

РЕШЕНИЕ ЗАДАЧИ КЛАССИФИКАЦИИ ЭМОЦИОНАЛЬНОГО ТОНА СООБЩЕНИЯ С ОПРЕДЕЛЕНИЕМ НАИБОЛЕЕ ПОДХОДЯЩЕЙ АРХИТЕКТУРЫ НЕЙРОННОЙ СЕТИ

Д. И. Багаутдинов¹ [0000-0003-4673-3253], Р. Салман² [0000-0002-1552-4910],
В. А. Алексеев³ [0000-0002-2208-9515], Р. М. Усмонов⁴ [0009-0002-1289-5465]

¹⁻⁴ *Институт информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета, ул. Кремлевская, 35, г. Казань, 420008*

¹d4niskabag@yandex.ru, ²Riham.salman.98@gmail.com, ³aleksvlad99@gmail.com, ⁴rustam190401@mail.ru

Аннотация

Для определения наиболее эффективного подхода к решению задачи классификации эмоционального тона сообщения проведено обучение выбранных моделей нейронной сети на различных наборах обучающих данных. На основе такого показателя, как процентное соотношение правильно данных ответов на тестовом наборе данных, сравнены комбинации наборов обучающих данных и различных моделей, обученных на основе этих данных. Произведено обучение четырех моделей нейронной сети на трех различных наборах обучающих данных. В результате сравнения точности ответов каждой модели, обученной на разных обучающих данных, сделаны выводы о выборе модели нейронной сети, наиболее подходящей для решения поставленной задачи.

Ключевые слова: *NLP, sentiment detection, neural networks, comparison of neural network models, LSTM, CNN, BiLSTM.*

ВВЕДЕНИЕ

Распознавание эмоционального тона сообщений [1] является важной задачей в области обработки естественного языка (NLP), поскольку может предоставить ценную информацию о значении и намерении, скрывающимися за тексто-

выми сообщениями. В последние годы использование нейронных сетей для решения этой задачи стало все более популярным благодаря их способности к обучению и обобщению данных [2, 3].

В настоящем исследовании мы сравнили различные модели нейронных сетей и наборы данных с целью их использования для решения задачи распознавания эмоционального тона в сообщениях. Мы исследовали эффективность различных архитектур нейронных сетей, таких как сверточные нейронные сети (CNN), нейронные сети с долгой краткосрочной памятью (LSTM), двунаправленная модель нейросети с долгой краткосрочной памятью (BiLSTM), на разных наборах данных с различными размерами и количеством классов эмоционального тона входных сообщений. В рамках данной работы проведено распознавание пяти базовых классов эмоционального тона сообщений: радость, грусть, нейтральная эмоция, злость, любовь. Для этого мы нашли и выбрали несколько наборов данных сообщений, помеченных соответствующим классом эмоционального тона, от маломасштабных наборов данных с несколькими тысячами примеров до больших наборов данных с 1.6 миллиона примеров. Затем, после обработки обучающих наборов данных, мы обучили несколько моделей нейронных сетей на каждом наборе. Для сравнения качества работы этих моделей мы подготовили набор тестовых сообщений, каждое из которых подавалось на вход обученным нейронным сетям [4, 5]. В конечном итоге комбинация нейронной сети и обучающего набора данных, у которой процентное соотношение верных ответов по тестовому набору сообщений стало наивысшим, признана наиболее эффективной для решения задачи классификации эмоционального тона сообщений.

БОТ ДЛЯ АНАЛИЗА ТОНАЛЬНОСТИ ТЕКСТА

Итоговой задачей нашего исследования является анализ влияния архитектуры нейронной сети и использованного для ее обучения датасета на примере разработки telegram-бота для предсказания эмоциональной тональности текста. Акцент сделан именно на рассмотрении использованных архитектур нейронной сети.

Использованный инструментарий

Основным использованным языком программирования являлся Python 3-ей версии, в процессе работы были использованы следующие библиотеки.

Keras (TensorFlow) [6] — открытая программная библиотека для машинного обучения, разработанная компанией Google для решения задач построения и тренировки нейронной сети с целью автоматического нахождения и классификации образов с достижением качества человеческого восприятия. Эта библиотека написана на языке Python и обеспечивает взаимодействие с искусственными нейронными сетями и в данный момент является частью TensorFlow [7].

NumPy [8] — библиотека для работы с математическими функциями, многомерными массивами и матрицами. В частности, она была использована нами для генерации векторов.

Pandas [9] — библиотека для обработки и анализа данных, ее работа с данными строится поверх библиотеки NumPy, являющейся инструментом более низкого уровня.

Aiogram [10] — это современный и полностью асинхронный фреймворк для Telegram Bot API, написанный на Python с использованием `asyncio` [11] и `aiohttp`[12].

Также в работе была применена предобученная модель GloVe [13] для получения векторных представлений слов размерности 200. GloVe – это алгоритм обучения без учителя для получения векторных представлений слов. Часть вычислений производилась с использованием платформы **Google Colab** [14] – бесплатном облачном сервисе на основе Jupyter Notebook [15]. Благодаря использованию GPU этого сервиса, время обучения моделей было сокращено в разы [13].

