УДК 004.912 *DOI 10.55421/3034-4689_2025_28_4_97*

Д. А. Колобова

АНАЛИЗ ЭФФЕКТИВНОСТИ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ В ЗАДАЧЕ КАТЕГОРИЗАЦИИ ТЕКСТОВ СРЕДСТВ МАССОВОЙ ИНФОРМАЦИИ

Ключевые слова: нейронные сети, классификация текстов, обработка естественного языка, токенизация, рекуррентные нейронные сети, токенизация, предобработка текста.

В статье исследуются современные методы и подходы к решению задачи классификации новостных текстов, что является актуальной проблемой в условиях большого объема информации, доступной пользователям. Классификация новостей играет ключевую роль в оптимизации процесса поиска информации, способствует созданию персонализированного контента и помогает анализировать общественные тренды, что особенно важно в эпоху цифровизации. В ходе работы рассматриваются основные концепции и принципы, связанные с обработкой и анализом текста, включая этапы предобработки текста, составления словаря, токенизации, создания батчей из текстовых последовательностей и классификации текстов. Особенное внимание уделяется различным архитектурам рекуррентных нейронных сетей (RNN), их особенностями, преимуществами и недостатками в контексте задачи классификации текста. Рекуррентные нейронные сети являются мощным инструментом для обработки последовательных данных, таких как текст, и позволяют учитывать контекст при классификации. Проведены эксперименты с различными моделями рекуррентных нейронных сетей, выполнен подбор оптимальных параметров, обеспечивающих высокую точность классификации новостных текстов, и выявлена наилучшая модель - GRU_model512_2layers_dropout_epoch10, состоящая из двух рекуррентных слоев архитектуры GRU, содержащая по 512 нейронов в скрытом слое, с дропаутом 20%, обученная на 10 эпохах. Она занимает меньше места в памяти (на 10 мб), чем модель с архитектурой LSTM и такими же параметрами, поскольку архитектура GRU имеет более простое строение. В связи с этим она также быстрее обучается (на 17 с/эпоха быстрее, чем модель с архитектурой LSTM). Также она показывает более высокую точность (91,6 %), чем модели с более простой архитектурой, склонные к переобучению. Для программной реализации алгоритма классификации новостных текстов используется язык программирования Python, а также фреймворк машинного обучения с открытым исходным кодом PyTorch и библиотека обработки естественного языка NLTK. Процесс классификации новостного текста выполняется в следующей последовательности: загрузка текста, его обработка, классификация и вывод категории, к которой данный текст принадлежит. Для обучения моделей и проверки результатов используется набор данных, содержащий образцы новостных текстов четырех категорий.

D. A. Kolobova

ANALYSIS OF THE EFFECTIVENESS OF RECURRENT NEURAL NETWORKS IN THE TASK OF CATEGORIZING MEDIA TEXTS

Keywords: neural networks, text classification, natural language processing, tokenization, recurrent neural networks, tokenization, text preprocessing.

The article examines modern methods and approaches to solving the problem of classifying news texts, which is an urgent problem in the context of a large amount of information available to users. News classification plays a key role in optimizing the information retrieval process, contributes to the creation of personalized content and helps analyze social trends, which is especially important in the era of digitalization. In the course of the work, the main concepts and principles related to text processing and analysis are considered, including the stages of text preprocessing, dictionary compilation, tokenization, creation of batches from text sequences and text classification. Special attention is paid to various architectures of recurrent neural networks (RNNs), their features, advantages and disadvantages in the context of the text classification task. Recurrent neural networks are a powerful tool for processing sequential data, such as text, and allow for context-based classification. Experiments have been conducted with various models of recurrent neural networks, optimal parameters have been selected to ensure high classification accuracy of news texts, and the best model has been identified - GRU_model512_2layers_dropout_epoch10, consisting of two recurrent layers of the GRU architecture, containing 512 neurons each in a hidden layer, with a dropout of 20%, trained on 10 epochs. It takes up less memory space (by 10 MB) than a model with the LSTM architecture and the same parameters, since the GRU architecture has a simpler structure. In this regard, it is also faster to learn (17 s/epoch faster than the LSTM architecture model). It also shows higher accuracy (91.6%) than models with simpler architectures, which are prone to overfitting. For the software implementation of the news text classification algorithm, the Python programming language is used, as well as the open source PyTorch machine learning framework and the NLTK natural language processing library. The process of classifying a news text is performed in the following sequence: loading the text, processing it, classifying it, and outputting the category to which the text belongs. To train the models and verify the results, a dataset containing samples of four categories of news texts is used.

