

МЕТОДЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

УДК 004.89

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ РАЗЛИЧНЫХ
АРХИТЕКТУР РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ
ДЛЯ РЕШЕНИЯ ЗАДАЧИ АНАЛИЗА ТОНАЛЬНОСТИ
РУССКОЯЗЫЧНЫХ КОММЕНТАРИЕВ ПОЛЬЗОВАТЕЛЕЙ
СОЦИАЛЬНЫХ СЕТЕЙ© А. Н. Жданова¹, А. В. Куприянов^{1,2}, А. А. Голова¹,
А. С. Булгаков¹, Д. С. Баканов¹¹Самарский национальный исследовательский университет им. академика С. П. Королева,
443086, г. Самара, Московское шоссе, 34²ИСОИ РАН — филиал ФНИЦ «Кристаллография и фотоника» РАН,
443001, г. Самара, ул. Молодогвардейская, 151
E-mail: danilenko.an@ssau.ru

Продemonстрировано применение методов машинного обучения для анализа тональности текстов и исследования эффективности различных архитектур нейронных сетей. Показано, что данное направление является актуальным в связи с развитием социальных сетей и онлайн-рекомендательных сервисов, где множество пользователей высказывает своё мнение о товарах и услугах. Представлены результаты прогноза и сравнения структур нейронных сетей на реальных данных из социальных сетей, что позволяет определить наиболее эффективную архитектуру для анализа тональности текстов. Исследование может быть полезным разработчикам социальных сетей для рекомендательных сервисов, а также исследователям, занимающимся обработкой естественного языка. Результаты могут помочь улучшить качество анализа мнений пользователей и повысить удовлетворённость пользователей услугами и продуктами. Таким образом, данное исследование вносит вклад в развитие области машинного обучения и анализа текстовых данных.

Ключевые слова: сентиментальный анализ, рекуррентные нейронные сети, анализ данных, тональность текста, прогнозирование.

DOI: 10.15372/AUT20230404

Введение. Анализ тональности текстов, т. е. определение мнения автора по отношению к объекту, в последнее десятилетие является распространённой задачей автоматической обработки естественного языка. Актуальность этого направления во многом связана с развитием социальных сетей, онлайн-рекомендательных сервисов, содержащих большое количество мнений пользователей по разным вопросам, в частности о товарах и услугах.

Практическая потребность в эффективном анализе большого количества текстов первоначально появилась со стороны онлайн-бизнеса в целях улучшения качества обслуживания клиентов. Высокий уровень конкуренции среди онлайн-ритейлеров, продающих однотипную продукцию, мотивирует компании искать дополнительные способы повышения лояльности покупателей. Менее активно аналогичными инструментами пользуются некоторые игроки ресторанного и банковского бизнеса. В названных предметных областях эффективность применения автоматического анализа мнений пользователей достигла наибольшего результата. Авторы прикладных исследований, которые предлагают новые методы анализа тональности текстов, в основном пользуются данными из этих предметных областей.

Растущий спрос на подобные системы подтолкнул к созданию стартапов и подразделений внутри крупных компаний, разрабатывающих свои технологии в данной области: Microsoft, Google, Hewlett-Packard, SAS, Сбербанк и т. д. Интерес к анализу тональности мнений также существует со стороны научных организаций и университетов, где часто делается акцент на анализе социальных взаимоотношений внутри общества.

Проблемы анализа эмоциональной окраски текста.

Сентиментальный анализ или анализ тональности текста — это комплекс методов обработки естественного языка, направленный на определение эмоциональной окраски текста по отношению к описанным объектам.

Анализ тональности ставит перед собой цель нахождения мнений в заданном тексте и определение их свойств. В зависимости от поставленной задачи можно рассматривать разные свойства. Например, может быть интересно, кто является автором комментария, о чём говорится в комментарии, тональность комментария, т. е. позитивное или негативное мнение автора.

На данный момент известно четыре подхода к изучению анализа тональности:

- 1) подход на правилах,
- 2) подход со словарём,
- 3) машинное обучение,
- 4) обучение без учителя.

