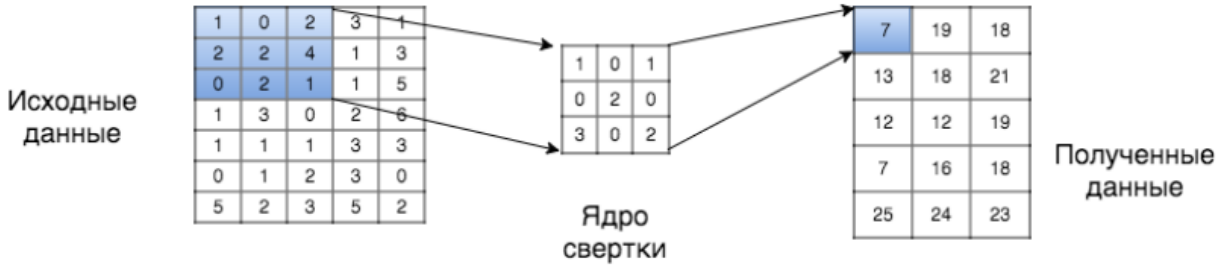


Рисунок 1.5 – Свёрточный слой

Субдискретизирующий слой

Слои этого типа выполняют уменьшение размерности (обычно в несколько раз). Это можно делать разными способами, но зачастую используется метод выбора максимального элемента (max-pooling) — вся карта признаков разделяется на ячейки, из которых выбираются максимальные по значению [13].

На рисунке 1.6 показан пример слоя подвыборки с методом выбором максимального элемента.

Рисунок 1.6 – Субдискретизирующий слой [13]

Полносвязный слой

Слой, в котором каждый нейрон соединен со всеми нейронами на предыдущем уровне, причем каждая связь имеет свой весовой коэффициент. На рисунке 1.7 показан пример полносвязного слоя.

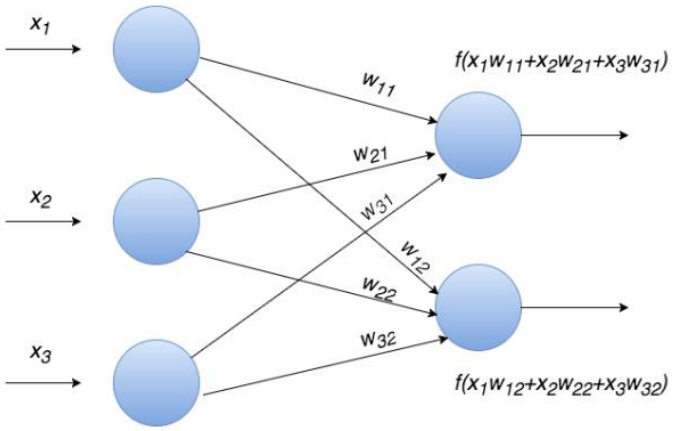


Рисунок 1.7 – Полносвязный слой [13]

w_{ij} — весовые коэффициенты.

$f(s)$ — функция активации.

Dropout слой

Dropout слой (dropout регуляризация) способ борьбы с переобучением в нейронных сетях, обучение которых обычно производят стохастическим градиентным спуском, случайно выбирая некоторые объекты из выборки. Dropout регуляризация заключается в изменение структуры сети: каждый нейрон выбрасывается с некоторой вероятностью p. По такой прореженной сети производится обучение, для оставшихся весов делается градиентный шаг, после чего все выброшенные нейроны возвращаются в нейронную сеть. Таким образом, на каждом шаге стохастического градиента мы настраиваем одну из возможных 2^N архитектур сети, где под архитектурой мы понимаем структуру связей между нейронами, а через N обозначаем суммарное число нейронов. При тестировании нейронной сети нейроны уже не выбрасываются, но выход

каждого нейрона умножается на $(1-p)$ — благодаря этому на выходе нейрона мы будем получать матожидание его ответа по всем 2^N архитектурам. Таким образом, обученную с помощью dropout-регуляризации нейросеть можно рассматривать как результат усреднения 2^N сетей [13].

2 Архитектура программного обеспечения

Для реализации программного обеспечения для определения эмоциональной окраски текста необходимо реализовать из следующие компоненты:

– Хранилище данных для хранения тренировочного набора данных;

– Бинарный классификатор эмоционального тона текста на основе нейронной сети;

– Веб-приложение, включающий в себя функционал для анализа сообщений пользователей и обновления тренировочных данных;

– Планировщик обучения нейронной сети.

Диаграмма компонентов представлена на рисунке 2.1.

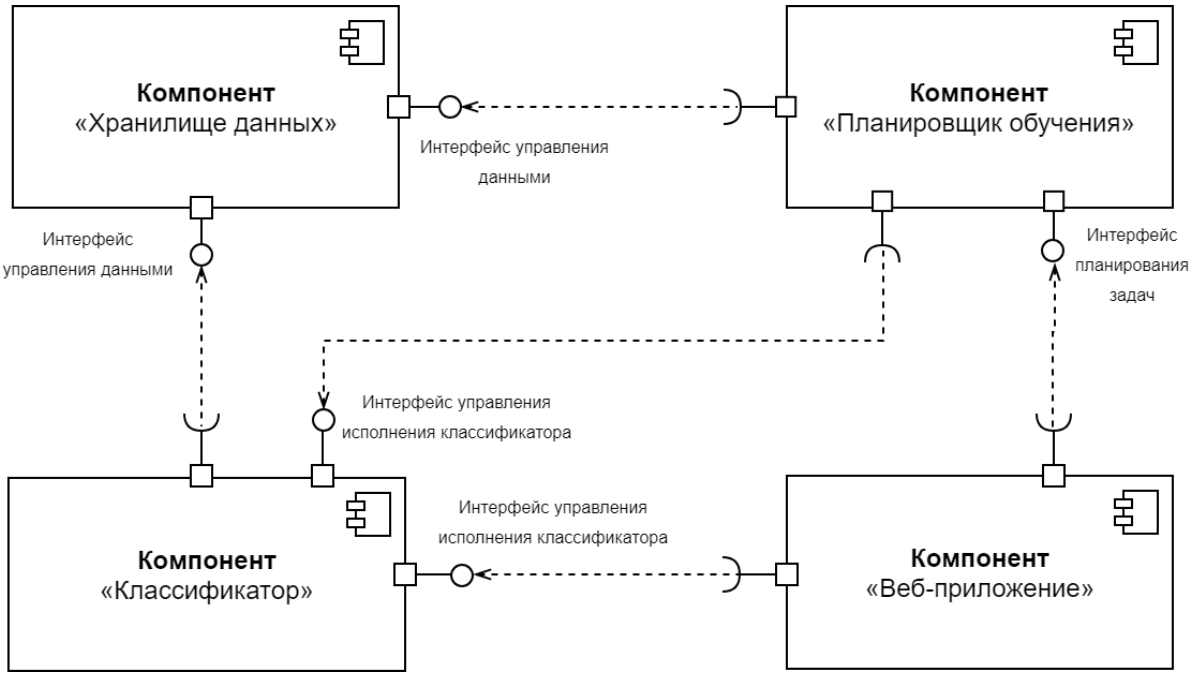


Рисунок 2.1 – Компоненты программного обеспечения для классификации текста сообщений из социальной сети.

Компонент «Классификатор» служит для получения тренировочных данных, обучения модели и классификации текста, поступающего с компонента «Веб-приложение»;

Компонент «Хранилище данных» является базой данных содержащий набор тренировочных данных.

Компонент «Планировщик обучения» служит для автоматического обучения классификатора на основе обновленных данных полученных с компонента «Хранилище данных».

Компонент «Веб-приложение» является клиентским веб-приложением для взаимодействия с компонентом «Классификатор» и служит для получения результатов классификации текста и для запуска обучения с помощью компонента «Планировщик обучения».

2.1 Хранилище данных

Работа классификатора заключается в обучении на большом тренировочном наборе данных. Исходя из этого, существует потребность в хранилище данных.

В основном, для хранения тренировочных данных исследователи используют готовые наборы в файлах с табличным форматом *.CSV. Т.к. с наборами данных будут проводиться большое количество операций, чтение и запись в файл не подходит. Данная задача решается развертыванием базы данных.

2.1.1 Выбор средств реализации

В качестве средства для хранения данных была выбрана документоориентированная база данных MongoDB, в связи с тем, что задача хранения данных для обучения нейронной сети не требует больше чем одну сущность (коллекцию). Наряду с этим, документоориентированные базы данных обладают большей производительностью при простых запросах без операции соединения реляционной алгебры при получении данных [21].

MongoDB (от англ. humongous — огромный) — документоориентированная система управления базами данных (СУБД) с открытым исходным кодом, не требующая описания схемы таблиц. Классифицирована как NoSQL, использует JSON-подобные документы [21].

2.1.2 Схема данных

База данных состоит из двух коллекции:

– Tweets – коллекция для хранения сообщений (набора данных);

– Users – коллекция для хранения информации о пользователях

Схема базы данных показана на рисунке 2.2.

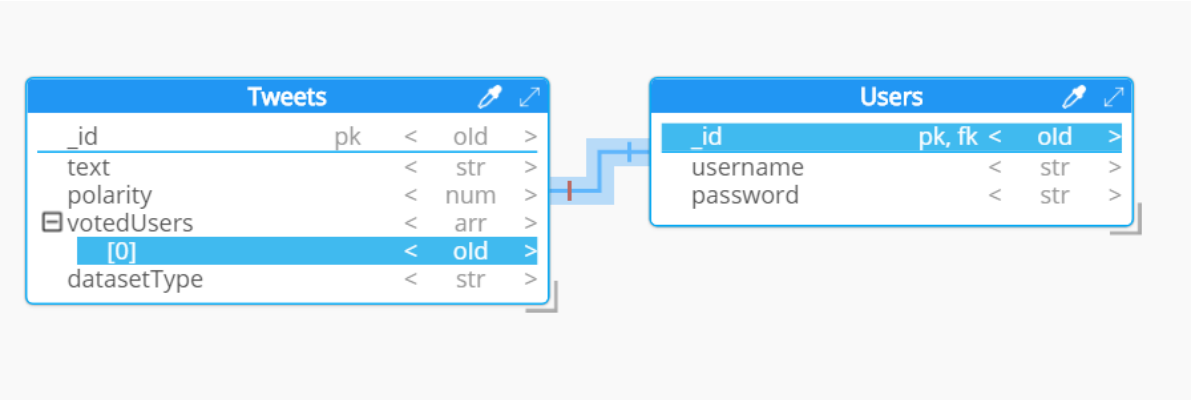


Рисунок 2.2 – База данных для хранения тренировочного набора данных и информации о пользователях

MongoDB при записи данных в коллекцию автоматически добавляет уникальное поле «_id».

В коллекции «Tweets»:

Поле «text» имеет строковый тип данных и хранит текст сообщения.

Поле «polarity» имеет числовой тип данных и хранит эмоциональный класс в бинарном виде (0 - отрицательный, 1-положительный).

Поле «votedUsers» является массивом и хранит «идентификатор пользователей» проголосовавших за один из эмоциональных классов.

Поле «datasetType» имеет строковый тип данных и хранит информацию о том, к какому набору данных относится сообщение.

В коллекции «Users»:

Поле «username» имеет строковый тип данных и хранит имя пользователя.

Поле «password» имеет строковый тип данных и хранит пароль пользователя.

2.1.3 Создание коллекций для хранения данных

В первую очередь, необходимо разделить данные на тестовую и обучающую выборку из корпуса текста указанной в работе [21] состоящий из 226834 русскоязычных сообщений пользователей социальной сети в формате базы SQL, для проверки работы классификатора на отличающихся от тренировочных данных. Чтобы создать тестовую выборку для проверки результатов необходимо отделить по 10,000 записей с каждой выборки. Таким образом, в тестовой выборке будет по 10,000 записей, относящихся к каждому из эмоциональных классов. Далее, необходимо сбалансировать количество положительных и отрицательных данных оставив в обучающей выборке по 100,000 записей на каждый эмоциональный класс.

Далее необходимо установить необходимое ПО для развертывания локального хранилища.

После успешного развертывания хранилища необходимо создать коллекцию. Создать коллекцию можно двумя методами:

- через консоль сервера;
- через интерфейсы программных языков.

Консоль сервера базы данных позволяет создавать коллекции и проводить стандартные CRUD (Create Read Update Delete) операции над данными. Пример консольных запросов к серверу MongoDB представлен на рисунке 2.3.

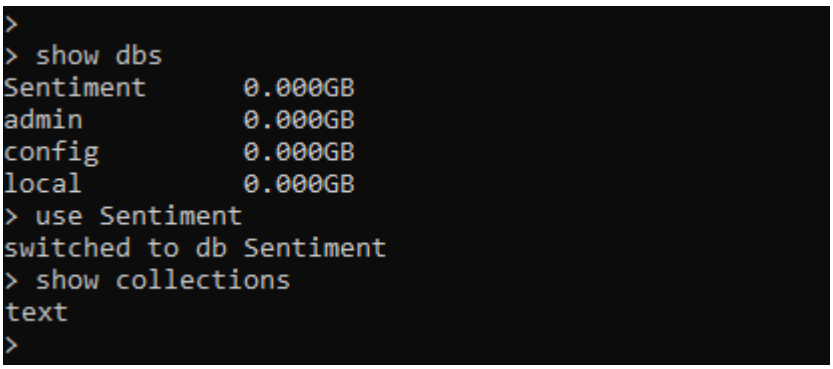


Рисунок 2.3 – Консольный режим доступа к MongoDB

Способ взаимодействия с базой данных через интерфейсы программных языков позволяет автоматизировать указанные выше процессы. На данный момент имеется набор библиотек для популярных языков программирования, позволяющие взаимодействовать с базой данных MongoDB.

Для того, чтобы записать разделенные по файлам данные в базу данных, был разработан небольшой скрипт, позволяющий итерационно записать каждое сообщение.

Скрипт для создания и заполнения коллекции тренировочными данными представлен в листинге приложении А.

2.2 Классификатор

Классификатор основан на архитектуре свёрточной нейронной сети

Классификатор состоит следующих классов:

– Класс «PrepareData» для подготовки и посимвольного кодирования данных;

– Класс «Train» отвечает за обучение нейронной сети;

– Класс «CNN» для создания модели сверточной нейронной сети;

– Класс «Evaluation» для классификации пользовательских сообщений.

Диаграмма классов представлена на рисунке 2.4.

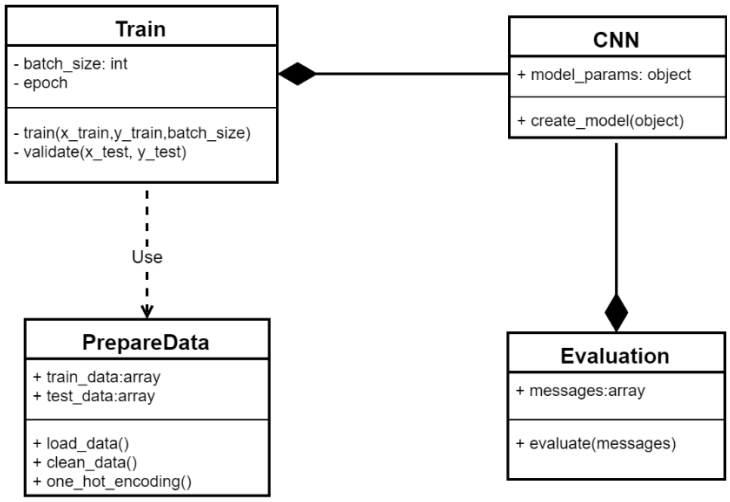


Рисунок 2.4 – Классы компонента «Классификатор».

2.2.1 Проектирование архитектуры классификатора

Классификатор представляет собой свёрточную нейронную сеть для решения задачи бинарной классификации. Архитектура нейронной сети авторов статьи [3] была модифицирована для работы с русскоязычным текстом. Были изменены методы считывания данных, добавлен русский алфавит в словарь метода кодирования, также были изменены параметры сети для обучения на коротких текстах (140 символов).

Модель свёрточной нейронной сети состоит из нескольких свёрточных слоев с различной шириной фильтра (в этой реализации 25 фильтров размером 1, 50 фильтров с размером 2, 175 фильтров размера 7 - высота всегда такая же, как и длина алфавита). Каждый из них принимает в качестве входных данных одно слово из предложения за одну итерацию процесса обучения.

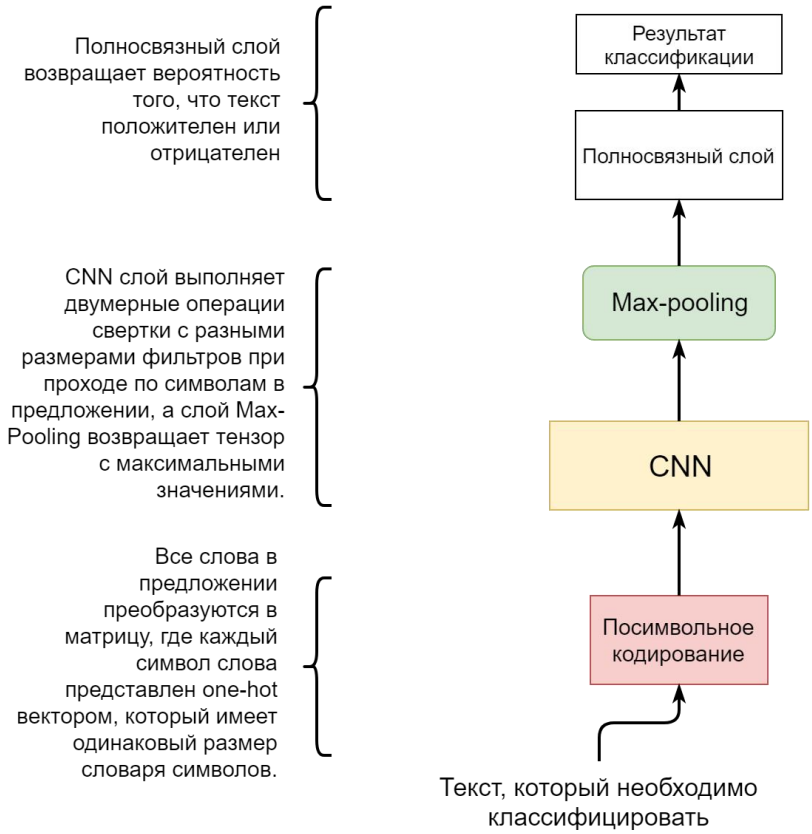


Рисунок 2.5 – Схема модели свёрточной нейронной сети для бинарной классификации текста на уровне символов [12]

Более подробная таблица слоев свёрточной нейронной сети описана в Таблице 2.1. Между полносвязными слоями 6, 7 и 7, 8 использовалась dropout регуляризация с параметром p = 0.5. При тестировании сети параметр p = 1.0. Функция активации на всех слоях кроме последнего — Relu, на последнем — Softmax. Данная архитектура описана в статье [5].

Таблица 2.1 – Слои свёрточной нейронной сети

№ слоя	Количество нейронов	Размер фильтра	Размер фильтра max-pooling
1	256	7	3
2	256	7	3
3	256	3	—
4	256	3	—

Продолжение таблицы 2.1.

5	256	3	—
6	256	3	3
7	1024	—	—
8	1024	—	—
9	Число классов	—	—

2.2.2 Выбор средств разработки

Реализация модели выполнена с помощью языка программирования Python и библиотеки Tensorflow и Keras.

Python стал общепринятым языком для многих сфер применения науки о данных (data science). Он сочетает в себе мощь языков программирования с простотой использования предметно- ориентированных скриптовых языков типа MATLAB или R. В Python есть библиотеки для загрузки данных, визуализации, статистических вычислений, обработки естественного языка, обработки изображений и многого другого. Этот обширный набор инструментов предлагает специалистам по работе с данными (data scientists) большой набор инструментов общего и специального назначения. Одним из основных преимуществ использования Python является возможность напрямую работать с программным кодом с помощью терминала или других инструментов типа Jupyter Notebook [22].

В связи с простотой использования и увеличения скорости разработки была выбрана библиотека Keras. Keras - это высокоуровневый API нейронных сетей, написанный на Python и способный работать поверх TensorFlow, CNTK или Theano. Он был разработан с упором на возможность быстрого экспериментирования [23].

Основная структура данных Keras - это модель, способ организации слоев. Простейшим типом модели является последовательная модель,

линейная совокупность слоев. Для более сложных архитектур есть возможность использовать функциональный API Keras, который позволяет создавать произвольные графики слоев.

Пример создания и складывание слоев:

```
model.add(Convolution2D(256,1,7))
model.add(MaxPooling2D(pool_size=(1,3)))
```

В примере выше указан код для создания 2 слоев: свёрточного и слоя объединения максимальных значений (max-pooling).

2.2.3 Реализация

Основываясь на статью [5], было выбрано кодирование слов на уровне символов. Кодирование выполняется путем назначения алфавита размера m для языка ввода, а затем каждый символ из корпуса текста входящий в состав алфавита преобразуется в последовательность чисел (one-hot-encoding). Последовательность символов в рамках одного текста фиксируется длиной l_0. Любой символ, превышающий длину l_0, игнорируется, и любые символы которые не находятся в алфавите, включая пустые символы, преобразуются как векторы с нулевым значением.

