Титул выпускная квалификационная работа

Задание

РЕФЕРАТ

Выпускная квалификационная работа магистра 60 с., 11 рис., 7 табл., 34 источн., 0 прил.

МАШИННОЕ ОБУЧЕНИЕ, ОБРАБОТКА ЕСТЕСТВЕННОГО ЯЗЫКА, АНАЛИЗ ТЕКСТА, КЛАССИФИКАЦИЯ, СОЗДАНИЕ ПРИЗНАКОВ, НЕЙРОННЫЕ СЕТИ, ПОДБОР ГИПЕРПАРАМЕТРОВ, МЕТРИКИ МАШИННОГО ОБУЧЕНИЯ

Объект исследования – процессы динамичности и энергичности в художественных произведениях, измерение изменчивости событий и эмоциональной насыщенности на протяжении книг.

Цель работы – разработка методики оценки динамичности и энергичности художественных произведений и их классификация на основе методик для улучшения понимания влияния динамичности на свойства произведения, заключение о возможности получения объективной оценки.

Методы исследования: генерация идей, сравнение, анализ и синтез, предобработка данных, количественный анализ данных (частотный анализ, статистический анализ, модели машинного обучения для классификации и прогнозирования.

Результат работы: получены выводы по характеристикам книг, алгоритмам их получения, обработаны, протестированы и выбраны подходящие модели классификации.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 6](#_Toc167655508)

[1 Анализ поставленной задачи и выбор направления 7](#_Toc167655509)

[2 Введение в машинное обучение и методов обработки естественного языка (NLP) 8](#_Toc167655510)

[3 Основные методы предобработки текста 9](#_Toc167655511)

[4 Извлечение характеристик 10](#_Toc167655512)

[5 Алгоритмы классификации 12](#_Toc167655513)

[ОСНОВНАЯ ЧАСТЬ 15](#_Toc167655514)

[1 Исследование динамики и энергетики текста: проблемы и подход 15](#_Toc167655515)

[1.1 Проблематика и актуальность 15](#_Toc167655516)

[1.2 Методология анализа динамики и энергетики текста 16](#_Toc167655517)

[1.3 Теоретическая часть моделей 17](#_Toc167655518)

[1.4 Выбор метрик 21](#_Toc167655519)

[2. Данные 23](#_Toc167655520)

[2.1 Общее описание художественных произведений 23](#_Toc167655521)

[2.2 Выбранные характеристик 23](#_Toc167655522)

[2.3 Алгоритм получения характеристик 24](#_Toc167655523)

[2.4 Обоснование выбора средних значений характеристик 25](#_Toc167655524)

[3. Оценка полученных характеристик 27](#_Toc167655525)

[3.1 Общие сведения 27](#_Toc167655526)

[3.2 Среднее общее количество глаголов в абзаце 28](#_Toc167655527)

[3.3 Среднее общее количество слов в абзаце 30](#_Toc167655528)

[3.4 Среднее количество глаголов на предложение 32](#_Toc167655529)

[3.5 Среднее количество слов на предложение 34](#_Toc167655530)

[3.6 Среднее количество предложений в абзаце 36](#_Toc167655531)

[3.7 Наиболее часто встречающаяся тональность 38](#_Toc167655532)

[3.8 Заключение по характеристикам 39](#_Toc167655533)

[4. ML классификация 41](#_Toc167655534)

[4.1 Выбор моделей 41](#_Toc167655535)

[4.2 Обучение и анализ моделей 41](#_Toc167655536)

[4.3 Заключение по моделям 50](#_Toc167655537)

[5. Заключение по основной части 51](#_Toc167655538)

[ЗАКЛЮЧЕНИЕ 54](#_Toc167655539)

[БИБЛИОГРАФИЧЕСКИЙ СПИСОК 56](#_Toc167655540)

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

ИИ – Искусственный Интеллект

**NLP – Natural Language Processing – Обработка естественного языка**

ML – Machine Learning – Машинное обучение

NLTK – Natural Language Toolkit – пакет библиотек и программ для работы с естественным языком

LSTM – Long Short–Term Memory – Долгая краткосрочная память

CNN – Convolutional Neural Network – Свёрточная нейронная сеть

SVM – Support Vector Machine Classifier – метод опорных вектором

ВВЕДЕНИЕ

Искусственный интеллект в современном мире используется во многих отраслях, от анализа и обработки информации, примером могут послужить чат–боты, переводчики и поисковики, до реального применения в производствах, например, автоматизированный процесс ковки на предприятиях General Electric.

Применение ИИ в этих сферах позволяет улучшить скорость, эффективность и точность выполнения задач. Он может обрабатывать большие объемы данных, которые человек не может обработать вручную, и принимать решения на основе этих данных. ИИ также может учиться на основе предыдущего опыта и адаптироваться к изменяющимся условиям, что делает его очень полезным инструментом в современном мире.

Также, одно из преимуществ ИИ, это выдача ответа/результата на основе объективных выводов по поступающей информации. В отличие от человека, ИИ не имеет настроения, субъективного мнения и не подвержен эмоциональным влияниям. Это позволяет ему обрабатывать большие массивы данных без искажения результатов из–за личных предпочтений или предубеждений. ИИ может быть обучен на основе объективных данных и алгоритмов, что делает его более точным и надежным в принятии решений.

Благодаря данной особенности, ИИ может применятся в сферах, где объективная оценка объекта будет напрямую влиять на эффективность производства.

Одна из таких сфер – оценка литературных произведений издательскими компаниями. Оценку работы автора и выбор, работать с ним или нет заключает человек.

Показательным примером, плохой оценки произведения, с точки зрения денежной прибыли для издателя, являет роман “Гарри Поттер и философский камень” Джоан Роулинг. Данное произведение разошлось тиражом более 120 миллионов экземпляров. Однако изначально Роулинг было отказано в 11 издательствах.

Помочь человеку делать более объективные оценки произведений поможет ИИ и метод обработки естественного языка (NLP).

Задача NLP – анализировать текст, и выделять из него необходимые признаки. Метод может справляться с морфологическим, синтаксическим и семантическим анализом.

Цель ВКР – разработать методику оценки динамичности и энергичности художественных произведений и оценить модели классификаторы для лучшего понимания влияния динамичности на свойства произведения и возможности получения объективной оценки для ООО "Издательство "Эксмо".

Результатом работы будет считаться возможность получения различных признаков из текста и возможности модели получать удовлетворительные результаты, которая поможет заказчику автоматизировать оценивание произведений на основе полученных статистик.

В дополнение работа может стать основой для более полномасштабного исследования и разработки ML NLP системы классификаторов.

1 Анализ поставленной задачи и выбор направления

Задача оценки динамичности текста может быть рассмотрена как задача классификации, поскольку она требует отнесения текста к одной из двух категорий: "динамичный" или "не динамичный". Это классическая задача машинного обучения, которая может быть решена с использованием различных методов и алгоритмов классификации, таких как логистическая регрессия, градиентный бустинг, нейронные сети и другие.

В отличие от кластеризации, где целью является группировка объектов без заранее определенных категорий, в задаче классификации целью является отнесение объекта к одной из заранее определенных категорий.

В контексте задачи классификации, каждый текст представляется как объект, который имеет определенные характеристики или признаки. Этими признаками могут быть, например, частота использования определенных слов или фраз, длина предложений, наличие определенных грамматических конструкций и так далее.

Затем, на основе этих признаков, алгоритм классификации определяет, к какой категории относится данный текст. Например, если текст содержит много глаголов в активном залоге и предложений с большим количеством действий, то он может быть классифицирован как "динамичный". Если же текст содержит больше описательных фраз и пассивных конструкций, то он может быть классифицирован как "нединамичный".

Таким образом, задача оценки динамичности текста является задачей классификации, где каждый текст представляется как объект с определенными признаками, а алгоритм классификации определяет, к какой категории относится данный текст на основе этих признаков.

Примером и ориентиром задачи классификации может послужить задача анализа тональности текста, разработке которой посвящено уже не мало времени и разработаны различные подходы.

Ключевой задачей в анализе тональности является классификация эмоциональной окраски текста, определение, является ли выраженное мнение в тексте положительным, отрицательным или нейтральным [1].

Подход на основе ML для классификации является более универсальным и не требует создания дополнительных данных, таких как словари и правила [2].

Поэтому, для успешного решения поставленной задачи – динамичности и энергичности текста, можно взять за ориентир приведённый выше пример.

2 Введение в машинное обучение и методов обработки естественного языка (NLP)

Машинное обучение (ML) – это область искусственного интеллекта, где алгоритмы учат машину думать и функционировать как человек. ML–инженер не программирует напрямую, он работает с данными и передает машине желаемый результат. Задача машинного обучения – научить модель самостоятельно находить решение. [3].

Обработка естественного языка (NLP) – это область искусственного интеллекта и компьютерной лингвистики, которая занимается анализом и синтезом речи и текстов с помощью ЭВМ. Это актуальное направление, так как объем данных постоянно растет, и большая часть из них – неструктурированная информация [4].

ML и NLP связаны между собой. Используя их связки, мы можем классифицировать текст, анализировать тональность текста и распознавать именные сущности.

При классификации текста, как в нашей задаче, над текстом обычно проводятся следующие последовательные шаги: предварительная обработка текста, извлечение характеристик, классификация текста [5].

3 Основные методы предобработки текста

Даже самой современной модели необходима предобработка данных [1].

Предварительная обработка текста включает несколько шагов: токенизацию (разделение текста на более мелкие фрагменты), нормализацию (приведение всех слов к одному регистру, удаление знаков пунктуации и преобразование чисел в текстовые эквиваленты), удаление стоп–слов (слов, не несущих смысловой нагрузки) и стемминг/лемматизацию (приведение слов к их основе или корневому формату). Эти методы упрощают понимание текста и обеспечивают качественную обработку для дальнейшего анализа [6]–[7].

При задаче классификации тональности текста, возникла проблема, что, убирая стоп слова, такие как “нет”, тональность менялась на противоположную [2]. В контексте нашей задачи данное обстоятельство не представляет существенных трудностей.

Для нашей задачи препятствием могут возникнуть две предобработки – это нормализация и лемматизация.

Убрав знаки препинания, можно разрушить структуру нашего текста из абзацев и предложений, и, следовательно, потеряв большое количество данных. Поэтому этот шаг предобработки нужно будет делать вручную, а не использовать готовые библиотеки.

При лемматизации часть деепричастий и причастий с некоторой вероятность станут глаголами, что может сказаться на результате модели [8].

4 Извлечение характеристик

Для обучения модели нам необходимы не только данные, но и характеристики.

Характеристики – это параметры, которые описывают данные. Например, для текста это длина предложений, для изображения – пиксели, для звука – частоты. Характеристики помогают модели понять, как данные связаны друг с другом [9].

Процесс сопоставления данных и характеристик учит модель выявлять закономерности и делать прогнозы. Если обучаем модель классификации текстов, данные – это тексты, а характеристики – это слова в текстах, длина предложений и другое.

Модель определяет, какие слова чаще встречаются в текстах определенного класса.

Условно, при выборе характеристик мы можем разделить их на 2 группы: на основе семантики (анализ по тексту) и на основе векторных представлений.

Word2vec и GloVe – это модели, которые позволяют представить слова в виде векторов, где близкие по смыслу слова будут иметь близкие векторы [5].

На основе данных методов строилась классификация тональности текста. Однако для получения характеристик в нашей задаче, нужен другой подход.

Динамичность и энергичность рассматривается как семантический феномен [10].

Характеристиками для построения модели под поставленную задачу могут служить:

1. Частота глаголов [10]–[11]
2. Длина абзацев/предложений [11]
3. Тональность абзацев [11]

Большинство библиотек для работы и анализа текста позволяют извлечь данные признаки.

Крупная библиотека NLTK (Natural Language Toolkit) является мощным инструментом для обработки и анализа текстов на естественном языке. Она предоставляет широкий набор компонентов для задач текстовой аналитики, включая токенизацию, стемминг, лемматизацию, синтаксический разбор и многие другие. Однако, несмотря на возможность работы с русским языком, изначальное направление на английский язык может стать проблемой [12].