Первый подход базируется на определённом наборе правил, применив которые система может сделать заключение о тональности текста. Этот подход является наиболее точным, используется в коммерческих системах, однако требует затрат. Например, к предложению «Я обожаю фильмы» можно применить следующее правило: если сказуемое (обожаю) входит в положительный набор глаголов (нравится, люблю, восторгаюсь, ...) и в предложении не имеется отрицаний, то необходимо классифицировать тональность как положительную. Данный подход затратен, так как для хорошей работы системы требуется прописать множество правил. Обычно правила привязаны к определённой тематике, и при смене тематики требуется также снова составлять и правила.

Подход, основанный на словарях, прост в применении, однако не универсален. При данном методе используют специальные тональные словари, которые представляют собой списки слов со значениями тональности (к каждому слову). Алгоритм данного метода заключается в том, что сначала каждому слову присваивается его значение тональности из словаря, затем эти значения суммируются и получается общая тональность текста.

Следующий подход — машинное обучение — позволяет автоматизировать процесс вычисления тональности текста, однако требует данные для обучения. Этот метод является наиболее распространённым и часто используется в исследованиях. Суть заключается в том, что нейросеть необходимо сначала обучить на заранее заготовленных размеченных текстах, а затем полученную модель можно использовать для анализа тональности новых текстов.

Последний подход так же, как и предыдущий, является автоматизированным, но не требует данные для обучения. Однако его точность считается самой низкой. Автоматическая кластеризация документов может послужить примером использования такого метода.

В работе используется машинное обучение с учителем. Такой процесс похож на процесс создания других систем, в которых применяется машинное обучение. Для начала нам необходимо будет собрать коллекцию документов для обучения, затем каждый документ должен быть представлен в виде вектора признаков, после чего для каждого документа требуется указать нужный тип тональности (позитивный или негативный) и по «правильным ответам», выбрав алгоритм, обучить классификатор. После того как все шаги будут пройдены, можно будет приступать к тестированию системы.

Определение тональности текста обычно сводится к задаче классификации — отнесение объекта к одному из заданных классов. Данный вид нейросети был выбран потому, что именно рекуррентные нейронные сети, или RNN, были разработаны для решения задач обработки последовательности [1].

Основные типы задач обработки последовательностей включают в себя:

1. Один-ко-многим: наблюдение в качестве входных данных, сопоставленных с последовательностью с несколькими шагами в качестве выходных данных.
2. Многие-к-одному: последовательность из нескольких шагов в качестве входных данных, сопоставленных с прогнозом класса или количества.
3. Многие-ко-многим: последовательность из нескольких шагов в качестве входных данных, сопоставленных с последовательностью из нескольких шагов в качестве выходных данных. Часто упоминается как последовательность к последовательности, или seq2seq

Модели и методы.

Под рекуррентными нейронными сетями обычно понимают семейство похожих архитектур: простые рекуррентные нейронные сети (RNN), сети с долгой и краткосрочной памятью (Long Short-Term Memory — LSTM), сети с управляемым рекуррентным блоком (Gated Recurrent Unit — GRU). Основная общая черта перечисленных методов — работа с последовательностями и решение таких типовых задач, как генерация описания изображений, последовательностей ответов на вопросы (чат-боты), разметка частей речи в предложении и т. д.

Сеть RNN в целом и LSTM в частности достигли наибольшего успеха при работе с последовательностями слов и абзацев, обычно называемыми обработкой на естественном языке.

Это включает в себя как последовательности текста, так и последовательности разговорного языка, представленные в виде временных рядов. Они также используются в качестве генеративных моделей, которые требуют вывода последовательности не только с текстом, но и в таких приложениях, как создание почерка.

Сеть RNN следует использовать для текстовых данных, речевых данных, генеративных моделей, классификации прогнозных задач [2]. В данном случае будет рассматриваться бинарная классификация, т. е. количество классов равно 2 (положительная и негативная окраска).