Введение

Обработка естественного языка (NLP) – это направление искусственного интеллекта, изучающее проблемы компьютерного анализа и синтеза текстов на естественном языке, который люди используют для общения между собой. NLP находится на стыке компьютерной лингвистики и технологий машинного обучения [1]. Одной из задач NLP является классификация текстов.

Классификация текстов – задача компьютерной лингвистики, заключающаяся в отнесении документа к одной из нескольких категорий на основании содержания документа [2]. Классификация текстов применяется во многих приложениях для обнаружение спама, анализа тональности, а также для разделения веб-страниц и сайтов по тематическим каталогам. Примером таких сайтов может служить новостной блог.

Классификация новостей на категории нужна для удобства пользователей. Она экономит время, позволяя быстро найти нужную информацию среди огромного объема новостей. Благодаря классификации новостей на категории, платформы могут предлагать пользователям персонализированный контент. Также классификация помогает в анализе новостного контента. Путем отслеживания количества новостей в каждой категории можно выявить популярные темы, тренды или изменения в обществе [3].

Поскольку классификация новостей позволяет решить большое количество задач, необходимо обеспечить ее точность. Это позволит избежать неудобства пользователей при поиске конкретной новости, сделает рекомендации более персонализированными, а также увеличит качество исследования тенденций в обществе.

Таким образом, цель данной работы – выбрать наилучший по точности метод классификации новостных текстов на категории и реализовать его.

Обзор существующих методов

Существует множество методов классификации текстов: методы машинного обучения, такие как Байесовская классификация, метод опорных векторов (SVM), метод ближайших соседей, деревья решений; метод TF-IDF[4]; классификаторы на основе правил [5] и др. Однако данные методы подходят для простых и быстрых решений, когда объем данных ограничен и не требуется учитывать последовательность слов. В отличие от описанных выше методов, метод на основе рекуррентных нейронных сетей способен обрабатывать большие объемы информации и учитывать закономерности в тексте, что позволит эффективно решать задачу классификации текстов.

Рекуррентные нейронные сети (RNN) позволяют учитывать контекст и порядок слов в тексте, что делает их эффективными для классификации новостей на категории. Архитектура RNN состоит из повторяющихся блоков, которые позволяют сети запоминать предыдущие состояния и использовать их для прогнозирования следующего состояния [6]. На рис.1 показано строение рекуррентного слоя:

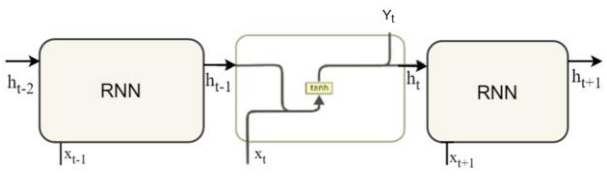

Рис. 1 – Строение рекуррентного слоя [6]

Fig. 1 – Structure of the recurrence layer [6]

Здесь h_t – вектор скрытого состояния слоя в момент времени t, h_{t-1} – вектор скрытого состояния слоя в момент времени t-1, x_t – входные данные.

Вектор h_t подсчитывается по формуле:

$$h_t = \tanh(Wx_t + Uh_{t-1} + b_h),$$

где W,U,b_h – обучаемые вектора.

Выходной вектор подсчитывается по формуле:

$$y_t = \tanh(Vh_t + b_y),$$

где V,b_y – обучаемые вектора.