Проекты DeepPavlov и NATASHA – библиотеки, которые были изначально ориентированы на работы с русским языком в NLP.

DeepPavlov – это проект с открытым исходным кодом, для анализа текста и разработки приложений, с большим количеством инструментов [13].

DeepPavlov имеет несколько предобученных моделей на основе различных архитектур ИИ, для решения задач, включая задачу классификацию текста.

NATASHA – проект который, также решает основные задачи в области NLP, нацеленный на русский язык: разделение на токены и предложения, морфологический и синтаксический анализ, лемматизация, извлечение, нормализация именованных сущностей [14].

На сайте разработчиков данного проекта, приведены различия с DeepPavlov:

Таблица 1 – Сравнение проектов DeepPavlov и NATASHA

|  |  |
| --- | --- |
| Natasha | DeepPavlov |
| Решение нескольких базовых задач: сегментация, морфология, синтаксис, NER | Много моделей, специализация на диалоговых системах, решает базовые и сложные задачи: QA, диалог |
| Коммерческая разработка, практичность | Наука |
| Компактные модели: малый размер, в том числе работают на CPU | Тяжёлые модели: большое файлы с весами, много RAM, требуют GPU |
| Только русский язык | Русский, английский, некоторые славянские |
| Набор библиотек | Монолит |
| PyTorch | TensorFlow |

Каждый из проектов может подойти для получение необходимых признаков. Но вот для обучения классификатора на основе уже числовых данных эти библиотеки не предназначены.

DeepPavlov можно выбрать для нашей задачи по двум причинам:

а) Основная – возможность в будущем на уже отработанном решении расширить модель не только для русского языка, но и для других;

б) Дополнительная – без использования дополнительных библиотек создать популярные диалоговые системы.

5 **Алгоритмы классификации**

Классификация текста – это процесс разделения текстовых данных на категории или классы. Алгоритмы классификации на основе машинного обучения позволяют автоматизировать этот процесс.

В определённом контексте, можно утверждать, что переход от текстовых данных к количественным данным приводит нас к традиционной задаче классификации, таким образом уйдя от NLP. В классической задаче классификации эти данные представлены числами или числовыми характеристиками. Это означает, что каждый объект имеет свои числовые параметры, которые описывают его свойства. Например, в задаче классификации текста каждый текст может быть представлен в виде числового набора, где каждое число может отражать частоту встречаемости слов или длину предложений в тексте.

LSTM (Long Short–Term Memory) и CNN (Convolutional Neural Network) традиционно ассоциируются с обработкой текстов и изображений, однако их применение не ограничивается лишь этими доменами. Эти модели также могут быть эффективно использованы для классификации на основе числовых данных, благодаря своим уникальным архитектурным особенностям [15].

LSTM, специализированный тип рекуррентной нейронной сети, является мощным инструментом для анализа последовательных данных. Она состоит из ячеек, которые взаимодействуют друг с другом по определенным правилам. Входные данные поступают в разные моменты времени, а сеть выдает значения в каждый момент времени. Это делает LSTM идеальной для случаев, когда числовые данные связаны с временными рядами или когда структура данных содержит важные последовательные зависимости, которые необходимо учитывать для правильного прогнозирования [16].

CNN, хоть и преимущественно известны своей способностью к анализу изображений, могут также быть эффективно применены к числовым данным. Сверточные сети способны выделять ключевые паттерны из данных благодаря своей структуре с сверточными слоями, что делает их подходящими для любых задач, где числовые данные могут быть представлены в виде многомерных массивов или матриц [17]–[18].

Однако также следует обратить внимание на более классические классификаторы числовых данных:

а) LogisticRegression – логистическая регрессия представляет собой статистический метод, используемый для прогнозирования вероятности принадлежности объекта к определённому классу, что делает её популярным выбором для двоичной и многоклассовой классификации [19];

б) SVM (Support Vector Machine Classifier) – SVM отличается высокой эффективностью в решении задач бинарной классификации. Метод основан на нахождении оптимальной гиперплоскости, разделяющей классы данных [20];

в) RandomForestClassifier – случайный лес состоит из множества деревьев решений, результаты которых агрегируются для улучшения точности и устойчивости модели. Этот метод особенно хорош в обработке данных с множеством признаков и различных типов данных [21];

г) GradientBoostingClassifier – метод градиентного бустинга последовательно строит ансамбль слабых моделей, обычно деревьев решений, оптимизируя при этом функцию потерь. Он эффективен для широкого диапазона задач и является одним из самых мощных доступных методов классификации [22];

д) KNeighborsClassifier – классификатор на основе ближайших соседей. Метод принимает решение на основе меток самых близких объектов в пространстве признаков. КНН прост в реализации и может быть очень эффективен, особенно при правильном выборе метрики расстояния и количества соседей [23].

Все эти модели получили широкое распространение благодаря своей адаптивности, высокой точности и способности работать с разными типами данных. Каждая из них имеет свои особенности, что делает их подходящими для различных сценариев исследования данных в зависимости от конкретных потребностей задачи классификации.

ОСНОВНАЯ ЧАСТЬ

1 Исследование динамики и энергетики текста: проблемы и подход

1.1 Проблематика и актуальность

1.1.1 Введение в проблематику оценки художественных текстов

В современной филологии и компьютерной лингвистике большое внимание уделяется разработке методов анализа текста, позволяющих оценить его насыщенность, стилистические и эмоциональные характеристики.

С развитием различных нейросетей и технологий, включая GPT, а также с расширением накопленной базы уроков и знаний, процесс создания собственного художественного произведения стал более доступным и эффективным.

Однако уровень таких произведений не всегда достигает хорошего результата.

А также случаи, когда ставшие впоследствии популярные книги, как например «Хроники Нарнии» К. С. Льюиса, получившего 37 отказов, или «Дюна» Фрэнка Герберта с 23 отказами, показывают необходимость получения независимых отзывов и оценок.

1.1.2 Актуальность темы

Постоянное увеличение объема текстовой информации и стремление к более объективной оценке, отходя от субъективных мнений, могут способствовать более эффективному анализу литературных произведений. Например, использование искусственного интеллекта (ИИ) основанного на алгоритмах машинного обучения может помочь в этом процессе. Такие системы могут быть полезны для редакторов, у которых может быть плохое настроение или предпочтения к определенным стилистикам, а также для обработки больших объемов текстовых данных. Как пример возможное дополнение, такие оценки и метки помогут улучшать рекомендательные алгоритмы.

Примерами таких оценок и меток могут служить:

а) Оценка простоты/сложности изложения текста;

б) Оценка детализации описания сцен, персонажей и локаций;

в) Анализ информации о месте и времени происходящих событий.

Также, одним из перспективных направлений исследований в данной области является анализ динамичности и энергичности текста, который может иметь значительные теоретические и практические последствия для изучения художественных произведений.

Динамичность текста может быть интерпретирована как степень его способности удерживать внимание читателя, а также как мера изменчивости и активности сюжета или стиля. Энергичность же описывает степень эмоциональной насыщенности и силы выражения.

Такой анализ представляет собой сложную задачу, так как подразумевает не только фиксацию частотности определенных слов или фраз, но и попытку понять, как различные элементы текста сотрудничают для создания общего впечатления о тексте. Поэтому традиционные методы лингвистического анализа дополняются современными компьютерными методами, такими как машинное обучение и нейросетевые алгоритмы, что позволяет переопределить и углубить понимание структуры и смысла художественного текста.

1.2 Методология анализа динамики и энергетики текста

Для анализа и создания модели необходимо определится с признаками/ характеристиками.

Для классификации тональности текста было взято отношения между словами и контекст. Для построения классификатора по выбранной теме, с точки зрения лингвистики, динамичность и энергичность текста будет рассматривается как семантический феномен [10].

В рамках данного исследования идёт фокусировка на анализе следующих параметров текста: количество слов и глаголов, длина абзацев и предложений. Эти параметры позволяют нам получить количественные характеристики текста, которые могут быть использованы для оценки его динамичности и энергичности. Такой подход основан на извлечении и анализе конкретных структурных особенностей текста, что позволяет нам оценить его общий характер и степень выразительности без необходимости определения конкретных динамичных глаголов или связей между словами.

Для более быстрой и лучшей обработки данных будут исследованы как классические алгоритмами машинного обучения – Logistic Regression, SVM, Random Forest, Gradient Boosting и K–Nearest Neighbors; так и популярные в NLP классификации, но в нашей задаче уже на числовых, нейронные сети LSTM и CNN.

1.3 Теоретическая часть моделей

LSTM – это нейронная сеть, которая используется для обработки последовательностей данных. Она состоит из ячеек, которые взаимодействуют друг с другом по определенным правилам. Входные данные поступают в разные моменты времени, а сеть выдает значения в каждый момент времени. В LSTM есть несколько вентилей, которые определяют, какие значения из предыдущего состояния ячейки должны быть забыты, сохранены или добавлены в новое состояние. Новое состояние ячейки – это сумма предыдущего состояния и новых значений–кандидатов. Выходной вентиль определяет, какие значения из нового состояния ячейки должны быть выданы в качестве выходного сигнала [15].

LSTM используется в задачах классификации из–за своей способности учитывать долгосрочные зависимости в последовательных данных. Плюсом использования LSTM в задачах классификации является его способность моделировать контекст и выявлять сложные закономерности в последовательностях текста, что улучшает качество классификации. Однако, минусом может быть более высокая вычислительная сложность по сравнению с другими моделями, что может привести к увеличению времени обучения модели [15].

CNN – это тип нейронной сети, который успешно применяется для обработки изображений и текста. Он использует операции свертки и агрегирования, которые позволяют выделять важные признаки из данных. Структура CNN адаптирована под работу мозга и органов зрения, что позволяет ей эффективно распознавать визуальные образы. Однако также благодаря векторному представлению слов, CNN может применяться и для текстовых данных. Принцип работы CNN основан на концепции рецептивных полей, где каждый нейрон обрабатывает информацию только из части данных, что позволяет извлекать содержательные признаки. Этот подход помогает понимать значение слов и контекста в тексте, делая CNN эффективным инструментом для анализа и классификации текстовых данных [15].

Преимущество использования CNN в классификации текста заключается в его способности автоматически извлекать сложные признаки и шаблоны из текстовых данных, что может повысить точность классификации. Однако недостатком является то, что CNN требует большого количества данных для обучения и настройки, что может быть затруднительно для небольших наборов данных [16].

Логистическая регрессия (Logistic Regression) является одним из основных статистических методов для бинарной классификации. Эта модель предсказывает вероятность принадлежности к одному из двух классов, что делает её особенно полезной в случаях, где необходимо определить, принадлежит ли объект к определённой категории или нет. Основной принцип модели заключается в применении логистической функции (сигмоида) к линейной комбинации входных факторов, чтобы получить результат, ограниченный между 0 и 1, который интерпретируется как вероятность. Благодаря своей простоте и эффективности во многих случаях, логистическая регрессия находит широкое применение в финансах, медицине, социальных науках и многих других областях [19].

Логистическая регрессия хорошо подходит для задач, где данные линейно разделимы, то есть, когда существует гиперплоскость, разделяющая два класса. Однако, эта модель может стать менее эффективной, если данные содержат сложные нелинейные зависимости. В таких случаях требуются методы регуляризации или использование других, более сложных моделей [19].

Метод опорных векторов (SVM) является одним из наиболее мощных и гибких методов классификации. Основной идеей этой модели является нахождение так называемой "оптимальной разделяющей гиперплоскости", которая максимизирует расстояние (зазор) между двумя классами входных данных. Метод опорных векторов хорошо работает на данных с чёткой маргинальной разделимостью и превосходит многие другие классификаторы, особенно когда важно избежать переобучения в условиях ограниченной выборки [20].

SVM может использовать различные ядерные функции для обработки данных с нелинейными свойствами, включая полиномиальные, радиально–базисные функции (RBF) и сигмоидальные. Это делает его особенно полезным для задач, где данные нелинейно разделены. Однако, выбор и настройка правильного ядерного параметра требует опыта и экспериментов [20].