Пусть $X = (x_1, x_2, \dots, x_n)$ — это множество заданных предложений, а $Y = (y_1, y_2) = (0, 1)$ — это классы, обозначающие окраску текста. Тогда задача состоит в построении отображения (1): $X \rightarrow Y$.

Для задачи поиска аспектов применяется распространённая кодировка IOB2, представляющая собой вектор one-hot из трёх элементов. Начало термина обозначается буквой B, продолжение — I, во всех остальных случаях — O. На каждом шаге модель получает новое слово-вектор, комбинирует его с прошлым скрытым состоянием и делает прогноз для текущего слова — является ли оно аспектом. Для задачи классификации нейронная сеть предсказывает значение класса только на последнем шаге после «прочтения» всей последовательности.

Существует несколько известных вариаций RNN, с которыми далее будут проведены эксперименты. Самая первая и популярная конфигурация — Elman-type RNN, в которой текущее скрытое состояние ht является нелинейной трансформацией прошлого состояния и текущего наблюдаемого значения xt : $ht = f(Uht - Y + Vxt + b)$, где f — функция активации, b — вектор сдвига, U и V — матрицы весов.

Скрытое состояние ht можно интерпретировать как накопленные знания о прошлом, которые могут накапливаться и меняться во времени.

Немного другая архитектура — Jordan-type RNN. Отличие состоит в том, что скрытое текущее состояние зависит от предсказанного значения на прошлом шаге $yt - Y$: $ht = f(Uyt - Y + Vxt + b)$.

В работе с текстовыми последовательностями иногда важно учитывать не только прямой, но и обратный порядок слов, так как знание о будущем может содержать ценную информацию. Для этого на практике применяется архитектура RNN с обратными связями (bidirectional RNN). Матрицы весов U и V , а также сдвиг b и скрытые состояния h независимы для слоёв с прямыми и обратными связями. Предсказания создаются путём комбинации двух скрытых состояний: $yt = g(Wht + Wht + c)$.

Использование RNN на практике связано с известной проблемой затухания градиента (vanishing gradient problem) при обучении нейронной сети методом обратного распространения ошибки. Проблема возникает из-за многочисленных звеньев в поиске градиента сложной функции. Одно из неприятных последствий — неспособность RNN хранить информацию о длинных последовательностях. Для разрешения ситуации исследователями была предложена архитектура LSTM и более современная упрощённая альтернатива — GRU.

Основное отличие LSTM и GRU от RNN заключается в изменённом методе расчёта активаций скрытого слоя. Скрытое состояние LSTM на текущем шаге ht обычно задаётся набором из следующих уравнений [3]:

$$\begin{aligned}ft &= \sigma d(Wx dxt + W) dht - Y + Wcd(ct - Y + bd), \\ct &= ftct - Y + it \tanh(Wxcxt + W)(cht - Y + bc), \\ot &= \sigma o(Wx oxt + W) oht - Y + W(co ct + bo), \\ht &= ot \tanh(ct),\end{aligned}$$

где σ — сигмоидная активация, \tanh — гиперболический тангенс, W — матрица весов, b — вектор сдвига, i — вектор input gate, f — вектор forget gate, c — вектор cell, o — вектор output gate.

Модель GRU является упрощением LSTM и обладает меньшим числом параметров, что положительно сказывается на скорости обучения при сопоставимом качестве результатов. Активация скрытого слоя рассчитывается по формулам [4]:

$$\begin{aligned}rt &= \sigma r(Wxrxt + W)rht - Y + br, \\ut &= \sigma u(Wxuxt + W)uht - Y + bu, \\ct &= \sigma c(Wxcxt + rt \odot W)cht - Y + bc, \\ht &= (1 - ut) \odot ht - Y + ut \odot ct,\end{aligned}$$

где \odot означает поэлементное умножение; r , u , c — векторы reset gate, update gate, cell memoгу соответственно.