Однако классические RNN имеют проблему затухания градиента, когда градиенты становятся слишком маленькими или слишком большими при обучении на длинных последовательностях [7]. Для решения этой проблемы были разработаны более продвинутые архитектуры, такие как LSTM (Long Short-Term Memory) и GRU (Gated Recurrent Unit), которые способны лучше сохранять и использовать информацию о длинных зависимостях в данных, т.к. в LSTM реализован механизм долгосрочной памяти, а в GRU – механизм контроля информации, позволяющие хранить информацию с предыдущих состояний и использовать ее в прогнозировании следующих [7].

Реализация моделей с использованием архитектур RNN, LSTM и GRU для решения задачи классификации новостных текстов представлена ниже.

Реализация метода на основе рекуррентных нейронных сетей

В данной работе в качестве данных был использован датасет AG News. Он содержит тексты, представляющие собой заголовок и описание новостных статей, и метки – принадлежность статьи одному из четырех классов («World», «Sports», «Business», «Sci/Tech»). Набор содержит 30000 тренировочных и 1900 тестовых образцов для каждого класса. Пример строки датасета: ‘News: UK scientists roll out Wi-Fi proof wallpaper British boffins have developed wallpaper that blocks Wi-Fi traffic but still allows other wireless transmissions to pass through in a bid to prevent unauthorised access to sensitive data via the WLAN.’ Данная строка относится к категории 3 – ‘Sci/Tech’.

Разнообразие элементов, из которых состоит текст, усложняет его анализ. Для работы с текстовыми данными их необходимо упростить и привести к стандартной форме, подходящей для используемого метода. Данный процесс называют предобработкой текста [8].

Этап предварительной обработки данных

В ходе подготовительного этапа был составлен словарь из слов тренировочных текстов. Для этого каждый текст был приведен к нижнему регистру, из него была удалена пунктуация, подсчитана частота вхождения каждого слова в текст, после чего в словарь были добавлены слова, повторяющиеся более 25 раз (значение 25 было подобрано экспериментальным путем, чтобы словарь был достаточно объемным для обучения нейронной сети, но в то же время не содержал несущественных слов).

Токенизация в NLP – это процесс преобразования последовательности текста в более мелкие части, называемые токенами [5]. Также существуют служебные токены, такие как токен начала и конца последовательности, неизвестный токен и токен, позволяющий объединять последовательности разной длины в один батч.

Слова, соответствующие служебным токенам, также были добавлены в словарь. Каждому слову в словаре был поставлен в соответствие его порядковый номер [9].

В ходе предварительной обработки данных каждое предложение текстов прошло токенизацию. В начало и конец предложения были добавлены служебные токены, а слова заменены на значения, соответствующие им в словаре. Если слова, встреченного в предложении, нет в словаре, то ему в соответствие было поставлено значение неизвестного токена [10].

Также была реализована функция для создания батча из текстовых последовательностей разной длины. Батч нужен для удобства работы с большим набором текстовых данных для ускорения обучения модели нейронной сети. Последовательности, длина которых меньше длины максимальной последовательности, дополняются служебным токеном, позволяющим объединять последовательности разной длины в один батч, после чего новая последовательность добавляется в батч. Метки классов так же добавляются в один батч. В ходе проведения экспериментов размер батча подбирался вручную как гиперпараметр.

Архитектура нейронной сети

Модель нейронной сети состоит из следующих слоев:

1) Эмбеддинг-слой. Эмбеддинг — это способ представления слов в виде векторов, которые отображают смысл и контекст слова [11]. Данный слой получает на вход номера слов (были получены в ходе предварительной обработки данных), а на выходе выдаёт их векторные представления (до начала обучения они случайные).

2) Слой рекуррентной нейронной сети (RNN, LSTM или GRU);

3) Два полносвязных слоя.

Также есть dropout-слой, позволяющий “отключить” определенный процент нейронов во время обучения для предотвращения переобучения сети [12]. В качестве функции активации был использован гиперболический тангенс, т.к. он эффективно распознает сложные зависимости в данных и часто применяется для решения задач NLP [13]. Для обучения был выбран оптимизатор Adam, т.к. данный оптимизатор имеет высокую скорость обучения [14]. В качестве функции потерь использовалась кросс-энтропия [15].

