Использование искусственных нейронных сетей для решения задачи классификации текста

Е.С. Попова[1], В.Г. Спицын[1], Ю.А. Иванова[1]
esp9@tpu.ru|spvg@tpu.ru|jbolotova@tpu.ru
[1]Национальный Исследовательский Томский Политехнический Университет, Томск, Россия

Статья посвящена нейросетевым алгоритмам классификации текстов. Актуальность данной темы обусловлена постоянно растущим объемом информации в интернете и потребностью в ней ориентироваться. В данной работе помимо алгоритма классификации, так же приводится описание способов предобработки и векторизации текста, данные этапы являются стартовой точкой для большинства NLP задач и делают нейросетевые алгоритмы эффективным на небольших наборах данных. В работе в качестве набора данных для обучения и тестирования нейронной сети будет использоваться выборка состоящая из 50 000 обзоров фильмов IMDB на английском языке. Для решения поставленной задачи был использован подход основанный на использовании свёрточной нейронной сети. Максимально достигнутая точность для тестовой выборки составила 90.16%.

Ключевые слова: *понимание текстов, обработка естественных языков, сверточные нейронные сети, классификация текстов.*

Using artificial neural networks to solve text classification problems

E.S. Popova[1], V.G. Spitsyn[1], Yu.A. Ivanova[1]
author1@domain|author2@domain|author3@domain
[1]National Research Tomsk Polytechnic University, Tomsk, Russia

The article is devoted to neural network text classification algorithms. The relevance of this topic is due to the ever-growing volume of information on the Internet and the need to navigate it. In this paper, in addition to the classification algorithm, a description is also given of the methods of text preprocessing and vectorization, these steps are the starting point for most NLP tasks and make neural network algorithms efficient on small data sets. In the work, a sampling of 50,000 English IMDB movie reviews will be used as a dataset for training and testing the neural network. To solve this problem, an approach based on the use of a convolutional neural network was used. The maximum achieved accuracy for the test sample was 90.16%.

Keywords: *text comprehension, natural language processing, convolutional neural networks, text classification.*

1. Введение

Задачи обработки естественных языков (natural language processing, NLP) становятся все более актуальными в связи с постоянно растущим объемом информации в интернете и потребностью в ней ориентироваться. Одной из широко используемых задач по обработке естественного языка и контролируемому машинному обучению является классификация текстов. При этом для обучения классификатора используется маркированный набор данных, содержащий текстовые документы и их метки.

Целью является автоматическая классификация текстовых документов по одной или нескольким предопределенным категориям. Ниже приведены частные случаи задачи классификации текстов:

1. Понимание настроения аудитории из социальных сетей.
2. Обнаружение спама.
3. Автоматическая пометка запросов клиентов.
4. Категоризация новостных статей на предопределенные темы.

Классификация текста широко используется в сентиментальном анализе (IMDB, классификация обзоров YELP), анализе фондового рынка, в «умном» ответе по электронной почте GOOGLE. Сфера обработки естественных языков является активно развивающейся областью исследований, как в академических кругах, так и в промышленности.

На сегодняшний день помимо классических алгоритмов интеллектуального анализа текстов, большое распространение получили методы, основанные на глубоком обучении нейронных сетей (deep learning), которые предлагают гибкий, универсальный и обучаемый подход для представления окружающей среды в виде визуальной и лингвистической информации.

Далее приведены нейросетевые архитектуры, которые могут быть применены для решения задачи классификации текстов:

1. Сверточная нейронная сеть (Convolutional Neural Network, CNN).
2. Рекуррентная нейронная сеть (Recurrent Neural Network, RNN).
3. Иерархическая сеть внимания (Hierarchical Attention Network, HAN).

В данной статье для решения поставленной задачи будут использованы сверточные нейронные сети, которые впервые были представлены в 1998 году французским исследователем Яном Лекуном, как развитие модели неокогнитрона и предназначены для эффективного распознавания изображений.

