МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**
Кафедра математической кибернетики и компьютерных наук

КЛАССИФИКАЦИЯ И ГЕНЕРАЦИЯ ОТЗЫВОВ С ПОМОЩЬЮ НЕЙРОННЫХ СЕТЕЙ

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 451 группы
направления 09.03.04 — Программная инженерия
факультета КНиИТ
Павловой Ксении Сергеевны

Научный руководитель
доцент, к. ф.-м. н. _____ Ю. Н. Кондратова

Заведующий кафедрой
к. ф.-м. н., доцент _____ С. В. Миронов

Саратов 2023

ВВЕДЕНИЕ

В современном мире пользователи все чаще совершают покупки онлайн. Отзывы становятся решающим фактором при принятии решения о покупке. Классификация и генерация отзывов на товары могут стать полезными инструментами компаний для укрепления их позиций на рынке и увеличения объемов продаж. Одним из наиболее перспективных методов для решения данной проблемы является использование нейронных сетей.

Модель нейронной сети для классификации отзывов может значительно экономить время и силы, ускоряя и улучшая анализ текстов и получение полезной информации для улучшения качества продуктов, услуг и обслуживания. Компаниям модель может помочь определить, что служба поддержки работает неэффективно, доступность товаров не соответствует ожиданиям покупателей. Заведениям – понять предпочтения посетителей. Исследователям – анализировать большие объемы текстов на основе определенных параметров.

Модель нейронной сети для генерации отзывов на товары может стать мощным инструментом для создания качественного контента и получения обратной связи. Генерация отзывов может значительно сэкономить ресурсы компании и позволить ей сосредоточиться на других задачах. Модель эффективна для создания персонализированных отзывов на основе предпочтений и поведения клиентов. Это позволит увеличить эффективность маркетинговых стратегий и улучшить взаимодействие с клиентами.

Целью данной работы является разработка моделей для классификации и генерации отзывов с помощью нейронных сетей.

Для достижения цели были поставлены следующие **задачи**:

- изучение теории нейронных сетей и способов их обучения;
- изучение существующих методов классификации и генерации текстов;
- сбор и обработка данных для обучения моделей;
- разработка и обучение модели для классификации отзывов по количеству звёзд;
- разработка и обучение модели для генерации отзывов на основе заданных параметров (название товара, категория);
- тестирование и оценка качества разработанных моделей;
- разработка чат-бота для взаимодействия пользователя с моделями.

Характеристика материалов исследования. Так как для обучения мо-

делей необходим большой корпус текстовых данных на русском языке, данные были собраны методом скрапинга. Были определены источники данных, которые могут быть использованы для обучения нейронной сети: популярные сайты с отзывами на русском языке «Отзовик» и «СпасибоВсем», где миллионы реальных покупателей, владельцев, туристов, сотрудников и пациентов обмениваются полезным опытом друг с другом.

В работе используется архитектура рекуррентной нейронной сети LSTM, которая позволяет модели запоминать длинные зависимости между последовательными элементами, четыре стандартных алгоритма классификации и механизм дообучения нейронной сети. Суть методов заключается в построении предсказания на основе полученного от пользователя текста отзыва.

Структура бакалаврской работы. Работа состоит из введения, двух разделов, заключения, списка использованных источников, содержащего 20 наименований и семи приложений. В первом разделе приводятся определения синтаксического, морфологического, графематического анализа анализа естественного языка, способы TF-IDF, One Hot Encoding, Bag of words, Embedding формирования цифрового представления текста и их сравнение, а также методы решения задач классификации и генерации текста. Во втором разделе подробно описываются используемые программные средства, процесс сбора данных для исследования, построение моделей классификации и генерации отзывов, их обучение и интеграция с чат-ботом на основе API Вконтакте.

1 Основное содержание работы

Задача классификации текста заключается в определении категории, к которой относится данный текст. Для решения задачи необходимо использовать алгоритмы машинного обучения, которые способны обрабатывать большие объемы текстовых данных и выделять в них ключевые признаки, которые помогают определить категорию текста. Среди таких алгоритмов можно назвать наивный байесовский классификатор, решающие деревья, нейронные сети.