Алфавит, используемый в модели, состоит из 91 символов, в том числе из 33 букв русского алфавита, 26 букв английского алфавита, символов пунктуации и вспомогательных символов:

```
alphabet:!\#%&'*+,./0123456789:;<=>?@^_abcdefghijklmnopqrstuvwxyz«»абвг
дежзийклмнопрстуфхцчшщъыьэюя
```

Пример кода для посимвольного кодирования:

```
def one_hot_encoding(data):
    data = data.lower()
    alphabet
=':!\#%&'*+,./0123456789:;<=>?@^_abcdefghijklmnopqrstuvwxyz\
                «»абвгдежзийклмнопрстуфхцчшщъыьэюя '
    onehot_encoded = list()
```

```
    try:
        char_to_int = dict((c, i) for i, c in enumerate(alphabet))
        print(char_to_int.values())
        integer_encoded = []
        for char in data:
            if char in char_to_int.keys():
                integer_encoded.append(char_to_int[char])
        # one hot encode
        for value in integer_encoded:
            letter = [0 for _ in range(len(alphabet))]
            letter[value] = 1
            onehot_encoded.append(letter)
    except Exception:
        pass
    return onehot_encoded
```

Следующим шагом является реализация необходимых слоев в коде с помощью библиотеки Keras. Как можно заметить ниже в примере реализации кода на Keras, все что необходимо сделать, это передать параметры в функцию `add()`:

```
def model_defn():
    print('Build model...')
    fully_connected = [1024,1024,1]
    model = Sequential()
    model.add(Convolution2D(256,67,7,input_shape=(1,67,1014)))
    model.add(MaxPooling2D(pool_size=(1,3)))
    model.add(Convolution2D(256,1,7))
    model.add(MaxPooling2D(pool_size=(1,3)))
    model.add(Convolution2D(256,1,3))
    model.add(Convolution2D(256,1,3))
    model.add(Convolution2D(256,1,3))
    model.add(Convolution2D(256,1,3))
    model.add(MaxPooling2D(pool_size=(1,3))
    model.add(Flatten())
    model.add(Dense(fully_connected[0]))
    model.add(Dropout(0.5))
    model.add(Activation('relu'))
```

```
            model.add(Dense(fully_connected[1]))
            model.add(Dropout(0.5))
            model.add(Activation('relu'))
            model.add(Dense(fully_connected[2]))
            model.add(Activation('sigmoid'))
            sgd = SGD(lr=0.01, decay=1e-5, momentum=0.9, nesterov=True)
            model.compile(loss='binary_crossentropy', optimizer=sgd,
class_mode="binary")
            return model,sgd
```

Обучение модели реализуется при помощи функции `model.fit()`:

```
model.fit(X_train, y_train, epochs=10, batch_size=32, verbose=1,
validation_data=(X_val, y_val))
```

параметры функции `fit()`:

– `x_train` и `y_train` – входные данные и разметки классов;

– `epochs` – количество эпох для обучения сети;

– `batch_size` – количество примеров, обрабатываемой сетью за одну итерацию;

– `verbose` – подробный режим вывода информации об обучении (1-полный вывод, 0-сокращенный);

– `validation_data` – данные из тестовой выборки, для тестирования модели на новых данных.

Функция потерь (или целевая функция или функция оценки эффективности) является одним из двух параметров, необходимых для компиляции модели.

Вторым обязательным параметром для обучения является метрика точности. Метрика точности - это функция, которая используется для оценки производительности модели. Метрические функции должны предоставляться в параметре метрики при компиляции модели. Для оценки производительности модели была использована F-мера. Для вычисления F-

меры используются точность (precision) и полнота (recall), а также следующие понятия о результатах [24]:

– TP (True Positive) — истиноположительный.
– FP (False Positive) — ложноположительный.
– FN (False Negative) — ложноотрицательный.
– TN (True Negative) — истиноотрицательный.

Точность показывает отношение верно угаданных объектов класса ко всем объектам [24]:

$$precision = \frac{TP}{TP+FP} \quad (2.1)$$

Полнота показывает отношение верно угаданных объектов класса ко всем представителям этого класса [26]:

$$recall = \frac{TP}{TP+FN} \quad (2.2)$$

F-мера является средне гармонической точности и полноты [24]:

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision+recall} \quad (2.3)$$

Классификация пользовательских текстов выполняется с использованием функции `predict()`:

```
model.predict(X_val, batch_size=10)
```

Параметры функции `predict()`:

– `x_val` – текст, который нужно классифицировать;
– `batch_size` – количество примеров, обрабатываемой сетью за одну итерацию (должен совпадать с `batch_size` обученной модели).

2.3 Веб-приложение

Реализованное веб-приложение предоставляет пользователю следующие возможности:

– Классификация текста в режиме ввода текста;

– Классификация текста найденных в социальных сетях, путем поиска сообщений определенного пользователя и выбора количества последних сообщений;

– Подробный просмотр результатов классификации;

– Формирование новых тренировочных данных, путем разметки сообщений из социальной сети на основе голосования.

На рисунке 2.6 представлены варианты использования веб-приложения.

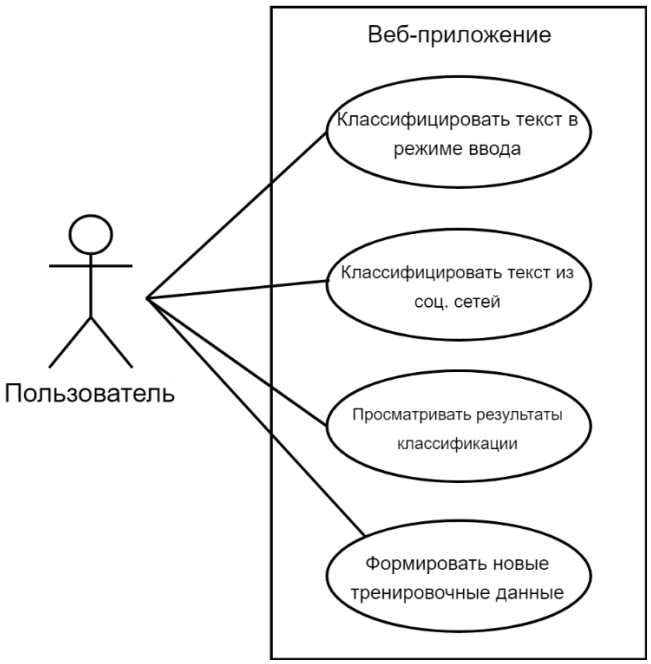


Рисунок 2.6 – Варианты использования веб-приложения

2.3.1 Проектирование схемы интерфейса

Пользовательский интерфейс веб-приложения представляет собой SPA (Single Page Application) приложение со следующей структурой интерфейса:

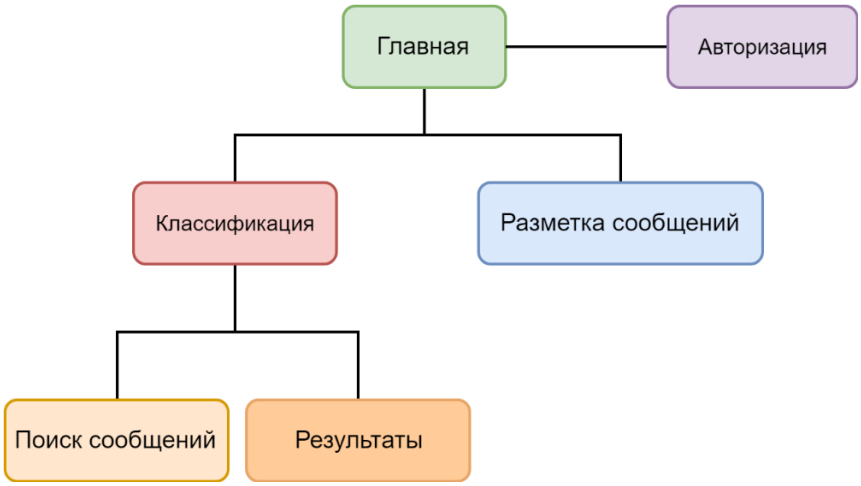


Рисунок 2.7 – Структура веб-приложения

Главная страница несет информационный характер и служит для знакомства пользователя с приложением. Макет главной страницы представлен на рисунке 2.8

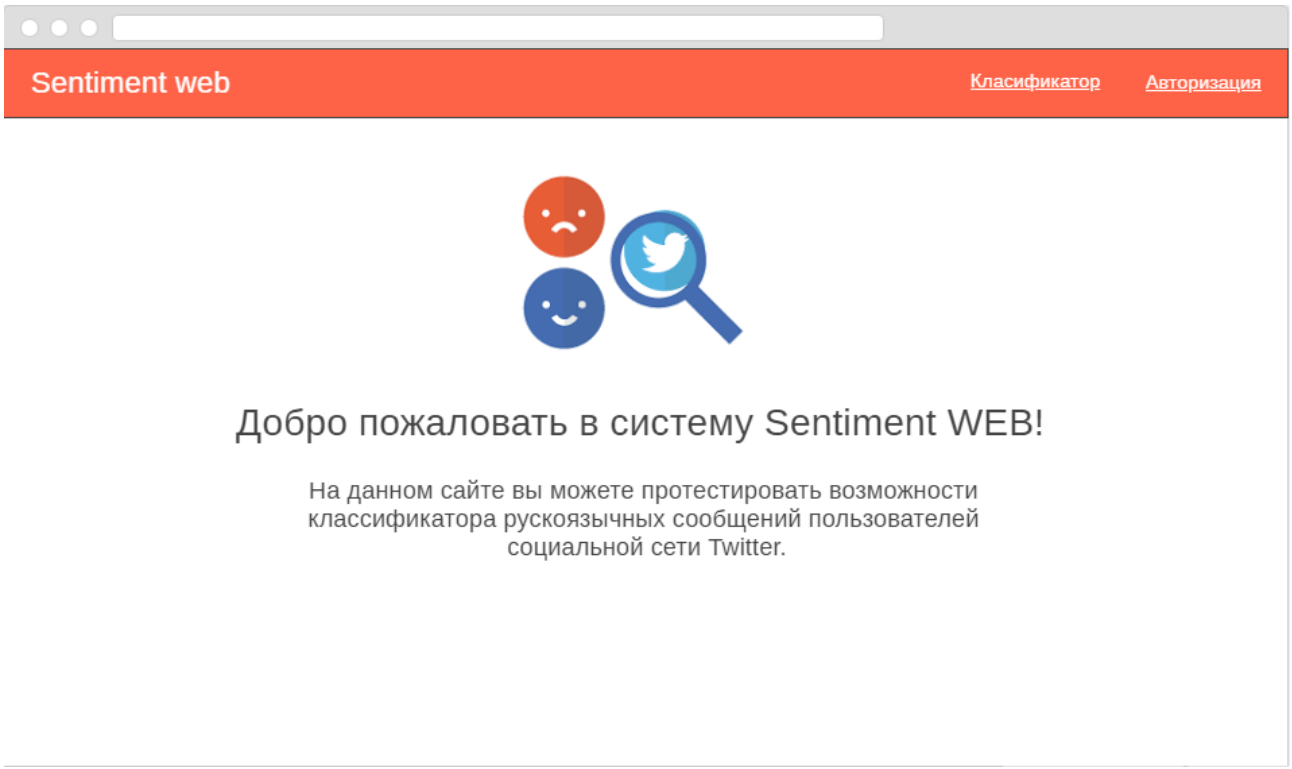


Рисунок 2.8 – Главная страница

На рисунке 2.8 можно заметить, что для навигации по сайту существует меню в правом верхнем углу.

Страница «Авторизации» служит для аутентификации пользователя, для хранения статистики разметки сообщений пользователей, чтобы на странице разметки сообщений показывались тексты, за которые пользователь еще не голосовал. Также страница «Авторизации» включает в себя функционал регистрации, в случае, если пользователя нет в системе. Сделано это с целью упрощения идентификации пользователей. Макет страница авторизации представлен на рисунке 2.9. Модальное окно согласия на регистрацию в системе представлена на рисунке 2.10.

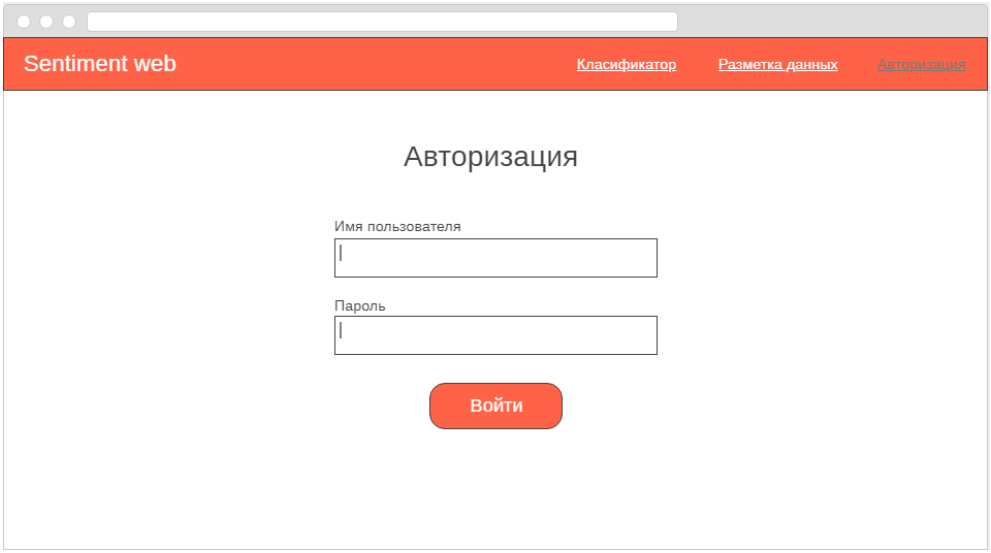


Рисунок 2.9 – Страница авторизации

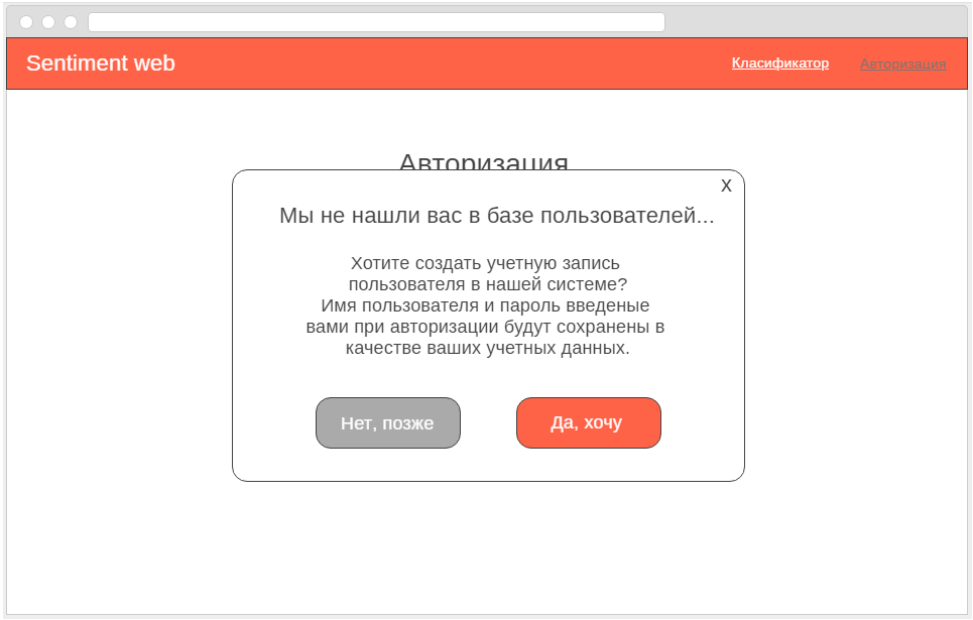


Рисунок 2.10 –Модальное окно согласия на регистрацию в системе

Страница «Классификации» включает в себя отображения интерфейса для поиска сообщений пользователей социальной сети и показ результатов классификации. Макет страницы классификации представлен на рисунке 2.11.

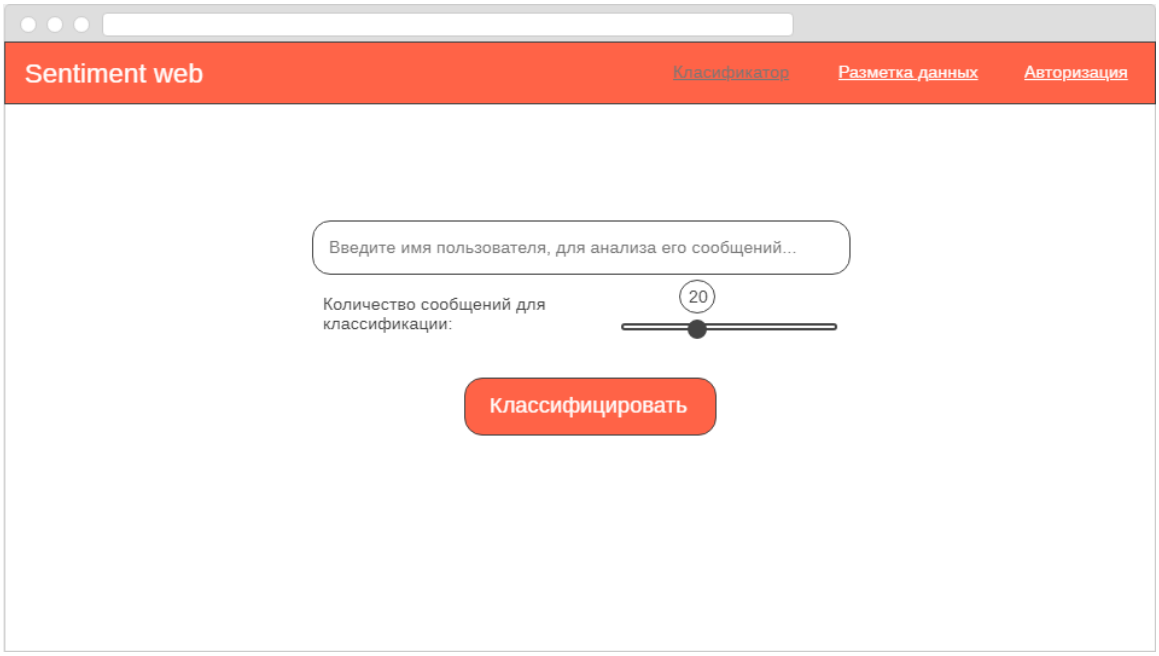


Рисунок 2.11 – Страница классификации

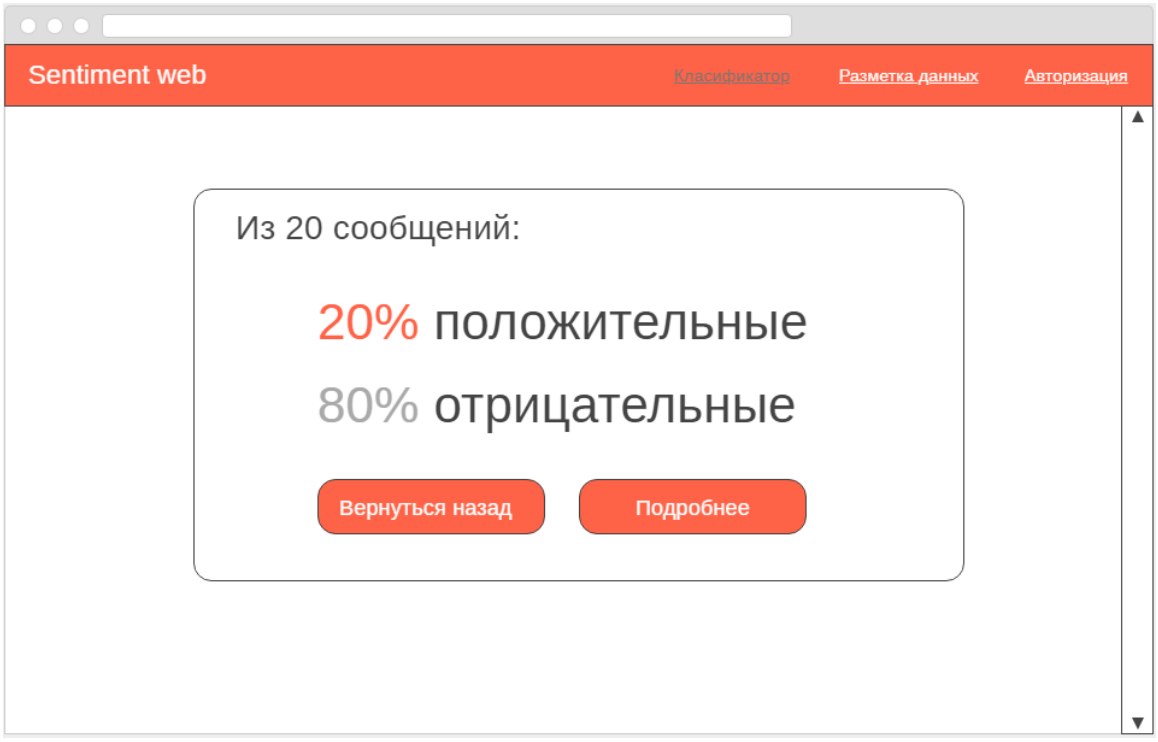


Рисунок 2.12 – Результаты классификации

Страница «Разметки сообщений» содержит функционал для разметки сообщений. Макет страницы разметки сообщений представлен на рисунке 2.13.