Анализ данных. Для обучения нейронной сети необходимо: провести предобработку данных (удаление малозначимых слов, приведение к общему регистру, удаление стоп-слов), разбить текст на токены, сформировать последовательности результатов токенизации, обучить модели.

Современные компьютеры далеки от понимания человеческих слов в привычном для нас смысле. Человеческий мозг способен мыслить словами, излагать их на бумаге и даже переводить с одного языка на другой. Электронно-вычислительная техника общается между собой с помощью чисел. Поэтому прежде чем компьютер сможет оценить наше

текстовое творение, его нужно перевести в числовое представление. Для этого необходимо разбить текст на более мелкие части (токены), чтобы в дальнейшем присвоить каждому токену индивидуальный код. Именно этот процесс — разбиение текста на более мелкие части — называется токенизацией [5].

Для выполнения задачи токенизации был использован метод `fit_on_texts` из класса `Tokenizer`. Данный метод позволяет обойти весь текст и обновить словарь токенов в соответствии с частотой вхождения слов так, чтобы наиболее популярные слова в тексте получили наименьшие индексы. Например, в предложении: «А за окном то снег, то дождь» у частицы «то» будет нулевой индекс, т. е. `word_index['to'] = 0`.

После токенизации текста, проведённой на предыдущем этапе, требуется сформировать из полученных результатов последовательности. Эта задача выполняется с помощью метода `texts_to_sequences`, после чего этот метод возвращает как результат числовую последовательность частотных характеристик слов, что также помогает сохранить семантическую связь между словами текстов.

Так как в данной задаче решается бинарная классификация, то в качестве метрики качества была использована ROC-кривая.

ROC-кривая — график чувствительности против специфичности.

Специфичность — процент (или доля) правильно классифицированных нулей.

Чувствительность — процент (или доля) правильно классифицированных единиц.

В качестве числового значения используется ROC-AUC — площадь под графиком ROC-кривой.

Обработка данных. Для начала была построена рекуррентная нейронная сеть с простым рекуррентным слоем (SimpleRNN), архитектура которой показана в табл. 1.

Рассмотрим архитектуру данной сети подробнее:

`Embedding` — слой, преобразующий векторы индексов в плотные векторы вложений слов.

`Simple_rnn` — слой простой рекуррентной связи.

`Dense` — плотный слой прямого распространения.

Параметры:

Общее число параметров: 12, 805, 185. Число параметров для обучения: 12, 805, 185.

Число параметров, которые не участвуют в обучении: 0.

Для данной нейронной сети была построена ROC-кривая (рис. 1), по которой точность составила примерно 0,833.

Матрица неточностей для данной сети выглядит, как показано на рис. 2.

Из рис. 2 можно видеть, что данная сеть неплохо предсказывает истинные значения. Также нужно отметить, что отклонения при отрицательных ответах ниже на порядок, чем на положительных, что показывает достаточно высокую эффективность.

Далее рассмотрим нейронную сеть с LSTM-слоем. Архитектура представлена в табл. 2.

Параметры:

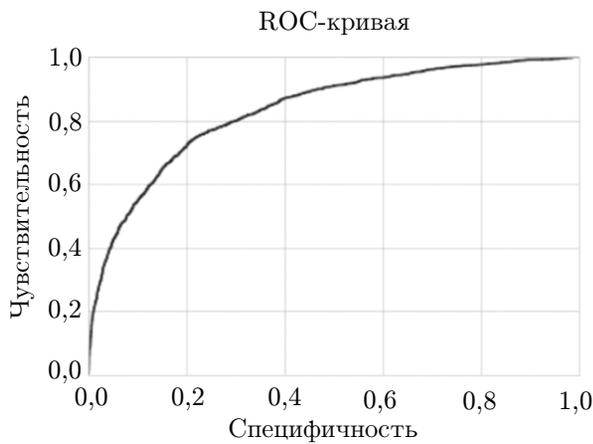
Общее число параметров: 12, 820, 641. Число параметров для обучения: 12, 820, 641.

Число параметров, которые не участвуют в обучении: 0.