Наивный байесовский классификатор – это алгоритм машинного обучения для классификации данных, данный алгоритм не учитывает возможные зависимости между признаками. Он предполагает, что каждый признак является независимым от остальных. Классификатор работает на основе теоремы Байеса, которая позволяет определить вероятность отнесения входного объекта к определенному классу. Наивный байесовский классификатор предполагает, что каждый признак (например, слова в тексте) является независимым от остальных признаков. Таким образом, вероятность того, что объект относится к конкретному классу, рассчитывается как произведение вероятностей отнесения к этому классу всех признаков объекта (всех слов) по формуле:

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)},$$

где x_i – признак объекта, а y – метка класса. После этого, когда классификатор получает новый объект, вычисляются апостериорные вероятности для каждого класса. Апостериорная вероятность $P(y|X)$ высчитывается так: сначала создаётся частотная таблица для каждого параметра относительно искомого результата. Затем из частотных таблиц формируются таблицы правдоподобия, после чего с помощью уравнения Байеса высчитывается апостериорная вероятность для каждого класса. Класс с наибольшей апостериорной вероятностью и будет прогнозируемым результатом.

Логистическая регрессия – это метод машинного обучения, который используется для классификации данных: отнесения объекта к одному из классов. Основным применением логистической регрессии является оценка вероятности принадлежности объекта к определенному классу. В логистической регрессии линейная зависимость задается через логистическую функцию, называемую сиг-

моидой:

$$f(x) = \frac{1}{1 + e^{-x}},$$

где x – это входное значение функции. Значение функции $f(x)$ лежит в интервале от 0 до 1. Если $f(x) \approx 1$, то это значит, что вероятность принадлежности к классу очень высока. Если же $f(x) \approx 0$, то вероятность очень низкая.

Случайный лес – это алгоритм машинного обучения, используемый для решения задач классификации и регрессии. Случайный лес представляет собой ансамбль решающих деревьев, созданных на основе случайной выборки образцов из набора данных и случайных наборов параметров при разделении узлов. Каждое дерево предсказывает выходное значение, и окончательный результат получается путем агрегирования предсказаний всех деревьев. В случае задачи классификации случайный лес делает предсказание на основе голосования по большинству. То есть каждое дерево голосует за определенный класс, и окончательный результат определяется классом, набравшим наибольшее количество голосов. Формула итогового классификатора:

$$a(x) = \frac{1}{n} \sum_{i=1}^{N} b_i(x),$$

где N – количество деревьев, b – решающее дерево, x – сгенерированная на основе данных выборка.

Градиентный бустинг – это алгоритм машинного обучения, который используется для решения задач классификации и регрессии. Основная идея градиентного бустинга заключается в том, что на каждой итерации алгоритм добавляет в ансамбль слабую модель, которая пытается исправить ошибки, допущенные на предыдущей итерации. Градиентный бустинг использует градиентный спуск для минимизации ошибки на каждой итерации. Формула функции ошибки для многоклассовой классификации:

$$Loss = -\sum_{i=1}^{N} y_i * log(\hat{y}_i),$$

где N – количество классов, y – вектор правильных ответов, $\hat{y}$ – вектор предсказаний модели. Значение функции ошибки будет высоким, если предсказанный вектор $\hat{y}$ существенно отличается от вектора правильных ответов y, и низким,

если они близки.

Для решения задачи классификации были рассмотрены классификаторы из библиотеки sklearn. Сравнительный анализ решений представлен в таблице 1.

Таблица 1 – Значение accuracy на классификаторах из sklearn

Классификатор	**Значение accuracy на тестовых данных**
sklearn.naive_bayes.GaussianNB	44 %
sklearn.linear_model.LogisticRegression	62%
sklearn.ensemble.RandomForestClassifier	68%
GradientBoostingClassifier	61%

Данные классификаторы не способны учитывать контекст, эмоциональный окрас и другие нюансы текста, поэтому точность предсказаний остаётся низкой.

Для классификации текстов используется архитектура LSTM рекуррентной нейронной сети. LSTM способна запоминать долгосрочные зависимости между словами, что позволяет ей лучше обрабатывать тексты с большим количеством слов и повышать точность классификации. Благодаря рекуррентным связям, LSTM может обрабатывать последовательности и сохранять информацию о предыдущих состояниях.

Создание модели нейронной сети для классификации отзывов по количеству звёзд состоит из:

1. обработки данных для повышения качества предсказаний с помощью методов обработки естественного языка (NLP);
2. определения параметров нейронной сети, создания ее структуры, выбора функции активации, гиперпараметров, алгоритма оптимизации и функции потерь;
3. обучения модели на размеченных данных;
4. оценки качества модели путём тестирования на новых данных.

Сбор данных был произведён следующим образом: Определены источники данных, которые могут быть использованы для обучения нейронной сети. Были выбраны популярные сайты с отзывами на русском языке «Отзовик» и «СпасибоВсем», где миллионы реальных покупателей, владельцев, туристов, сотрудников и пациентов обмениваются полезным опытом друг с другом. Затем были написаны скрипты для скрапинга данных. Для этого были использованы

библиотеки Python (Beautiful Soup, requests и csv). Всего было собрано 81263 отзыва. Все данные сохранены в файле scrap_data.csv.