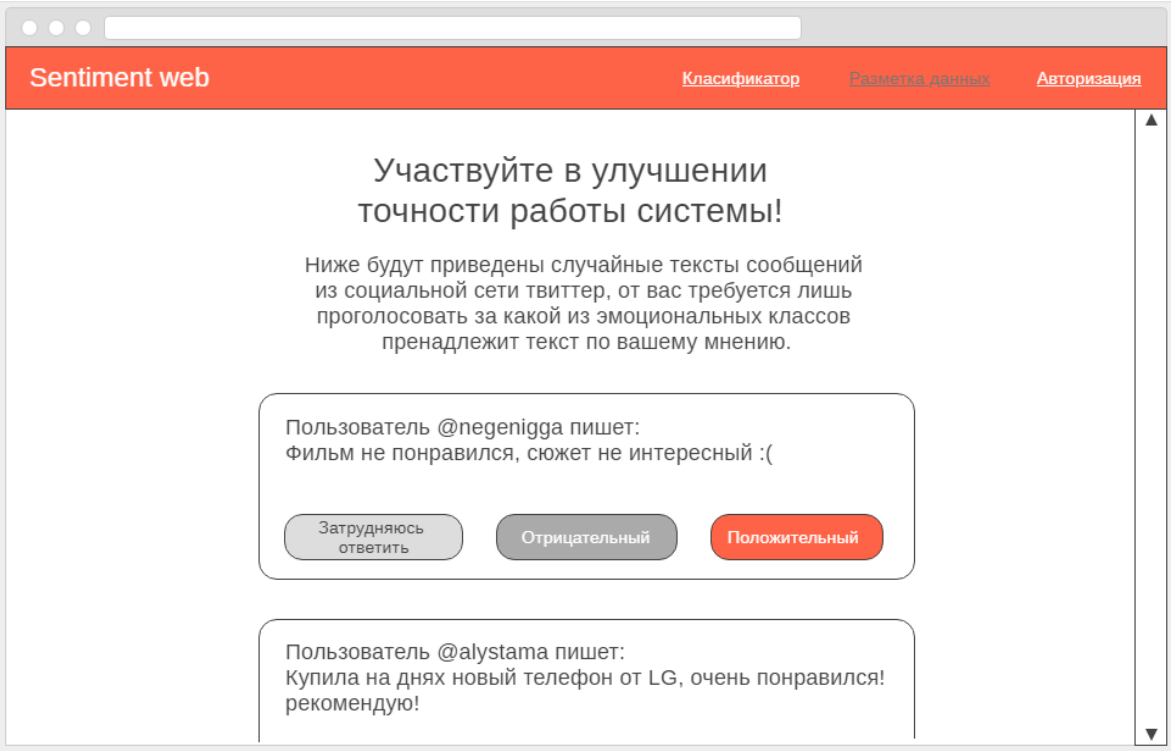


Рисунок 2.13 – Страница разметки текста сообщений

Алгоритм работы компонентной части для обновления тренировочных данных

Для того, чтобы улучшать точность работы классификатора, нейронную сеть необходимо периодически обучать на новых тренировочных данных.

Разработанное веб-приложение наряду с функционалом, позволяющим классифицировать тексты сообщений, позволит самим пользователям пополнять набор тренировочных данных. Так как оценка эмоционального тона будет основываться на субъективное мнение пользователя, было решено что отнесение сообщение к одному из эмоциональных классов будет на основе голосования.

В интерфейсе (рисунок 2.13) выводятся сообщения пользователей социальной сети и кнопки для отнесения сообщения к одному из

эмоциональных классов. Также существует кнопка «Затрудняюсь ответить» для тех случаев, когда сложно определить эмоциональный окрас сообщения. В случае нажатия пользователем кнопки «Затрудняюсь ответить» сообщение будет размечено как нейтральное. Если сообщение помечено как нейтральное больше пяти раз, то сообщение вносится в архив и не участвует в улучшении модели классификации.

Алгоритм работы приложения: пользователям выводится определенное сообщение на русском языке, случайным образом загруженная из социальной и дается возможность отнести сообщения к одному из классов эмоции путем голосования за одну из них. Когда собирается определенное количество (по умолчанию 10) голосов, сообщение попадает в общую базу размеченных тренировочных данных в зависимости от результатов голосования. На рисунке 2.14 показана блок-схема работы приложения.

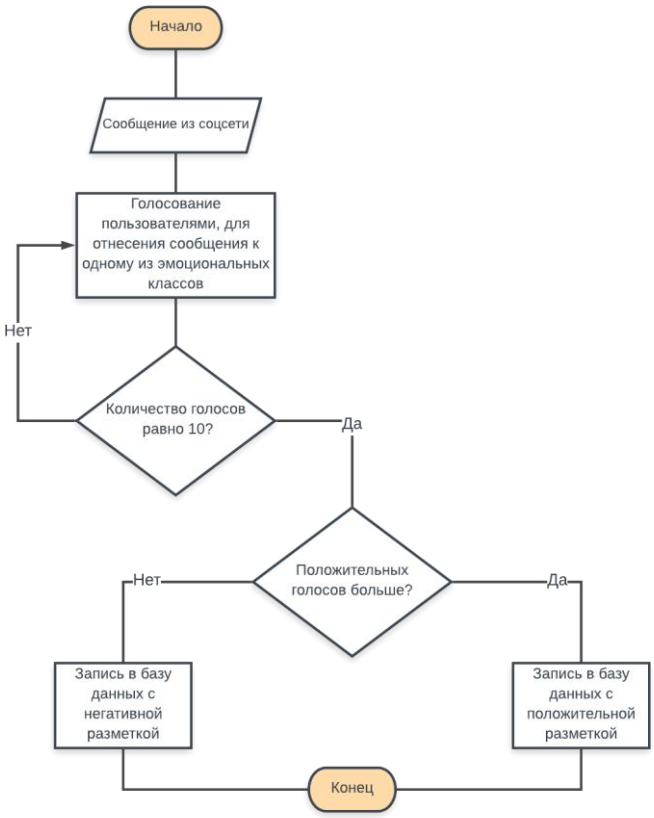


Рисунок 2.14 – Алгоритм формирования пользователем тренировочных данных.

2.3.2 Выбор средств реализации

Реализация веб-приложения выполнена с использованием веб-фреймворка Angular версии 5.0.

Angular — это фреймворк, который позволяет быстро и удобно создавать одностраничные веб-приложения. По своей сути он использует язык программирования TypeScript (производное от Javascript). TypeScript является надстройкой для добавления строгой типизации для языка Javascript.

Взаимодействие между клиентом и сервером осуществляется на основе сервис-ориентированного архитектурного стиля. На данный момент наиболее востребованы REST-сервисы.

Для реализации REST API, с помощью которого веб-клиент взаимодействует с API социальной сети Twitter и основным функционалом, была выбрана серверная технология NodeJS. NodeJS — среда выполнения программного кода на языке Javascript за пределами браузера. Архитектура веб-сервиса реализована с использованием библиотеки Express.js в среде выполнения NodeJS [25].

2.3.3 Реализация

Приложения на фреймворке Angular 5 строятся на основе компонентных частей как веб-компоненты. Веб-компонент включает в себя модель и представление. В модели компонента реализуется основной логика приложения. Представление служит для отображения данных приходящих с модели и отправкой данных в модель. Связь между представлением и моделью реализуется посредством двусторонней либо односторонней привязки данных.

Компоненты служат для разделения функциональности. Например, для функции авторизации пользователя следует создать отдельный компонент «AuthComponent», а для вывода сообщений пользователей компонент

«Tweets». Этот подход позволяет легко и удобно масштабировать проект, а также улучшать уже существующий функционал.

Файловая структура проекта выглядит следующим образом:

```
.
|-- app
|   |-- app.component.css
|   |-- app.component.html
|   |-- app.component.spec.ts
|   |-- app.component.ts
|   `-- app.module.ts
|-- assets
|-- environments
|   |-- environment.prod.ts
|   `-- environment.ts
|-- favicon.ico
|-- index.html
|-- main.ts
|-- polyfills.ts
|-- styles.css
|-- test.ts
|-- tsconfig.app.json
|-- tsconfig.spec.json
`-- typings.d.ts
```

app/app.component.{ts,html,css,spec.ts} — специфицирует AppComponent компонент html-шаблоном, стилями и юнит-тестами. Это корневой компонент, для которого по мере развития приложения появится дерево вложенных компонентов.

app/app.module.ts — специфицирует AppModule. Корневой модуль, который сообщает Angular, как будет собрано приложение. Для начала в нем будет объявлен только AppComponent. Впоследствии по мере необходимости в этом файле можно объявлять другие компоненты.

assets/* — директория, в которой размещаются изображения и все медиа файлы, которую необходимо скопировать в конечную директорию сборки при создании приложения.

environments/* — эта директория содержит файлы целей сборки (dev или prod режимы), каждый из которых экспортирует простые env-переменные конфигурации, необходимые для использования в приложении.

main.ts — точка входа вашего приложения. По умолчанию, приложение компилируется в поставке с JIT-компилятором. Данный файл загружает корневой модуль приложения (AppModule) и запускает его в браузере.

polyfills.ts — различные браузеры имеют различные уровни поддержки тех или иных веб-стандартов. Полифиллы помогают нормализовать эти различия.

styles.css —файл, где хранятся глобальные стили.

test.ts — это точка входа всех юнит-тестов. Этот файл имеет настраиваемую конфигурацию, но как правило редко редактируется.

tsconfig.{app|spec}.json — конфигурация компилятора TypeScript описывается в файле tsconfig.app.json, для юнит-тестов же используется конфигурация tsconfig.spec.json.

Реализация веб-сервиса

Сервер приложений представляет собой набор REST веб-сервисов, реализующих операции чтения, создания и удаления данных, которые соответствуют методам протокола HTTP.

Интерфейс для доступа к социальной сети Twiter API

Социальная сеть Twitter предоставляет доступ через публичный API к основному функционалу для работы с ним. Для того чтобы получить доступ к публичному API достаточно создать приложение в их системе управлении

пользовательскими приложениями по адресу https://apps.twitter.com/. После создания приложения системой генерируются ключи для доступа к публичному API. Далее, разрабатывается модуль приложения для получения сообщений пользователей при помощи REST-запросов на сервер социальной сети. Модуль для аутентификации и получения сообщений пользователей реализовано при помощи технологии NodeJS, с использованием библиотеки Express.js и Twitter.js.

В файле routes.js реализованы веб-сервисы для классификации сообщений, подсчета голосов пользователей для обновления тренировочных данных:

– GET /tweets/ – получение сообщений пользователей из социальной сети Twitter;

– POST /classify – классификация сообщений пользователей социальной сети Twitter;

– POST /signin – регистрация нового пользователя в системе;

– POST /signin – аутентификация пользователя в системе.

Исходный код реализации веб-сервисов приведен в приложении А.

2.4 Планировщик обучения нейронной сети

Планировщик является частью операционной системы, которая отвечает за одновременное выполнение задач, потоков, процессов. Планировщик выделяет время процессорного времени, память, стек и другие ресурсы. Планировщик может принудительно управлять потоком (например, таймером, или, когда появляется поток с более высоким приоритетом) или просто ждать, пока сам поток явно (путем вызова определенной системной процедуры) или неявно (после завершения) будет дать управление планировщику.

В операционных системах в комплекте идут следующие планировщики задач:

– Cron – планировщик задач для операционных систем семейства Unix(Linux,MacOS,FreeBSD);

– Планировщик заданий Windows для операционных систем Windows и Windows Server.

2.4.1 Выбор средств реализации

По причине того, что приложение будет развертываться на операционной системе Ubuntu Server, которая относится к операционным системам семейства Unix, подходящим решением является планировщик задач cron.

Cron – классическая компьютерная программа входящий в состав операционных систем класса Unix, использующийся для периодического выполнения заданий в определённое время. Регулярные действия описываются инструкциями, помещенными в файлы crontab и в специальные директории [26].

Crontab (сокращение от «cron table») представляет собой список команд, которые планируется запустить с регулярными временными интервалами в компьютерной системе. Команда crontab открывает таблицу планировщика для редактирования и позволяет добавлять, удалять или изменять запланированные задачи.

2.4.2 Реализация

Планировщик обучения нейронной сети реализован с использованием механизма планировщика запуска операционной системы Linux. Серверная операционная система Linux, где будет развернуто приложение, поддерживает планировщик задач cron.

Планировщик обучения является скриптом, оформленный в соответствии с правилами и синтаксисом bash-скриптов выполняющихся в

операционных системах Unix, который запускает программу, написанную на языке Python для запуска процесса обучения нейронной сети с новыми тренировочными данными.

Чтобы отредактировать файл crontab для внесения задачи необходимо выполнить команду:

```
$ crontab –e
```

Синтаксис crontab:

```
1 2 3 4 5 /path/to/command arg1 arg2
```

Где цифры обозначают:

- 1: Минуты (0-59)
- 2: Часы (0-23)
- 3: День (0-31)
- 4: Месяц (0-12 [12 == Декабрь])
- 5: День недели (0-7 [7 или 0 == sunday])
- /path/to/command — имя скрипта или команды для выполнения по расписанию

Для запуска скрипта обучения каждое воскресенье в 00:00 необходимо записать в файл crontab следующую строку:

```
0 0 * * 7 /home/cloud/sentiment/scripts/train_cron.py
```

Выполняемый скрипт проверяет наличие новых данных в базе данных, если количество новых данных не превышает 200, то скрипт завершает работу. В противном случае скрипт запустит процесс обучение на новом тренировочном наборе данных.

3 Тестирование

Для обучения данные были разделены на обучающую (90%) и тестовую (10%) выборку.

Результаты обучения оценивались с помощью метрики достоверности (accuracy), т. е. считалась доля верно классифицированных объектов к общему количеству объектов.

Ниже в таблице приведены итоги тестирования описанной свёрточной нейронной сети на данных из статьи [27].

Таблица 3.1 – Итоги тестирования классификации

Количество данных	**Количество фильтров**	**Количество итераций, шаг**	**Точность, %**	**Потери, %**	**Время обучения, мин**
Без предобработки данных					
10000	32	1000	~63,5	0,71	15
20000	32	1000	~64,2	0,66	30
230000	32	2000	~74,03	0,57	120
С применением стемминга при обработке данных					
230000	64	1400	~74,12	0,52	60
С применением лемматизации при обработке данных					
230000	128	1400	~76,37	0,48	240

Из таблицы 3.1 видно, что с увеличением количества данных точность классификации растет. Это связано с тем, что для свёрточных нейронных сетей необходимо большое количество данных для выявления карты признаков [20].

На рисунках 3.1 и 3.2 показаны графики точности обучения и тестирования с количеством данных равным 230000 и количеством фильтров равным 32.

Точность данного процесса обучения составила 74,03%.

Данные для тестирования не содержатся в обучающей выборке

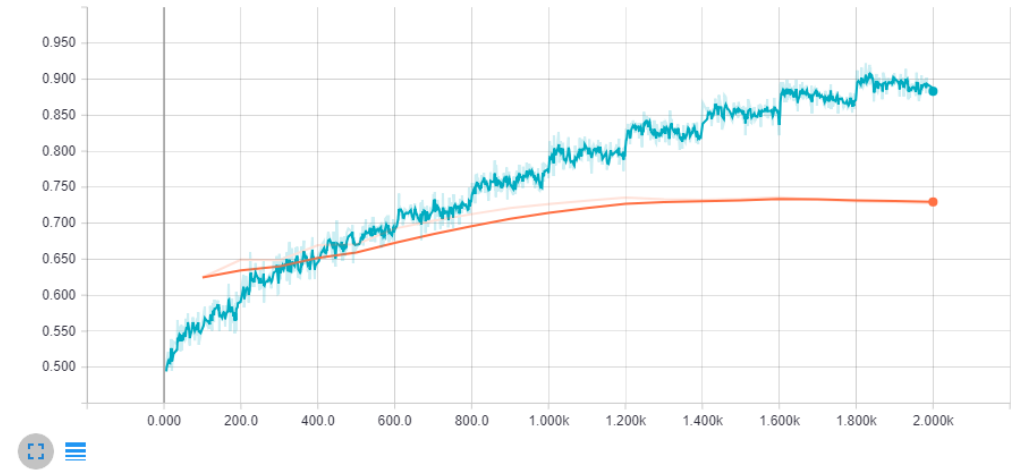


Рисунок 3.1 – Точность обучения и тестирования

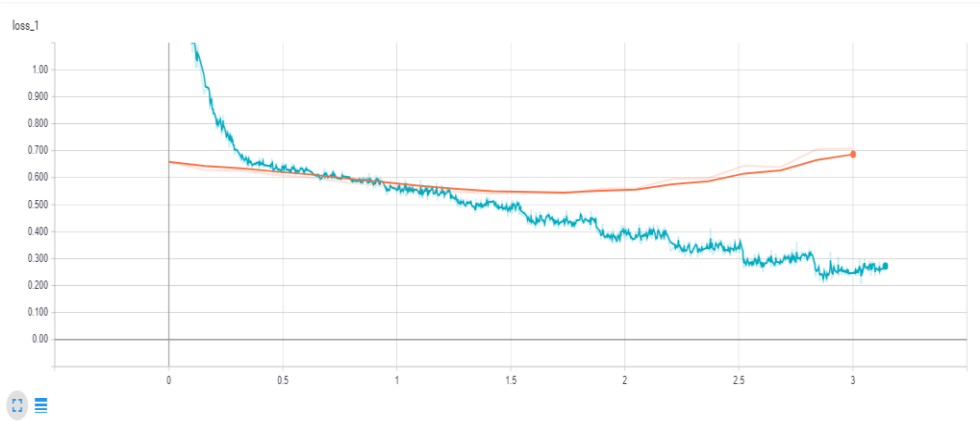


Рисунок 3.2 – Значение потерь при обучении и тестировании

Классификатор на основе символьного кодирования входных данных показал результат ~74,12% на тестовом наборе данных. График потерь показывает, что после в интервале от 1200 до 1500 итерации наблюдается повышение значения функции потерь. Это свидетельствует о том, что нейронная сеть начала переобучаться (слишком быстрая аппроксимация данных) [24]. Для предотвращения таких случаев, в библиотеке Keras предусмотрены функции ранней остановки обучения (earlystopping), для того чтобы избежать деградации модели [23].

При помощи использования стемминга для тренировочных данных, реализованной в библиотеке pymorphy2 [9], удалось достичь потерь равной

0,52 при точности 74,12%, при этом уменьшив время обучения до 60 минут. Графики обучения с использованием стемминга показаны на рисунках 3.3 и 3.4.

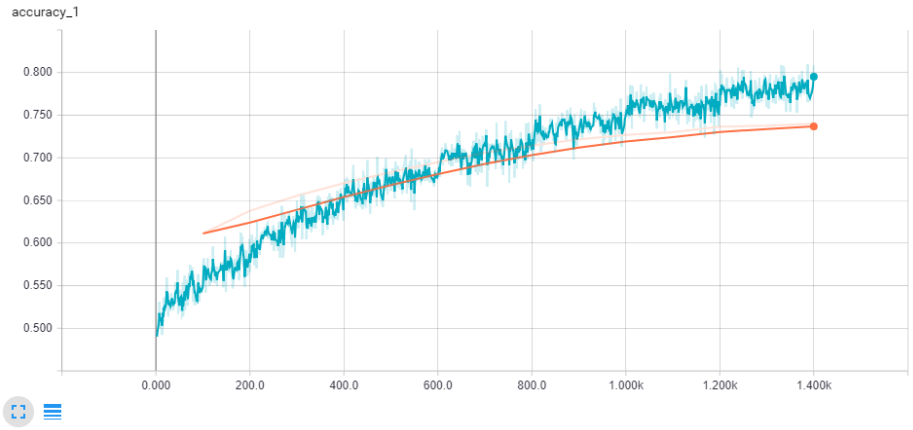


Рисунок 3.3 – Точность обучения и тестирования с использованием стемминга для данных.

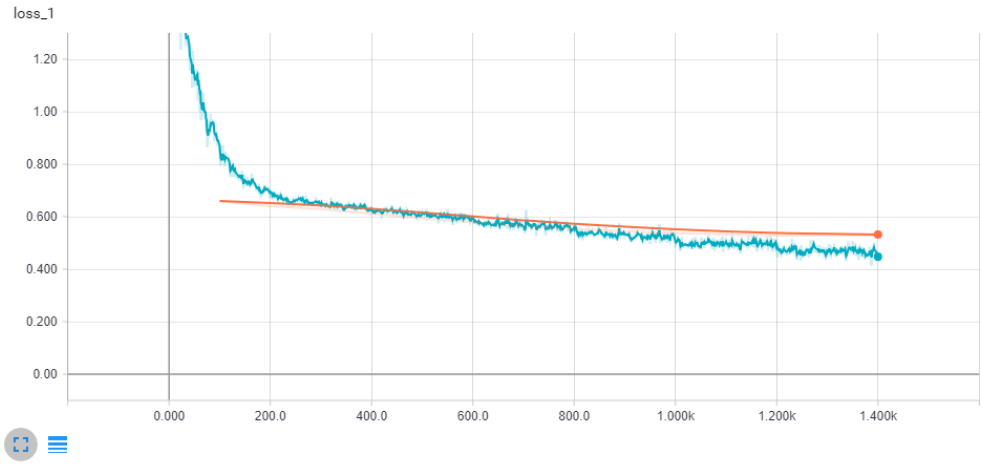


Рисунок 3.4 – Значение потерь при обучении и тестировании с использованием стемминга для данных.

С использованием метода лемматизации при обработке тренировочных данных и увеличения количества фильтров до 128, удалось достичь точности классификации 76,37%, при этом потери составили 0,48. Из-за увеличения количества фильтров, время обучения увеличилось до 240 мин.