Случайный лес (Random Forest) – это алгоритм ансамблевого обучения, основанный на методе деревьев решений. Каждое дерево в "лесу" обучается на случайно выбранной подвыборке исходного набора данных с использованием случайного подмножества признаков, что делает модель устойчивой к переобучению и обеспечивает высокую точность классификации. Ансамблируя результаты множества деревьев, случайный лес уменьшает вариативность, что часто приводит к улучшению общей производительности по сравнению с отдельными деревьями решений [21].

Одним из основных преимуществ использования случайного леса является его способность эффективно обрабатывать большие наборы данных с большим количеством переменных (функций), не требуя масштабирования данных или сложной настройки. Однако, несмотря на простоту использования, модель требует тщательного подбора параметров, таких как количество деревьев в лесу и глубина каждого дерева, чтобы оптимизировать производительность и избежать чрезмерного расхода вычислительных ресурсов [21].

Градиентный бустинг (Gradient Boosting) – это техника машинного обучения для задач регрессии и классификации, которая строит модель путем построения ансамбля слабых предсказательных моделей, обычно деревьев решений. Основная идея метода заключается в последовательном добавлении новых моделей, каждая из которых исправляет ошибки предыдущих, таким образом улучшая общую производительность. Градиентный бустинг обладает высокой предсказательной силой и может эффективно обрабатывать различные типы данных, при этом оставаясь устойчивым к переобучению при правильно подобранных параметрах [22].

При работе с градиентным бустингом ключевым аспектом является настройка параметров, таких как скорость обучения, количество деревьев в ансамбле и глубина каждого дерева. Высокая сложность модели может привести к большим затратам времени на обучение и требованием к вычислительным ресурсам, но, при грамотной настройке, метод способен дать выдающиеся результаты [22].

Метод ближайших соседей (K–Nearest Neighbors, KNN) – это один из самых простых алгоритмов машинного обучения, использующихся для классификации и регрессии. В KNN классификация объекта осуществляется путем подсчета голосов его соседей, с объектом, присваиваемым классу, наиболее частому среди его k ближайших соседей. Это метод, основанный на экземплярах, где функция решения аппроксимируется локально и все вычисления откладываются до стадии классификации [23].

Основное преимущество KNN заключается в простоте его реализации и понимания, а также в способности эффективно адаптироваться к реальным изменениям данных. Недостатки метода включают его чувствительность к масштабу признаков и к выбору параметра k. KNN может быть неэффективным для больших объемов данных [23].

1.4 Выбор метрик

Для оценки эффективности моделей были выбраны следующие метрики: Accuracy, Precision, Recall, F1–score, AUC–ROC и Confusion Matrix.

Выбор этих метрик для анализа бинарного классификатора объясняется их способностью комплексно оценивать эффективность классификации, предоставляя как общее понимание точности модели (Accuracy), так и более детальные аспекты, такие как способность модели корректно идентифицировать положительные случаи (Precision, Recall), равновесие между этими показателями (F1–score), общую способность различать классы при разных порогах (AUC–ROC) и визуальное представление всех результатов классификации (Confusion Matrix). Это комплексное использование помогает учитывать как сильные, так и слабые стороны модели, что крайне важно для точной настройки и требуемой оптимизации в реальных условиях применения [24].

Подробнее о каждой метрике.

Accuracy (точность) используется для оценки общей доли правильно классифицированных случаев и является одним из наиболее интуитивно понятных показателей эффективности классификатора. Однако этот показатель может вводить в заблуждение при несбалансированных классах, поскольку он не учитывает распределение ошибок между классами [24].

Precision (точность классификации) важен для случаев, когда стоимость ложноположительной классификации высока. Этот показатель показывает долю истинно положительных результатов среди всех результатов, классифицированных как положительные, и помогает понять, насколько наш классификатор точен при прогнозировании класса событий [25].

Recall (полнота) необходима, когда важно улавливать как можно больше положительных случаев (например, в медицинских приложениях). Он показывает, какую долю из всех действительных положительных случаев классификатор смог правильно идентифицировать, что критически важно, если пропуск положительных случаев может иметь серьезные последствия [26].

F1–score (F1–мера) объединяет показатели precision и recall в одно число, предоставляя сбалансированную меру классификатора, особенно в случаях, когда распределение классов является несимметричным. Это гармоническое среднее между точностью и полнотой, что делает эту метрику особенно полезной для оценки общей эффективности классификатора [27].

AUC–ROC (площадь под кривой ошибок) является ценной метрикой для оценки, как хорошо классификатор может различать два класса. Больший AUC означает, что классификатор лучше разделяет положительные и отрицательные классы. Этот показатель используется для оценки производительности независимо от порога классификации и дает представление о качестве прогнозных вероятностей [28].

Confusion Matrix (матрица ошибок) предоставляет полное представление о производительности классификатора, включая количество истинно положительных, истинно отрицательных, ложноположительных и ложноотрицательных результатов. Это позволяет не только оценить ошибки классификации, но и оптимизировать порог принятия решений классификатора, учитывая специфику приложения [29].

Каждая метрика, отображается в числовом формате. Для всех метрик, кроме Confusion Matrix, это значения от 0 до 1, и чем ближе к 1, тем лучше. Confusion Matrix также является неотрицательной, но значения вариативны в зависимости от размера выборки и классов.

2. Данные

2.1 Общее описание художественных произведений

Для исследования было взято 100 текстовых произведений в одном жанре “Отечественная фантастика” с сайта <http://www.lib.ru/> .

Произведение были короткого содержания. От 340 слов до 7400 слов.

Всего было 50 произведений, размеченных как “динамичные”, и столько же как “не динамичные”.

Произведения можно разделить на следующие типы: короткое, длинное, с преобладающим количеством описаний, с преобладающим количеством диалогов, в равной пропорции и описания и диалоги.

При составлении выборки было уделено внимание равномерному распределению различных типов в каждом классе. Однако может имеется небольшое различие в численном составе типов на класс ввиду малой выборки и особенностей классов.

2.2 Выбранные характеристик

В качестве характеристик были выбранные данные значения:

а) 'average\_total\_verbs': среднее общее количество глаголов в абзаце;

б) 'average\_total\_words': среднее общее количество слов в абзаце;

в) 'average\_verbs\_per\_sentence': среднее количество глаголов на предложение;

г) 'average\_words\_per\_sentence': среднее количество слов на предложение (длина в словах);

д) 'average\_sentence\_count': среднее количество предложений в абзаце;

е) 'most\_common\_sentiment': наиболее часто встречающаяся тональность.

Эти характеристики выбраны на основе данных предположений.

Глагол – часть речи, прямо указывающая на действие. Это, вероятно, один из главных наших маркеров динамичности.

В коротких предложениях или абзацах часто проявляется динамичность, подчеркивая быстроту и напряжение событий. Они могут быть эффективным средством передачи динамичных действий или быстро меняющихся сцен. Напротив, длинные предложения и абзацы часто ассоциируются с более подробными описаниями и медленным темпом повествования. Они могут использоваться для более глубокого анализа и развития сюжета, а также для создания более детальных описательных сцен и контекста. Таким образом, выбор между короткими и длинными предложениями/абзацами зависит от желаемого эффекта и целей автора в передаче информации.

Быстрая смена тональности может обозначать и быструю смену событий, в то время как однородная тональность может показать более продолжительную и длинную сцену.

2.3 Алгоритм получения характеристик

На вход программы предобработки и получения характеристик подаётся файл csv формата с именем файла и его классом. Каждый файл проходит один и тот же алгоритм.

Текст каждого файла будет разделён на абзацы (на основе одной и более новых строк), а после и на предложения (на основе токенизации библиотеки nltk). Слова фильтруются на основе nltk.stopwords и приводятся к одному виду.

Эта структура представляет собой простую предобработку для наших данных. Нам изначально важны были знаки препинания, чтобы разделять наши сущности, поэтому процесс предобработки был простым.

Также мы считаем длину предложений, а после абзаца.

Для определения глаголов используется библиотека DeepPavlov и модель ru\_syntagrus\_joint\_parsing. Эта модель определяет часть речи по её начальной форме, поэтому дополнительная лемматизации нам не нужна.

Как итог данных действий – у нас есть характеристики по каждому абзацу.

После этого список полученных характеристик делится на три части для обобщения информации. По каждой части получается среднее значение для каждой характеристики. Для получения средней тональности мы берём самую частую полученную тональность. А также дополнительно среднее значение по всем трём частям произведения. Все эти данные записываются в итоговый файл.

Пример полученных данных приведён в таблице 2. Класс произведения для дальнейшей работы с ИИ кодирует по принципу: динамичный = 0, не динамичный = 1.

Таблица 2 – Пример полученных характеристик из текстового произведения

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| average\_total\_verbs | average\_total\_words | average\_verbs\_per\_sentence | average\_words\_per\_sentence | average\_sentence\_count | most\_common\_sentiment | class |
| 5.13, 2.68, 2.52, 3.44 | 17.27, 8.42, 7.95, 11.19 | 1.4, 1.27, 1.45, 1.37 | 4.66, 3.64, 4.56, 4.29 | 3.45, 2.12, 1.89, 2.48 | 1, 1, 1, 1 | 1 |

2.4 Обоснование выбора средних значений характеристик

Применение средних значений при статистическом анализе текстов имеет ряд важных преимуществ. В частности, средние значения помогают уменьшить влияние выбросов и случайных колебаний, которые могут возникать при детальном рассмотрении отдельных частей текста или же при анализе различных произведений. Это обеспечивает формирование более обобщенной и устойчивой статистической модели, что особенно критично при анализе большого количества данных.

Средние показатели таких метрик как частотность употребления слов, продолжительность предложений или интенсивность использования эмоциональных выражений позволяют получить представление о характерных чертах стиля текста, минимизируя при этом дисперсию, связанную с структурными особенностями отдельных произведений.

Выбор отказа от минимаксной нормализации и альтернативных методов корректировки данных обусловлен особенностями задач и свойствами исследуемых данных. Минимаксная нормализация приводит к сжатию всех показателей к одной шкале (от 0 до 1), что может стирать значимые различия в масштабах между данными. В контексте литературного анализа это может привести к потере уникальных стилистических особенностей, делающих каждое произведение выраженным и запоминающимся.

Усреднение, в отличие от нормализации, позволяет сохранить оригинальные пропорции и вариации между текстами, поддерживая тем самым реалистичность и точность анализа. Сохранение исходных масштабов данных важно для точного отражения стилистической уникальности и эмоциональной глубины произведений, что может быть утрачено при использовании радикальных методов преобразования данных.

Однако, в свою очередь, разбивка на слишком большие блоки и усреднение их по тональности негативно сказалось на выборе такой тактики. В итоге большая часть произведений получила одну и ту же нейтральную тональность.

3. Оценка полученных характеристик

3.1 Общие сведения

Сначала разберём среднее значения по каждой характеристике по всей книге.

В таблице 3 представлены результаты по всей книге, разбитые на класс.

Таблица 3 – Средние показатели характеристик по всей книге

|  |  |  |
| --- | --- | --- |
|  | Динамичный | Не динамичный |
| Среднее общее количество глаголов в абзаце | 4.5398 | 5.6324 |
| Среднее общее количество слов в абзаце | 15.9494 | 20.2474 |
| Среднее количество глаголов на предложение | 1.6740 | 1.8718 |
| Среднее количество слов на предложение | 5.6908 | 6.7354 |
| Среднее количество предложений в абзаце | 2.7942 | 3.1290 |

Из представленных данных видно, что средние значения параметров текста класса “динамичный текст” ниже, чем у текстов класса “не динамичный текст”, даже количество глаголов. Вероятно, это можно описать данным тезисами:

а) Эффективное использование действий: В динамичных текстах, авторы могут более точно выбирать глаголы, чтобы передать действие, делая текст более живым и энергичным. В то время как в более статичных текстах может быть больше описательных частей, не требующих такого активного использования глаголов;

б) Краткость и ясность: Динамичные тексты часто стремятся к ясной и краткой передаче информации. Это может привести к тому, что авторы удаляют излишние глаголы, оставляя только те, которые действительно необходимы для передачи смысла;

в) Синтаксическая структура: Динамичные тексты могут содержать более короткие и простые предложения, что в конечном итоге уменьшает количество глаголов на предложение. Это может быть связано с потребностью в быстрой передаче информации и создании более динамичного ритма чтения.