Данные для обучения

Обучение моделей производилось на трех различных наборах данных. Первый набор состоял только лишь из предложений от первого лица («Я люблю пиццу» и т. п.) [16], что было достаточно серьезным ограничением, и модель плохо распознавала предложения иной структуры, предложения были помечены 6-ю классами. Второй набор данных имел ограничение в количестве классов, которыми он был

размечен, предложения были более разнообразными по своей структуре, но датасет размечен только на два класса – «позитивный» и «негативный» [17]. Последний датасет собран из данных твитов социальной сети Twitter, размеченных 5-ю классами эмоций, поэтому в конечном итоге выбор был сделан в пользу этого обучающего набора данных [18].

Обзор архитектур нейронных сетей, использованных для анализа

В процессе разработки нашего бота, решающего задачу анализа тональности текста, было опробовано несколько архитектур нейронных сетей с целью улучшить результаты анализа тональности текста, а именно, сверточная нейронная сеть (CNN), нейронная сеть с долгой краткосрочной памятью (LSTM), двунаправленная модель нейросети с долгой краткосрочной памятью (BiLSTM).

CNN. Изначально была реализована простая архитектура сверточной нейронной сети, или CNN. Состоит данная модель из следующих слоев:

1) Первый слой модели – ‘Embedding’, предназначенный для представления входных данных в виде числового вектора. Этот слой принимает три аргумента: ‘num_words’, который представляет размер словаря, ‘32’, который представляет размерность числового вектора, и ‘input_length’, который представляет длину каждой входной последовательности. Этот слой создает обучающую таблицу поиска, которая преобразует каждое слово во входной последовательности в вектор размера 32.

2) Второй слой – это сверточный слой Conv1D, который выполняет одномерную свертку переданной входной последовательности. Этот слой применяет 250 фильтров размером 5 каждый и использует функцию активации ‘relu’. Этот слой помогает идентифицировать значимые признаки во входной последовательности на основе числового вектора, полученного из предыдущего слоя.

3) Третий слой – это слой максимальной подвыборки (GlobalMaxPooling1D Layer), который выполняет глобальную операцию максимальной подвыборки над выходом предыдущего слоя. Этот слой выбирает максимальное значение из каждого набора признаков, созданного слоем Conv1D, и выводит вектор фиксированного размера.

4) Четвертый слой – это полносвязный слой (Dense Layer) с 128 нейронами и функцией активации 'relu'. Этот слой принимает выход предыдущего слоя в качестве входа и применяет к нему линейное преобразование. Выходом этого слоя является вектор размера 128.

5) Последний слой – это еще один полносвязный слой Dense с размерностью выходного пространства 5 и функцией активации 'softmax'. Этот слой принимает выход предыдущего слоя в качестве входа и производит распределение вероятностей по 5 возможным классам эмоционального тона входных сообщений.

Архитектура данной модели представлена на Рис. 1.

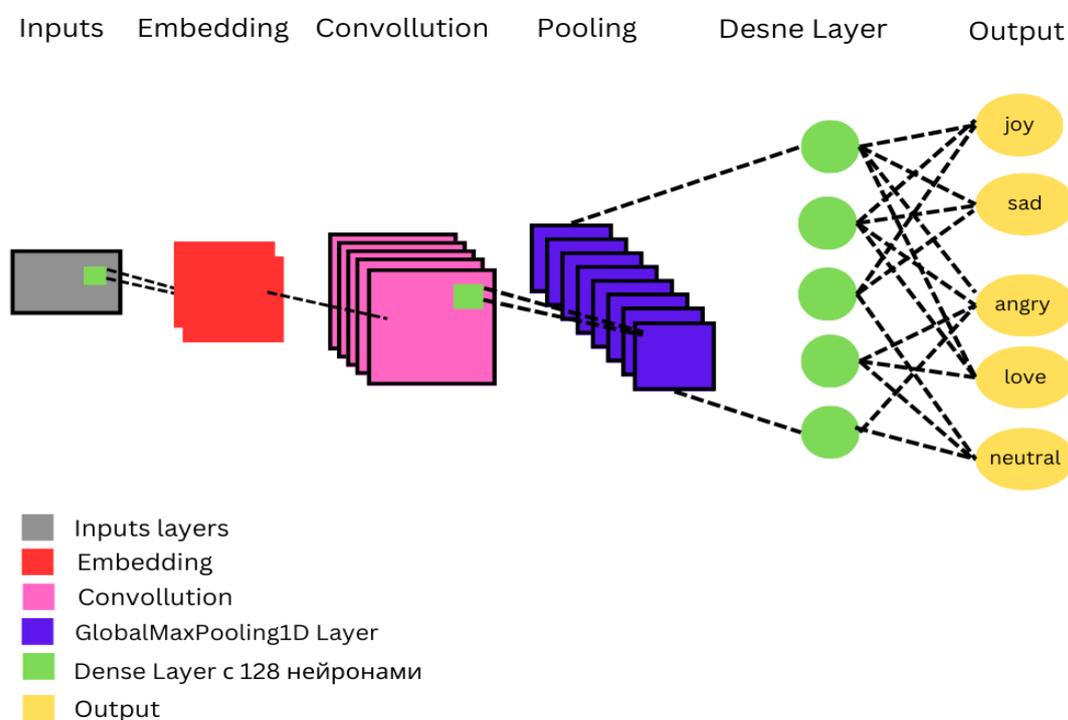


Рис. 1. Архитектура CNN модели

LSTM. Следующей архитектурой нейронной сети для реализации поставленной задачи была выбрана архитектура нейронной сети с долгой краткосрочной памятью (LSTM). Состоит данная модель из следующих нейронных слоев (Рис. 2).