4 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Маркетологу необходимо постоянно быть в курсе восприятия рынком предлагаемого продукта или услуги. Продуктовые компании получают отзывы на естественном языке о продуктах из опросов, жалоб клиентов, отзывов о магазине. В связи с этим существует потребность в автоматизировании анализа отзывов.

В данной работе были реализованы различные архитектуры нейронных сетей для того, чтобы маркетолог мог сократить этот огромный объем информации и получить сжатый информативный отчёт.

В данной работе анализируется тональность текста: какие эмоции испытывал человек, оставивший данный отзыв, то есть стоит задача бинарной классификации — определить, позитивный или негативный отзыв оставил клиент.

В данном разделе осуществляется оценка коммерческого потенциала, перспективности и альтернатив проведения разработки с позиции ресурсоэффективности и ресурсосбережения, а также планирование и формирование бюджета.

4.1 Предпроектный анализ

4.1.1 Потенциальные потребители результатов проекта

В век информации получить данные уже обработанных отзывов, рецензий, комментариев очень легко. Однако анализировать такие массивы информации чрезвычайно сложно.

Целевым рынком данной разработки являются маркетологи, компании, которым в первую очередь важно получить обратную связь о своем продукте. Что позволит им в дальнейшем улучшить качество продукта или

предоставления услуг, скорректировать курс бизнеса.

		Виды автоматизации анализа отзывов			
		Веб-интерфейс для работы с системой	*Сервер для актуализации корректировки модели нейронной сети*	*ИнтеграциясBI (Business Intelligence)*	*Доработка системы*
Размер организации	*Крупные*				
	Средние				
	Мелкие				

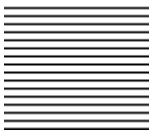
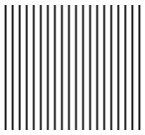
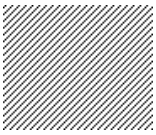
 - организация А,  - организация В,  - организация С

Рисунок 4.1 – Карта сегментирования рынка разработки

В малых предприятиях, достаточно веб-интерфейса для взаимодействия с системой, без затрат на серверное оборудование и их обслуживание. Для средних компаний предусматривается возможность доработки системы под конкретные нужды. Крупные компании могут позволить себе весь комплекс программного обеспечения, для того чтобы получить наиболее тщательный анализа сообщений клиентов.

4.1.2 Анализ конкурентных технических решений

Разработка программного обеспечения для анализа тональности текста сообщений пользователей социальной сети основывалась на анализе функционала существующих систем и выборе наиболее актуальных решений для автоматизации процесса анализа текстов.

В качестве конкурентов целесообразно рассмотреть онлайн сервисы для определения тональности текстов.

Таблица 4.1– оценочная карта сравнения конкурентных технических решений

№	Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
			Бф	Бк1	Бк2	Кф	К1	К2
	1	2	3	4	5	7	8	9
Технические критерии оценки ресурсоэффективности								
1	Повышение производительности труда пользователя	0,1	8	7	8	0,8	0,7	0,8
2	Удобство в эксплуатации (соответствует требованиям потребителей)	0,13	10	8	8	1,3	1,04	1,04
7	Функциональная мощность (предоставляемые возможности)	0,15	9	6	7	1,35	0,9	1,05
8	Простота эксплуатации	0,12	10	7	8	1,2	0,84	0,96
9	Качество интеллектуального интерфейса	0,15	10	9	10	1,5	1,35	1,5
Экономические критерии оценки эффективности								
1	Конкурентоспособность продукта	0,09	7	7	8	0,63	0,63	0,72
2	Уровень проникновения на рынок	0,05	9	8	9	0,45	0,4	0,45
3	Цена	0,1	10	6	6	1	0,6	0,6
4	Послепродажное обслуживание	0,03	10	6	6	0,3	0,18	0,18
5	Финансирование научной разработки	0,03	9	8	8	0,27	0,24	0,24
6	Срок выхода на рынок	0,05	10	7	7	0,5	0,35	0,35
	Итого	**1**				**9,3**	**7,23**	**7,89**

На основании представленного выше анализа конкурентов – «OpenText» и «ToneAnalyzer», можно сделать вывод, что недостатки конкурентных решений связана с высокими затратами на разработку и послепродажное обслуживание, а также с недостаточностью функциональных возможностей, к чему можно отнести один из главных недостатков конкурентных решений – отсутствие поддержки русского языка.

Наиболее сильным конкурирующим решением можно считать «ToneAnalyzer». Его отличительным достоинством является качество интерфейса для взаимодействия, что является немаловажным фактором при работе с системой, связанной с нейронными сетями.

Преимуществами собственной разработки являются поддержка работы с русским языком, простота в эксплуатации и возможность улучшения и послепродажного обслуживания с низким уровнем затрат.

4.1.3 SWOT – анализ

SWOT–анализ применяют для исследования внешней и внутренней среды проекта. Матрица составляется на основе анализа рынка и конкурентных технических решений, и показывает сильные и слабые стороны проекта, возможности и угрозы для разработки.

Первый этап заключается в описании сильных и слабых сторон проекта, в выявлении возможностей и угроз для реализации проекта, которые проявились или могут появиться в его внешней среде. Матрица SWOT представлена в таблице 4.2.

Таблица 4.2– SWOT-анализ

		Сильные стороны		**Слабые стороны**
	С 1	Кроссплатформенность, в связи с тем, что приложение имеет веб-интерфейс	СЛ 1	Неточности при определении сложных предложений.
	С 2	Ведение статистики для улучшения анализа текстов	СЛ 2	Низкая производительность прибольших объемах данных
	С 3	Дружественный интерфейс	СЛ 3	Высокие требования к вычислительным мощностям
	С 4	Интерфейс для улучшения точности анализа самим пользователем	СЛ 4	Высокие требования к объемам памяти на серверах

Продолжение таблицы 4.2

Возможности В 1 Интеграция с системами интеллектуального анализа данных	С 5 Поддержка программного обеспечения разработчиком В1С2 Ведение статистики позволит интегрировать системы интеллектуального анализа данных, что позволит улучшить процесс анализа отзывов.	В1СЛ4Для интеллектуального анализа нужны большие количества данных, что скажется на объеме памяти серверов
В 2 Возможность непрерывного обучения модели нейронной сети для обеспечения наилучшей точности классификации отзывов	В3С4 Интерфейс для улучшения точности анализа самим пользователем позволит обеспечить наилучшую точность классификации отзывов	В3СЛ3 Выходу системы на рынок может воспрепятствовать высокие требования к вычислительным мощностям. .
В 3 Выход системы на рынок	В3С1С3С5 Кроссплатформенность, дружественный интерфейс, поддержка программного обеспечения разработчиком способствуют распространению системы на рынок.	
Угрозы У 1 Отказ пользователя от программы улучшения качества обучения нейронной сети У 2 Медленная работа системы	У1С3С4 Дружественный интерфейс, грамотно спроектированный интерфейс для улучшения точности анализа позволит избежать отказа пользователя от улучшения качества обучения нейронной сети для классификации отзывов. У2С5 Поддержка программного обеспечения разработчиком снизит вероятность медленной работы системы	У2СЛ2 Медленная работы системы может проявляться из за большого количества данных которое подается на анализ

Второй этап состоит в выявлении соответствия сильных и слабых сторон научно-исследовательского проекта внешним условиям окружающей среды. Это соответствие или несоответствие должны помочь выявить степень необходимости проведения стратегических изменений.

Соотношения параметров представлены в таблицах 4.3-4.6.

Таблица 4.3– Интерактивная матрица для сильных сторон и возможностей

Сильные стороны						
Возможности		С1	С2	С3	С4	С5
	В1	-	+	-	-	+
	В2	+	-	-	+	+
	В3	+	-	+	+	+

Таблица 4.4 – интерактивная матрица для слабых сторон и возможностей

Слабые стороны					
Возможности		СЛ1	СЛ2	СЛ3	СЛ4
	В1	+	+	-	-
	В2	-	0	-	-
	В3	-	-	0	-

Таблица 4.5 – интерактивная матрица для сильных сторон и угроз

Сильные стороны						
Угрозы		С1	С2	С3	С4	С5
	В1	-	+	+	-	-
	В2	-	0	+	-	-
	В3	+	+	0	-	-

Таблица 4.6 – интерактивная матрица для слабых сторон и угроз

Слабые стороны					
Угрозы		СЛ1	СЛ2	СЛ3	СЛ4
	В1	+	-	+	+
	В2	-	-	+	+
	В3	-	+	+	-

4.1.4 Оценка готовности проекта к коммерциализации

На какой бы стадии жизненного цикла не находилась научная разработка полезно оценить степень ее готовности к коммерциализации и выяснить уровень собственных знаний для ее проведения (или завершения). Для этого необходимо заполнить специальную форму, содержащую показатели о степени проработанности проекта с позиции коммерциализации и компетенциям разработчика научного проекта. Перечень вопросов приведен в табл. 4.7.

Таблица 4.7 – Бланк оценки степени готовности научного проекта к коммерциализации

№ п/п	Наименование	Степень проработанности научного проекта	Уровень имеющихся знаний у разработчика
1.	Определен имеющийся научно-технический задел	5	4
2.	Определены перспективные направления коммерциализации научно-технического задела	5	4
3.	Определены отрасли и технологии (товары, услуги) для предложения на рынке	5	4
4.	Определена товарная форма научно-технического задела для представления на рынок	5	3
5.	Определены авторы и осуществлена охра-на их прав	4	2
6.	Проведена оценка стоимости интеллектуальной собственности	5	4
7.	Проведены маркетинговые исследования рынков сбыта	3	2
8.	Разработан бизнес-план коммерциализации научной разработки	1	1
9.	Определены пути продвижения научной разработки на рынок	5	3
10.	Разработана стратегия (форма) реализации научной разработки	1	1
11.	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	1	1

Продолжение таблицы 4.7.

12.	Проработаны вопросы использования услуг инфраструктуры поддержки, получения льгот	1	1
13.	Проработаны вопросы финансирования коммерциализации научной разработки	1	1
14.	Имеется команда для коммерциализации научной разработки	1	1
15.	Проработан механизм реализации научного проекта	5	3
	ИТОГО БАЛЛОВ	48	35

Итого получилось суммарное количество баллов по каждому направлению: 48 баллов – по степени проработанности научного проекта; 35 балла – по уровню, имеющихся знаний у разработчика. Согласно этим баллам, можно сказать, что перспективность данной разработки выше среднего.

4.1.5 Методы коммерциализации результатов научно– технического исследования

Перспективность данного научного исследования выше среднего, поэтому не все аспекты рассмотрены и изучены. Таким образом, для организации предприятия этого недостаточно (пункт 4 – 8 не подходят). Но так как основной научно-технический задел определен, этого достаточно для коммерциализации для следующих методов (пункты 1 - 3): Торговля патентной лицензией; передача ноу-хау и инжиниринг. Степени проработанности научного проекта и уровень знаний разработчика достаточно для реализации пунктов, которые были выбраны.

4.2 Инициация проекта

В рамках процессов инициации определяются изначальные цели и содержание и фиксируются изначальные финансовые ресурсы. Определяются

внутренние и внешние заинтересованные стороны проекта, которые будут взаимодействовать и влиять на общий результат научного проекта.

Таблица 4.8 – Заинтересованные стороны проекта

Заинтересованные стороны проекта	**Ожидания заинтересованных сторон**
Организация – заказчик	Разработанный функционал приложения по низкой цене
Пользователь	Удобное взаимодействие с системой анализа эмоционального тона сообщений пользователей
Разработчик	Прибыль
Научный руководитель, студент	Готовая магистерская диссертация

4.2.1 Цели и результаты проекта

Цели и результаты проекта представлены в таблице 4.9:

Таблица 4.9 – Цели и результат проекта

Цели проекта:	• Спроектировать архитектуру нейронной сети • Спроектировать пользовательский интерфейс • Разработать спроектированный функционал и пользовательский интерфейс • Произвести тестирование и анализ
Ожидаемые результаты проекта:	Данное исследование поможет автоматизировать процесс классификации текста сообщения пользователей социальной сети.
Критерии приемки результата проекта:	Высокая точность результатов
Требования к результату проекта:	Разработанное программное обеспечение, которое позволит классифицировать по эмоциональному тону текст сообщений пользователей социальной сети

4.2.2 Организационная структура проекта

Таблица 4.10 – Рабочая группа процесса

№ п/п	ФИО, основное место работы, должность	Роль в проекте	Функции	Трудо-затраты, час.
1	Банокин П.И, Ассистент	Руководитель	Консультирование	6
2	Байкадир Ж.Б, магистрант	Исполнитель	Выполнение НИР	5
3	Рыжакина Т.Г Кандидат экономических наук, доцент ТПУ	Эксперт проекта	Консультирование	2
ИТОГО:				13

4.2.3 Ограничения и допущения проекта

Ограничения проекта представлены в таблице 4.11:

Таблица 4.11 – Ограничения проекта

Фактор	Ограничения/ допущения
2.1. Бюджет проекта	19046,3 руб.
2.1.1. Источник финансирования	НИ ТПУ
2.2. Сроки проекта:	**10.02.2018-31.05.2018**
2.2.1. Дата утверждения плана управления проектом	**10.02.2018**
2.2.2. Дата завершения проекта	**31.05.2018**

4.3 Планирование управления научно – техническим проектом

4.3.1 Иерархическая структура работ проекта

Группа процессов планирования состоит из процессов, осуществляемых для определения общего содержания работ, уточнения целей и разработки последовательности действий, требуемых для достижения данных целей.

План управления научным проектом должен включать в себя следующие элементы:

- иерархическая структура работ проекта;
- контрольные события проекта;
- план проекта;
- бюджет научного исследования.

Иерархическая структура работ (ИСР) – детализация укрупненной структуры работ. В процессе создания ИСР структурируется и определяется содержание всего проекта. На рисунке 4.2 представлен шаблон иерархической структуры.

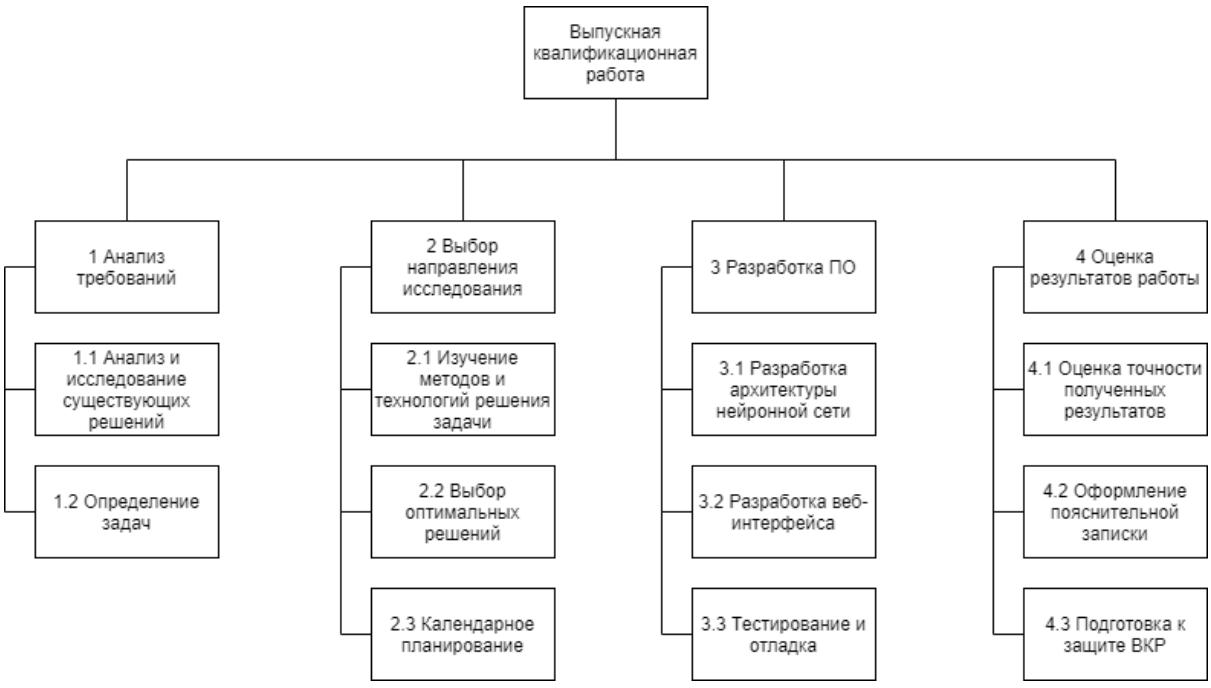


Рисунок 4.2 – Иерархическая структура по ВКР

4.3.2 План проекта

Таблица 4.12 – Календарный план проекта

Код рабо--ты	Название	Длите-льность, дни	Дата начало работ	Дата окончания работ	Состав участников
1	Выбор направления исследования	2	17.02.2018	19.02.2018	Банокин П.И. Байкадир Ж.Б.
1.1	Анализ и исследование готовых решений на рынке	1	19.02.2018	20.02.2018	Банокин П.И., Байкадир Ж.Б.
1.2	Определение задач	5	20.02.2018	25.02.2018	Банокин П.И.
2	Анализ требований	5	26.02.2018	03.03.2018	Банокин П.И., Байкадир Ж.Б.
2.1	Изучение методов и технологий для решения задачи	18	04.03.2018	22.03.2018	Байкадир Ж.Б.
2.2	Выбор оптимальных решений	2	24.03.2018	26.03.2018	Байкадир Ж.Б.
3	Разработка ПО	37	24.03.2018	28.04.2018	Байкадир Ж.Б.
4	Оценка результатов работы	5	07.05.2018	12.05.2018	Банокин П.И., Байкадир Ж.Б.
4.2	Оформление пояснительной записки	6	12.05.2018	19.05.2018	Байкадир Ж.Б.
4.3	Подготовка к защите ВКР	18	20.05.2018	08.06.2018	Байкадир Ж.Б.
Итого:		99			

Диаграмма Ганта – это тип столбчатых диаграмм(гистограмм), который используется для иллюстрации календарного плана проекта, на котором работы по теме представляются протяженными во времени отрезками, характеризующимися датами начала и окончания выполнения данных работ.

График строится в виде таблицы 10 с разбивкой по месяцам и декадам (10 дней) за период времени выполнения научного проекта. При этом работы на графике следует выделить различной штриховкой в зависимости от исполнителей, ответственных за ту или иную работу. Диаграмма Ганта представлена в приложении В.

Для удобства построения графика, длительность каждого из этапов работ из рабочих дней следует перевести в календарные дни.

Коэффициент календарности определяется по следующей формуле:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{\text{Т}_{\text{кал}} - \text{Т}_{\text{вых}} - \text{Т}_{\text{пр}}} = \frac{365}{365 - 104 - 14} = 1{,}48 \qquad (4.1)$$

$$T_{ki} = T_{pi} \cdot k_{\text{кал}}$$

Таблица 4.13 –Временные показатели проведения научного исследования.

№	Наименование работ	Трудоемкость работ			Исполнители
		t_{min}, чел-дни	t_{max}, чел-дни	$t_{ожі}$, чел-дни	
1	Выбор направления исследования	1 1	2 2	1,4 1,4	Р С
2	Анализ и исследование готовых решений на рынке	1 1	1 1	1 1	Р С
3	Определение задач	2	5	3,2	Р
4	Анализ требований	3	5	3,8	С
5	Изучение методов и технологий для решения задачи	8	18	12	С
6	Выбор оптимальных решений	1	2	1,4	С
7	Разработка ПО	20	37	26,8	С
8	Оценка результатов работы	2 2	5 5	3,2 3,2	С Р

Продолжение таблицы 4.13.

9	Оформление пояснительной записки	3	6	4,2	С
10	Подготовка к защите ВКР	12	18	1,4	С

Р – руководитель

С – студент

Таблица 4.14 – Временные показатели проведения научного исследования.

№	Наименование работ	Длительность работ в рабочих днях, T_{pi}	Длительность работ в календарных днях, T_{ki}	Исполнители
1	Выбор направления исследования	0,7 0,7	1 1	Р С
2	Анализ и исследование готовых решений на рынке	0,5 0,5	1 1	Р С
3	Определение задач	3,2	5	Р
4	Анализ требований	3,8	6	С
5	Изучение методов и технологий для решения задачи	12	18	С
6	Выбор оптимальных решений	1,4	2	С
7	Разработка ПО	26,8	40	С
8	Оценка результатов работы	1,6 1,6	2 2	Р С
9	Оформление пояснительной записки	4,2	6	С
10	Подготовка к защите ВКР	14,4	21	С

4.3.3 Бюджет научного исследования

В состав бюджета входит стоимость всех расходов, необходимых для выполнения работ по магистерской диссертации. При формировании бюджета используется группировка затрат по следующим статьям:

- материальные затраты;
- основная заработная плата исполнителей темы;
- дополнительная заработная плата исполнителей темы;
- отчисления во внебюджетные фонды (страховые отчисления);
- накладные расходы.

4.3.3.1 Расчет материальных затрат

Этот пункт включает в себя стоимость всех материалов, необходимых для выполнения НИР.

К категории материалов относят:

1) Хостинг, доменное имя;
2) Электроэнергия.

Для данной разработки требуется специальное оборудование в виде персонального компьютера, но так как в наличии имелся личный ноутбук он не будет заносится в статью материальных расходов.

Разработка проводилась в течении 4 месяцев (в среднем 20 дней в месяц) по 4 часа (320 часов), официально заявленная мощность оборудования 0,09 кВт/час.