Таким образом, хотя на первый взгляд может показаться, что динамичные тексты должны содержать больше глаголов, фактически меньшее количество глаголов может быть обусловлено спецификой их стиля и целевой аудиторией. Краткие, прямые и лаконичные стили могут быть более эффективным для удержания внимания и передачи информации в ситуациях, где требуется динамизм и активность.

3.2 Среднее общее количество глаголов в абзаце

Следует разобрать каждую характеристику по отдельности. Начинаем с глаголов.

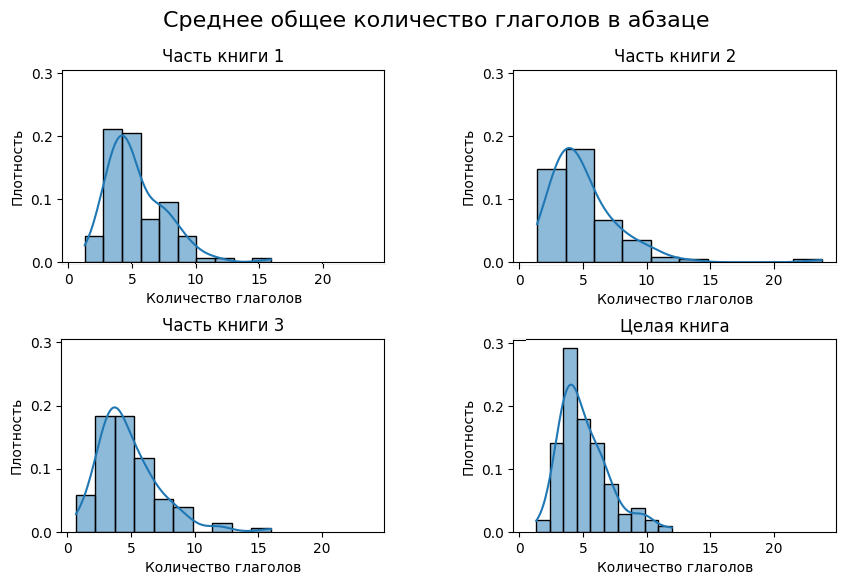


Рисунок 1 – Гистограммы по характеристики глаголов на абзац

Представленные графики на рисунке 1 отражают, как количество использованных глаголов распределено в абзацах трех частей книг, а также в общем для всей книги. Из графиков видно, что среди абзацев с умеренным числом глаголов они встречаются чаще, чем абзацы с очень низким или высоким числом глаголов.

По форме распределений можно сделать вывод, что первая и третья часть книги имеют схожее распределение глаголов, в то время как вторая часть имеет больший разброс значений и более высокую максимальную частоту употребления глаголов (до 40).

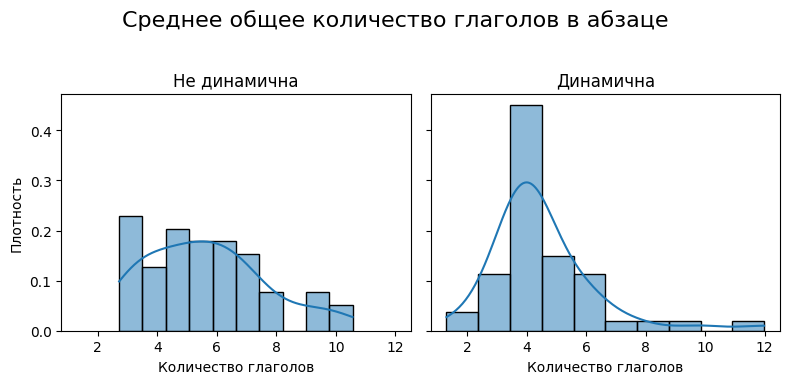


Рисунок 2 – Гистограммы по характеристики глаголов на абзацы с разбивкой на классы

На рисунке 2 выведены графики с информацией по всей книге, но с разбивкой по классам. Можно заметить, что распределение для динамичных книг имеет более выраженный пик около 4–6 глаголов на абзац, что указывает на то, что большинство абзацев в динамичных книгах содержат среднее количество глаголов, способствующее быстрому и активному повествованию.

Для не динамичных книг график распределения количества глаголов более плоский и широкий, с пиком в районе 6–8 глаголов на абзац. Это указывает на более разнообразное использование глаголов в абзацах, что может соответствовать более медленному, размышлительному стилю повествования в не динамичных книгах.

Таким образом, анализ среднего количества глаголов в абзаце может быть полезен для классификации книг на динамичные и не динамичные. Книги с более высокой частотой глаголов склонны быть более динамичными и активными, что видно по более узкому и высокому распределению глаголов, в то время как более широкое распределение в не динамичных книгах отражает более спокойный и разнообразный стиль.

3.3 Среднее общее количество слов в абзаце

На рисунке 3 представлены графики для анализа среднего количества слов в абзаце для каждой трети книги, а также для книги в целом.

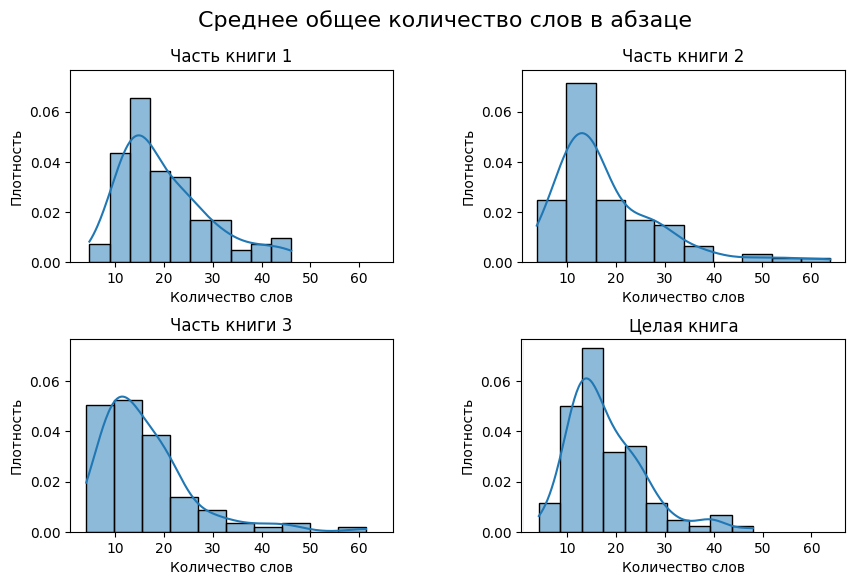


Рисунок 3 – Гистограммы по характеристики количества слов на абзац

Для каждой части книги и для всей книги распределение количества слов приблизительно следует нормальному закону распределения с разными степенями асимметрии.

Графики показывают, что пиковая частота количества слов в абзаце для различных частей и всей книги находится в диапазоне от 10 до 30 слов, с наиболее вероятным количеством около 20 слов. Однако для всей книги пик частоты немного смещен и более высок, указывая на более стабильное использование абзацев средней длины по всему тексту. Разделы могут иметь динамику в использовании более коротких или более длинных абзацев, что приводит к большим колебаниям в их длине.

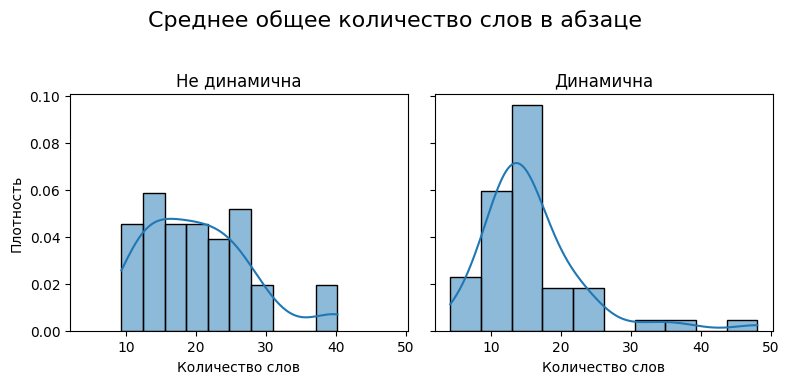


Рисунок 4 – Гистограммы по характеристики слов на абзац с разбивкой на классы

На представленных гистограммах на рисунке 4 видно распределение среднего общего количества слов в абзаце для динамичных и не динамичных книг. На графике для динамичных книг распределение имеет четкий пик, центрированный около 20 слов в абзаце, что указывает на более компактный и краткий стиль написания. Кривая распределения близка к нормальной, указывая на стабильность использования таких размеров абзацев в динамичных текстах.

Для не динамичных книг распределение более широкое и мультимодальное, с несколькими пиками в интервалах около 15, 20 и 25 слов в абзаце. Это может указывать на то, что в не динамичных книгах используется более разнообразный подход к структуре абзацев, в том числе комбинируя краткие и более длинные абзацы, что может способствовать более медленному, размышляющему темпу повествования.

По итогу можно предполагать, что динамичные книги характеризуются более общим и кратким подходом к написанию абзацев, в то время как не динамичные произведения используют более разностороннюю структуру, потенциально способствующую медленному и развернутому повествованию.

3.4 Среднее количество глаголов на предложение

На рисунке 5 представлены графики для анализа среднего количества глаголов на предложение для каждой трети книги, а также для книги в целом.

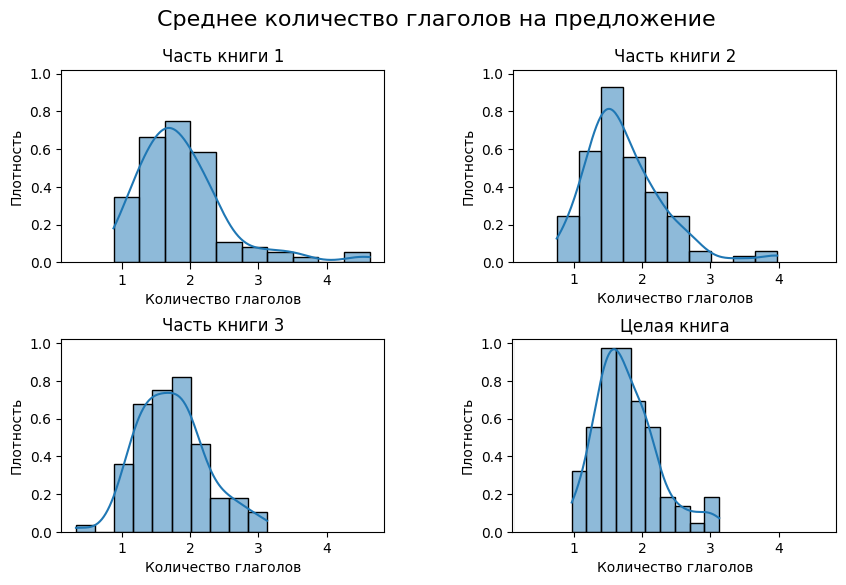


Рисунок 5 – Гистограммы по характеристики глаголов на предложения

Графики распределения глаголов в разных частях книги отражают различия в их структуре. Первая и вторая часть имеют схожее распределение, с высокой частотой значения в окрестностях 1,5, хотя первая часть демонстрирует более высокие значения. Третья часть выделяется симметричным распределением вокруг 2 глаголов на предложение.

График для всей книги показывает более узкий и высокий пик около 2 глаголов, что говорит о более однородном использовании глаголов в тексте. В сравнении с отдельными частями, он менее "шумный", что свидетельствует о меньшей вариабельности и, возможно, о стиле автора или жанровой особенности.

На рисунке 6 представлена информация по целой книге уже с разбивкой на классы.