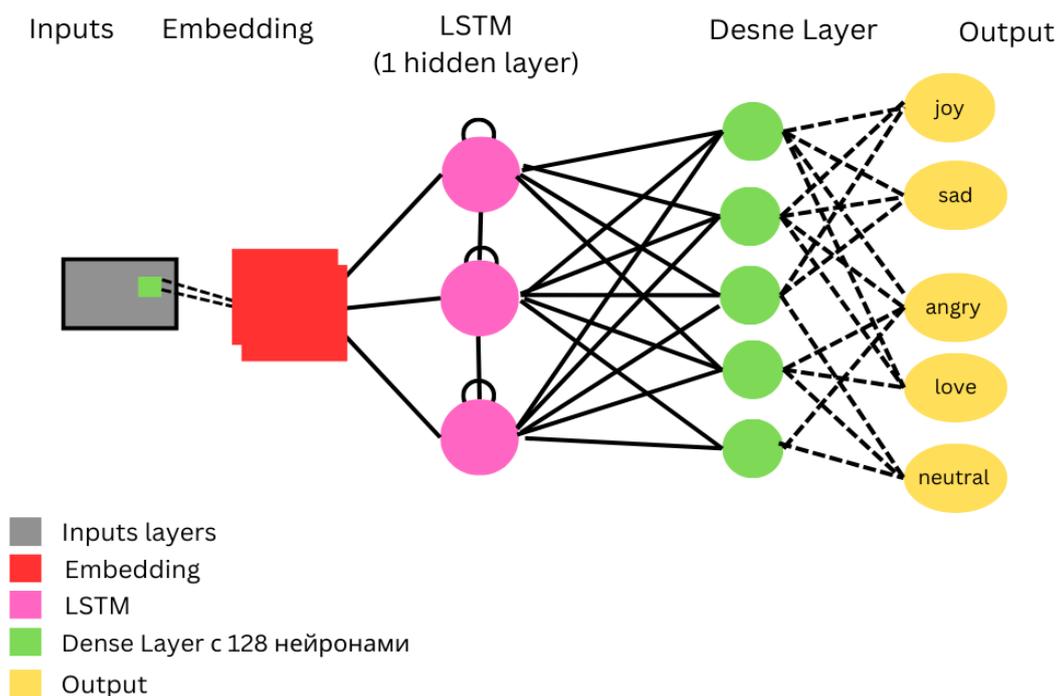


Рис. 2. Архитектура модели LSTM

- 1) Первый слой модели, так же, как у предыдущей, – ‘Embedding’, предназначенный для представления входных данных в виде числового вектора;
- 2) Второй слой – слой LSTM. Это рекуррентный слой нейронной сети, который обрабатывает последовательность числовых векторов, выводимых слоем Embedding. Уровень LSTM имеет 128 нейронов и коэффициент ‘dropout’, равный 0,2 для предотвращения переобучения. Аргумент ‘recurrent_dropout’ задает частоту выпадения повторяющихся соединений между блоками памяти.
- 3) Третий слой – полносвязный слой (Dense Layer). Этот уровень состоит из 128 нейронов и использует функцию активации ‘relu’. Это полносвязный слой, который принимает выходные данные из слоя LSTM в качестве входных данных и преобразует их в новое представление. Выходом этого слоя является вектор размера 128.
- 4) Последний слой – это еще один полносвязный слой Dense с размерностью выходного пространства 5 и функцией активации ‘softmax’. Этот слой принимает выход предыдущего слоя в качестве входа и производит распределение вероятностей по 5 возможным классам эмоционального тона входных сообще-

ний. В примере, представленном на Рис. 2, указаны именно 5 классов эмоционального тона, так как это значение разнится в зависимости от используемого набора обучающих данных; в данном примере показана модель для обучения на данных с 5-ю классами эмоционального тона сообщений.

BiLSTM. В качестве еще одной архитектуры модели нейронной сети была выбрана модель BiLSTM [19] (Двунаправленная LSTM) (см. Рис. 3). Она принимает на вход последовательность данных и последовательно обрабатывает ее слева направо и справа налево, чтобы отслеживать зависимости в обоих направлениях в контексте задачи.

1) Первым слоем модели аналогично предыдущим используемым архитектурам является слой 'Embedding' векторных представлений слов;

2) Далее создаются два LSTM «канала», которые складываются после прохождения через сверточные и рекуррентные слои. Как сказано выше, это рекуррентные слои нейронной сети, которые обрабатывают последовательность числовых векторов, выводимых слоем Embedding (только в данной случае – с двух сторон; имеет 128 нейронов и коэффициент 'dropout', равный 0,2 для предотвращения переобучения). Имеет по 6 нейронов и коэффициент 'dropout', равный 0,3, для предотвращения переобучения;

3) Далее входная последовательность в двух каналах обрабатывается в сверточных слоях с различными размерами ядер (2, 3, 4, 5, 6 и 8), чтобы обнаружить в тексте различные паттерны. Путем прохождения через Conv1D и Dropout (защита от переобучения) слои с различными значениями параметра kernel (как было указано выше, 2, 3, 4, 5, 6 и 8);

4) Далее сверточные слои объединяются в LSTM-слои, и выполняется конкатенация выходов LSTM-слоя и сверточно-рекуррентного слоя, а затем производится понижение размерности через слои MaxPooling1D и Flatten;

5) Наконец, на выходном слое сети используется полносвязный Dense-слой с размерностью выхода 5 (по количеству классифицируемых классов), использующий функцию активации 'softmax', который преобразует входы в вероятности для каждого класса, где модель должна выбрать один из пяти классов предсказания.