Затраты на электроэнергию рассчитываются по формуле:

$$С_{эл} = Ц_{эл} \times Р \times F_{об}\ , \qquad (4.2)$$

где $Ц_{эл}$– тариф на электроэнергию (*3,5 руб за 1 кВт-ч*);

P – мощность оборудования, кВт;

$F_{об}$ – время использования оборудования, ч.

$$С_{эл} = 3{,}5 \times 0{,}09 \times 320 = 100{,}8 \text{ руб.}$$

Так же в статью материальных расходов можно занести покупку хостинга и доменного имени:

$$С_{м} = С_{эл} + С_{х} + С_{ди} \quad (4.3)$$

$С_{х}$ – затраты на хостинг (*619,11 руб. в месяц*);

$С_{ди}$ – затраты на доменное имя (*595 руб. в год*);

Необходимо рассчитать затраты на хостинг за 12 месяцев пользования:

$$С_{х} = 12 \times 619{,}11 = 7429{,}32 \text{ руб.}$$

$$С_{м} = 100{,}8 + 7429{,}32 + 595 = 8125{,}12 \text{ руб.}$$

Таблица 4.15 – Материальные расходы

Материальные затраты	Сумма, руб
Затраты на электроэнергию	100,8
Хостинг	7429,32
Доменное имя	595
Компьютер	—
Итог	8125,12

Таблица 4.16 – Расчет затрат по статье «Спецоборудование для научных работ»

№ п/п	Наименование оборудования	Кол-во единиц оборудования	Цена единицы оборудования, тыс.руб.	Общая стоимость оборудования, тыс.руб.
1.	Персональный компьютеры	1	-	-
2.	Linux сервер	5	2 000	10 000
3	Электроэнергия	-	-	100,8
4.	Среда разработки Visual Studio Code	1	-	-
	Итого		-	-

4.3.3.2 Основная заработная плата

В данную статью включается основная заработная платаруководителя от предприятия и студента, также премия, выплачиваемая ежемесячно из фонда заработной платы в размере 20 –30 % от тарифа или оклада. Расчет выполняется на основе трудоемкости выполнения каждого этапа и величины месячного оклада исполнителя.

Расчет основной заработной платы приведен в таблице 4.17.

Таблица 4.17 – Расчет основной заработной платы

Исполнители	Исполнители по категориям	Трудо-емкость, чел.-дн.,$Т_р$	Заработная плата, приходящаяся на один чел.-дн., тыс. руб.здн	Всего заработная плата по тарифу (окладам), тыс. руб.зосн
Дипломник		65,4		2600
Руководитель	ассистент	6	687,93	4127,58
Итого				**6727,58**

$$С_{зп} = З_{осн} + З_{доп}, \quad (4.4)$$

где $З_{осн}$ – основная заработная плата;

$З_{доп}$ – дополнительная заработная плата.

Основная заработная плата $З_{осн}$ руководителя рассчитывается по следующей формуле:

$$З_{осн} = З_{дн} * Т_{раб} \quad (4.5)$$

где $Т_{раб}$– продолжительность работ, выполняемых научно-техническим работником, раб.дн. (таблица 14);

$З_{дн}$– среднедневная заработная плата работника, руб.

Значит, для руководителя:

$$З_{осн} = 687{,}93 \times 6 = 4127{,}58 \text{ рублей}$$

Среднедневная заработная плата рассчитывается по формуле:

$$З_{дн}=(З_{м}*М)/F_{д} \quad (4.6)$$

где $З_{м}$ – месячный должностной оклад работника, руб (в качестве месячного оклада магистра выступает стипендия, которая составляет 2410руб);

М – количество месяцев работы без отпуска в течение года:

при отпуске в 45 раб. дней М=10,4 месяца, 6 - дневная неделя;

$F_{д}$ – действительный годовой фонд рабочего времени научно-технического персонала (в рабочих днях) (табл.4.18). Тогда,

Для руководителя:

$$З_{дн} = \frac{17000 \times 10{,}4}{257} = 687{,}93 \text{ рублей}$$

Баланс рабочего времени представлен в таблице 4.18.

Таблица 4.18 – Баланс рабочего времени

Показатели рабочего времени	Руководитель
Календарное число дней	365
Количество нерабочих дней – выходные дни – праздничные дни	 46 14
Потери рабочего времени – отпуск – невыходы по болезни	 48 —
Действительный годовой фонд рабочего времени	257

Месячный должностной оклад работника рассчитывается по формуле:

$$З_{м} = З_{б} * к_{р} \quad (4.7)$$

где $З_{б}$ – базовый оклад, руб;

$k_{р}$ – районный коэффициент, равный 1,3.

Заработная плата старшего преподавателя составляет 17000 руб., согласно «Положению об оплате труда» ТПУ.

Для руководителя: $З_{м} = 17000 * 1{,}3 = 22100$ руб.

Результаты расчета основной заработанной платы представлены в таблице 4.19.

Таблица 4.19 - Результаты расчета основной заработанной платы

Исполнители	$З_б$,руб.	$к_р$	$З_м$,руб	$З_{дн}$,руб.	$Т_{раб.}$раб. дн.	$З_{осн}$,руб.
Руководитель	17000	1,3	22100	687,93	6	4127,58
Итого $З_{осн}$						4127,58

4.3.3.3 Дополнительная заработная плата научно-производственного персонала

Затраты по дополнительной заработной плате исполнителей темы учитывают величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда, а также выплат, связанных с обеспечением гарантий и компенсаций.

Расчет дополнительной заработной платы ведется по следующей формуле:

$$З_{доп} = k_{доп} * З_{осн} \quad (4.8)$$

где $З_{доп}$ – дополнительная заработная плата, руб.;

$k_{доп}$ – коэффициент дополнительной зарплаты ($k_{доп}$ =0,12);

$З_{осн}$ – основная заработная плата, руб.

Для руководителя:

$$З_{доп} = 4127{,}58 * 0{,}12 = 495{,}30 \text{ рублей}$$

В таблице 16 приведен расчёт основной и дополнительной заработной платы.

Таблица 4.20 – Заработная плата исполнителей ВКР

Заработная плата	Руководитель	Дипломник
Основная зарплата	4127,58	2410
Дополнительная зарплата	495,30	–

Продолжение таблицы 4.20.

Зарплата исполнителя	4622,88	2410
Итого по статье $С_{зп}$	7032,88	

4.3.3.4 Отчисления на социальные нужды

Статья включает в себя отчисления во внебюджетные фонды.

$$С_{внеб}= k_{внеб}\cdot(З_{осн} + З_{доп})=0,3\cdot(4127,58 + 495,30)= 1386,86 \text{ руб.} \quad (4.9)$$

где $k_{внеб}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

4.3.3.5 Накладные расходы

В эту статью относятся расходы по содержанию, эксплуатации и ремонту оборудования, производственного инструмента и инвентаря, зданий, сооружений и др. В расчетах эти расходы принимаются в размере 70 - 90 % от суммы основной заработной платы научно-производственного персонала данной научно-технической организации.

Накладные расходы составляют 80-100 % от суммы основной и дополнительной заработной платы, работников, непосредственно участвующих в выполнение темы.

Расчет накладных расходов ведется по следующей формуле:

$$С_{накл}= к_{накл} * (З_{осн} + З_{доп}) \quad (4.10)$$

где $k_{накл}$ – коэффициент накладных расходов.

$$С_{накл} = 0,8 \times (4127,58 + 495,30) = 3698,30 \text{ руб.}$$

4.3.3.6 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в качестве нижнего предела затрат на разработку научно-технической продукции.

Таблица 4.21 – Группировка затрат по статьям

	Статьи						
Вид работ	Материальные расходы	Специальное оборудование для научных (экспериментальных) работ	Основная заработная плата	Дополнительная заработная плата	Расходы на социальные нужды	Накладные расходы	Итого плановая себестоимость
1.	8125,12	–	6727,58	495,30	1386,86	3698,30	19046,3
2.	11000	–	10000	1000	3300	5000	27000

В результате было получено, что бюджет затрат НТИ составит 19046,3 руб. При этом затраты у конкурентов составляют 27000 рублей, из чего можно сделать вывод что полученный продукт будет экономичней, чем у конкурентов.

4.3.4 Организационная структура проекта

В практике используется несколько базовых вариантов организационных структур: функциональная, проектная, матричная.

Для выбора наиболее подходящей организационной структуры можно использовать табл. 4.22.

Таблица 4.22 – Выбор организационной структуры научного проекта

Критерии выбора	Функциональная	Матричная	Проектная
Степень неопределенности условий реализации проекта	Низкая	Высокая	Высокая
Технология проекта	Стандартная	Сложная	Новая
Сложность проекта	Низкая	Средняя	Высокая
Взаимозависимость между отдельными частями проекта	Низкая	Средняя	Высокая
Критичность фактора времени (обязательства по срокам завершения работ)	Низкая	Средняя	Высокая
Взаимосвязь и взаимозависимость проекта от организаций более высокого уровня	Высокая	Средняя	Низкая

4.3.5 План управления коммуникациями проекта

План управления коммуникациями отражает требования к коммуникациям со стороны участников проекта. Пример плана управления коммуникациями приведен в таблице 4.23.

Таблица 4.23 – План управления коммуникациями

№ п/п	**Какая** информация передается	**Кто** передает информацию	**Кому** передается информация	**Когда** передает информацию
1.	Статус проекта	Руководитель проекта	Представителю заказчика	Ежемесячно
2.	Обмен информацией о текущем состоянии проекта	Исполнитель проекта	Руководителю проекта	Еженедельно (понедельник)
3.	Документы и информация по проекту	Ответственное лицо по направлению	Руководителю проекта	Не позже сроков графиков и к. точек
4.	О выполнении контрольной точки	Исполнитель проекта	Руководителю проекта	Не позже дня контрольного события по плану управления

4.3.6 Реестр рисков проекта

Идентифицированные риски проекта включают в себя возможные неопределенные события, которые могут возникнуть в проекте и вызвать последствия, которые повлекут за собой нежелательные эффекты. Информацию по данному разделу сведены в таблицу 4.24.

Таблица 4.24 – Реестр рисков

№	Риск	Потенциальное воздействие	Вероятность наступления (1-5)	Влияние риска (1-5)	Уровень риска*	Способы смягчения риска	Условия наступления
1	Изменение требований к продукту	Задержка срока выхода продукта на рынок, увеличение себестоимости	4	3	3	Разделить требования на "абсолютно необходимые" и "хорошо бы было иметь", до запуска системы выполнять только абсолютно необходимые требования	Смена ответственного лица со стороны заказчика, переориентация конечного продукта
2	Затруднения с финансированием	Отказ разработчика поддерживать продукт	2	5	5	Поднять вопрос на уровень директора компании	Нехватка бюджета заказчика

4.4 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности разработки

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

4.4.1 Динамические методы экономической оценки инвестиций

Динамические методы оценки инвестиций базируются на применении показателей:

- чистая текущая стоимость (**NPV**);
- срок окупаемости (**D PP**);
- внутренняя ставка доходности (**IRR**);
- индекс доходности (**PI**).

Все перечисленные показатели основываются на сопоставлении чистых денежных поступлений от операционной и инвестиционной деятельности, и их приведении к определенному моменту времени. Теоретически чистые денежные поступления можно приводить к любому моменту времени (к будущему либо текущему периоду). Но для практических целей оценку инвестиции удобнее осуществлять на момент принятия решений об инвестировании средств.

4.4.1.1 Чистая текущая стоимость (NPV)

Данный метод основан на сопоставлении дисконтированных чистых денежных поступлений от операционной и инвестиционной деятельности.

Если инвестиции носят разовый характер, то **NPV** определяется по формуле:

$$\mathbf{NPV} = \sum_{\mathbf{t}=1}^{\mathbf{n}} \frac{\mathbf{ЧДП_{оп_t}}}{(1+\mathbf{i})^{\mathbf{t}}} - \mathbf{I_0}, \qquad (4.11)$$

где $\mathbf{ЧДП_{оп_t}}$ – чистые денежные поступления от операционной деятельности;

$\mathbf{I_0}$ – разовые инвестиции, осуществляемые в нулевом году;

$\mathbf{t}$ – номер шага расчета ($\mathbf{t}$=0, 1, 2… $\mathbf{n}$);

n – горизонт расчета;

i – ставка дисконтирования (желаемый уровень доходности инвестируемых средств).

Чистая текущая стоимость является абсолютным показателем. Условием экономичности инвестиционного проекта по данному показателю является выполнение следующего неравенства: **NPV** >0.

Чем больше **NPV**, тем больше влияние инвестиционного проекта на экономический потенциал предприятия, реализующего данный проект, и на экономическую ценность этого предприятия.

Таким образом, инвестиционный проект считается выгодным, если **NPV** является положительной.

Таблица 4.25 - Расчет чистой текущей стоимости по проекту в целом

№	Наименование показателей	Шаг расчета				
		0	1	2	3	4
1.	Выручка от реализации, тыс.руб	0	28,569	28,569	28,569	28,569
2.	Итого приток, тыс.руб	0	28,569	28,569	28,569	28,569
3.	Инвестиционные издержки, тыс.руб.	-19,046	0	0	0	0
4.	Операционные затраты, тыс. руб С+Ам+ФОТ	0	9,294	9,294	9,294	9,294
5.	Налогооб. прибыль		19,275	19,275	19,275	19,275
6.	Налоги, тыс. руб Выр-опер=донал.приб*20%	0	3,855	3,855	3,855	3,855
7.	Итого отток, тыс.руб. Опер.затр+налоги	-19,046	13,149	13,149	13,149	13,149
8.	Чистый денежный поток, тыс.руб. ЧДП=Пчист+Ам Пчист=Пдонал.-налог	-19,046	15,42	15,42	15,42	15,42
9.	Коэффициент дисконтирования (приведения при **i** =20%)	1,0	0,833	0,694	0,578	0,482

Продолжение таблицы 4.25.

10.	Дисконтированный чистый денежный поток, тыс.руб. (с8*с9)	-19,046	12,844	10,701	8,912	7,432
11.	То же нарастающим итогом, тыс.руб. (**NPV** =20,845тыс.руб.)	-19,046	-6,202	4,499	13,441	20,845

Таким образом, чистая текущая стоимость по проекту в целом составляет 20,845 тыс. руб., что позволяет судить о его эффективности.

4.4.1.2 Дисконтированный срок окупаемости

Как отмечалось ранее, одним из недостатков показателя простого срока окупаемости является игнорирование в процессе его расчета разной ценности денег во времени.

Этот недостаток устраняется путем определения дисконтированного срока окупаемости.

Рассчитывается данный показатель примерно по той же методике, что и простой срок окупаемости, с той лишь разницей, что последний не учитывает фактор времени.

Наиболее приемлемым методом установления дисконтированного срока окупаемости является расчет кумулятивного (нарастающим итогом) денежного потока (см. табл. 4.26).

Таблица 4.26– Дисконтированный срок окупаемости.

№	Наименование показателя	Шаг расчета				
		0	1	2	3	4
1.	Дисконтированный чистый денежный поток (**i** =0,20)	-19,046	12,844	10,701	8,912	7,432
2.	То же нарастающим итогом	-19,046	-6,202	4,499	13,441	20,843
3.	Дисконтированный срок окупаемости	**$PP_{дск}$** = 1+1,43/14,674=0,67 года				

4.4.1.3 Внутренняя ставка доходности (IRR)

Для установления показателя чистой текущей стоимости (NPV) необходимо располагать информацией о ставке дисконтирования, определение которой является проблемой, поскольку зависит от оценки экспертов. Поэтому, чтобы уменьшить субъективизм в оценке эффективности инвестиций на практике широкое распространение получил метод, основанный на расчете внутренней ставки доходности (IRR).

Между чистой текущей стоимостью (NPV) и ставкой дисконтирования (**i**) существует обратная зависимость. Эта зависимость следует из таблицы 20 и графика, представленного на рисунке 4.3.

Таблица 4.27 – Зависимость **NPV** от ставки дисконтирования.

№ п/п	Наименование показателя	0	1	2	3	4	NPV
1	Чистые денежные потоки	-19,046	15,42	15,42	15,42	15,42	
2	коэффициент дисконтирования						
	i=0,1	1	0,909	0,826	0,751	0,683	
	i=0,2	1	0,833	0,694	0,578	0,482	
	i=0,3	1	0,769	0,592	0,455	0,35	
	i=0,4	1	0,714	0,51	0,364	0,26	
	i=0,5	1	0,667	0,444	0,295	0,198	
	i=0,6	1	0,625	0,39	0,244	0,095	
	i=0,7	1	0,588	0,335	0,203	0,07	
	i=0,8	1	0,556	0,309	0,171	0,095	
	i=0,9	1	0,526	0,277	0,146	0,077	
3	Дисконтированный денежный поток, тыс. руб						
	i=0,1	-19,046	14,017	12,737	11,58	10,532	29,82
	i=0,2	-19,046	12,845	10,701	8,913	7,432	20,845
	i=0,3	-19,046	11,858	9,129	7,016	5,397	14,354
	i=0,4	-19,046	11,01	7,864	5,613	4,009	9,45
	i=0,5	-19,046	10,285	6,846	4,549	3,053	5,687
	i=0,6	-19,046	9,638	6,014	3,762	1,465	1,833
	i=0,7	-19,046	9,067	5,166	3,13	1,079	-0,604
	i=0,8	-19,046	8,574	4,765	2,637	1,465	-1,605
	i=0,9	-19,046	8,111	4,271	2,251	1,187	-3,226

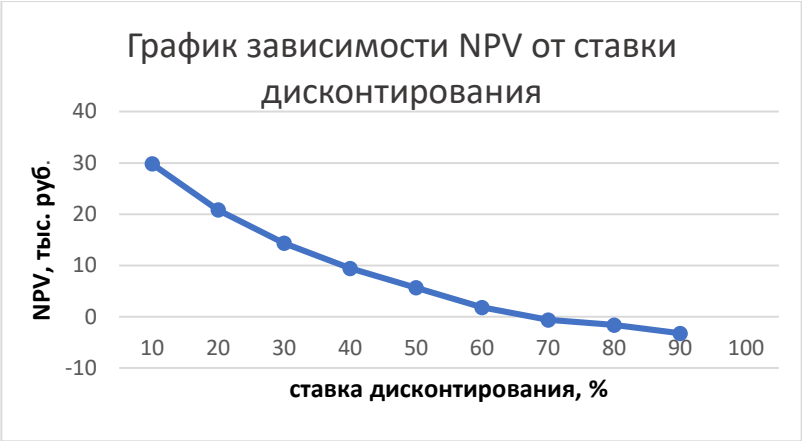


Рисунок 4.3 – Зависимость NPV от ставки дисконтирования.

Из таблицы и графика следует, что по мере роста ставки дисконтирования чистая текущая стоимость уменьшается, становясь отрицательной. Значение ставки, при которой **NPV** обращается в нуль, носит название «внутренней ставки доходности» или «внутренней нормы прибыли». Из таблицы и графика следует, что по мере роста ставки дисконтирования чистая текущая стоимость уменьшается, становясь отрицательной. Значение ставки, при которой обращается в нуль, носит название «внутренней ставки доходности» или «внутренней нормы прибыли». Из графика получаем, что IRR составляет 0,67.

4.4.1.4 Индекс доходности (рентабельности) инвестиций (PI)

Индекс доходности показывает, сколько приходится дисконтированных денежных поступлений на рубль инвестиций.

Расчет этого показателя осуществляется по формуле

$$PI = \sum_{t=1}^{n} \frac{ЧПД_t}{(1+i)^t} / I_0, \quad (4.12)$$

где I_0 – первоначальные инвестиции.

$$\text{PI} = \frac{12{,}844 + 10{,}701 + 8{,}912 + 7{,}432}{19{,}046} = 2{,}094$$

PI=2094>1, следовательно, проект эффективен при i=0,2; NPV=20,845 тыс. руб.

4.4.2 Оценка сравнительной эффективности исследования

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсэффективности.

Интегральный показатель финансовой эффективности научного исследования получают в ходе оценки бюджета затрат трех (или более) вариантов исполнения научного исследования. Для этого наибольший интегральный показатель реализации технической задачи принимается за базу расчета (как знаменатель), с которым соотносится финансовые значения по всем вариантам исполнения.

Интегральный финансовый показатель разработки определяется:

$$I_{финр}^{исп.i} = \frac{Ф_{pi}}{Ф_{\max}}, \tag{4.13}$$

где $I_{финр}^{испi}$ – интегральный финансовый показатель разработки;

$Ф_{pi}$– стоимость i-го варианта исполнения;

$Ф_{\max}$– максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в разах (значение больше единицы), либо соответствующее численное удешевление стоимости разработки в разах (значение меньше единицы, но больше нуля).

Так как разработка имеет одно исполнение, то:

$$I_{\text{финр}}^{\text{р}} = \frac{19{,}046}{19{,}046} = 1;$$

Для аналогов (с использованием дополнительного оборудования, которое стоит 10000 руб и 15000 руб) соответственно:

$$I_{\text{фин1}}^{a1} = \frac{\Phi_{\text{a1}}}{\Phi_{max}} \tag{4.14}$$

$$I_{\text{фин1}}^{a1} = \frac{\Phi_{\text{a1}}}{\Phi_{max}} = \frac{24406}{19406} = 1{,}25$$

$$I_{\text{фин2}}^{a2} = \frac{\Phi_{\text{a2}}}{\Phi_{max}} = \frac{25406}{19406} = 1{,}30$$

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{pi} = \sum a_i \cdot b_i \,, \tag{4.15}$$

где I_{pi} – интегральный показатель ресурсоэффективности для i-го варианта

исполнения разработки;

a_i – весовой коэффициент *i*-го варианта исполнения разработки;

b_i^a, b_i^p – бальная оценка *i*-го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Расчёт интегрального показателя ресурсоэффективности представлен в таблице 4.28.