Результаты проведенных экспериментов

Для подбора архитектуры и параметров нейронной сети, решающей задачу классификации новостных текстов датасета AG News с максимально возможной точностью, был проведен ряд экспериментов. В ходе проведения экспериментов изменялись следующие параметры: процент “выключаемых” нейронов (dropout), количество нейронов в скрытом слое (hidden-dim), количество эпох обучения(epoch). Эксперименты проводились на трех типах архитектур рекуррентных нейронных сетей – RNN, GRU и LSTM. Также варьировался тип агрегации информации с предыдущих состояний – max и mean. Результаты отражены в табл.1. Здесь в первых трех строках представлены три базовых модели, отличающиеся лишь архитектурой рекуррентного слоя, которые послужили основой для дальнейшего подбора параметров для достижения наилучшего качества каждой модели. Наборы значений параметров, при которых удалось достичь максимальной точности каждой модели, представлены в трех последний строках табл.1.

В табл. 1 рассматриваются модели:

1 – RNN_model с архитектурой RNN и количеством рекуррентных слоев равным 1;

2 – GRU_model с архитектурой GRU и количеством рекуррентных слоев равным 1;

3 – LSTM_model с архитектурой RNN и количеством рекуррентных слоев равным 1;

4 – RNN_model_2layers с архитектурой RNN и количеством рекуррентных слоев равным 2;

5 – GRU_model512_2layers_dropout_epoch10 с архитектурой GRU и количеством рекуррентных слоев равным 2;

6 – LSTM_model512_2layers_dropout с архитектурой LSTM и количеством рекуррентных слоев равным 2.

Таблица 1 – Результаты экспериментов

Table 1 – Experimental results

Модель	**Изменяемые параметры**			**Accuracy**	
	dropout	hidden_dim	epoch	max	mean
1	10%	256	5	0,902	0,910
2	10%	256	5	0,912	0,914
3	10%	256	5	0,910	0,914
4	10%	256	5	0,915	0,906
5	20%	512	10	0,916	0,916
6	20%	512	5	0,911	0,916

Наилучшие результаты показала архитектура GRU с 2 рекуррентными слоями по 512 нейронов в скрытом слое, с процентом дропаута 20%, обученная на 10 эпохах (см. модель 5 табл. 1). На графике изменения функции потерь (см. рис.2) видно, что loss падает с увеличением числа эпох при любом типе агрегации. На графике изменения функции точности (см. рис.3) видно, что точность достигает максимального значения 0,916 при типе агрегации mean на 3 эпохе, при типе агрегации max на 5 эпохе, после чего точность падает, следовательно, наступает переобучение модели.