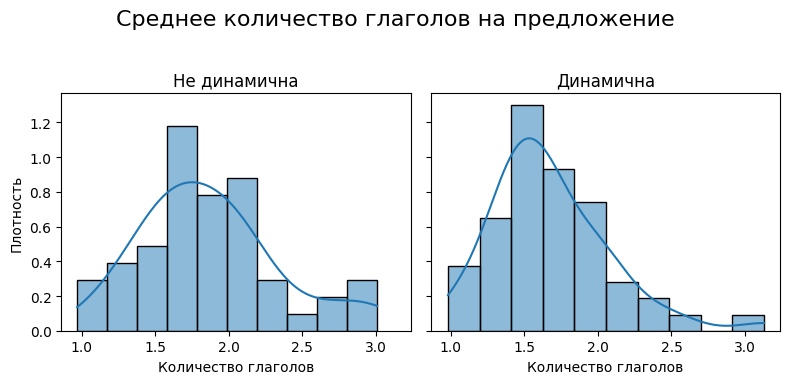


Рисунок 6 – Гистограммы по характеристики глаголов на предложения с разбивкой на классы

График динамичных книг показывает более высокую среднюю частоту глаголов на предложение, где большинство книг сосредоточено в диапазоне от 1,5 до 2,5 глаголов, с пиком около 2 глаголов. Это может указывать на то, что в динамичных книгах предложения наполнены большим количеством действий, что, в свою очередь, поддерживает быстрый ход нарратива.

На графике для не динамичных книг пик значительно ниже, и большая часть данных сконцентрирована около 1–2 глаголов, что указывает на более простое и спокойное повествование. Отмечается также, что распределение проявляет бимодальные черты с менее выраженным вторичным пиком около 2–3 глагола. Это может указывать на присутствие различных стилистических подходов внутри данной категории.

Анализируя данные, можно предположить, что количество глаголов на предложение является индикатором динамичности текста: книги с более высоким средним количеством глаголов склонны к более динамичному, активному стилю повествования, что делает их более захватывающими и напряженными. Это может быть полезным для классификации книг с точки зрения их динамичности, а также для измерения темпа и активности сюжета в литературных произведениях.

Взяв дополнительно данные глаголов по абзацам, можно сделать вывод что: книги класса "Динамичный" характеризуются более высокой частотой использования глаголов как в абзацах, так и в предложениях, что соответствует более активному и быстрому повествованию. Это подтверждается графиками, на которых отчетливо видны узкие пики распределения количества глаголов на абзац и предложение. В отличие от этого, книги класса "Не динамичный" имеют более разнообразное и расплывчатое использование глаголов, что отражает более медленный и размышляющий стиль повествования.

3.5 Среднее количество слов на предложение

На рисунке 7 представлены графики для анализа среднего количества слов на предложение для каждой трети книги, а также для книги в целом.

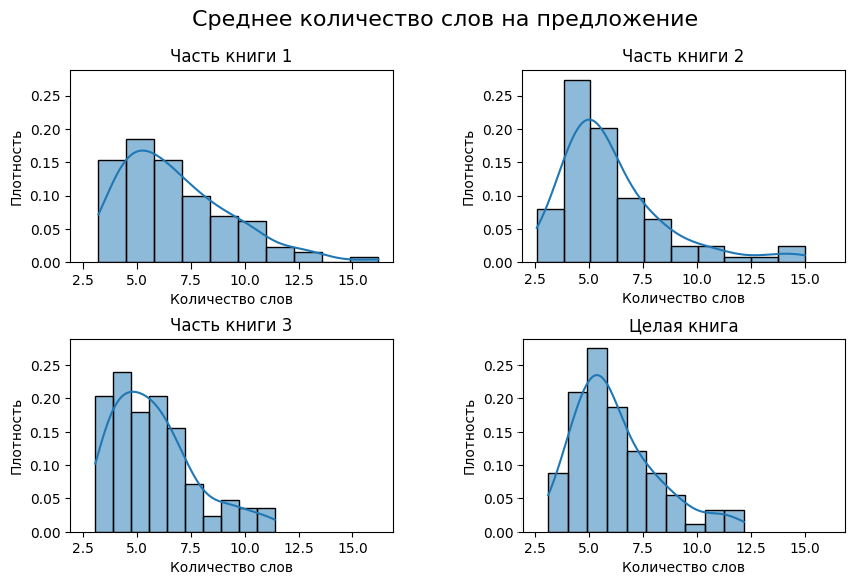


Рисунок 7 – Гистограммы по характеристики слов на предложения

Из графиков видно, что распределение среднего количества слов на предложение в трех частях каждой книги и в целой книге имеет асимметричную форму с пиком и постепенным убыванием. Мода в первой части книги находится в районе 7–8 слов, во второй части – около 5 слов, а в третьей части – около 6 слов на предложение. Для целой книги характерно более низкое среднее количество слов – пик приходится на 6–7 слов. Это предполагает, что внутри отдельных частей книги может быть больше длинных предложений, в то время как общая тенденция книги склоняется к более коротким предложениям.

Анализ целой книги показывает более равномерное и сглаженное распределение по сравнению с отдельными частями. Это может указывать на то, что различия в стилях написания (например, описание против диалогов) более заметны в каких–то конкретных частях, чем на протяжении всей книги. Отсюда можно сделать вывод, что общий стиль книги более устойчив и равномерен, несмотря на изменения в отдельных разделах.

На рисунке 8 приведена характеристика для целой книги, но с разбивкой на классы.

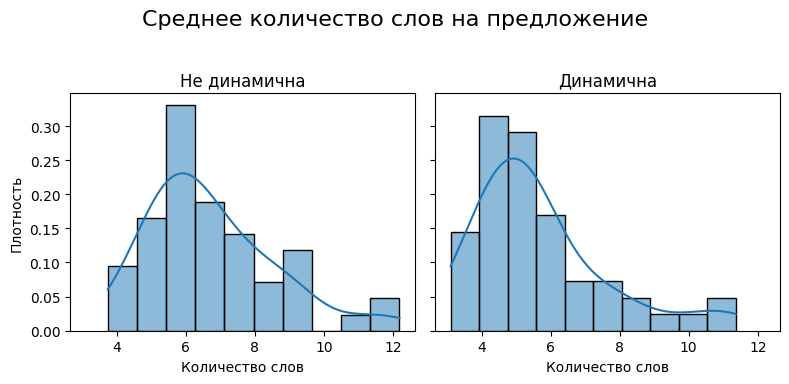


Рисунок 8 – Гистограммы по характеристики слов на предложения с разбивкой на классы

Из графиков можно увидеть чёткие различия в стилях письма. Для динамичных книг распределение показывает пик примерно на 5–6 словах в предложении, что свидетельствует о более лаконичном и сжатом стиле изложения. Такой стиль может способствовать более быстрому темпу чтения и восприятию, что вполне уместно для динамичного жанра.

С другой стороны, книги не динамичного класса имеют пик распределения около 7–8 слов, что указывает на более развёрнутый и подробный стиль изложения. Это может объясняться использованием более сложных предложений и детализированного описания, что зачастую встречается в книгах, требующих медленного и осмысленного чтения.

Анализ двух типов распределений позволяет предположить, что среднее количество слов в предложении может быть полезным параметром для классификации книг на динамичные и не динамичные. Книги с более короткими предложениями склонны быть динамичными и наоборот, что коррелирует с общим темпом и структурой нарратива. Такие выводы могут быть важны при анализе текстовых данных и классификации литературных произведений.

3.6 Среднее количество предложений в абзаце

На рисунке 9 представлены графики для анализа среднего количества предложений в абзацах для каждой трети книги, а также для книги в целом.

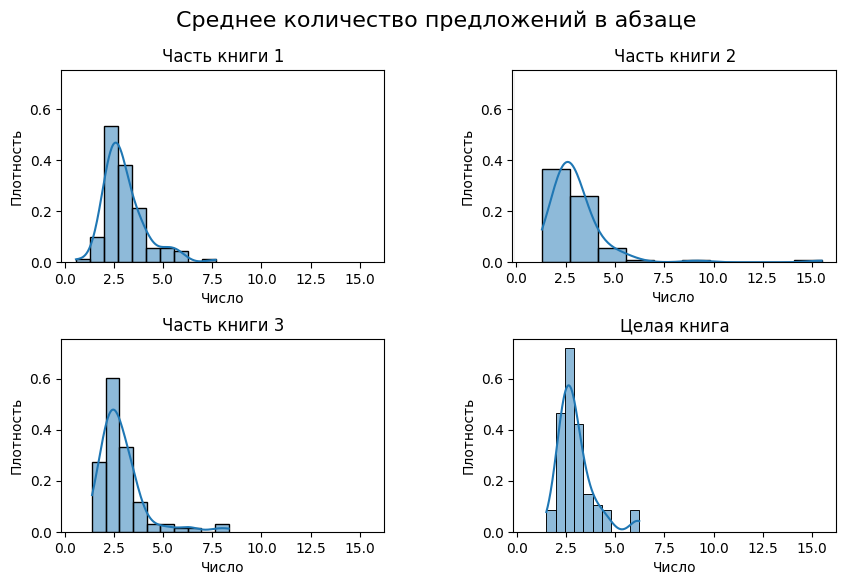


Рисунок 9 – Гистограммы по количеству предложений

Видно, что распределения в первой и третьей части имеют выраженные пики около 2–3 предложений в абзаце, в то время как во второй части график расширяется, и пик сдвигается к 4–5 предложениям, охватывая больший диапазон значений до 15 предложений в абзаце. График для всей книги показывает более сглаженное распределение, также с пиком в районе 2 предложений в абзаце, но с более широким распределением до 6 предложений.

Исходя из этого анализа, можно сказать, что книга представляет собой текст, в котором начальные и заключительные части более сжаты и содержательны по форме изложения, в отличие от середины книги, где текст становится более развернутым и подробным. Это может указывать на динамичный характер повествования книги, где активные и напряженные моменты сосредоточены в начале и в конце, в то время как середина более расслаблена и медлительна, что часто свойственно для развития сюжета и персонажей. Такое строение текста может способствовать захватывающему вовлечению читателя с акцентом на кульминационные моменты в начале и конце.

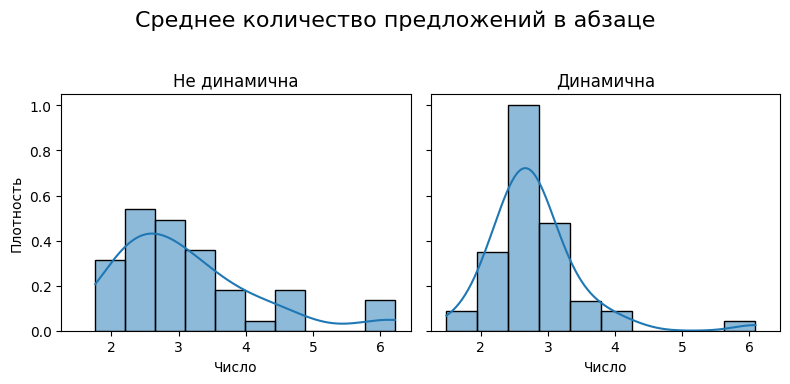


Рисунок 10 – Гистограммы по количеству предложений с разбивкой на классы

Анализируя графики с рисунка 10 можно сделать такие выводы: для динамичных книг показывает пик в районе 3 предложений на абзац и затем быстрое уменьшение частоты с увеличением количества предложений. Это указывает на то, что динамичные книги, вероятно, предпочитают короткие абзацы, что может способствовать быстрому темпу и лёгкости восприятия текста.

График для не динамичных книг более разнообразен с несколькими пиками, наибольший из которых приходится на 4 предложения в абзаце. Распределение более плоское и обширное, с учётом вариативности в количестве предложений в абзаце, что может свидетельствовать о более медленном, размышляющем или описательном стиле написания.