Таблица 4.28 – Сравнительная оценка характеристик вариантов исполнения проекта

Объект исследования / Критерии	Весовой коэффициент параметра	Текущий проект	Аналог 1	Аналог 2
1.Способствует росту производительности труда пользователя	0,4	5	3	2
2. Удобство в эксплуатации (соответствует требованиям потребителей)	0,2	5	3	3
3. Помехоустойчивость	0,1	4	4	4
4. Надёжность	0,25	4	4	4
5. Материалоёмкость	0,15	5	3	3
ИТОГО	1	5,15	2,92	2,5

$I_{тп} = 5 \cdot 0,4 + 5 \cdot 0,2 + 4 \cdot 0,1 + 4 \cdot 0,25 + 5 \cdot 0,15 = 5,15;$

Аналог 1 = $3 \cdot 0,4 + 3 \cdot 0,2 + 4 \cdot 0,1 + 4 \cdot 0,25 + 3 \cdot 0,15 = 3,65;$

Аналог 2 = $2 \cdot 0,4 + 3 \cdot 0,2 + 4 \cdot 0,1 + 4 \cdot 0,25 + 3 \cdot 0,15 = 3,25.$

Интегральный показатель эффективности вариантов исполнения разработки ($I^{p}_{финр}$) и аналога ($I^{ai}_{финai}$) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I^{p}_{финр} = \frac{I^{p}_{m}}{I^{p}_{финр}}; \quad I^{ai}_{финai} = \frac{I^{ai}_{m}}{I^{ai}_{финai}}; \qquad (4.15,4.16)$$

В результате:

$$I^{p}_{финр} = \frac{I^{p}_{m}}{I^{p}_{финр}} = \frac{5,15}{1} = 5,15;$$

$$I^{a1}_{фина1} = \frac{I^{a1}_{m}}{I^{a1}_{фина1}} = \frac{3,65}{1,25} = 2,92;$$

$$I^{a2}_{фина2} = \frac{I^{a2}_{m}}{I^{a2}_{фина2}} = \frac{3,25}{1,30} = 2,5.$$

Сравнение интегрального показателя эффективности текущего проекта и аналогов позволит определить сравнительную эффективность проекта.

Сравнительная эффективность проекта:

$$Э_{ср} = \frac{I^{p}_{финр}}{I^{ai}_{финai}} \tag{4.17}$$

Результат вычисления сравнительной эффективности проекта и сравнительная эффективность анализа представлены в таблице 4.29.

Таблица 4.29 – Сравнительная эффективность разработки

№	Показатели	Аналог 1	Аналог 2	Разработка
1	Интегральный финансовый показатель разработки	1,25	1,30	1
2	Интегральный показатель ресурсоэффективности разработки	3,65	3,25	5,15
3	Интегральный показатель эффективности	2,92	2,5	5,15
4	Сравнительная эффективность вариантов исполнения	2,3	1,92	5,15

Таким образом, основываясь на определении ресурсосберегающей, финансовой, бюджетной, социальной и экономической эффективности исследования, проведя необходимый сравнительный анализ, можно сделать вывод о превосходстве выполненной разработки над аналогами.

4.4.3 Оценка абсолютной эффективности проекта

Социальная эффективность научного проекта учитывает социально-экономические последствия осуществления научного проекта для общества в целом или отдельных категорий населений или групп лиц, в том числе как непосредственные результаты проекта, так и «внешние» результаты в смежных секторах экономики: социальные, экологические и иные внеэкономические эффекты.

Для оценки социальной эффективности научного проекта магистранту необходимо выявить критерии социальной эффективности, на которые влияет реализация научного проекта и оценить степень их влияния.

Таблица 4.30 – Критерии социальной эффективности

ДО	ПОСЛЕ
Затрудненность в извлечении и анализе общественного мнения в сети по тому или иному вопросу.	Процесс извлечения мнений и отзывов полностью автоматизирован.
Низкая популярность анализа эмоционального тона текстов на русском языке	Предложен метод для анализа эмоционального тона текста на русском языке
Отсутствие возможностей для улучшения существующих моделей нейронной сети для получения наиболее точных результатов определения эмоциональной окраски текста	Добавлена возможность обучения нейронной сети на новых данных, для поддержки актуальности данных и увеличения точности определения эмоционального окраса

Вывод

В процессе выполнения части работы по финансовому менеджменту, ресурсоэффективности и ресурсосбережению был проведен тщательный анализ разрабатываемого исследования. Во-первых, оценен коммерческий потенциал и перспективность проведения исследования. Полученные результаты говорят о потенциале и перспективности на уровне выше среднего. Во-вторых, проведено планирование НИР, а именно: определена структура и календарный план работы, трудоемкость и бюджет НТИ по трем исполнениям для сравнения. Результаты соответствуют требованиям к ВКР по срокам и иным параметрам. В-третьих, определена эффективность исследования в разрезах ресурсной, финансовой, бюджетной, социальной и экономической эффективности.

5 Социальная ответственность

Введение

В данном разделе рассмотрены вопросы производственной и экологической, правовой и организационной безопасности, также безопасность в чрезвычайных ситуациях при выполнении и оформлении магистерской диссертации в соответствии с требованиями законодательных и правовых актов, технических регламентов в области безопасности производства, охраны труда и защиты окружающей среды.

Выполнение разработки производилось в офисе на территории работодателя, снабженном настольными персональными компьютерами (ПК). Выполнение проекта заключалось в разработке мобильного приложения для абитуриентов, соответственно, производилось взаимодействие разработчика-программиста с персональным компьютером.

5.1 Производственная безопасность

Для рассмотрения производственной безопасности проекта необходимо выявить вредные и опасные факторы, которые могут возникнуть на рабочем месте, и описать мероприятия по защите исследователя и пользователей конечных продуктов от действия опасных и вредных факторов.

5.1.1 Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте при выполнении проекта

На данном этапе выполнения работы необходимо выявить источники опасности, то есть части производственных систем, производственного оборудования и элементы среды, формирующие эти опасности.

Производственные условия на рабочем месте характеризуются наличием некоторых опасных и вредных факторов, которые В соответствии с ГОСТ 12.0.003-74 опасные и вредные производственные факторы подразделяются по природе действия на следующие группы: физические, химические, биологические и психофизиологические [30].

В общих случаях к определенным признакам опасных и вредных факторов относятся: затруднение осуществления физиологических функций дыхания, возможность непосредственного воздействия на организм, кровообращения, работы центральной нервной системы, органов пищеварения, выделения.

В соответствии с [30] приведены основные опасные и вредные производственные факторы, воздействующие на персонал, пользующийся компьютером при выполнении данной разработки.

Вредные:

а) микроклимат (ГОСТ 12.1.005-88 с изм. №1 от 2000 г., СанПиН 2.2.4.548-96);

б) освещенность (ГОСТ 12.4.026-76) [2];

в) шум (ГОСТ 12.1.003-83) [30];

г) нервно-психические нагрузки

д) электромагнитное поле [30].

Опасные:

а) статическое электричество;

б) электрический ток.

в) короткое замыкание

5.1.2 Производственная санитария

Производственное помещение – это пространство, производственная среда, где осуществляется трудовая деятельность человека. В

производственных помещениях должны быть обеспечены и соблюдены нормативные санитарно-технические условия.

При планировании рабочего помещения необходимо соблюдать нормы полезной площади и объема помещения.

Размеры рабочего кабинета представлены в таблице 5.1.

Таблица 5.1 - Размеры рабочего кабинета

длина помещения	8 м
ширина	3 м
высота	3 м

Рабочее помещение представляет собой комнату площадью 24 м^2 и объемом 72 м^3. Одновременно в рабочем помещении находится 3 человек, следовательно на одного человека приходится около 24 м^3 объема помещения и 8 м^2 площади, что удовлетворяет требованиям санитарных норм [33**Ошибка! Источник ссылки не найден.**], согласно которым для одного работника должны быть предусмотрены площадь величиной не менее 6 м^2 и объем не менее 20 м^3 с учетом максимального числа одновременно работающих в смену [33].

Выбор типа производственного помещения определяется производственным процессом, и при анализе опасных и вредных факторов необходимо ориентироваться на конкретное рабочее место и конкретные условия труда.

5.1.2.1 Производственный шум

Одним из важнейших параметров, наносящим большой ущерб для здоровья и резко снижающим производительность труда, является шум. Действие шума различно: затрудняет разборчивость речи, вызывает снижение работоспособности, повышает утомляемость, вызывает необратимые изменения в органах слуха человека. Шум воздействует не только на органы слуха, но и на весь организм человека через центральную нервную систему.

Ослабляется внимание, ухудшается память, снижается реакция, увеличивается число ошибок при работе [35].

Производственные помещения, в которых для работы используются ПЭВМ, не должны граничить с помещениями, в которых уровень шума и вибрации превышают нормируемые значения. При выполнении основной работы на ПЭВМ уровень шума на рабочем месте не должен превышать 50 дБ. Допустимые уровни звукового давления в помещениях для персонала, осуществляющего эксплуатацию ЭВМ при разных значениях частот [36].

По субъективным ощущениям шумовая обстановка на рабочем месте соответствует норме и не превышает 50 дБ.

5.1.2.2 Электромагнитные поля

Как любые электрические приборы, видеотерминалы (ВДТ) и системные блоки производят электромагнитное излучение. Большая часть его происходит не от экрана монитора (ВДТ), а от видеокабеля и системного блока. В портативных компьютерах практически все электромагнитное излучение идет от системного блока, располагающегося под клавиатурой. Современные машины выпускаются заводом-изготовителем со специальной металлической защитой внутри системного блока для уменьшения фона электромагнитного излучения.

При воздействии полей, имеющих напряженность выше предельно допустимого уровня, развиваются нарушения со стороны нервной, сердечно-сосудистой систем, органов пищеварения и некоторых биологических показателей крови.

Согласно СанПиН 2.2.2/2.4.1340-03 напряженность электромагнитного поля на расстоянии 50 см вокруг ВДТ по электрической составляющей должна быть не более:

В диапазоне частот 2 кГц ÷ 400кГц – 2,5 В/м.

В диапазоне частот 5 Гц ÷ 2 кГц – 25 В/м.

Плотность магнитного потока должна быть не более:

– В диапазоне частот 2 кГц ÷ 400кГц – 25 нТл.

– В диапазоне частот 5 Гц ÷ 2 кГц – 250 нТл.

Возможные способы защиты от ЭМП:

– Основной способ – увеличение расстояния от источника, во избежание последствий экран видеомонитора должен находиться на расстоянии не менее 50 см от пользователя.

– Применение экранных фильтров, специальных экранов и других средств индивидуальной защиты, прошедших испытание в аккредитованных лабораториях и имеющих соответствующий гигиенический сертификат.

5.1.2.3 Психофизиологические факторы

Как известно, любой вид деятельности человека порождает возникновение различных видов опасностей. Наибольшее количество опасностей возникает, в первую очередь, в процессе трудовой деятельности. Это обусловлено двумя причинами: в течение суток человек занимается трудовой деятельностью (работа, учеба, спорт, активный отдых и т д), то есть повышается вероятность проявления опасностей; производственные процессы, в которых осуществляется преобразование веществ, энергии и информации и возникают основные техногенные опасности. В любой трудовой деятельности человека можно выделить два компонента: физиологический и психический. Физиологический компонент связан с физиологическими возможностями каждого индивидуума и определяется работой его мышц, системы кровообращения, дыхания, сердечно-сосудистой системы, опорно- двигательного аппарата. Действие этих систем координируется центральной нервной системой. В этом процессе используется большое количество энергии, кислорода для активизации обменных процессов. Отрасль физиологии, которая изучает изменения функционального состояния человека в зависимости от характера и типа

трудовой деятельности и разрабатывает оптимальные режимы (условия) труда и отдыха, называется физиологией труда.

Рабочее место – это часть рабочей зоны. Оно представляет собой место постоянного или временного пребывания, работающего в процессе трудовой деятельности. Рабочее место должно удовлетворять следующим требованиям:

- обеспечивать возможность удобного выполнения работ;
- учитывать физическую тяжесть работ;
- учитывать размеры рабочей зоны и необходимость передвижения в ней работающего;

учитывать технологические особенности процесса выполнения работ.

Психический компонент определяется психическими процессами и психическими свойствами личности. Психологические состояния отличаются разнообразием и характером. Они обусловливают особенности психической деятельности в конкретный период времени и могут положительно или отрицательно влиять на протекание всех психических процессов.

При оформлении помещения большое значение имеет цветовое решение. Психофизическое воздействие цвета - первый и наиболее важный фактор, учитываемый при выборе цветового решения. Учитывая характер работ, следует выбирать неяркие, малоконтрастные оттенки, которые не рассеивали бы внимание в рабочей зоне. Так как работа требует спокойствия и сосредоточенности, предпочтительно использовать оттенки «холодных» цветов.

Меры по устранению психофизиологических факторов:

- соблюдать чистоту и порядок на рабочем месте;
- не создавать шума;
- не нарушать инструкции по техники безопасности.

Помещение, в котором находится рабочее место, относится к категории помещений без повышенной опасности. Его можно охарактеризовать, как сухое, непыльное, с нормальной температурой воздуха.

Температурный режим, влажность воздуха, химическая среда не способствуют разрушению изоляции электрооборудования.

5.1.2.4 Микроклимат в помещении

Микроклимат - искусственно создаваемые климатические условия в закрытых помещениях (напр., в жилище) для защиты от неблагоприятных внешних воздействий и создания зоны комфорта. Зона комфорта – оптимальное для организма человека сочетание температуры, влажности, скорости движения воздуха и воздействия лучистого тепла (напр., в состоянии покоя или при выполнении легкой физической работы: температура зимой 18-22 °С, летом 23- 25 °С; скорость движения воздуха зимой 0,15, летом 0,2-0,4 м/с; относительная влажность 40-60%). Тесно соприкасаясь с воздушной средой, организм человека подвергается воздействию ее физических и химических факторов: состава воздуха, температуры, влажности, скорости движения воздуха, барометрического давления и др. Особое внимание следует уделить параметрам микроклимата помещений — аудиторий, производственных и жилых зданий. 79 Микроклимат, оказывая непосредственное воздействие на один из важнейших физиологических процессов — терморегуляцию, имеет огромное значение для поддержания комфортного состояния организма. Нормы оптимальных и допустимых показателей микроклимата при работе с ЭВМ устанавливает СанПиН 2.2.2/2.4.1340-03. Все категории работ разграничиваются на основе интенсивности энергозатрат организма в ккал/ч (Вт). Работа, производимая сидя и сопровождающаяся незначительным физическим напряжением, относится к категории Iа – работа с интенсивностью энергозатрат до 120 ккал/ч (до 139 Вт).

5.1.3 Экологическая безопасность

Экологическая безопасность и охрана окружающей среды являются одними из важнейших факторов при выполнении работ любого характера. При работе в офисном помещении за персональным ПК отсутствуют выбросы в окружающую среду и нет влияния на жилищную зону.

Поскольку при разработке данной магистерской диссертации использовался компьютер, необходимо помнить о правильной утилизации компьютерного лома после выхода из строя данного ПК. В соответствии с постановлением правительства №340 [34] юридическим лицам запрещено самостоятельно утилизировать компьютерную технику. Необходимо найти организацию, которая занимается утилизацией в частном порядке. Это относится к следующим видам отходов:

- образование твердых отходов, относящихся к IV классу опасности (системный блок компьютера, принтеры, сканеры, клавиатура, манипулятор "мышь") и жидких отходов; образование твердых отходов, относящихся к IV классу опасности (системный блок компьютера, принтеры, сканеры, клавиатура, манипулятор "мышь") и жидких отходов;
- Жидкие отходы: сточные воды;
- Люминесцентные лампы.

5.1.3.1 Безопасность в чрезвычайных ситуациях

Персональный компьютер - электроприбор. Основным отличием персонального компьютера от других электроприборов является длительное время эксплуатации без отключения от электрической сети. В связи с этим следует особое внимание уделить качеству организации электропитания. Во время работы за компьютером запрещается:

- прикасаться к задней панели системного блока и переключать разъемы кабелей периферийных устройств при включенном питании;

– загромождать верхние панели устройств бумагами и посторонними предметами;

– производить отключение питания во время выполнения активных задач;

– включать охлажденное оборудование;

– работать на офисной технике при снятых кожухах, при отсутствии или неисправности предусмотренных конструкцией оборудования предохранительных приспособлений, блокировок;

– отключать оборудование от электросети и выдергивать электрическую вилку, держась за шнур;

– самостоятельно производить вскрытие, обслуживание и ремонт офисной техники.

Во избежание воздействия возможных опасных факторов необходимо:

– занулять электрооборудование;

– следить за своевременным техническим обслуживанием оборудования; корпус системного блока необходимо очищать от пыли не реже 1 раза в шесть месяцев, во избежание замыкания электрических контактов;

– использовать только исправное электрооборудование; исключить резкие перегибы электрических проводов.

Также не следует работать с персональными электронно-вычислительными машинами (ПЭВМ) в условиях повышенной влажности (относительная влажность воздуха длительно превышает 75%), высокой температуры (более 35°С), наличии токопроводящей пыли, токопроводящих полов и возможности одновременного соприкосновения к имеющим соединение с землёй металлическим элементам и металлическим корпусом электрооборудования. Правилами запрещено работать с ПЭВМ в таких условиях. Таким образом, работа с ПЭВМ может проводиться только в помещениях без повышенной опасности, и возможность поражения током может быть только при прикосновении непосредственно с элементами ПЭВМ.

Помещение, в котором проводились работы, относится к помещению класса 1 - без повышенной опасности поражения электрическим током, то есть отсутствуют условия, создающие повышенную опасность согласно ПУЭ 1.1.13.

Офис, в котором производилась разработка, по степени пожаро-взрывоопасности относится к категории «В», поскольку горючие вещества и материалы находятся в твердом состоянии без выделения пыли. Поэтому необходимо предусмотреть ряд профилактических мероприятий технического, эксплуатационного, организационного плана.

5.1.3.2 Мероприятия по предотвращению ЧС

Для предупреждения пожаров от коротких замыканий и перегрузок необходимы правильный выбор, монтаж и соблюдение установленного режима эксплуатации электрических сетей, дисплеев и других электрических средств автоматизации.

Следовательно, необходимо предусмотреть ряд профилактических мероприятий технического, эксплуатационного, организационного плана.

На каждом этаже здания, на видном месте вывешен план эвакуации с этажа (здания). На плане эвакуации кроме путей выхода (стрелками), указываются места размещения средств пожаротушения, телефонов.

Для предупреждения возникновения пожара необходимо соблюдать следующие правила пожарной безопасности:

а) исключение образования горючей среды (герметизация оборудования, контроль воздушной среды, рабочая и аварийная вентиляция);

б) применение при строительстве и отделке зданий несгораемых или трудно сгораемых материалов.

Необходимо в офисе проводить следующие пожарно-профилактические мероприятия:

Организационные мероприятия:

– противопожарный инструктаж обслуживающего персонала;

– обучение персонала правилам техники безопасности;

– издание инструкций, плакатов, планов эвакуации.

– Эксплуатационные мероприятия:

– соблюдение эксплуатационных норм оборудования;

– обеспечение свободного подхода к оборудованию.

В помещениях рабочие места размещены так, что расстояние между рабочими местами с видеотерминалами (от поверхности экрана одного, до поверхности экрана другого) составляет порядка 4,5 м, расстояния между боковыми поверхностями порядка 1 м, что соответствует нормам, а поэтому дополнительных мер защиты не требуется;

– содержание в исправности изоляции токоведущих проводников.

– технические мероприятия:

– соблюдение противопожарных мероприятий при устройстве электропроводок, оборудования, систем отопления, вентиляции и освещения. В помещении аудитории имеется порошковый огнетушитель типа ОУ-5, установлен рубильник, обесточивающий все помещение, на двери аудитории приведен план эвакуации в случае пожара, и на досягаемом расстоянии находится пожарный щит. Если возгорание произошло в электроустановке, для его устранения должны использоваться углекислотные огнетушители или порошковые;

– профилактический осмотр, ремонт и испытание оборудования.

Кроме устранения самого очага пожара, нужно своевременно организовать эвакуацию людей.

5.1.4 Правовые и организационные вопросы обеспечения безопасности

В соответствии с ГОСТ 12.2.032-78 ССБТ «Рабочее место при выполнении работ сидя. Общие эргономические требования» к рабочему месту предъявляются следующие основные требования:

– Конструкцией рабочего места должно быть обеспечено выполнение трудовых операций в пределах зоны досягаемости моторного поля;

– При организации рабочего места следует учитывать антропометрические показатели женщин (если работают только женщины) и мужчин (если работают только мужчины); если работают и женщины и мужчины – общие средние показатели женщин и мужчин;

– Конструкцией рабочего места должно быть обеспечено оптимальное положение работающего, которое достигается регулированием высоты рабочей поверхности, сиденья и пространства для ног [36].