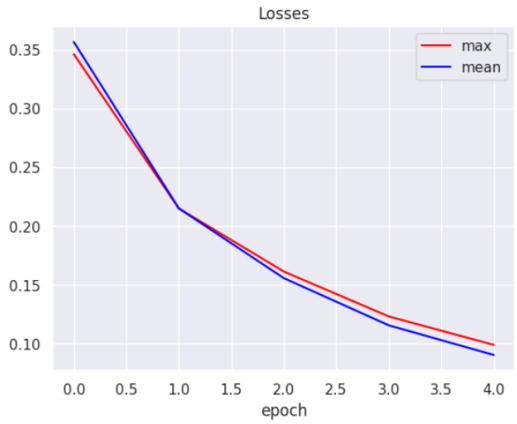

Рис. 2 – Изменение функции потерь

Fig. 2 – Change in the loss function

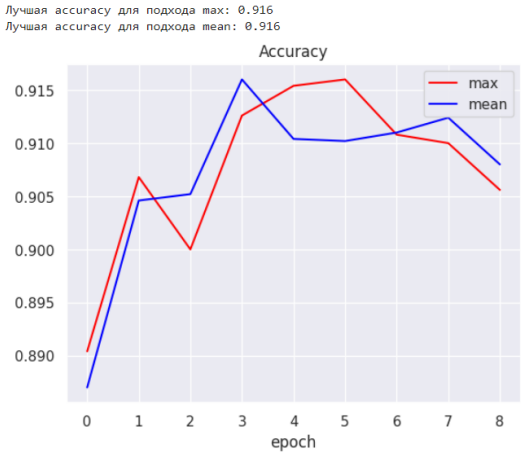

Рис. 3 – Изменение точности

Fig. 3 – Change in accuracy

Разработка велась на языке программирования Python с использованием фреймворка PyTorch и библиотеки обработки естественного языка NLTK. Вычисления проводились в облачной среде Google Colab, с использованием предоставляемого GPU, в браузере Google Chrome на компьютере с шестиядерным процессором Intel(R) Core(TM) i5-10400 CPU @ 2.90GHz и оперативной памятью объемом 16 Гб. Обучение занимало в среднем 1 мин 48 сек на каждую эпоху.

Описанный в работе подход к решению задачи классификации новостных текстов показал высокую эффективность и точность. В дальнейшем он может быть использован для облегчения навигации на новостных сайтах.

Заключение

Таким образом, использование рекуррентных нейронных сетей для классификации новостных текстов продемонстрировало свою эффективность. В ходе исследования была подобрана оптимальная по точности, скорости обучения и размеру модель GRU_model512_2layers_dropout_epoch10, показавшая наивысшую среднюю точность категоризации новостных текстов, равную 91,6%.

В дальнейшем планируется дообучение модели на датасетах новостных текстов на русском языке, что позволит значительно улучшить её способность к анализу контекста и структуры языка, а также адаптировать её под особенности русскоязычных медиа.

Литература

1. IBM. What is NLP? [Электронный ресурс] URL: https://www.ibm.com/think/topics/natural-language-processing (дата обращения: 31.08.2024).
2. ИТМО. Классификация текстов и анализ тональности. [Электронный ресурс] URL: https://clck.ru/3Cwnwi (дата обращения: 17.07.2024).
3. S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao. *ACM computing surveys (CSUR)*, **54**, 3, 1-40 (2021). DOI: 10.48550/arXiv.2004.03705.
4. Т. В. Батура. *Программные продукты и системы/Software & Systems*, **30,** 1, 85-99 (2017). DOI: 10.15827/0236-235X.117.085-099.
5. NFT.RU. Что такое токенизация? [Электронный ресурс] URL: https://nft.ru/article/chto-takoe-tokenizatsiia-1011 (дата обращения: 3.08.2024).
6. M.S. Ibrahim, W. Abbas, M. Waseem, C. Lu, H.H. Lee, J. Fan, K.-H. Loo. *Mathematics*, **11**, 15, Article 3283 (2023). DOI: 10.3390/math11153283.
7. А.В. Глазкова, *Программные продукты и системы*. **32**, 2, 263–267 (2019). DOI: 10.15827/0236-235X.126.263-267.
8. Р. К. Акжолов, А. В. Верига. *Вестник науки.* **24**, 3, 66-68 (2020).
9. C.W. Schmidt, V. Reddy, H. Zhang, A. Alameddine, O. Uzan, Y. Pinter, C. Tanner. *Conference on Empirical Methods in Natural Language Processing*, Miami, Florida, 2024, С. 2.
10. Большакова Е.И., Воронцов К.В., Ефремова Н.Э., Клышинский Э.С., Лукашевич Н.В., Сапин А.С. *Автоматическая обработка текстов на естественном языке и анализ данных.* НИУ ВШЭ, Москва, 2017, С. 67-70.
11. SberAI. Что такое эмбеддинги и как они работают? [Электронный ресурс] URL: https://ai.sber.ru/post/chto-takoe-embeddingi-i-kak-oni-rabotaut (дата обращения: 9.10.24).
12. Хабр. Dropout — метод решения проблемы переобучения в нейронных сетях. [Электронный ресурс] URL:https://habr.com/ru/companies/wunderfund/articles/330814/ (дата обращения: 9.10.24).
13. Хабр. Выбор слоя активации в нейронных сетях: как правильно выбрать для вашей задачи. [Электронный ресурс] URL: https://habr.com/ru/articles/727506/ (дата обращения: 15.08.2024).
14. В.А. Мальцев, В сб. *Научный форум: Инновационная наука*, Изд. «МЦНО», Москва, 2019, С. 63.
15. J. Torkunova, D. Milovanov. (2023). *International Journal of Advanced Studies*, **13**, 4, 142-158 (2023). DOI: https://doi.org/10.12731/2227-930X-2023-13-4-142-158.