В заключение, можно сказать, что короткие абзацы в динамичных книгах могут способствовать быстрому вовлечению читателя и поддержанию интереса к тексту, в то время как более длинные и разнообразные абзацы в не динамичных книгах предлагают глубокое погружение в материал. Эти наблюдения можно использовать для дальнейшей настройки критериев оценки при классификации книг по динамичности.

3.7 Наиболее часто встречающаяся тональность

Оценку тональности произведения с текущим алгоритмом нельзя назвать удовлетворительной.

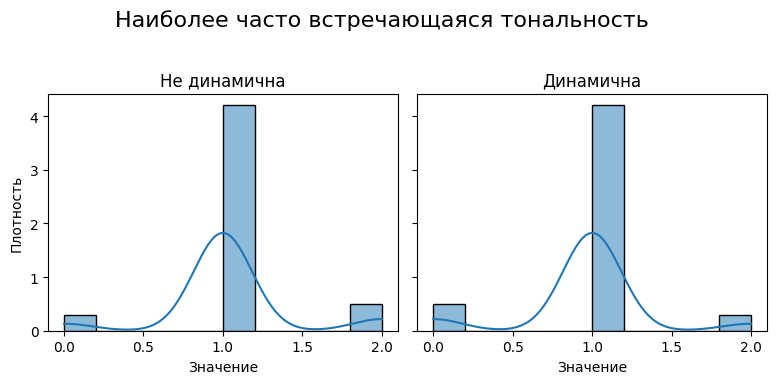
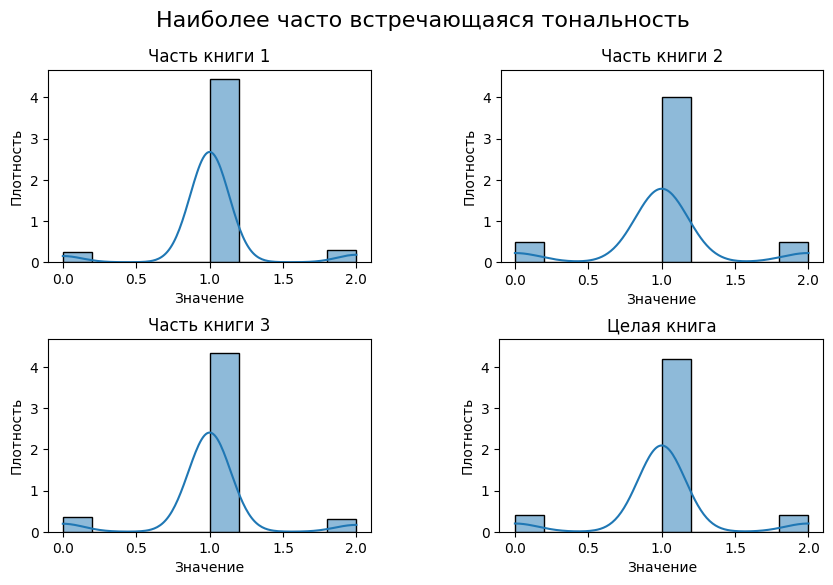


Рисунок 11 – Гистограммы по тональности

Как видно на рисунке 11, в подавляющем числе нейтральная тональность являются сильно преобладающей.

Это может быть связано со слишком большой разбивкой. Небольшие абзацы с нейтральной тональностью, обрамляющие основные, диалоговые системы, в итоге, дадут общую нейтральную тональность. Диалоги, по отдельности, тоже в итоге могу превратится в нейтральную тональность как целое.

Для улучшения работы, для определения тональности можно делать более мелкие разбивки.

Продумать объединение диалогов в один литой абзац и определять их тональность как одну.

3.8 Заключение по характеристикам

В заключение можно сказать, что представленные характеристики помогают определить динамичность или не динамичность текста.

Однако нужно учесть несколько факторов:

а) Дата сет состоял из коротких произведений и одного жанра. Это может повлиять на представление результатов. Для определения целой картины необходима выборка в большое количество данных. Одна краткость выбранных произведений можно вывести в положительную сторону, а именно, крупные произведения можно также делить на короткие, и, по предложению, эффективно анализировать произведения;

б) Количество глаголов не прямо отображает деление на классы. Важно их распределение. Можно предположить, что, объединяя статистики можно получать данные о лаконичности текста, и с помощью этого определять динамику текста;

в) Нужна крупная доработка и отдельный алгоритм для получения тональностей по группе текстов.

После подтверждения, что характеристики могут показывать различия для классов, можно приступать к выбору и обучению моделей.

4. ML классификация

4.1 Выбор моделей

Для классификации текста есть два пути. Первый – использование уже проверенных алгоритмов машинного обучения. Второй путь – построение нейросетей.

В ходе работы были выбраны оба пути. Для алгоритмов были выбраны следующие модели: Logistic Regression, SVM, Random Forest, Gradient Boosting и K–Nearest Neighbors.

Выбрав эти разные модели классификации, которые обладают разными преимуществами, позволяют учесть различные аспекты данных и эффективно решать задачи классификации благодаря своим уникальным подходам. Комбинирование этих моделей дает комплексное понимание данных и помогает выбрать подходящую стратегию классификации для конкретной задачи.

Для нейросетей: LSTM и CNN. Эти нейросети чаще использовались для анализа в теме NLP, однако они подходят и для других задач.

Нашей целью будет также сравнить, какие модели будут эффективнее.

Обучение будет проходить на характеристиках по целой книге, а не по отдельным частям.

4.2 Обучение и анализ моделей

4.2.1 Стандартные модели

В первую очередь рассмотрим классические алгоритмы машинного обучения. Далее подвергнем улучшению выбранный алгоритм и уже после сравним с сетями.

Все пять алгоритмов были использованы на основе библиотеки “sklearn”.

Scikit–learn (sklearn) – это мощная библиотека для машинного обучения на языке Python, широко используемая для построения и оценки моделей. В частности, для задач классификации она предлагает разнообразие алгоритмов и инструментов, которые позволяют эффективно решать проблемы классификации, оценивать качество моделей, и оптимизировать их параметры. Простота в использовании и хорошая документация делают sklearn особенно привлекательной как для начинающих, так и для опытных специалистов в области анализа данных [30].

Обучались и тестировались данные на одной и той же разбивке: 30/70. Без прописки дополнительных гиперпараметров.

В результате обучения были получены такие значения для метрик:

Таблица 4 – Метрики по алгоритмам ML

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Accuracy | Precision | Recall | F1–score | AUC–ROC | Confusion Matrix |
| LogisticRegression | 0.5333 | 0.6154 | 0.4706 | 0.5333 | 0.5792 | [8, 5] [9, 8] |
| SVM | 0.5667 | 0.7000 | 0.4118 | 0.5185 | – | [10, 3] [10, 7] |
| RandomForest | 0.6333 | 0.7143 | 0.5882 | 0.6452 | 0.6991 | [9, 4] [7, 10] |
| GradientBoosting | 0.6333 | 0.7143 | 0.5882 | 0.6452 | 0.7240 | [9, 4] [7, 10] |
| KNeighbors | 0.5667 | 0.6429 | 0.5294 | 0.5806 | 0.6335 | [8, 5] [8, 9] |

Каждая из моделей имеет определённые сильные и слабые стороны. Однако, что модель GradientBoosting, что RandomForest, выделяются среди других. Они демонстрируют лучшие показатели.

Значения точности у обеих моделей идентичны (0.7143), это может говорить о высокой степени правильности классификации положительных случаев по отношению ко всем случаям, признанным положительными. При этом стоит заметить, что ни одна из других моделей не достигает такого уровня точности.

В контексте полноты модели также демонстрируют равные и высшие результаты среди других моделей.

В F1–мере, которая является гармоническим средним точности и полноты, обнаруживает ничем не различимые значения в 0.6452 для обеих моделей. Этот показатель подчеркивает сбалансированность моделей в том, как они обрабатывают положительные и отрицательные классы.

Отдельно следует выделить показатель AUC–ROC, который оценивает, насколько хорошо модель может различать классы. В этой метрике GradientBoosting отличается от RandomForest, получая значение 0.7240 против 0.6991. Хотя разница между показателями не кажется существенной, даже незначительное превосходство в этой метрике может быть решающим фактором при выборе модели для задач с критической зависимостью от способности различать классы на различных порогах вероятностей.

Обе модели показали себя замечательно, однако GradientBoosting выделяется благодаря чуть более высокому значению AUC–ROC. Этот показатель говорит о том, что модель GradientBoosting обладает лучшей способностью к обобщению и различению между классами на различных уровнях вероятности, что делает её предпочтительным выбором для более сложных или чувствительных к ошибкам классификаций. Таким образом, GradientBoosting можно считать наилучшей моделью среди рассматриваемых в данной ситуации.

4.2.2 Улучшение модели GradientBoosting

На этом заключение выбираем GradientBoosting модель как лучшую, и попытаемся улучшить результат.

“Sklearn” позволяет нам корректировать различные параметры и различные функции, вот одни из них:

а) max\_depth: Этот параметр контролирует максимальную глубину каждого дерева в ансамбле. Установка этого параметра может помочь предотвратить переобучение модели.

б) min\_samples\_split: Этот параметр определяет минимальное количество образцов, необходимых для разделения внутреннего узла. Увеличение этого параметра может помочь в борьбе с переобучением за счёт увеличения уровня обобщения модели.

в) min\_samples\_leaf: Этот параметр определяет минимальное количество образцов, необходимых для оставления узла в качестве листа (конечного узла). Увеличение этого параметра может также помочь в контроле за переобучением.

г) loss: Этот параметр определяет функцию потерь, используемую в градиентном бустинге.

д) max\_features: Этот параметр определяет количество признаков, рассматриваемых при каждом разделении. Установив в параметр 'sqrt' обозначили использование квадратного корня из общего количества признаков.

Также нужно отметить, что XGBoost и CatBoost являются развитием концепции GradientBoosting, направленными на улучшение производительности и расширение функциональности исходной модели. Также рассмотрим и их.

XGBoost (Extreme Gradient Boosting) вносит значительные улучшения в алгоритмическую структуру и управление ресурсами. Основой его эффективности служит оптимизация расчетов градиентного бустинга, позволяющая обрабатывать большие объемы данных с высокой скоростью. XGBoost использует модель параллельных вычислений, которая значительно сокращает время тренировки модели по сравнению с традиционным Gradient Boosting. Кроме того, в XGBoost имплементированы улучшенные стратегии обработки пропущенных значений и встроенные методы регуляризации, которые препятствуют переобучению, делая модель более стабильной и надежной при работе с разнообразными наборами данных [31].

CatBoost (Category Boosting) отличается превосходной обработкой категориальных данных, что делает эту модель особенно полезной в задачах, где категориальные признаки играют важную роль. Эта модель автоматически преобразует категориальные переменные в числовой формат, что облегчает подготовку данных и увеличивает точность модели без дополнительных манипуляций с данными со стороны пользователя. Благодаря этому CatBoost показывает высокую производительность в реальных бизнес–задачах. Модель также вводит улучшенные техники бустинга, такие как симметричное древовидное строение для более эффективных расчетов и усовершенствованное предотвращение переобучения, что делает CatBoost надежным выбором даже при ограниченных обучающих данных [32].

По итогу у нас есть шесть моделей:

а) GradientBoosting\_Default – стандартная модель;

б) GradientBoosting\_Tuned: С параметрами: max\_depth=5, min\_samples\_split=10, min\_samples\_leaf=5;

в) GradientBoosting\_Exp: Функция потерь экспоненциальная;

г) GradientBoosting\_Sqrt: max\_features='sqrt';

д) XGBoost\_Default

е) CatBoost\_Default

В результате работы были получены метрики, приведённые в таблиц 5.