Выводы и рекомендации

В данной главе были рассмотрены вопросы обеспечения безопасных, безвредных условий труда, необходимых при написании ВКР. Были выделены факторы, оказывающие вредное и опасное влияние на студента в ходе написания работы. В итоге было получено, что помещение, где писалась ВКР, является помещением без повышенной опасности по степени вероятности поражения электрическим током. С точки зрения комфортности микроклимата рассматриваемого помещения есть смысл применять искусственную (механическую) вентиляцию (кондиционеры). Рассматриваемое помещение с точки зрения пожарной безопасности также соответствует необходимым нормам.

Заключение

Результатом настоящей работы является спроектированное и реализованное программное обеспечение для классификации эмоционального тона сообщений пользователей социальной сети Twitter.

В данной работе проводилось исследование основных методов классификации текстов.

В результате были разработаны:

– классификатор тональности сообщений социальной сети Twitter работающий на русскоязычных текстах;

– веб-приложение для взаимодействия пользователя с системой для определения эмоционального класса сообщений пользователей социальной сети.

– способ формирования тренировочных данных с участием пользователей. Участие пользователей позволяет обновлять тренировочный набор данных новыми текстами из социальной сети, что в свою очередь способствует повышению точности классификации.

В результате исследования было выяснено, что для достижения приемлемых показателей точности больше 75%, с использованием свёрточных нейронных сетей с посимвольным кодированием, требуется не менее 200 000 данных для обучения сети. В рамках исследования была получена точность классификации 76,37% для текстов на русском языке. Разработанный классификатор может быть использован для анализа текстов социальных сетей на других языках, где нет обширных баз векторных представлений слов. Кроме того, созданный классификатор может быть использован для создания и анализа социальных графов пользователей с учетом тональности их сообщений.

Conclusion

The result of this work is designed and implemented software for the emotional tone classifier of Twitter social media users' messages.

In this paper, the main methods of classification of texts were considered.

As a result, the following were developed components:

- the tone classifier of the social network Twitter messages working on Russian-language texts;
- a web application for interacting with a network user to determine the emotional class of messages for social network users.
- a way of forming training data with the participation of users. User participation allows you to update the training set of data from social networks, which in turn contributes to the observance of the classification.

As a result of the study it was found that to achieve acceptable accuracy rates of 75%, using convolutional networks with binding, at least 200 000 data are required for network training.

Список использованной литературы

1. S. Owens Your Guide to Twitter Marketing // [Электронный ресурс]: URL: https://moluch.ru/archive/116/31390/ (дата обращения: 05.05.2018).
2. В.В. Осокин. Анализ тональности русскоязычного текста / В.В. Осокин, М.В. Шегай. – Издательство «Интеллектуальные системы», 2016.
3. Tone Analyzer. // [Электронный ресурс]: URL: https://www.ibm.com/watson/services/tone-analyzer/ (дата обращения: 15.05.2018)
4. Repustate. Sentiment Analysis System. // [Электронный ресурс]: URL: https://www.repustate.com/russian-sentiment-analysis/ (дата обращения: 16.05.2018)
5. Xiang Zhang. Character-level Convolutional Networks for Text Classification / Xiang Zhang, Junbo Zhao, Yann LeCun – New York University719 Broadway, 12th Floor, New York, NY 10003.
6. Turney P.D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews // Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002. P. 417 – 424.
7. Котельников Е.В. Автоматический анализ тональности текстов на основе методов машинного обучения. / Котельников Е.В., Клековкина М.В РОМИП 2011.
8. NLTK Documentation. // [Электронный ресурс]: URL: https://www.nltk.org/ (дата обращения: 16.05.2018)
9. Korobov M.: Morphological Analyzer and Generator for Russian and Ukrainian Languages // Analysis of Images, Social Networks and Texts, pp 320-332 (2015).
10. Tomas Mikolov. Distributed Representations of Words and Phrases and their Compositionality / Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean. // [Электронный ресурс]: URL: https://papers.nips.cc/paper/5021-

distributed-representations-of-words-and-phrases-and-their-compositionality.pdf (дата обращения: 17.05.2018)

11. Нугуманова А.Б. Обогащение модели Bag of words семантическими связями для повышения качества классификации текстов предметной области. / Нугуманова А.Б., Бессмертный И.А., Байбурин Е.М., Пецина П. 2016. № 2. С. 89-99.

12. Charles Ashby. Character level sentiment Analysis [Электронный ресурс]: URL: https://charlesashby.github.io/2017/06/05/sentiment-analysis-with-char-lstm/ (дата обращения: 05.04.2018)

13. С. Хайкин. Нейронные сети: Полный курс. // ООО «И. Д. Вильямс», 2006. – 1104 с.

14. Naive Bayes Classification. Wikipedia // [Электронный ресурс]: URL: https://en.wikipedia.org/wiki/Naive_Bayes_classifier (дата обращения: 15.05.2018)

15. Юрий Лифшиц. Метод опорных векторов Лекция № 7 курса «Алгоритмы для Интернета» // [Электронный ресурс]: URL: http://docplayer.ru/27533576-Metod-opornyh-vektorov-lekciya-7-kursa-algoritmy-dlya-interneta.html (дата обращения: 20.05.2018)

16. Mark R. Segal. Machine Learning Benchmarks and Random Forest Regression //Division of Biostatistics, University of California, San Francisco, CA 94143-0560,April 14, 2003.

17. McCulloch, W. S. A logical calculus of the ideas immanent in nervous activity / Warren S. McCulloch, Walter Pitts // Springer New York. — 1943.

18. Yann LeCun. Gradient-based learning applied to document recognition / Yoshua Bengio, Yann LeCun, Leon Bottou, Patrick Haffner // IEEE. — 1998.

19. Krizhevsky A. Imagenet classification with deep convolutional neural networks / Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton // NIPS. — 2012. — 1106-1114 p

20. MongoDB official documentation. Getting Started // [Электронный ресурс]: URL: https://docs.mongodb.com/ (дата обращения: 30.04.18)

21. Ю. В. Рубцова. Построение корпуса текстов для настройки тонового классификатора // Программные продукты и системы, 2015, №1(109), –С.72-78

22. А.Мюллер, Введение в машинное обучение с помощью Python / А.Мюллер, Сара Гвидо – 2017.

23. Keras documentation. // [Электронный ресурс]: URL: https://keras.io/, (дата обращения: 07.05.2018)

24. Нейронные сети. Statsoft // [Электронный ресурс]: URL: http://statsoft.ru/home/textbook/modules/stneunet.html (дата обращения 30.05.2018)

25. Mozilla Developer Nenwork. // [Электронный ресурс]: URL: https://developer.mozilla.org/ru/ (дата обращения: 08.05.2018)

26. Cron. Wikipedia. // [Электронный ресурс]: URL: https://ru.wikipedia.org/wiki/Cron (дата обращения: 06.05.2018)

27. Korobov M.: Morphological Analyzer and Generator for Russian and Ukrainian Languages // Analysis of Images, Social Networks and Texts, pp 320-332 (2015).

28. Н. В. Воробьев. Классификация текстов с помощью сверточных нейронных сетей / Н. В. Воробьев, Е. В. Пучков – 2017.

29. Mark R. Segal. Machine Learning Benchmarks and Random Forest Regression //Division of Biostatistics, University of California, San Francisco, CA 94143-0560, April 14, 2003.

30. ГОСТ 12.0.003-74 ССБТ. Опасные и вредные производственные факторы. Классификация.

31. СНиП 23-05-95 Естественное и искусственное освещение.

32. СанПиН 2.2.2/2.4.1340-03 Гигиенические требования к персональным электронно-вычислительным машинам и организации работы.

33. СанПиН 2.2.2. 542-96 «Гигиенические требования к видео дисплейным терминалам, персональным электронно-вычислительным машинам и организации работы»

34. СанПиН 2.2.4.548-96 «Гигиенические требования к микроклимату производственных помещений»

35. СНиП 2.2.4/2.1.8.562-96 «Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки»

36. Снип 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования.

37. ОНТП 24–86 «Определение категории помещений и зданий по взрывопожарной и пожарной опасности»

38. Технологический регламент по обращению с отработанными люминесцентными ртутьсодержащими лампами. [Электронный ресурс]: - Режим доступа: http://eco-profi.info/othod/instr/602-instr-3533010013011-4.html, свободный. – Загл. с экрана.

Приложение А

(Обязательное)

Исходный код реализации веб-сервисов

```
const express = require('express');
const router = express.Router();
const Tweets = require('./models/Tweets');
const PythonShell = require('python-shell');
const mongoose = require('mongoose');
const jwt = require('jsonwebtoken');
const bcrypt = require("bcrypt");
const checkAuth = require('./middleware/check-auth');
const Users = require('./models/Users');
const twitterUtils = require('./twitter/twitter-utils');

PythonShell.defaultOptions = { scriptPath: 'cnn' };

function classifyTweets(text, callback) {
  let options = {
    mode: 'text',
    pythonPath: 'cnn/venv/bin/python3.5',
    args: ['-s', text]
  };
  PythonShell.run('eval.py', options, (err, results) => {
    if (err) throw err;
    callback(results[0]);
  });
}

function getPositiveTweetsPercent(tweetsCount, positivesCount) {
  return positivesCount * 100 / tweetsCount;
}

router.route('/vote/:id').post((req, res) => {
  var id = req.params.id
  Tweets.findById(id, (error, tweet) => {
    if (error) {
      return next(new Error('Tweet was not found'))
    } else {
        tweet.sentiment = req.body.vote;
        tweet.save({
            function (error, tweet) {
              if (error) {
                  res.status(400).send('Unable to vote');
              } else {
                  res.status(200).json(tweet);
              }
            }
        });
    }
  });
});

router.post('/classify',checkAuth, (req, res, next) => {
  classifyTweets(req.body.text, (result) => {
    result = JSON.parse(result);
    res.json(result);
  });
});
```

```
router.post('/signup', (req, res) => {
  Users.find( {username: req.body.username} )
  .then(user => {
    if (user.length >= 1) {
      return res.status(409).json({
        message: "Пользователь с таким именем уже существует"
      });
    } else {
      bcrypt.hash(req.body.password, 10, (err, hash) => {
        if (err) {
          return res.status(500).json({
            error: err
          });
        } else {
          const user = new Users({
            username: req.body.username,
            password: hash
          });

          user.save().then(result => {
            res.status(200).json({
              message: 'Регистрация успешно завершена'
            })
          }).catch(error => {
            res.status(500).json({
              error: error
            });
          });
        }
      });
    }
  });
});

router.post('/signin', (req, res) => {
  Users.findOne({ username: req.body.username })
  .then(user => {
    bcrypt.compare(req.body.password, user.password, (err, result) => {
      if (err) {
        return res.status(401).json({
          failed: "Unauthorized access"
        });
      } else {
        const JWTToken = jwt.sign({
          username: user.username
        }, "secret", {
          expiresIn: "24h"
        });

        return res.status(200).json({
          message: "Authenticated",
          token: JWTToken
        });
      }
    });
  }).catch(error => {
    res.status(401).json({
      message: "Неправильный логин или пароль"
    });
  });
});
```

```
router.post('/tweets',checkAuth, (req, res, next) => {
  let count = req.body.count;
  twitterUtils.getTweetsBySearchText(req.body.searchText, count)
    .then((result) => {
      let tweetsToAnalyze = [];
      result.data.statuses.forEach((tweet) => {
          tweetsToAnalyze.push(tweet.full_text);
      });
      classifyTweets(tweetsToAnalyze.join('|'), (results) => {
        results = JSON.parse(results);
        console.log(results)
        let tweets = [];
        let positiveTweetsCount = 0;
        results.forEach((result) => {
          tweets.push(result);
          positiveTweetsCount += result.sentiment;
        });
        let positiveTweetsPercent = getPositiveTweetsPercent(count,
positiveTweetsCount);
        let negativeTweetsPercent = 100 - positiveTweetsPercent;
        let tweetsSentimentInfo = {
            tweets,
            info: {
              count,
              positiveTweetsPercent,
              negativeTweetsPercent
            }
        }
        return res.json(tweetsSentimentInfo);
      });
    })
    .catch(err => {
      console.log(err.stack);
    })
});

module.exports = router
```

Приложение Б

(Обязательное)

Emotional tone classifier of Twitter social media users' messages

Студент

Группа	ФИО	Подпись	Дата
8ВМ6Г	Байкадир Жансерик Багдатулы		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент	Цапко Сергей Геннадьевич	к.т.н		

Консультант-лингвист отделения иностранных языков ШБИП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ст. преподаватель	Кудряшова Александра Владимировна			

Introducing

Over the past decade, the use of various online resources, in particular social networks such as Twitter, has increased significantly. Many companies and organizations define these resources as significant for marketing research [1]. Usually, in order to get feedback and understanding of how customers relate to their products, companies conduct interviews, questionnaires and surveys. These standard methods often require a lot of time and money; moreover, they do not always bring the desired result.

To solve the problem of automatic determination of the emotional coloring of the text, algorithms for processing natural languages are used. Deep learning algorithms are among the most popular ones at the moment. There are a lot of works devoted to natural language processing and, in particular, to analyzing tonality using neural networks. But most of them are adapted to being applied to the English language [2].

At the moment, there are Web services for determining the key of the text, but most of them work only with English. There are also services that support Russian-language texts, but these services are commercial, and are not available for a wide range of consumers and researchers. In this paper, the main points related to the implementation of the task of analyzing the tonality of texts in Russian are discussed.

The objectives of this work are:

- To develop classifier for the emotional tone of Russian-language messages from users of the social network Twitter;
- To develop a method for the formation of training data.

1. Overview of technology and solutions

1.1 Purposes and objectives

The purpose of this work is the implementation of the emotional tone classifier of Twitter social media users' messages. To achieve the goal, it is necessary to do the following:

1. Analyze the main methods for solving the task of analyzing the tonality of the text;
2. Analyze the existing algorithms of machine learning;
3. Choose machine learning architecture;
4. Implement encoding of input data;
5. Develop a tone classifier for social network messages from Twitter;
6. Ensure classification accuracy of at least 75%;
7. Test and compare the obtained model with the existing methods of solutions;
8. Develop a web application for user interaction with the system to determine the emotional class of user messages from social network;
9. To develop a method for the formation of training data with the participation of users.

1.2 Existing studies in the analysis of the text tonality

In publications the solutions to the problem of classifying a text by the method of studying with a teacher were considered [4]. These articles consider reviews of various products: the task was to find out whether the reviewer recommends the product to which the review is devoted. In these works, the task was formulated, the main terminology and the first solutions to this problem were proposed.

In [6] it is told that CNN shows 89.6% accuracy of sentences classification on different data samples.

In [7], the authors classify texts at the character level. A set of 70 characters was used to represent each symbol as a one-hot vector. Also, a fixed text 1014 characters long was set. Thus, the text is represented by a binary matrix of size 70x1014. The network has no idea of words and sees them as a combination of characters, and information about the semantic proximity of words is not provided to the network, as in cases pre-trained by Word2Vec vectors.

1.3 Preprocessing and Data Encoding Methods

Preprocessing text affects the quality of the classification using machine learning algorithms with the teacher in a positive way. At the moment, several methods of preprocessing are used:

1. Stemming, which is removing endings and bringing a word to the base. Implemented with the nltk.stem package;

2. Lemmatization, which is bringing a word to its initial form. Implemented with the nltk.stem package;

3. Removing stop words from the list nltk.corpus.

1.3.1 Methods of data encoding

Data that is fed to the neural network input must be numeric. As part of the study, the following methods of converting a text to a vector were considered:

1. Word2Vec;

2. Bag of Words

3. One-hot-encoding

Word2vec is a software tool for analyzing the semantics of natural languages, which takes a large text case as input data and associates each word with a vector, giving out the coordinates of the words at the output. First he creates a dictionary, "learning" on the input text data, and then calculates the vector representation of words. The vector representation is based on contextual affinity: words that occur in

the text next to the same words (hence having a similar meaning) will have close coordinates of the word vectors in the vector representation. The resulting word vectors can be used for processing natural language and machine learning [10]. Disadvantages of word2vec is the re-education of the model for each language and the need for large text corpus. Also, word2vec provides a low level in the analysis of word combinations.

Bag of Words is a model of texts in natural language, in which each document or text looks like an unordered set of words without information about the connections between them. It can be represented in the form of a matrix, each line in which corresponds to a separate document or text, and each column to a specific word. The cell at the intersection of the row and column contains the number of occurrences of the word in the corresponding document. The main disadvantage of this method is that to obtain an ordered matrix. Ignoring the semantic links between words is the main drawback of the Bag-of-words model. Another significant drawback is that texts like word sets are projected into a space of high dimensionality and high sparsity, which is caused by the volume of the dictionary used [11].

These approaches have the following disadvantages when processing texts in Russian:

- low accuracy when working with certain language constructs (negation, punctuation marks, word combinations);
- lack of number of dictionaries in Russian;
- the demand for retraining of the model.

Along with this, messages from social networks can have spelling errors or new words, for this reason, the Word2Vec approach will miss a lot of information, since this method ignores the words which are not in the dictionary [12].

One-hot-encoding is the representation of symbols or a vector with binary values. This approach allows you to obtain a significant increase in accuracy in the task of classifying text in languages where there are complex constructions of words (endings, suffixes, prepositions, etc.). An example of representing a word in Russian at the character level using character-by-symbol encoding is shown in Figure 1.1

	а	б	в	г	д	е	ё	ж	з	и	й	к	л	м	н	о	п	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
х	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
о	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
р	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
о	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ш	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
о	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 1.1 Example of character-by-character encoding

The main advantages of using this approach are:

– the model will be much smaller (about 50-100 mb for the whole model compared to more than 3 GB for the classic Word2Vec);

– the model can understand the basic emotion of repetitive letters;

– the model has a low level of susceptibility to typos;

– suitable for learning the model on texts of any languages.

1.4 Review of methods for classifying text based on the method of learning with a teacher

Teaching methods with the teacher allow you to classify data based on a pre-trained data set called a training set. Such methods should implement two functions: training on training data and classification on new data.

Teaching methods with the teacher are subject to the problem of retraining. Retraining is a phenomenon in which the constructed model classifies the examples from the training set with great accuracy, but when working with examples that differ from the training data, it shows low accuracy. This phenomenon occurs due to the fact that during the learning process the model reveals some regularities in the training set of data that are not present in the total population. The methods of combating retraining depend on specific methods and models.

Also the following classifiers belong to the learning methods with the teacher:

– Naive Bayesian classifier

– Support vector method

1.4.1 The Naive Bayesian Classifier

Naive Bayesian classifier, as the name implies, uses the Bayes theorem – operates with conditional probabilities. Naive means that the words in the sentence do not depend on each other.

According to the article [2], based on a naive Bayesian classifier, an accuracy of 78.4% was achieved on a given test set of data (Table 1). The article uses the concept of stamping. Stamping is the removal of endings and stop words, which have no semantic load.

Table 1.1 Accuracy of text classification using the naive Bayesian classifier [2]

Description of features	**Accuracy [%]**
Ungrams without stamping	76.79
Ungrams with stamping	75.66
Processing negations without stamping	78.26
Removing unnecessary symptoms without stamping	78.39

1.4.1 Support vector method

The main idea of the method is translation of the initial vectors into a space of higher dimension and the search for a separating hyperplane with the maximum gap in this space. The effectiveness of the support vector method is significantly reduced if the number of indicative descriptions is very large.

In [5], using the reference vector method, an accuracy of 81% was obtained on unigrams and combinations between unigrams and bigrams (one word) and 76% on bigrams (two words, a phrase) (Table 1.2).

Table 1.2 Accuracy of the classifier on the basis of the support vector method [5]

Description of features	**Accuracy [%]**

ungramms	81
digrams	76
combination	81

1.4.2 Neural networks

Attempts to reproduce the ability of biological nervous systems to learn and correct mistakes led to the creation of artificial neural networks. Artificial neural networks are a family of models built on the principle of the organization and functioning of biological neural networks - nerve cell networks of a living organism.

The concept of an artificial neural network was proposed back in 1943 by W. McCulloch and W. Pitts in [8]. In particular, they proposed a model of an artificial neuron.

To reflect the essence of biological neural systems, an artificial neuron is constructed as follows. It receives input signals (input data or output signals of other neurons of the neural network) through several input channels. Each input signal passes through a connection having a certain weight. A certain threshold value is associated with each neuron. The weighted sum of the inputs is calculated, the threshold value is subtracted from it and, as a result, the activation value of the neuron is obtained. The activation signal is converted using the activation function and as a result, the output signal of the neuron is obtained.

Figure 1.2 shows an example of an artificial neuron

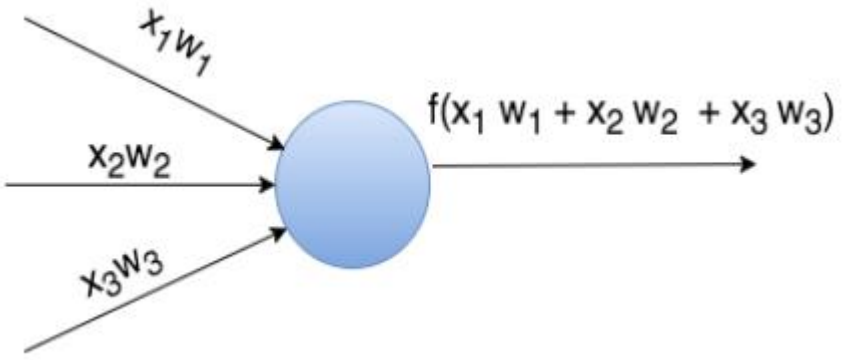


Figure 1.2 Artificial neuron

x_i – input signal

w_i – input signal weight

$f(s)$ – activation function

1.4.3 Convolutional neural networks

With the advent of large amounts of data and large computing capabilities, neural networks began to be actively used. Especially popular were convolutional neural networks, the architecture of which was proposed by Yann Lekun [15] and is aimed at effective image recognition. The network architecture received its name because of the existence of a fold operation, the essence of which is that each fragment of the image is multiplied by the matrix (core) of convolution elementwise, and the result is summed and written to the same position of the output image. The architecture of the network contains a priori knowledge from the subject field of computer vision: the image pixel is more closely connected with the neighboring (local correlation) and the object in the image can meet in any part of the image.