Таблица 1

Архитектура нейронной сети с SimpleRNN

Типы слоёв	Выходной размер	Количество параметров
Embedding (Embedding)	(None, None, 128)	12800000
Simple_rnn (SimpleRNN)	(None, 32)	5152
Dense (Dense)	(None, 1)	33



ROC-AUC Значение: 0,833

Рис. 1

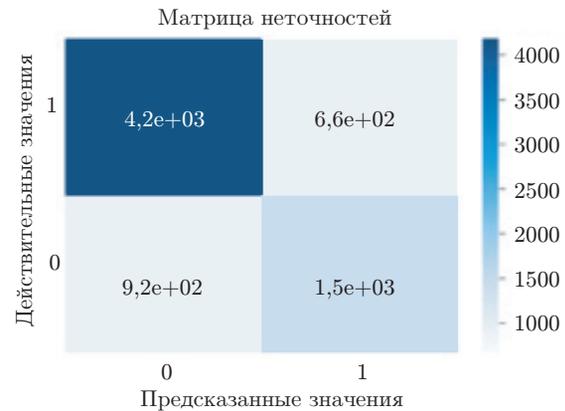
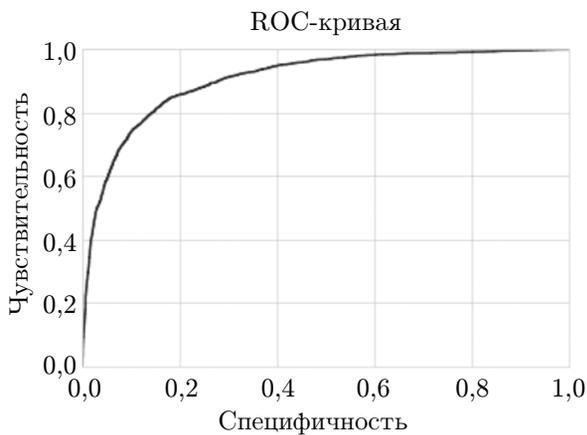


Рис. 2

Рис. 1. ROC-кривая для нейронной сети с SimpleRNN

Рис. 2. Матрица неточностей для нейронной сети с SimpleRNN



ROC-AUC Значение: 0,9065

Рис. 3

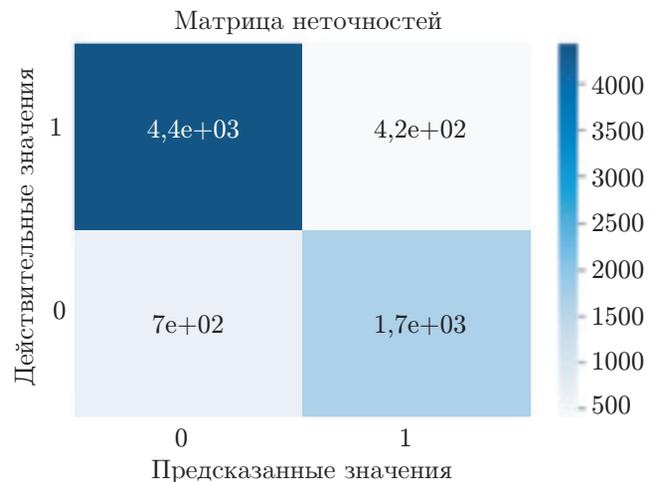


Рис. 4

Рис. 3. ROC-кривая для нейронной сети с LSTM-слоем

Рис. 4. Матрица неточностей для рекуррентной сети с LSTM-слоем

Данную архитектуру отличает слой lstm — рекуррентный слой с возможностью сохранения промежуточного состояния.

ROC-кривая для нейронной сети изображена на рис. 3, численное значение точности составило 0,91.

Матрица неточностей для сети с LSTM-слоем представлена на рис. 4.

Из рис. 4 можно видеть, что данная нейронная сеть тоже хорошо предсказывает истинные значения, но в отличие от SimpleRNN чаще ошибочно относит негативные к положительным.