Чтобы избежать предвзятости и неправильных решений из-за дисбаланса классов, в задачах классификации применяется балансировка классов для уравновешивания количества представителей каждого класса. Для достижения балансировки классов была применена генерация синтетических данных и взвешивание учебных данных для учета разных размеров классов. Генерация данных производилась для отзывов класса 0 и 1, из классов 4 и 5 отзывы с наименьшей длиной были удалены.

Данные были разделены на обучающую и тестовую выборки. Целевой столбец с количеством звёзд на отзыве был закодирован в формате one hot encoding (при данном подходе каждый объект может быть представлен как вектор, где все значения будут равны нулю, а в единицу будет установлен только тот признак, который соответствует исходному значению). Так как нейронные сети работают с числами, необходимо преобразовать текст в последовательности чисел.

Токенизация текста – это процесс разделения текста на отдельные единицы, называемые токенами. Разбитый на токены текст позволяет нейронной сети лучше понять смысл текста и выделить его ключевые аспекты. Средняя длина отзывов в полученном файле 142. Общее количество различных слов во всех отзывах – 99063. Был создан токенизатор для 9906 самых популярных слов. Длина каждого отзыва обрезана до 150 слов, при недостатке слов в конце отзыва дописаны нули с помощью параметра padding.

В результате слой Long Short Term Memory (LSTM) имеет параметры: количество скрытых нейронов – 121, доля отбрасываемых для линейного преобразования повторяющегося состояния единиц – 0.2.

На выходе сети вектор из 5 чисел от 0 до 1, каждое из них – вероятность принадлежности отзыва к классу, функция активации – softmax.

В качестве оптимизатора был выбран алгоритм adam. Функция потерь – categorical_crossentropy, так как предстоит решать задачу многоклассовой классификации (более 2 классов). Метрика, которую необходимо оценить и взвесить во время обучения и тестирования – accuracy. Итоговое количество параметров для обучения: 392138.

Модель обучалась в 20 эпох, размер мини-выборки – 128, 10% данных

отдаётся на оценку в конце эпохи обучения. График значений метрики accuracy на каждой эпохе обучения представлен на рисунке 1.

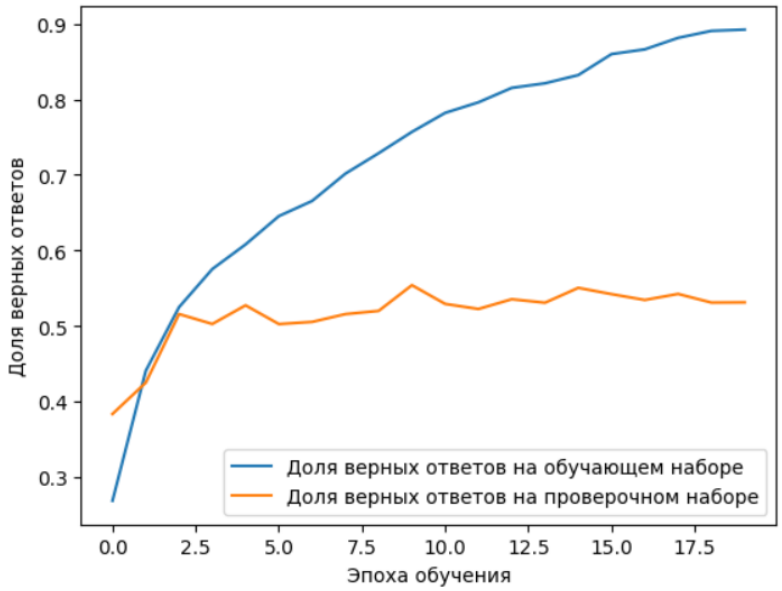

Рисунок 1 – Значения accuracy во время обучения

На тестовой выборке значение метрики accuracy составило 89.22%, на валидационной – 53.11%. Низкое значение точности на валидационной выборке обусловлено тем, что модель часто путает оценки 2 и 3, 3 и 4, 4 и 5. Значение метрики accuracy на тестовых данных составило 79.57%.

В качестве решения задачи генерации отзывов был выбран метод Transfer Learning. Transfer Learning – это метод обучения нейронных сетей, в котором модель, обученная на одной задаче, применяется для решения другой задачи.