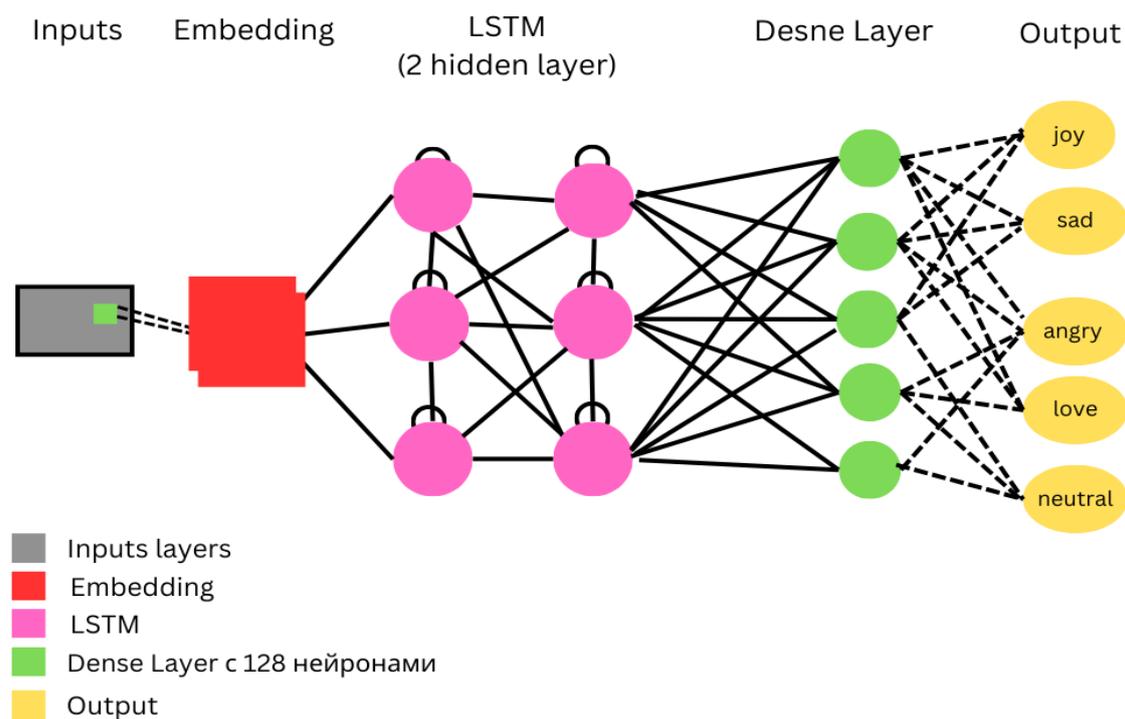


Рис. 3. Архитектура модели BiLSTM

Результаты, полученные в рамках исследования

Получены следующие результаты, отраженные в Таблице 1. В качестве метрик качества для сравнения сетей выбрана validation accuracy (доля правильных ответов алгоритма). Зеленым цветом в таблице 1 выделено самое большое значение точности, полученное в модели Bi-LSTM.

Таблица 1. Сравнительная таблица accuracy для обученных моделей.

Модель	accuracy	validation accuracy
CNN	0.9907	0.6002
LSTM	0.9109	0.6060
Bi-LSTM	0.6242	0.6149

Далее для сравнения каждой из использованных топологий нейронных сетей на Рисунках 4 и 5 показаны графики обучения для моделей CNN, обученных на 1-ом и 3-ем датасетах, на 5-й и 80-й эпохах соответственно.