Particular attention was paid to the light-neural networks after the ImageNet contest, which took place in October 2012 and was devoted to the classification of objects in photographs. The contest required the recognition of images in 1000 categories. The winner of this competition - Alex Krizhevsky, using a convolutional neural network, it was possible to reduce the number of errors to 15% [16].

The success of the application of light-neural networks to the classification of images has led to a number of attempts to use this method to other problems. Recently, they have become actively used for the task of classifying texts.

1.4.4 The architecture of convolutional neural networks

A convolutional neural network is usually an alternation of convolution layers, subsampling layers and in the presence of fully-connected layers at the output. All three types of layers can alternate in random order [15].

In the convolution layer, neurons that use the same weights are combined into feature maps, and each feature card neuron is associated with a part of the neurons of the previous layer. When calculating the network, it turns out that each neuron 13 performs the convolution of some area of the previous layer (determined by the set of neurons associated with the given neuron).

An example of the architecture of a neural network is shown in Figure 1.3

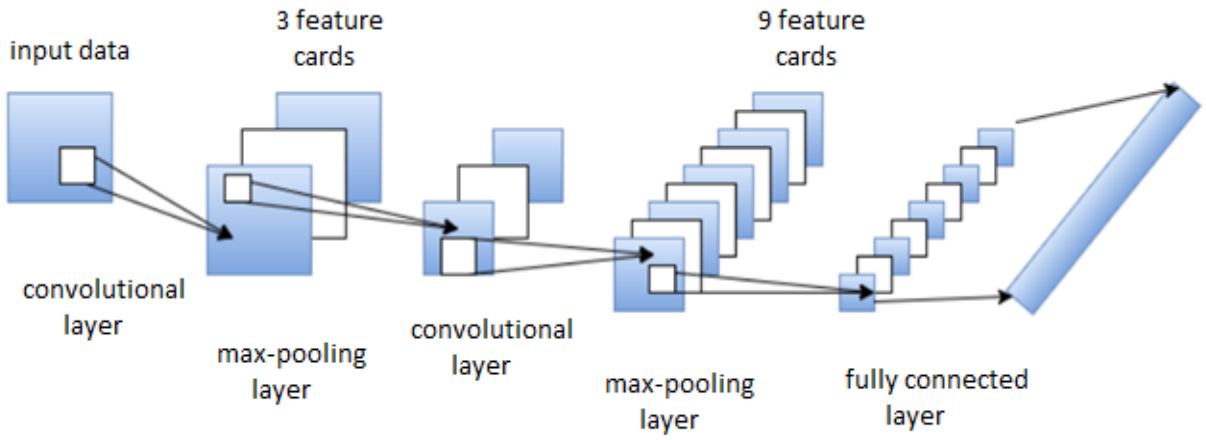


Figure 1.3 The architecture of a convolutional neural network

Fully connected layer

A layer in which each neuron is connected to all neurons at the previous level, with each link having its own weight coefficient. Figure 1.4 shows an example of a fully connected layer.

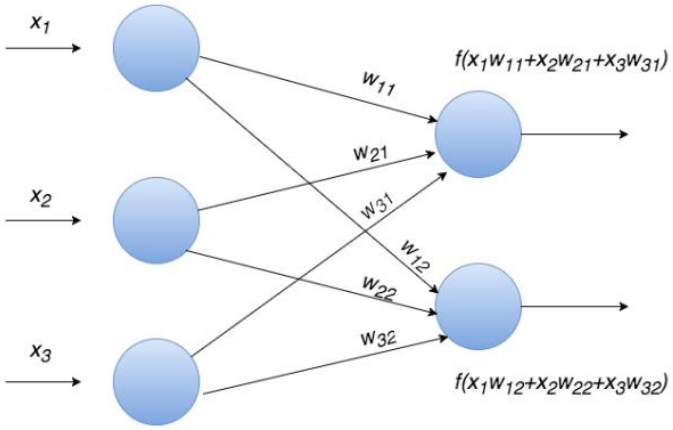


Figure 1.4 Fully connected layer

w_{ij} — weight coefficients.

$f(s)$ — activation function.

Convolution layer

Unlike the fully connected, in the convolution layer, a neuron is connected only to a limited number of neurons of the previous level, i.e., the convolution layer is analogous to the use of the convolution operation, where only a small scale matrix is used (the convolution core), which is "moved" along the entire processed layer.

Another feature of the convolution layer is that it slightly reduces the image due to edge effects.

Figure 1.5 shows an example of a convolutional layer with a 3 × 3 convolution core.

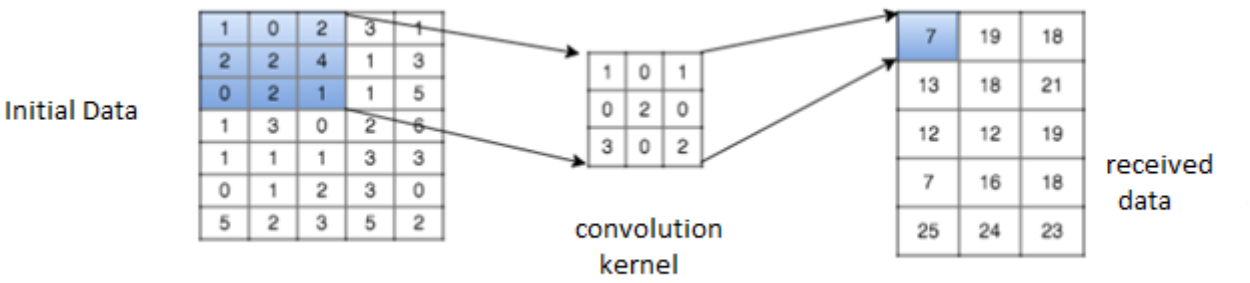


Figure 1.5 Convolution layer

Pooling layer

Layers of this type perform a reduction in dimension (usually several times). This can be done in many ways, but often the max-pooling method is used – the entire feature card is divided into cells from which the maximum values are selected.

Figure 1.6 shows an example of a subsampling layer with the method of selecting the maximum element.

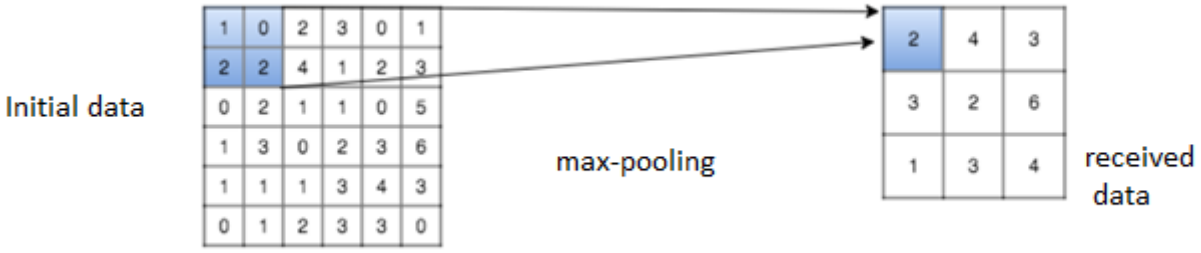


Figure 1.6 Pooling layer

Dropout layer

Dropout layer (dropout regularization) is a method of combating retraining in neural networks, whose training usually produces a stochastic gradient descent by randomly selecting some objects from the sample. Dropout regularization consists in changing the network structure: each neuron is ejected with some probability p. On such a thinned network training is carried out, for the remaining scales a gradient step is made, after which all discarded neurons return to the neural network. Thus, at each step of the stochastic gradient, we set up one of the possible 2^N network architectures, where by architecture we mean the structure of the neuron connections, and N denotes the total number of neurons. When testing a neural network, neurons are no longer discarded, but the output of each neuron is multiplied by $(1 - p)$ – due to this, at the output of the neuron we will receive the expectation

of its response across all 2^N architectures. Thus, the neural network learned with the help of dropout-regularization can be considered as a result of averaging of 2^N networks [17].

2 Software Architecture

To implement the software to determine the emotional color of the text, it was necessary to implement the following components:

– Data warehouse for storing the training data set;

– Binary classifier of the emotional tone of the text based on the neural network;

– A web application that includes functionality for analyzing user messages and updating training data;

– Neural network training scheduler.

The component diagram is shown in Figure 2.1.

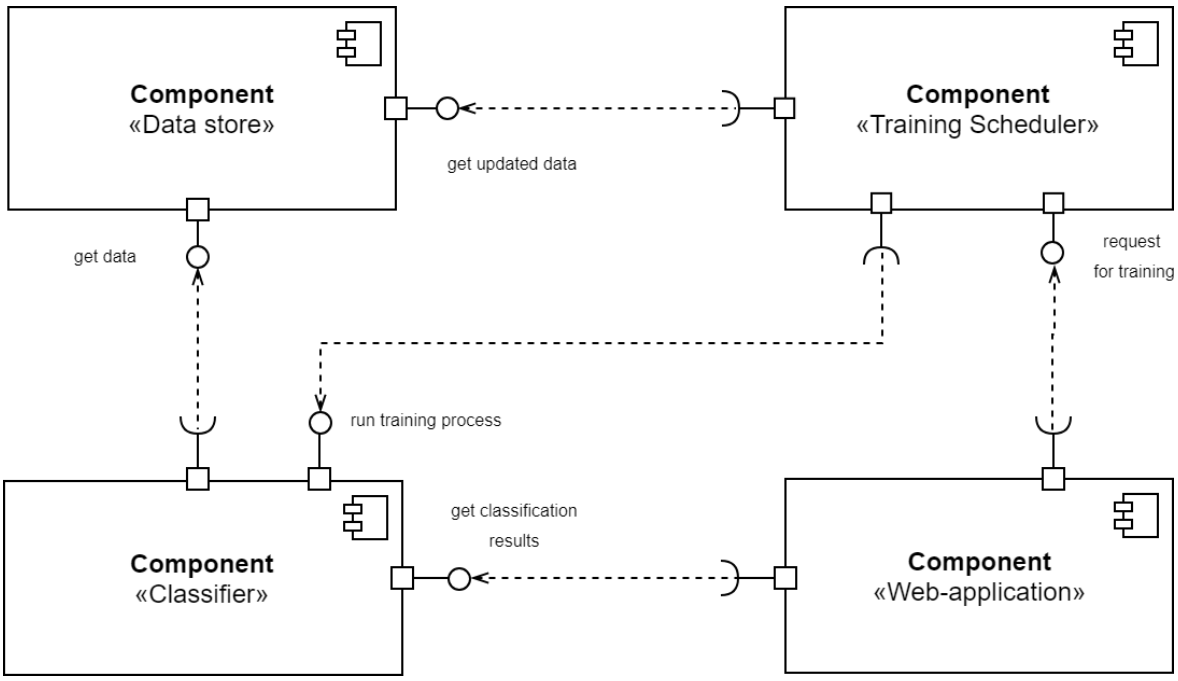


Figure 2.1 Software components for classification of text messages from the social network.

The "Classifier" component is used to obtain training data, learn the model and classify the text coming from the "Web application" component;

The Data Warehouse component is the MongoDB database with training data collections.

The "Training Scheduler" component is used to automatically learn the classifier based on the updated data received from the "Data Warehouse" component.

The "Web application" component is a client web application for interacting with the "Classifier" component and serves for obtaining the results of text classification and for starting the training using the "Training Scheduler" component.

2.1 Data Store

The work of the classifier consists in training on a large training set of data. Based on this, there is a need for a data warehouse.

Basically, for the storage of training data, researchers use ready-made sets in files with tabular format * .CSV. Because with a set of data will be a large number of operations, reading and writing to a file is not suitable. This task is solved by deploying the database.

2.1.1 Selection of funds for implementation

As a means for data storage, a document-oriented MongoDB database was chosen, because the data storage task for learning a neural network does not require more than one entity (collection). Along with this, document-oriented databases have more performance with the same requests for data [8].

MongoDB (from English humongous - huge) is a document-based database management system (DBMS) with open source code that does not require a description of the table schema. Classified as NoSQL, uses JSON-like documents [8].

2.1.2 Definition of the collection

The collection for storing the training data set is an entity from the following fields in the form of JSON (JavaScript Object Notation) notation:

```
{
     “text”: “message”,
     “polarity”: “0”,
     “votedUsers”: [],
     “datasetType”: “type”
}
```

MongoDB automatically adds a unique "id" field when writing data to the collection, for this reason, the "identifier" field for messages is not required.

The "text" field has a string data type and stores the message text.

The field "polarity" has a numeric data type and stores the emotional class in binary form (0 - negative, 1-positive).

The "votedUsers" field is an array and stores the "user ID" of those who voted for one of the emotional classes.

The "datasetType" field has a string data type and stores information about which set of data the message relates to.

2.1.2 Creating collections for data storage

First of all, it is necessary to divide the data into a test and training sample from the body of the text specified in [9], to test the performance of the classifier on those different from the training data. To create a test sample for checking the results, it’s need to separate 10,000 entries from each sample. Thus, in the test sample there will be 10,000 entries pertaining to each of the emotional classes. Further, it is necessary to balance the amount of positive and negative data leaving in the training sample 100,000 entries for each emotional class.

Next, you need to install the necessary software to deploy the local storage. According to MongoDB documentation, in order to successfully deploy local storage, needed to do the following:

1. Run the command to import the key for access to the application download repositories:

```
$ sudo apt-key adv –keyserver hkp://keyserver.ubuntu.com:80 --recv
2930ADAE8CAF5059EE73BB4B58712A2291FA4AD5
```

2. Run the command to import links for downloading MongoDB:

```
$ echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu
xenial/mongodb-org/3.6 multiverse" | sudo tee
/etc/apt/sources.list.d/mongodb-org-3.6.list
```

3. Update the status of local repository lists:

```
$ sudo apt-get update
```

4. Install the stable version of MongoDB with the following command:

```
$ sudo apt-get install -y mongodb-org
```

For start MongoDB run the following command:

```
$ sudo service mongod start
```

After that, the following message to be logged on the command line window that the local storage server is running on port 27017:

```
[initandlisten] waiting for connections on port 27017
```

After a successful deployment of the repository, it's needed to create a collection. Collection can have created in two ways:

– through the server console;

– through the interfaces of program languages.

The database server console allows to create collections and perform standard CRUD (Create Read Update Delete) operations on the data. An example of console requests to the MongoDB server is shown in Figure 2.2.

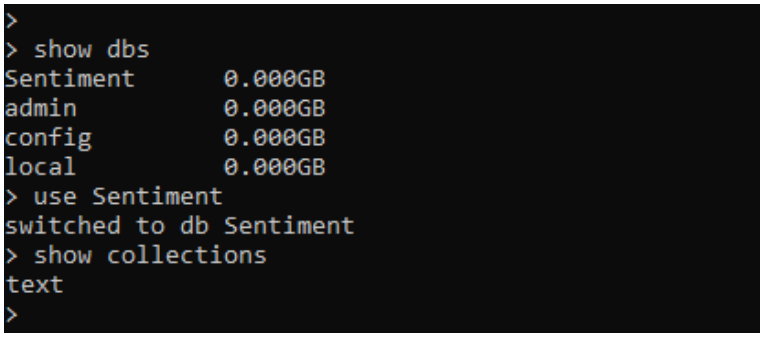


Figure 2.2 Console mode access to MongoDB

The way of interaction with the database through the interfaces of program languages allows you to automate the above processes. At the moment there is a set of libraries for popular programming languages that allow you to interact with the MongoDB database.

In order to write the data divided into files into the database, a small script was developed, allowing iteratively to write down each message.

The script for creating and filling the collection with training data is presented in Attachment A.

2.2 Classifier

The classifier is based on the architecture of a convolutional neural network

The classifier consists of the following classes:

- Class for character encoding of data;
- Class for learning the model;
- Class for testing the model on the test set of data;
- Class for classifying user messages.

2.2.1 Architectural architecture design.

The classifier is a convolutional neural network for solving the problem of binary classification. The architecture of the neural network of the authors of the article [3] was modified to work with the Russian-language text.

The model of a convolutional neural network consists of several convolution layers with different filter widths (in this implementation there are 25 filters of size 1, 50 filters with size 2, 175 filters of size 7 - the height is always the same as the length of the alphabet). Each of them takes as input one word from the sentence for one iteration of the learning process.

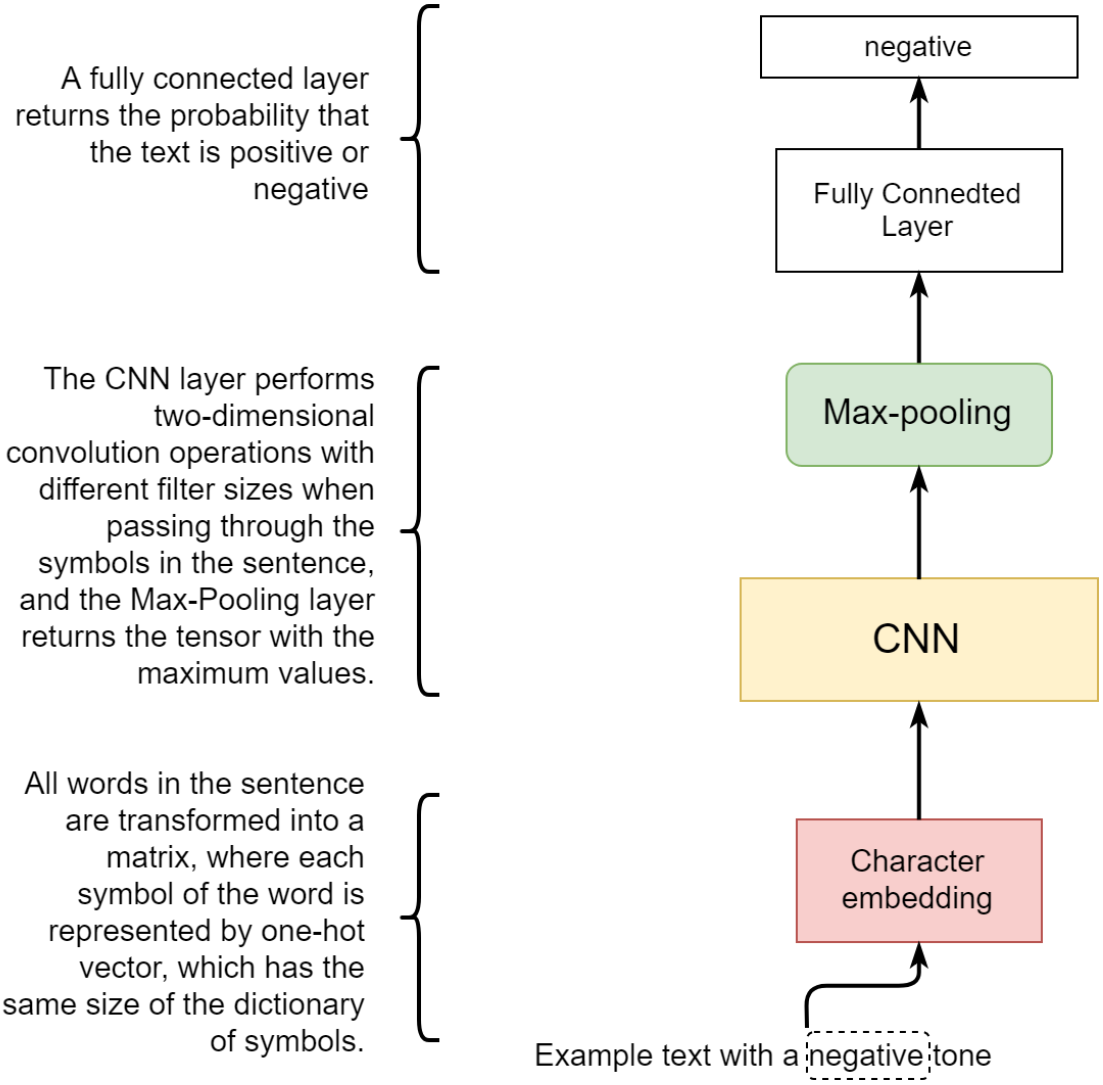


Figure 2.3 Diagram of the convolutional neural network model for binary text-level character classification [12]

Приложение В

(Обязательное)

Диаграмма ганта

№ работ	Вид работ	Исполнители	$T_{кi}$, кал. Дн.	Продолжительность выполнения работ													
				февр.			март			апрель			май			июнь	
				1	2	3	1	2	3	1	2	3	1	2	3	1	2
1	Выбор направления исследования	Р, С	1														
2	Анализ и исследование готовых решений на рынке	Р, С	1														
3	Определение задач	Р	5														
4	Анализ требований	С	6														
5	Изучение методов и технологий для решения задачи	С	18														
6	Выбор оптимальных решений	С	2														
7	Разработка ПО	С	40														
8	Оценка результатов работы	Р, С	4														
9	Оформление пояснительной записки	С	6														
10	Подготовка к защите ВКР	С	21														

Студент (С)

Руководитель от предприятия (РП)