Наконец, рассмотрим рекуррентную нейронную сеть с GRU-слоем. Архитектура показана в табл. 3.

Параметры:

Общее число параметров: 12, 815, 585. Число параметров для обучения: 12, 815, 585.

Таблица 2

Архитектура нейронной сети с LSTM-слоем

Типы слоёв	Выходной размер	Количество параметров
Embedding (Embedding)	(None, None, 128)	12800000
lstm (LSTM)	(None, 32)	20608
Dense (Dense)	(None, 1)	33

Таблица 3

Архитектура нейронной сети с GRU-слоем

Типы слоёв	Выходной размер	Количество параметров
Embedding (Embedding)	(None, None, 128)	12800000
gru (GRU)	(None, 32)	15552
Dense (Dense)	(None, 1)	33

Число параметров, которые не участвуют в обучении: 0.

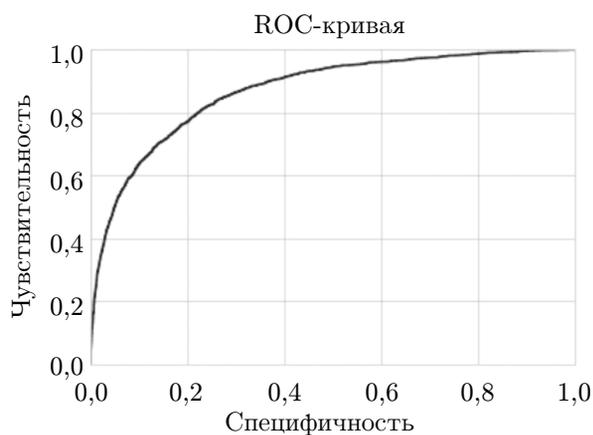
Данную архитектуру отличает слой gru: рекуррентный слой с возможностью сохранения промежуточного состояния, но в отличие от lstm имеет меньше параметров за счёт полного раскрытия внутреннего состояния.

ROC-кривая для рекуррентной нейронной сети с GRU-слоем представлена на рис. 5, её точность составила 0,87.

Матрица неточностей для неё показана на рис. 6.

На рис. 6 видно, что данная нейронная сеть лучше определяет истинные значения отзывов, но чаще делает ошибки на негативных отзывах, чем на положительных.

Результаты исследования. Обучающая выборка сделана на основе датасета [6]. Набор данных представляет собой 14412 комментариев на русском языке, взятых из социальных сетей. Особенностью набора является то, что все предложения взяты из неформального общения пользователей социальной сети на разные темы. В предложениях присутствуют жаргонизмы и слова со скрытым подтекстом. Поэтому эти особенности ограничивают



ROC-AUC Значение: 0,8708

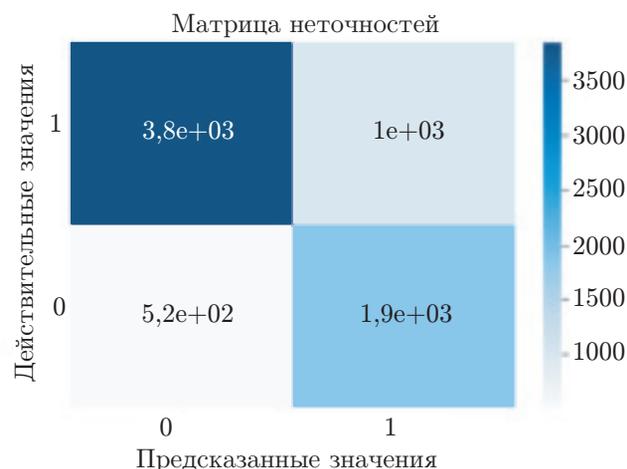


Рис. 5

Рис. 6

Рис. 5. ROC-кривая для нейронной сети с GRU-слоем

Рис. 6. Матрица неточностей для нейронной сети с GRU-слоем

Таблица 4

**Распределение длины отзыва и распределение количества слов
в отзыве (числовая характеристика)**

Длина отзыва		Количество слов в отзыве	
Количество образцов	14412	Количество образцов	14412
Среднее	176,5	Среднее	27,9
Стандартное отклонение	271,6	Стандартное отклонение	41,4
Минимальное:	21	Минимальное:	1
перцентиль — 25 %	57	перцентиль — 25 %	9
перцентиль — 50 %	101	перцентиль — 50 %	16
перцентиль — 75 %	197	перцентиль — 75 %	32
Максимальное	7404	Максимальное	1078