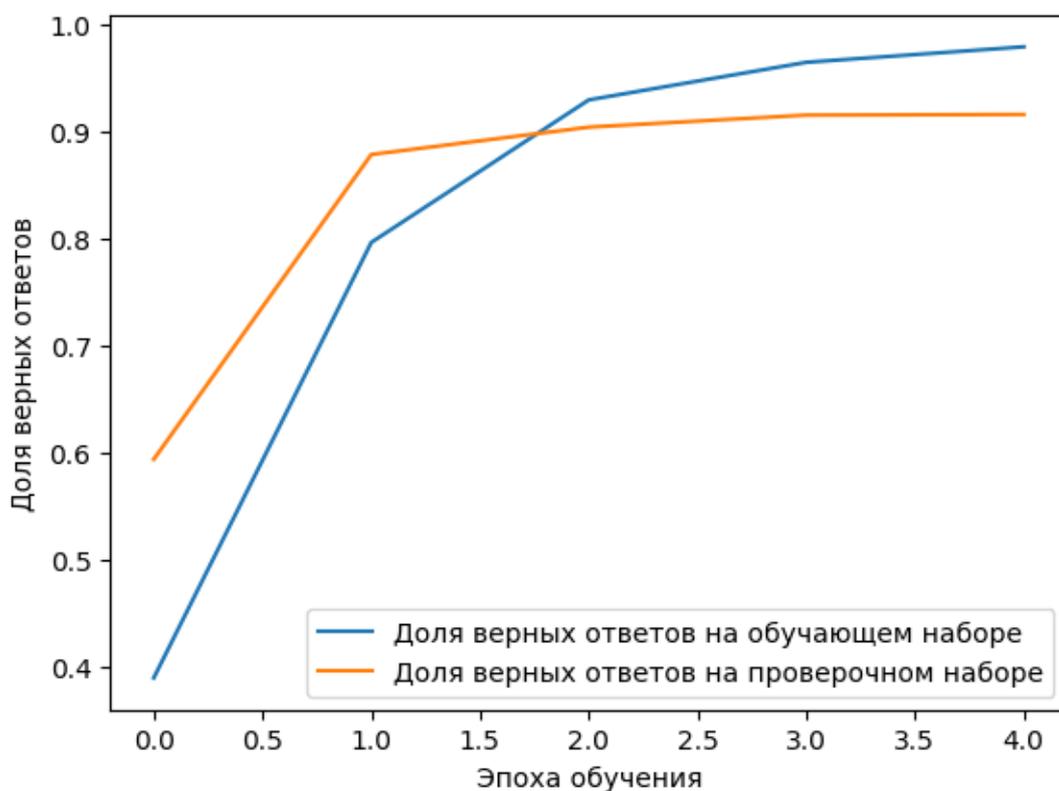


Рис. 4. График обучения модели CNN с 6-ю классами эмоций

На Рис. 5 приведен явный пример «переобучения» модели, так как прогресс в обучении останавливается примерно на 10-ой эпохе, и дальнейшее обучение было бессмысленно. На Рис. 6 показан график обучения на 20-ти эпохах – видим, что ассигасу растет достаточно быстро, но при этом валидационное ассигасу падает, что опять же может говорить о переобучении.