Создание модели нейронной сети для генерации отзывов состоит из:

1. обработки данных и приведения их к единому формату;
2. выбора предобученной языковой модели, загрузки модели и сохранения ее в формате, совместимом с библиотекой для дообучения;
3. определения параметров нейронной сети;
4. обучения модели на размеченных данных;
5. оценки качества модели путём тестирования на новых данных.

В качестве языковой модели была выбрана ruGPT-3 Small – это языковая модель глубокого обучения, созданная Сбербанком, которая способна генерировать тексты на русском языке в естественной форме.

Подготовка данных к обучению модели ruGPT-3 Small заключается в приведении всех отзывов к единому формату, который будет использоваться при

дальнейшей генерации. Модели на вход будет подаваться желаемая оценка продукта, его название и категория. Начало и конец текста в ruGPT-3 Small обозначается тэгом <s>. Для обучения нейронной сети была использована платформа параллельной обработки от NVIDIA – CUDA, которая позволяет использовать графические процессоры (GPU) для ускорения вычислений на компьютере.

Модель обучалась 90 минут, в директории сохранились файлы конфигурации (config.json), весов модели (pytorch_model.bin, training_args.bin), а так же словарь токенизатора (vocab.json).

Модель генерирует связный текст для продуктов распространённых категорий (сайты, игры, продукты питания, техника, товары для детей). Для продуктов из редко встречающихся категорий (дом, дача) модель генерирует связный, но часто бессмысленный текст, вставляя в него фрагменты из художественной литературы или фильмов (так как изначальных корпус текстов для обучения содержал большое число книг, описаний фильмов и биографий людей).

В данной работе чат-бот выполняет функцию интерфейса для взаимодействия пользователя с разработанными моделями классификации и генерации отзывов. Для взаимодействия клиента (пользователя Вконтакте) и сервера (код обработки сообщений, модель классификации и модель генерации) был выбран фреймворк Django. Фреймворк подчеркивает возможность повторного использования, меньший объем кода, низкую связь, быструю разработку. Django также предоставляет дополнительный административный интерфейс создания, чтения, обновления и удаления, который генерируется динамически с помощью самоанализа и настраивается с помощью моделей администратора.

Сервер django по умолчанию запускается на адресе 127.0.0.1:8000. Для предоставления Вконтакте доступа к этому локальному IP-адресу используется утилита ngrok. Необходимо создать сообщество Вконтакте и перейти в настройки, работа с API. Для подтверждения существования сервера сайт отправляет тестовый POST-запрос.

Для работы с моделями создана клавиатура с двумя кнопками (классифицировать и сгенерировать), которая отправляется пользователю от бота вместе с сообщением по умолчанию. Далее, в зависимости от действий пользователя, система ожидает текста отзыва для классификации или краткое описание продукта для генерации отзыва. Затем, получив ответ от нейронной сети – отправляет пользователю результат классификации или генерации.

Текст отзыва для классификации необходимо обработать теми же средствами, которыми обрабатывались данные перед обучением нейронной сети. Одна из функций подгружает уже обученную нейронную сеть классификации, а так же токенизатор текста, производит предварительную обработку данных и в качестве результата вычисляет возможное количество звёзд на отзыве от 1 до 5.

Для генерации отзыва модели необходима максимально подробная «затравка» в формате, на котором обучалась модель. Для этого чат-бот узнаёт у пользователя желаемое количество звёзд для оценки продукта от 1 до 5, название и категорию.

ЗАКЛЮЧЕНИЕ

По итогам проведённой работы были разработаны модели классификации и генерации отзывов с использованием нейронных сетей, создан чат-бот на основе API Вконтакте а именно:

- изучена теория нейронных сетей и способов их обучения;
- изучены существующие методы классификации и генерации текстов;
- собраны и обработаны 81263 отзыва для обучения моделей;
- разработана и обучена модель классификации отзывов по количеству звёзд со значением точности на тестовой выборке 79%;
- разработана и обучена модель генерации отзывов на основе заданных параметров (название товара, категория);
- проведено тестирование и оценка качества разработанных моделей;
- разработан чат-бот для взаимодействия пользователя с моделями.

Основными инструментами в данной работе стали: язык программирования Python, бесплатный облачный сервис Google Colab и Google Drive, библиотека transformers для работы с моделью генерации текста ruGPT-3 Small, веб-фреймворк Django, утилита Ngrok, среда разработки PyCharm, библиотеки numpy, torch и transformers.

Разработанные методы способны обеспечить высокое качество классификации и генерации отзывов. Результаты работы могут быть использованы в различных сферах, связанных с анализом текстовой информации, таких как маркетинг, реклама, социальные сети и многие другие.