Таблица 5

Примеры предсказаний нейронных сетей

Отзыв	Истинное значение	SimpleRNN-предсказание	GRU-предсказание	LSTM-предсказание
светофоры с вызовом таймером	0	0,0001	0	0,0003
спасибо за конструктивную критику приму сведению ваши слова	0	0,0002	0,0002	0,0003
рабочий день впереди	0	0,0004	0,0001	0,0003
мать брат сват ребёнок сиди пикабу вконтакте дурак	1	0,8279	0,4614	0,1727
просто похоже это горит красный это магазин иркутский фары далеке это энергетик	0	0,1031	0,0425	0,0018
яндекс мб норм проводят собеседования работать отнюдь прекрасно гугле ашле хорошие собеседования понравилось собеседование близзард например ищут джунов чаще говорим топовые компании середнячок	0	0,0049	0,0004	0,0001
южные культуры ещё существуют вроде говорили олимпиады вырубили нафиг	0	0,0002	0	0,0001
кожного дурака	1	0,955	0,9781	0,9434
шо своё лицо засвети	1	0,9981	0,9995	0,9988
моих коллег программистов сеньоры зареган соцсетях некоторые заводят резюме принципа правда отправляет резюме её запросам хороших знакомых всё равно умудряются хантить каким образом	0	0,0828	0	0,0001
фото парад уродов лол	1	0,9984	0,9997	0,9995
действительно столько всякой химозы использую путаю названия	0	0,0001	0	0,0001
спасибо большое дельный совет	0	0,0002	0	0

обучение модели глубокого обучения, увеличивая её сложность [7, 8].

Все записи были приведены в нижний регистр; все html-теги, встречающиеся в записях, удалены. После этого задача по текстовой нормализации считается завершённой.

Далее был проведён разведывательный анализ данных, чтобы говорить о репрезентативности обоих наборов и распределении образцов (записей в датасете) к значениям целевого признака эмоциональной окраски текста. Проведён подсчёт и визуализация основных параметров записей в датасете. Распределение количества слов в отзыве представлено в табл. 4.

Как можно видеть, среди данных нет какого-либо перевеса в пользу одного из вариантов тональности текста; также, исходя из полученных числовых характеристик записей, можно считать, что текстовая выборка достаточно велика: [21; 7404] букв и [1; 1078] слов. Из совокупности факторов следует, что обучающий набор сбалансирован.

Из всех рассмотренных архитектур нейронных сетей лучше всего себя показали нейронные сети со слоями LSTM и GRU, у них примерно одинаковая точность: 0,9 и 0,87 соответственно.

Столбец `comment` датасета — тексты комментариев, тональность которых анализируем выбранными нейронными сетями, `true_prediction` — это истинные значения предсказаний (где 1 — предложение, имеющее негативную окраску, 0 — признак отсутствия негативной окраски), `simple_rnn_prediction` — значения предсказаний сети с использованием слоя `simple rnn` (простая рекуррентная нейронная сеть), `gru_prediction` — значения предсказаний нейронной сети со слоем GRU, `lstm_prediction` — значения предсказаний нейронной сети со слоем `lstm`).

Для отображения предсказаний нейронных сетей было выбрано случайным образом несколько предложений и каждая нейронная сеть сделала для неё предсказания (табл. 5).

На основании данной таблицы можно заметить, что в большинстве случаев нейронные сети справились с задачей анализа тональности текста, поскольку предсказанные значения сходны с истинными значениями и имеют достаточно малое отклонение. Однако делают ошибки на предложениях, имеющих скрытый подтекст. Это следует из малого объёма данных. Также следует отметить, что GRU- и LSTM-сети реже ошибаются, чем SimpleRNN, так как имеют блоки памяти и применяют фильтры.