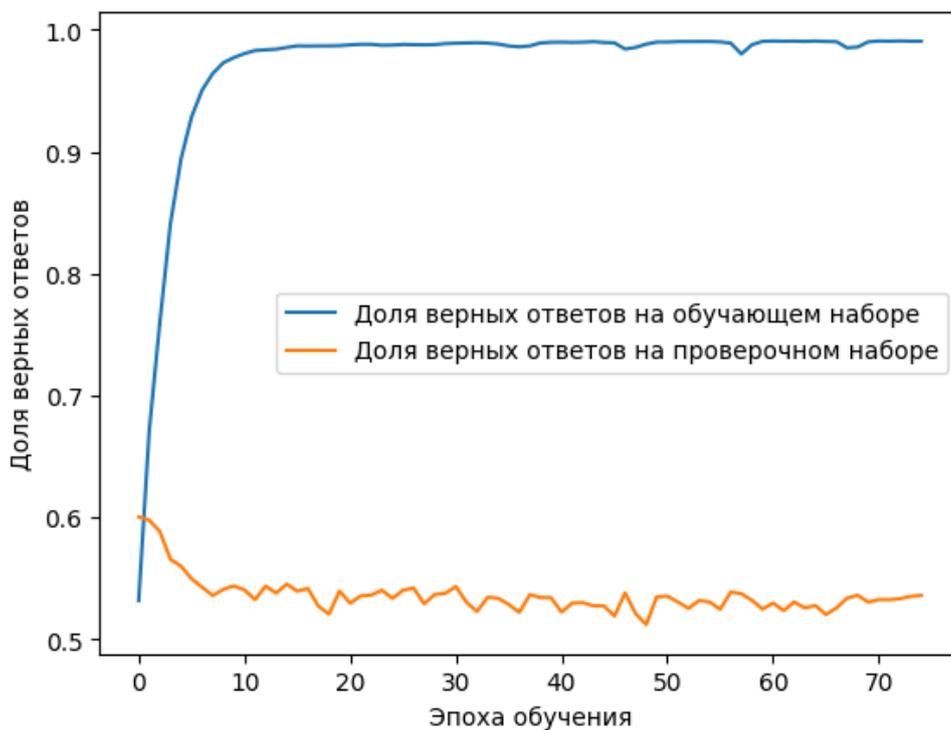


Рис. 5. График обучения модели CNN с-ю 5 классами эмоций

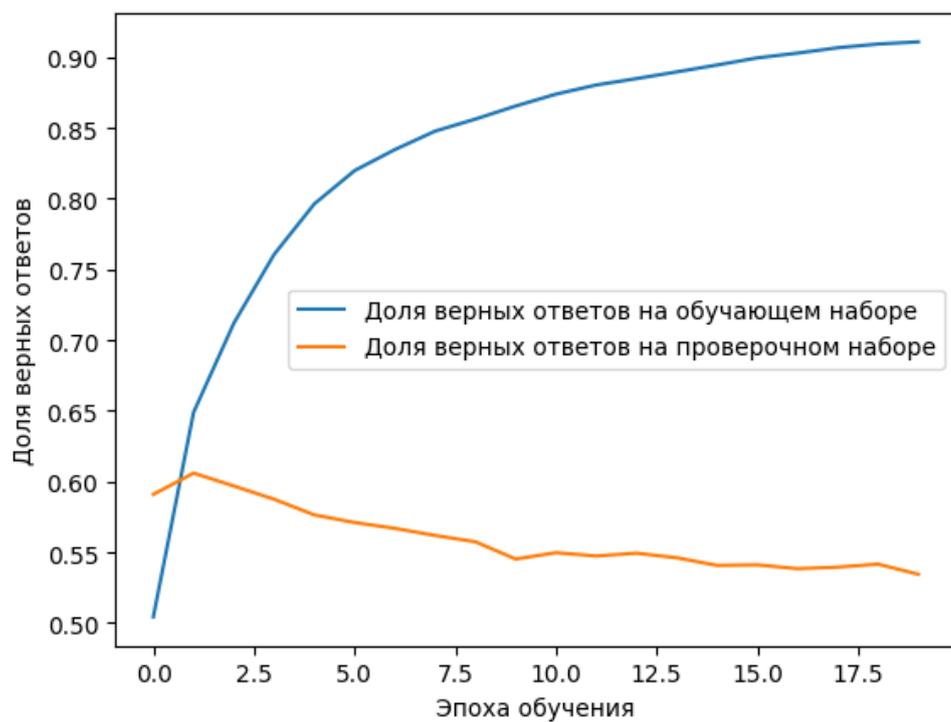


Рис. 6. График обучения модели LSTM с 5-ю классами эмоций

Самой удачной в динамике обучения оказалась модель Bi-LSTM. На Рис. 7 можно проследить изменение значений метрики точности, которая плавно возрастает и на обучающей, и на валидационной выборках. Как и ожидалось, функция потерь при этом снижается и для валидационной, и для обучающей выборок (Рис. 8).

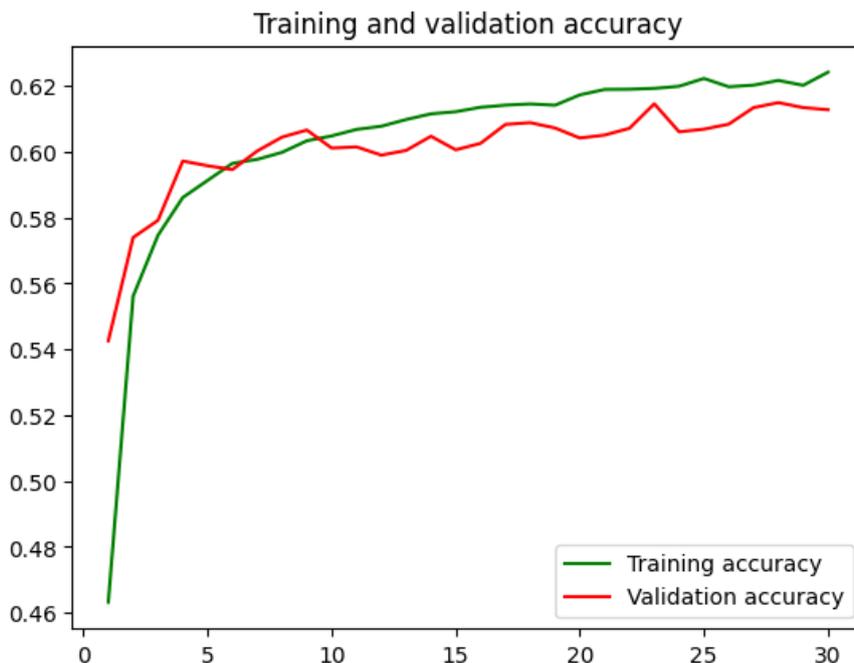


Рис. 7. График обучения модели BiLSTM с 5-ю классами эмоций

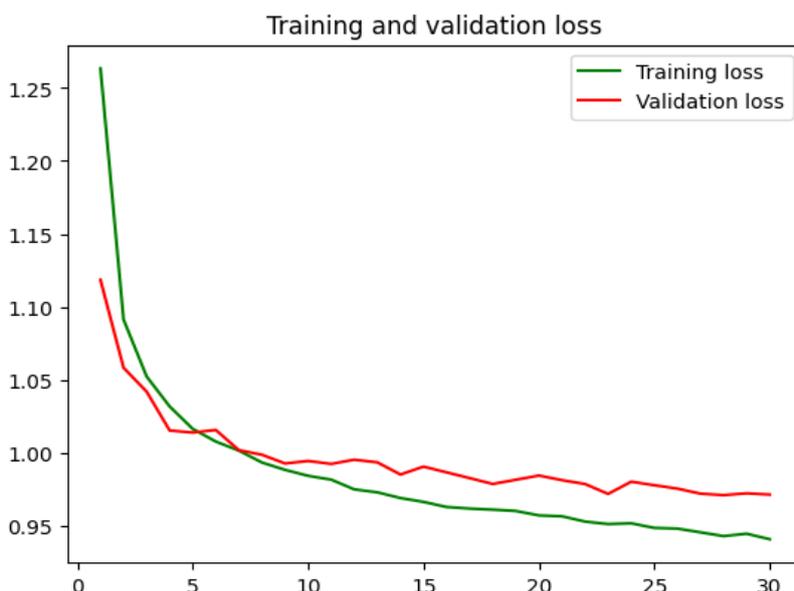


Рис. 8. График функции потерь модели BiLSTM с 5-ю классами эмоций

Реализация telegram-бота

В результате проведенной работы был создан telegram-бот (Рис. 9), который может распознавать основные эмоции по введенному сообщению. Бот был создан с помощью фреймворка aiogram. Бот работает на английском языке, т. к. модели обучались на датасетах на английском языке.