Таблица 5 – Метрики по улучшенному алгоритму GradientBoosting

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Accuracy | Precision | Recall | F1–score | AUC–ROC | Confusion Matrix |
| GradientBoosting\_Default | 0.6333 | 0.7143 | 0.5882 | 0.6452 | 0.7240 | [9, 4][7, 10] |
| GradientBoosting\_Tuned | 0.5667 | 0.6667 | 0.4706 | 0.5517 | 0.6606 | [9, 4][9, 8] |
| GradientBoosting\_Exp | 0.6667 | 0.7333 | 0.6471 | 0.6875 | 0.7376 | [9, 4][6, 11] |
| GradientBoosting\_Sqrt | 0.7000 | 0.7857 | 0.6471 | 0.7097 | 0.7873 | [10, 3][6, 11] |
| XGBoost\_Default | 0.6667 | 0.8182 | 0.5294 | 0.6429 | 0.7195 | [11, 2][8, 9] |
| CatBoost\_Default | 0.6667 | 0.7692 | 0.5882 | 0.6667 | 0.7014 | [10,3][7,10] |

Accuracy: видно, что наивысшую точность демонстрируют модели GradientBoosting\_Sqrt и GradientBoosting\_Exp. Высокий показатель точности у этих моделей указывает на их способность эффективно классифицировать как положительные, так и отрицательные примеры.

Precision: стоит выделить XGBoost\_Default и GradientBoosting\_Sqrt. Эти модели эффективнее других определяют действительно положительные случаи, минимизируя ошибки первого рода (ложные срабатывания), что делает их предпочтительными при высокой стоимости ошибочных положительных результатов.

Recall: где необходимо максимализировать идентификацию всех возможных положительных случаев, наилучшие результаты показывают модели GradientBoosting\_Exp и GradientBoosting\_Sqrt. Указанные модели обеспечивают более высокую вероятность того, что положительные случаи не будут упущены.

F1–мера: здесь вновь выделяются GradientBoosting\_Sqrt и GradientBoosting\_Exp. Это указывает на их способность эффективно балансировать между необходимостью минимизации ошибок первого и второго рода, что делает их универсальным выбором в различных условиях задачи.

AUC–ROC: способности модели различать классы независимо от порога классификации, выявила GradientBoosting\_Sqrt и GradientBoosting\_Exp как наилучшие. Высокие значения AUC–ROC подтверждают их эффективность в широком диапазоне условий и пороговых значений.

По итогу можно сказать, что модель GradientBoosting\_Sqrt представляется наиболее предпочтительной. Эта модель демонстрирует сильные результаты по всем ключевым параметрам, что делает её наиболее универсальным инструментом для решения задач классификации в различных условиях. Исследование подтверждает её превосходство в балансировке между различными типами ошибок и обеспечении высокой вероятности правильного классифицирования объектов.

Можно сказать, что контроль над количеством признаков, на основе использование квадратного корня – можно записать в “утверждённый” параметр.

4.2.3 LSTM и CNN

LSTM модель будет строится на основе слоев из tensorflow.keras.layers.

Библиотека tensorflow.keras.layers предоставляет набор инструментов для создания и настройки различных слоев нейронных сетей в TensorFlow. Она позволяет разработчикам легко строить сложные архитектуры нейронных сетей, используя различные типы слоев, такие как полносвязные, сверточные, рекуррентные и другие. Каждый слой представлен в виде объекта, который можно настраивать с помощью различных параметров, таких как количество нейронов, функции активации, методы инициализации весов и многое другое.

Слои модели идут в данном порядке:

model = Sequential([

Bidirectional(LSTM(100, return\_sequences=True), input\_shape=(X\_train.shape[1], X\_train.shape[2])),

Dropout(0.25),

Bidirectional(LSTM(100)),

Dropout(0.25),

Dense(1, activation='sigmoid')

])

Подробнее о каждом слое:

Первый слой Bidirectional: LSTM(100, returnsequences=True): Это слой долгой краткосрочной памяти с 100 скрытыми нейронами. LSTM используется для обучения на последовательных и временных данных, сохраняя информацию в долгосрочной памяти. Параметр `returnsequences=True говорит о том, что слой будет возвращать последовательность скрытых состояний для каждого временного шага, что позволяет слою передавать данные следующему рекуррентному слою, если он есть. Обертка Bidirectional создает два отдельных слоя LSTM, один обрабатывает данные в прямом направлении, а другой – в обратном. Это помогает модели лучше уловить контекст и зависимости как из прошлого в будущее, так и наоборот. Используется часто для улучшения производительности на многих задачах обработки последовательностей [33].

Первый слой Dropout: Слой "выбрасывает" (зануляет) случайным образом 25% входных единиц на каждом шаге во время обучения, что предотвращает переобучение модели. Это регуляризационная техника, которая помогает сделать модель менее чувствительной к конкретным весам отдельных нейронов, тем самым увеличивая её обобщающую способность [33].

Второй слой Bidirectional(LSTM(100)): Второй двунаправленный LSTM–слой. В отличие от первого, он не возвращает последовательности, что значит, что он выводит только последнее скрытое состояние, подведя итог всей входной последовательности. Этот слой суммирует всю информацию, полученную от предыдущего слоя, и готовит её к предсказанию [33].

Второй слой Dropout(0.25): Такой же слой Dropout, как и предыдущий, позволяющий дополнительно предотвратить переобучение после второго LSTM–слоя [33].

Dense: полносвязный слой нейронной сети, который принимает выходные данные от предыдущих слоев и преобразует их в один выходной сигнал. Здесь используется активация "сигмоид", что часто применяется для бинарной классификации (1 или 0).

Порядок слоёв: в конфигурации сначала идут два рекуррентных слоя, чтобы выявить и интегрировать зависимости во входных данных (с учетом обоих направлений времени), затем следуют слои Dropout, которые помогают уменьшить переобучение перед тем, как окончательно делать предсказание через полносвязный слой [33].

Обучение происходило на 40 эпохах с батче равный 50.

Эпохи: количество полных проходов через обучающий набор данных. Большее количество эпох может улучшить производительность сети до определенного предела, после чего может начаться переобучение.

Batchsize: размер пакета данных, используемый на каждом шаге обновления модели. Меньший размер пакета может привести к более быстрой сходимости за счет стохастического градиентного спуска, но может быть более подвержен шумам в данных.

Таблица 6 – Метрики модели LSTM

|  |  |  |
| --- | --- | --- |
|  | Обучение | Валидация |
| Accuracy | 0.6857 | 0.5333 |
| Precision | 0.6897 | 0.6667 |
| Recall | 0.6061 | 0.3529 |
| AUC | 0.7596 | 0.6561 |
| F1–мера | 0.6452 | 0.4615 |

Анализируя метрики модели, представленные в таблице 6, можно сделать следующие выводы.

Точность (accuracy) модели на тренировочных данных составляет 68.57%, тогда как на валидационных данных – всего 53.33%. Это значительное снижение между обучением и валидацией может указывать на переобучение модели, когда она хорошо запоминает тренировочные данные, но плохо справляется с новыми, ранее не виденными данными.

Precision на тренировочной выборке составляет 68.97%, а на валидации 66.67%. Это снижение не слишком значительное, но становится заметнее, когда смотрим на recall – 60.61% на обучении против 35.29% на валидации. Низкий показатель recall на валидации говорит о том, что модель пропускает большое количество реальных позитивных случаев, что нежелательно.

AUC для обучения составляет 75.96%, а для валидации 65.61%. Эти цифры показывают, что модель довольно неплохо разделяет классы на тренировочных данных, однако её способность к обобщению на валидационных данных снижается. Хорошим показателем AUC считается значение ближе к 1, поэтому текущий результат на валидации говорит о среднем качестве классификации.

F1–мера сочетает в себе как precision, так и recall в одном показателе. Она составляет 64.52% на обучении и снижается до 46.15% на валидации. Это снижение говорит о том, что эффективность модели в одинаковой степени рассматривает как точность, так и полноту, и обе ухудшаются на новых данных.

В дополнение к данным из таблицы идут данные по матрице: Confusion Matrix: [10,3] [11, 6]. Видно, что на валидационных данных у модели 10 истинно отрицательных прогнозов (True Negatives, TN), 6 истинно положительных (True Positives, TP), 3 ложноположительных (False Positives, FP) и 11 ложноотрицательных (False Negatives, FN) результатов. Большое количество FN вновь указывает на проблемы с recall.

По итогу, модель показала средний результат. Однако, можно сделать вывод, что основной проблемой стало переобучение.

Архитектура модели позволяет бороться с этим разными способами. Например: модификация количества LSTM–слоёв, размера скрытых единиц или увеличение коэффициента Dropout, использование техник регуляризации, такие как L1 или L2 регуляризации.

Также, не стоит забывать, что, как и GradientBoosting, мы можем варьировать с гиперпараметрами.

Представленная модель LSTM была очень простой, и, на взгляд автора, может справляться с задачей классификаций данных, если тщательнее проработать переобучению модели.

Также, на основе всё той же библиотеки и методах tensorflow.keras.layers была создана сеть классификатор по типу CNN.

Однако времени для её проработки потребовалось значительно больше, чем LSTM, при этом результат был хуже.

Не смотря на более лучший результаты в обучении, по итогу на валидации показывала результаты в два раза хуже. Модель также страдала от переобучения, однако требовала слишком “мягкого” подбора и других параметров, в результате чего, это модель пришлось отложить.

4.3 Заключение по моделям

Модель GradientBoosting показала хорошие результаты. Улучшая её гиперпараметры, можно добиться ещё более высоких результатов. Делать это можно, как вручную, так и в автоматическом виде с помощью Grid Search или Random Search, или использованием алгоритмов оптимизации, что даёт более большой спектр возможностей. Gradient Boosting популярная модель, и наработок в области классификации с её применением достаточно много.

Модель LSTM даже в базовом представление частично, но смогла справится с задачей. Её также не стоит исключать из выбора моделей. Использование этой модели потребует большего времени, однако из–за возможностей более тонкой проработки архитектуры, она даст больший спектр применения модели.

5. Заключение по основной части

Выполняя выпускную квалификационную работу, из книг были получены численные данные. Проанализировав, можно сказать, что все полученные характеристики могут влиять на то, какая книга, динамичная или напротив.

К предоставленным ранее графическим материалам, можно также добавить и вероятностную меру, p–value, помогающую понять, насколько результаты исследований или экспериментов являются статистически значимыми.

Критерий Уэлча – статистический тест предполагает, что обе группы данных взяты из популяций с нормальным распределением, однако не требует, чтобы дисперсии этих популяций были равными. Один из возможных примеров использования критерия Уэлча – проверка нулевой гипотезы о равенстве средних значений двух выборок. В отличие от классического t–теста, критерий Уэлча позволяет корректно учитывать возможные различия в дисперсиях двух выборок, что делает его более гибким и надежным в подобных контекстах [34].

Из–за его гибкости он и был выбран.

Нулевая гипотеза заключается в том, что выбранные признаки и целевая метрика независимы.

В таблице 7 приведены результаты, насколько характеристики связаны с нашей целевой метрикой.

Таблица 7 – p–value по столбцам

|  |  |
| --- | --- |
| Feature | P–Value |
| average\_total\_verbs | 0.004989 |
| average\_total\_words | 0.005368 |
| average\_verbs\_per\_sentence | 0.025463 |
| average\_words\_per\_sentence | 0.006942 |
| average\_sentence\_count | 0.067155 |

Для большинства наших характеристик, p–value меньше стандартного 0,05, а следовательно, можно ещё раз сказать, что по нашим выбранным характеристикам можно оценивать динамичность книг.

У числа предложений в абзаце чуть выше. Однако это не значит, что эта характеристика не является значимой.

Вероятнее, следует объединить проблематику значимости этой характеристики и характеристики тональности в одну.

Также стоит доработать алгоритм разбиения абзацев. Диалоговые системы нужно рассматривать как монолит, а не как отдельные структуры. Тогда мы сможем более чётко определять тональность частей книги, и так же иметь более лучшее представление о длине абзацев.

Для тональности также следует придумать свой отдельный алгоритм усреднения, так как на крупной разбивке не эффективен.

Классификационные модели, задействованные без предварительной настройки и оптимизации параметров, способны демонстрировать значимые результаты даже при обработке ограниченных объемов данных и отсутствии дополнительных предобработок этих данных. Это обнаружение указывает на возможность получения объективных и независимых метрик оценки, к которому и стремились.