Таким образом, были обучены легковесные модели нейронных сетей, которые показали достаточно высокую точность при решении задачи сентиментального анализа. Было показано, что сети, имеющие блоки памяти, лучше обрабатывают текстовые последовательности, чем простые рекуррентные сети [9, 10].

Заключение. В представленном исследовании описан процесс создания обучающей выборки на основе датасета, состоящего из 14412 комментариев на русском языке, взятых из социальных сетей. Особенностью набора данных является неформальное общение пользователей социальной сети на разные темы, что приводит к наличию данных жаргонизмов и слов со скрытым подтекстом. Была проведена текстовая нормализация данных и их разведывательный анализ для определения репрезентативности наборов данных, а также распределения образцов по значениям целевого признака эмоциональной окраски текста. Далее были обучены модели нейронных сетей с использованием слоёв SimpleRNN, GRU и LSTM для анализа тональности текста. Из результатов экспериментов следует, что модели, имеющие блоки памяти (GRU и LSTM), лучше обрабатывают текстовые последовательности, чем простые рекуррентные сети. Однако из-за наличия жаргонизмов и слов со скрытым подтекстом модели делают ошибки. В целом полученные результаты показывают, что обученные модели достаточно точно определяют эмоциональную окраску текста, что может быть полезно в различных приложениях обработки естественного языка.

СПИСОК ЛИТЕРАТУРЫ

1. **Абдулкадиров Р. И., Ляхов П. А.** Новый подход к обучению нейронных сетей с помощью натурального градиентного спуска с импульсом на основе Дирихле // Компьютерная оптика. 2023. **47**, № 1. С. 160–169.
2. **Когда** использовать нейронные сети MLP, CNN и RNN. URL: <https://www.machinelearningmastery.ru/when-to-use-mlp-cnn-and-rnn-neural-networks> (дата обращения: 12.05.2022).
3. **Рекуррентные** нейронные сети: типы, обучение, примеры и применение. URL: <https://neurohive.io/ru/osnovy-data-science/rekurrentnye-nejronnye-seti/> (дата обращения: 12.05.2022).
4. **Illustrated** Guide to LSTM's and GRU's: A step by step explanation. URL: <https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21> (дата обращения: 12.05.2022).
5. **Токенизация** слов при помощи nltk и keras. URL: <https://webtort.ru/токенизация-слов-при-помощи-nltk-и-keras/> (дата обращения: 12.05.2022).
6. **Russian** language toxic comments. URL: <https://www.kaggle.com/datasets/blackmoon/russian-language-toxic-comments> (дата обращения: 12.05.2022).
7. **Спицын В. Г., Болотова Ю. А., Фан Н. Х., Буй Т. Т. Ч.** Применение вейвлет-преобразования хаара, метода главных компонент и нейронных сетей для оптического распознавания символов на изображениях в присутствии импульсного шума // Компьютерная оптика. 2016. **40**, № 2. С. 249–257.
8. **Yamaev A. V., Chukalina M. V., Nikolaev D. P. et al.** Neural network regularization in the problem of few-view computed tomography // Компьютерная оптика. 2022. **46**, № 3. С. 422–428.
9. **Козик В. И., Нежевенко Е. С., Феоктистов А. С.** Исследование метода адаптивного прогнозирования развития лесных пожаров на основе рекуррентных нейронных сетей // Автометрия. 2014. **50**, № 4. С. 88–95.
10. **Терещенко С. Н., Перов А. А., Осипов А. Л.** Особенности применения предобученных свёрточных нейронных сетей к задачам стегоанализа графических изображений // Автометрия. 2021. **57**, № 4. С. 98–105. DOI: 10.15372/AUT20210412.

Поступила в редакцию 24.04.2023

После доработки 13.05.2023

Принята к публикации 17.05.2023