Рис. 9. а) Пример предсказаний моделью на коротких сообщениях моделью BiLSTM; б) Пример предсказаний моделью на коротких сообщениях моделью с CNN

Эксперименты с ботом проводились с использованием трех моделей нейронных сетей (CNN, LSTM и Vi-LSTM). После этого мы сравнили результаты, которые выдает бот: в следующей таблице показаны различия в результатах для каждой нейросети. В первой модели с CNN бот распознавал 6 эмоций (грусть, сча-

стью, радость, удивление, страх и любовь) (Рис. 9а), а вторая модель с использованием Bi-LSTM распознавала только 5 эмоций (грусть, счастье, радость, нейтральный и любовь) (Рис. 9б), но с большей точностью предсказаний. Также с использованием модели CNN и датасета с 2-мя классами эмоций была получена модель, которая достаточно точно определяла эмоциональный тон сообщения, но использовала лишь 2 класса эмоционального тона сообщений – positive и negative, чего недостаточно для полноценной реализации поставленной задачи

Таблица 2. Пример классификации предложений моделью CNN, LSTM и BiLSTM

	BiLSTM	LSTM	CNN
I feel bored	Sad 😞	Sad	anger
I am sad today	Sad 😞	Sad	sadness
I don't know what I should do	Neutral 😐	Neutral	fear
I have an exam	Neutral 😐	Sad	fear
I got an excellent deal	Happy 😊	Happy	fear
I like summer	Neutral 😐	Neutral	anger
I like food	Sad 😞	Sad	anger
I am happy	Happy 😊	Happy	joy
I like funny videos	Happy 😊	Happy	surprise
I forgot about the exam	Neutral 😐	Neutral	fear
I have a good friend	Neutral 😐	Neutral	joy
This professor has good ideas	Happy 😊	Happy	joy

ЗАКЛЮЧЕНИЕ

Результатом работы стал Telegram-бот, предсказывающий эмоциональный тон сообщения, введенного пользователем. Так как в результате анализа различных моделей наибольшее валидационное ассурасу было получено при использовании архитектуры Bi-LSTM, к боту в итоге была подключена именно данная модель, обученная на 3-ем наборе данных. Как можно увидеть на Рис. 9, модель с данной архитектурой лучше предсказывает короткие фразы, поданные на вход. Наши результаты показывают, что выбор модели нейронной сети и набора данных может оказать значительное влияние на производительность нейронной сети для решения задачи распознавания эмоционального тона.

СПИСОК ЛИТЕРАТУРЫ

1. Gosai D. D., Gohil H. J., Jayswal H. S. A Review on a Emotion Detection and Recognition from Text Using Natural Language Processing. URL: http://www.ripublication.com/ijaer18/ijaerv13n9_33.pdf
2. Acheampong F. A., Nunoo-Mensah H., Wenyu Chen. Transformer models for text-based emotion detection: a review of BERT-based approaches. URL: <https://link.springer.com/article/10.1007/s10462-021-09958-2>
3. Naresh S. S., Khethawat S. Emotion Detection from Text URL: <https://arxiv.org/abs/1205.4944>
4. Baccianella S., Esuli A., Sebastiani F. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining // Lrec. 2010. P. 2200–2204
5. Akbik A., Blythe D., Vollgraf R. Contextual string embeddings for sequence labeling // Proceedings of the 27th international conference on computational linguistics. 2010. P. 1638–1649.
6. Keras. URL: <https://keras.io/>
7. Tensorflow. URL: <https://www.tensorflow.org/>
8. Numpy. URL: <https://numpy.org/>
9. Pandas. URL: <https://pandas.pydata.org/>
10. Aiogram. URL: <https://aiogram.dev>
11. Asyncio. URL: <https://docs.python.org/3/library/asyncio.html>
12. Aiohttp. URL: <https://docs.aiohttp.org/en/stable/index.html>

13. GloVe: Global Vectors for Word Representation.

URL: <https://nlp.stanford.edu/projects/glove/>

14. Google Collab. URL: <https://colab.research.google.com>

15. Jupyter Notebook.

URL: <https://docs.jupyter.org/en/latest/install/notebook-classic.html>

16. Multi-Class Text Emotion Analysis.

URL: <https://github.com/kanchitank/Text-Emotion-Analysis>

17. Sentiment140 dataset with 1.6 million tweets.

URL: <https://www.kaggle.com/datasets/kazanov/sentiment140>

18. Twitter Emotion Classification.

URL: <https://www.kaggle.com/code/shtrausslearning/twitter-emotion-classification?scriptVersionId=126983369>

19. BiLSTM.

URL: <https://www.kaggle.com/code/mlwhiz/bilstm-pytorch-and-keras>

SOLVING THE PROBLEM OF CLASSIFYING THE EMOTIONAL TONE OF A MESSAGE WITH DETERMINING THE MOST APPROPRIATE NEURAL NETWORK ARCHITECTURE

D. I. Bagautdinov¹ [0000-0003-4673-3253], R. Salman² [0000-0002-1552-4910],

V. A. Alekseev³ [0000-0002-2208-9515], R. M. Usmonov⁴ [0009-0002-1289-5465]

¹⁻⁴*Institute of Information Technologies and Intelligent Systems, Kazan (Volga region) Federal University, ul. Kremlyovskaya, 35, Kazan, 420008*

¹d4niskabag@yandex.ru, ²Riham.salman.98@gmail.com, ³aleksvlad99@gmail.com,

⁴rustam190401@mail.ru

Abstract

To determine the most effective approach for solving the task of classifying the emotional tone of a message, we trained selected neural network models on various sets of training data. Next, based on the performance metric of the percentage of correctly classified responses on a test data set, we compared combinations of training

data sets and various models trained on them. During the writing of this article, we trained four neural network models on three different sets of training data. By comparing the accuracy of the responses from each model trained on different training data sets, conclusions were drawn regarding the neural network model best suited for solving the task at hand.