References

1. IBM. What is NLP? [Electronic resource] URL: https://www.ibm.com/think/topics/natural-language-processing (date of reference: 31.08.2024).
2. ITMO. Text classification and tone analysis. [Electronic resource] URL: https://clck.ru/3Cwnwi (date of reference: 17.07.2024).
3. S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao. *ACM computing surveys (CSUR)*, **54**, 3, 1-40 (2021). DOI: 10.48550/arXiv.2004.03705.

4. T. V. Batura. *Software products and systems/Software & Systems*, **30,** 1, 85-99 (2017). DOI: 10.15827/0236-235X.117.085-099.
5. NFT.RU. What is tokenization? [Electronic resource] URL: https://nft.ru/article/chto-takoe-tokenizatsiia-1011 (date of reference: 3.08.2024).
6. M.S. Ibrahim, W. Abbas, M. Waseem, C. Lu, H.H. Lee, J. Fan, K.-H. Loo. *Mathematics*, **11**, 15, Article 3283 (2023). DOI: 10.3390/math11153283.
7. A.V. Glazkova, *Software Products and Systems*. **32**, 2, 263-267 (2019). DOI: 10.15827/0236-235X.126.263-267.
8. R. K. Akzholov, A. V. Veriga. *Bulletin of Science.* **24**, 3, 66-68 (2020).
9. C.W. Schmidt, V. Reddy, H. Zhang, A. Alameddine, O. Uzan, Y. Pinter, C. Tanner. *Conference on Empirical Methods in Natural Language Processing*, Miami, Florida, 2024, P. 2.
10. Bolshakova E.I., Vorontsov K.V., Efremova N.E., Klyshinsky E.S., Lukashevich N.V., Sapin A.S. *Automatic processing of natural language texts and data analysis*. National Research University Higher School of Economics, Moscow, 2017, P. 67-70.
11. SberAI. What are embeddings and how do they work? [Electronic resource] URL: https://ai.sber.ru/post/chto-takoe-embeddingi-i-kak-oni-rabotaut(date of access: 9.10.24).
12. Hubr. Dropout - a method for solving the problem of overtraining in neural networks. [Electronic resource] URL:https://habr.com/ru/companies/wunderfund/articles/330814/(accessed on 9.10.24).
13. Hubr. Activation layer selection in neural networks: how to choose the right one for your task. [Electronic resource] URL:https://habr.com/ru/articles/727506/ (accessed 15.08.2024).
14. V.A. Maltsev, V sb. *Nauchny forum: Innovatsionnaya nauka*, Izd. "ICNO", Moscow, 2019, P. 63.
15. J. Torkunova, D. Milovanov. (2023). *International Journal of Advanced Studies*, **13**, 4, 142-158 (2023). DOI: https://doi.org/10.12731/2227-930X-2023-13-4-142-158.

© **Д. А. Колобова** – инженер кафедры Автоматизированных систем обработки информации и управления, Казанский национальный исследовательский технический университета им. А.Н.Туполева, Казань, Россия, darya.kolobova@inbox.ru.

© **D. A. Kolobova** – Engineer at the Department of Automated Information Processing and Management Systems, Kazan National Research Technical University named after A.N.Tupolev, Kazan, Russia, darya.kolobova@inbox.ru.

Дата поступления рукописи в редакцию – 27.03.25.
Дата принятия рукописи в печать – 07.04.25.