Однако следует подчеркнуть, что дальнейшее улучшение алгоритмических процедур и расширение объема используемой базы данных могут значительно повысить точность и эффективность таких моделей. Кардинальное увеличение объема обучающих данных, в параллели с глубокой настройкой и оптимизацией алгоритмов, обеспечит более высокую производительность классификационных систем, что позволит достигнуть улучшенных аналитических результатов.

Таким образом, увеличение объёма данных и их качественная обработка приведут к более точному и надежному прогнозированию, что откроет новые перспективы для принятия обоснованных решений на основе данных.

ЗАКЛЮЧЕНИЕ

В ходе данной выпускной квалификационной работы была поставлена цель, определить возможность оценивания художественных произведений с точки зрения динамичности и энергичности текста, с последующей целью создания модели независимой оценки произведений и предоставления возможности их дальнейшего развития.

В ходе реализации дипломной работы были достигнуты следующие результаты:

– подготовлен и собран набор данных в одном жанре, с разными авторами, с равным распределением по “типам” произведений;

– проведена предварительная обработка данных для получения характеристик и дальнейшего обучения. Потребовалась токенизация, удаление стоп–слов, и нормализация;

– был выполнен анализ данных и главный фактор темы для получения полного понимания характеристик и структуры набора данных;

– были проанализированы данные по характеристикам. Найдены вероятные проблемы в алгоритмах получения некоторых из них, и проиллюстрированы как характеристики распределены;

– были предопределены метрики для сравнения будущих моделей;

– были выбраны различные модели классификации, для сравнения полученных результатов и выделения приоритетных. Выбраны GradientBoosting (с вариативностью гиперпараметров и функций), а также нейросеть LSTM как возможный аналог.

В результате было заключено, что оценивать художественные произведения, путём получения из них численных статистик и анализируя и классифицируя можно получить независимые необходимые оценки.

Дополнительное улучшение алгоритмов и расширение используемой базы данных являются ключевыми факторами для повышения точности и эффективности моделей. Тщательное улучшение и настройка параметров алгоритмов, а также расширение базы данных для обучения, способствуют значительному повышению качества классификации. Эти меры подчеркивают важность инвестиций в сбор и обработку обширных объемов данных и техническое обновление, что, в свою очередь, расширяет возможности для аналитической деятельности и основывает принятие решений на достоверных данных.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. А. С. Маньков, А. А. Детков. Анализа тональности текста // Весенние дни науки: сборник докладов Международной конференции студентов и молодых ученых (Екатеринбург, 20–22 апреля 2023 г.). – Екатеринбург: УрФУ, 2023. – C. 63–65 URL: <https://elar.urfu.ru/handle/10995/125432> (дата обращения: 22.03.2024).

2 Гальченко Ю.В., Нестеров С.А. Классификация текстов по тональности методами машинного обучения // Системный анализ в проектировании и управлении. – 2023. – №3. – С. 369–378. URL: [https://cyberleninka.ru/article/n/klassifikatsiya–tekstov–po–tonalnosti–metodami–mashinnogo–obucheniya](https://cyberleninka.ru/article/n/klassifikatsiya-tekstov-po-tonalnosti-metodami-mashinnogo-obucheniya) (дата обращения: 22.03.2024).

3. Что такое машинное обучение [Электронный ресурс] // Блок Яндекс Практикума. – URL: https://practicum.yandex.ru/blog/chto–takoe–mashinnoe–obuchenie/ (дата обращения: 15.03.2024).

4. Прошина М.В. Современные методы обработки естественного языка: нейронные сети // Экономика строительства. – 2022. – №5. – С. 27–42. URL: [https://cyberleninka.ru/article/n/sovremennye–metody–obrabotki–estestvennogo–yazyka–neyronnye–seti](https://cyberleninka.ru/article/n/sovremennye-metody-obrabotki-estestvennogo-yazyka-neyronnye-seti) (дата обращения: 15.03.2024).

5. Логунова Т.В., Щербакова Л. В., Васюков В.М., Шимкун В.В. Анализ алгоритмов классификации текстов // Universum: технические науки. – 2023. – №2. – С. 4–20. URL: [https://cyberleninka.ru/article/n/analiz–algoritmov–klassifikatsii–tekstov](https://cyberleninka.ru/article/n/analiz-algoritmov-klassifikatsii-tekstov) (дата обращения: 16.03.2024).

6. Попова Е.С., Спицын В.Г., Иванова Ю.А. Использование искусственных нейронных сетей для решения задачи классификации текста // Труды международной конференции по компьютерной графике и зрению "графикон". – 2019. – №29. – С. 270–273. URL: <https://www.graphicon.ru/html/2019/papers/10/Volume1_paper_56.pdf> (дата обращения: 17.03.2024).

7. Введение в стемминг и лемматизацию (НЛП) [Электронный ресурс] // skine.ru. – URL: https://skine.ru/articles/15017/ (дата обращения: 17.03.2024).

8. Усков И.В. Лемматизация русских текстов компьютером // Материалы III Международной научно–технической интернет–конференции молодых ученых. – 2013. – №1. – С. 182–185. URL: https://elibrary.ru/item.asp?id=21219713 (дата обращения: 17.03.2024).

9. 1.2. Машинное обучение [Электронный ресурс] // Яндекс Образование Учебник по машинному обучению. – URL: https://education.yandex.ru/handbook/ml/article/mashinnoye–obucheniye (дата обращения: 17.03.2024).

10. Фролова М. В. Передача экспрессивности при переводе художественной прозы // Вестник Марийского государственного университета. – 2011. – №6. – С. 38–46. URL: https://cyberleninka.ru/article/n/peredacha-ekspressivnosti-pri-perevode-hudozhestvennoy-prozy (дата обращения: 22.03.2024).

11. Как передать динамику в тексте, Дополнение к "Как перевести образы в слова". [Электронный ресурс] // Фантасты.ru. – URL: http://fantasts.ru/forum/index.php?showtopic=4996 (дата обращения: 22.03.2024).

12. Documentation Natural Language Toolkit [Электронный ресурс] // NLTK. – URL: https://www.nltk.org/ (дата обращения: 22.03.2024).

13. DeepPavlov [Электронный ресурс] // DeepPavlov. – URL: https://deeppavlov.ai/ (дата обращения: 22.03.2024).

14. Проект Natasha [Электронный ресурс] // natasha.github.io. – URL: https://natasha.github.io/ (дата обращения: 22.03.2024).

15. Жалыбин А.А., Маликов А.В. Применение нейронных сетей для классификации текстов // Перспективы науки science prospects. – 2020. – №12. – С. 32–36. URL: [https://moofrnk.com/assets/files/journals/science–prospects/135/science–prospect–12(135)–main.pdf#page=32](https://moofrnk.com/assets/files/journals/science-prospects/135/science-prospect-12(135)-main.pdf#page=32) (дата обращения: 22.03.2024).

16. Олисеенко В.Д., Абрамов М.В., Тулупьев А.Л. Нейронные сети LSTM и GRU в приложении к задаче многоклассовой классификации текстовых постов пользователей социальных сетей // Вестник ВГУ. Серия: системный анализ и информационные технологии. – 2021. – №4. – С. 130–141. URL: https://elibrary.ru/item.asp?id=47402249 (дата обращения: 22.03.2024).

17. Самигулин Т.Р., Джурабаев А.Э.У. Анализ тональности текста методами машинного обучения // Научный результат. Информационные технологии. – 2021. – №1. – С. 55–63. URL: <https://www.elibrary.ru/item.asp?id=45765989> (дата обращения: 19.03.2024).

18. Захаренко Д.В. Использование нейронных сетей глубокого обучения для классификации токсичных комментариев в социальных сетях // Информатика. Экономика. Управление. – 2023. – №2. – С. 119–133. URL: https://cyberleninka.ru/article/n/ispolzovanie–neyronnyh–setey–glubokogo–obucheniya–dlya–klassifikatsii–toksichnyh–kommentariev–v–sotsialnyh–setyah (дата обращения: 22.03.2024).

19. Фальк В.Н. Бочаров И.А. Шаграев А.Г. Трансдуктивное обучение логистической регрессии в задаче классификации текстов // Программные продукты и системы. – 2014. – №2. – С. 114–118. URL: https://cyberleninka.ru/article/n/transduktivnoe–obuchenie–logisticheskoy–regressii–v–zadache–klassifikatsii–tekstov–1 (дата обращения: 18.04.2024).

20. Пылов П.А., Протодьяконов А.В., Киренберг А.Г. Support vector machine как мощный инструмент решения прикладной задачи искусственного интеллекта // Инновации. Наука. Образование. – 2021. ­ – №30. – С. 706–711. URL: https://elibrary.ru/item.asp?id=45699590 (дата обращения: 18.04.2024).

21. Летова М.С. Реализация регрессионных и классификационных задач с помощью метода Random Forest // E–Scio. – 2017. – № 8. – С. 15–21. URL: https://cyberleninka.ru/article/n/realizatsiya–regressionnyh–i–klassifikatsionnyh–zadach–s–pomoschyu–metoda–random–forest (дата обращения: 18.04.2024).

22. Сологуб Г.Б., Пухов В.А. Проблемы классификации текстов естественного языка методами классического машинного обучения // Моделирование и анализ данных. 2023. Том 13. № 2. С. 64–76. DOI: 10.17759/mda.2023130203 URL: https://psyjournals.ru/journals/mda/archive/2023\_n2/Sologub\_Pukhov (дата обращения: 18.04.2024).

23. Метод K–Nearest Neighbors. Разбор без использования бибилотек и с использованием бибилиотек [Электронный ресурс] // Хабр. – URL: https://habr.com/ru/articles/680004/ (дата обращения: 18.04.2024).

24. Метрики классификации и регрессии [Электронный ресурс] // Яндекс.Академия. – URL: https://education.yandex.ru/handbook/ml/article/metriki–klassifikacii–i–regressii (дата обращения: 25.04.2024).

25. Accuracy vs. precision vs. recall in machine learning: what's the difference? [Электронный ресурс] // EvidentlyAI. – URL: https://www.evidentlyai.com/classification–metrics/accuracy–precision–recall (дата обращения: 25.04.2024).

26. Recall score [Электронный ресурс] // cloudfactory. – URL: https://wiki.cloudfactory.com/docs/mp–wiki/metrics/recall (дата обращения: 25.04.2024).

27. Что считается «хорошей» оценкой F1? [Электронный ресурс] // Кодкамп: онлайн–институт цифровых навыков. – URL: https://www.codecamp.ru/blog/what–is–a–good–f1–score/ (дата обращения: 25.04.2024).

28. How to explain the ROC curve and ROC AUC score? [Электронный ресурс] // EvidentlyAI. – URL: https://www.evidentlyai.com/classification–metrics/explain–roc–curve (дата обращения: 25.04.2024).

29. Confusion Matrix in Machine Learning [Электронный ресурс] // GeeksforGeeks. – URL: https://www.geeksforgeeks.org/confusion–matrix–machine–learning/ (дата обращения: 25.04.2024).

30. About us [Электронный ресурс] // Scikit–learn Machine Learning in Python. – URL: https://scikit–learn.org/stable/about.html (дата обращения: 01.05.2024).

31. XGBoost [Электронный ресурс] // GeeksforGeeks. – URL: https://www.geeksforgeeks.org/xgboost/ (дата обращения: 01.05.2024).

32. Introduction to CatBoost [Электронный ресурс] // GeeksforGeeks. – URL: https://www.geeksforgeeks.org/introduction–to–catboost/ (дата обращения: 01.05.2024).

33. Module: tf.keras.layers [Электронный ресурс] // TensorFlow. – URL: https://www.tensorflow.org/api\_docs/python/tf/keras/layers (дата обращения: 01.05.2024).

34. Т–критерий Уэлча: когда его использовать + примеры [Электронный ресурс] // Кодкамп: онлайн–институт цифровых навыков. – URL: https://www.codecamp.ru/blog/welchs–t–test/ (дата обращения: 01.05.2024).