Keywords: *NLP, sentiment detection, neural networks, comparison of neural network models, LSTM, CNN, BiLSTM.*

REFERENCES

1. Gosai D. D., Gohil H. J., Jayswal H. S. A Review on a Emotion Detection and Recognition from Text Using Natural Language Processing. URL: http://www.ripublication.com/ijaer18/ijaerv13n9_33.pdf
2. Acheampong F. A., Nunoo-Mensah H., Wenyu Chen. Transformer models for text-based emotion detection: a review of BERT-based approaches. URL: <https://link.springer.com/article/10.1007/s10462-021-09958-2>
3. Naresh S. S., Khethawat S. *Emotion Detection from Text* URL: <https://arxiv.org/abs/1205.4944>
4. Baccianella S., Esuli A., Sebastiani F. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining // *Lrec. 2010. P. 2200–2204*
5. Akbik A., Blythe D., Vollgraf R. Contextual string embeddings for sequence labeling // *Proceedings of the 27th international conference on computational linguistics. 2010. P. 1638–1649.*
6. Keras. URL: <https://keras.io/>
7. Tensorflow. URL: <https://www.tensorflow.org/>
8. Numpy. URL: <https://numpy.org/>
9. Pandas. URL: <https://pandas.pydata.org/>
10. Aiogram. URL: <https://aiogram.dev>
11. Asyncio. URL: <https://docs.python.org/3/library/asyncio.html>
12. Aiohttp. URL: <https://docs.aiohttp.org/en/stable/index.html>
13. GloVe: Global Vectors for Word Representation.
URL: <https://nlp.stanford.edu/projects/glove/>
14. Google Collab. URL: <https://colab.research.google.com>

15. Jupyter Notebook.

URL: <https://docs.jupyter.org/en/latest/install/notebook-classic.html>

16. Multi-Class Text Emotion Analysis.

URL: <https://github.com/kanchitank/Text-Emotion-Analysis>

17. Sentiment140 dataset with 1.6 million tweets.

URL: <https://www.kaggle.com/datasets/kazanova/sentiment140>

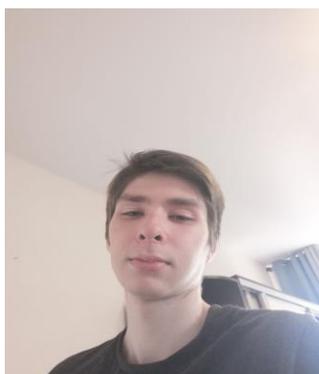
18. Twitter Emotion Classification.

URL: <https://www.kaggle.com/code/shtrausslearning/twitter-emotion-classification?scriptVersionId=126983369>

19. BiLSTM.

URL: <https://www.kaggle.com/code/mlwhiz/bilstm-pytorch-and-keras>

СВЕДЕНИЯ ОБ АВТОРАХ



БАГАУТДИНОВ Данис Ильмасович – студент магистратуры Института ИТИС КФУ. Сфера научных интересов – обработка естественных языков, машинное обучение, нейронные сети, алгоритмы и структуры данных.

Danis Ilmasovich BAGAUTDINOV – Master's student at the Institute of ITIS KFU. Research interests are natural language processing, machine learning, neural networks, algorithms and data structures.

email: d4niskabag@yandex.ru

ORCID 0000-0003-4673-3253



САЛМАН Рухам – студентка магистратуры Института ИТИС КФУ. Сфера научных интересов – обработка естественных языков, машинное обучение, нейронные сети.

Riham SALMAN – Master's student at the Institute of ITIS KFU. Research interests – natural language processing, machine learning, neural networks.

email: Riham.salman.98@gmail.com

ORCID 0000-0002-1552-4910



АЛЕКСЕЕВ Владислав Алексеевич – студент магистратуры Института ИТИС КФУ. Сфера научных интересов – обработка естественных языков, машинное обучение, нейронные сети, веб-разработка, разработка эффективных интерфейсов.

Vladislav Alekseevich ALEKSEEV – Master's student at the Institute of ITIS KFU. Research interests – natural language processing, machine learning, neural networks, web development, development of effective interfaces.

email: aleksvlad99@gmail.com

ORCID 0000-0002-2208-9515



УСМОНОВ Рустамджон Муроджонович – студент магистратуры Института ИТИС КФУ. Сфера научных интересов – обработка естественных языков, машинное обучение, нейронные сети, веб-разработка, разработка эффективных интерфейсов.

Rustamdzhon Murodzhonovich Usmonov – Master's student at the Institute of ITIS KFU. Research interests – natural language processing, machine learning, neural networks, web development, development of effective interfaces.

email: rustam190401@mail.ru

ORCID 0009-0002-1289-5465

Материал поступил в редакцию 23 мая 2023 года