CNN обычно используются в компьютерном зрении, однако в последнее время они стали активно применятся к различным задачам NLP и исходя из статьи [2] от коллектива авторов из Intel и Carnegie-Mellon University подходят для этого даже лучше, чем рекуррентные нейронные сети, которые безраздельно властвовали в этой области на протяжении последних лет.

В данной статье для решения поставленной задачи используется фреймворк машинного обучения Keras и язык программирования Python.

2. Обучающая выборка

В качестве набора данных для обучения и тестирования нейронной сети будет использоваться набор данных, состоящий из 50 000 обзоров фильмов IMDB на

английском языке, специально отобранных для анализа тональности.

Тональность в выборке двоичная, т.е. IMDB – рейтингу менее 5 была сопоставлена оценка 0, а рейтингу не менее 7 – оценка 1. На каждый фильм приходится не более 30 обзоров. Также необходимо отметить, что в выборке отсутствуют обзоры, имеющие рейтинги 5 или 6, так как их нельзя однозначно отнести к положительным или отрицательным отзывам, следовательно, они не вписываются в бинарную модель классификации. Все обзоры в выборке перемешаны в случайном порядке. Набор данных имеет следующую структуру:

1. Id – уникальный идентификатор каждого отзыва.
2. Sentiment – настроение обзора (1 за положительные отзывы и 0 за отрицательные отзывы).
3. Review – текст обзора.

3. Предобработка текста

Предобработка текста позволяет уменьшить исходное пространство признаков без потери полезной информации. Это положительно сказывается на качестве понимания и скорости обучения выбранного алгоритма.

Ниже приведены основные методы морфологической и синтаксической предобработки текста, входящие в состав лингвистического анализа, который является базовым для многих современных подходов к интеллектуальному анализу текста, и включает в себя следующие этапы [3]:

1. Токенизация – это самый первый шаг при обработке текста. Заключается в разбиении длинных строк текста на более мелкие: абзацы делим на предложения, предложения – на слова.
2. Нормализация – для качественной обработки текст должен быть нормализированным. Все слова приводятся к одному регистру, удаляются знаки пунктуации, расшифровываются сокращения, числа приводятся к их текстовому написанию и т.д. Нормализация необходима для унификации методов обработки текста.
3. Стэмминг – это устранение придатков к корню, то есть отделение суффикса, приставки, окончания и приведение слова к основе.
4. Лемматизация – близка к стеммингу. Отличие в том, что лемматизация приводит слово к смысловой канонической форме слова (инфинитив для глагола, именительный падеж единственного числа – для существительных и прилагательных). Например, зафрахтованный – фрахтовать, ценами – цена, лучший – хороший.
5. Удаление стоп-слов. Стоп-слова – слова, которые не несут никакой смысловой нагрузки. Их еще называют шумовыми словами. Например, в английском языке это артикли, в русском – междометия, союзы и т.д.

В данной работе для достижения лучшего качества классификации будут использованы некоторые из перечисленных выше методов.

Необходимо, чтобы в исходных текстах содержалось как можно меньше данных, не несущих полезной информации, например, в выборке встречаются HTML теги, такие как </br>, аббревиатуры и пунктуация.

Для удаления HTML тегов использовалась Python библиотека BeautifulSoup Package. Для удаления символов пунктуации был использован пакет регулярных выражений. Далее вся выборка была приведена к нижнему регистру.

Также в выборке присутствуют стоп-слова, такие как «a», «and» «is» «the» и другие, которые не несут смысловой нагрузки. Для их удаления использовалась библиотека Natural Language Toolkit (NLTK).

4. Векторное представление слов

Векторное представление считается стартовой точкой для большинства NLP задач и делает глубокое обучение эффективным на небольших наборах данных. Также оно лежит в основе многих систем обработки естественного языка, таких как Amazon Alexa, Google translate и т.д.

Данный метод соотносит текстовому слову некоторый числовой вектор фиксированной размерности. Вектора строятся таким образом, что встречающиеся в схожих контекстах слова имеют схожие векторные представления.

Векторные представления слов обладают разнообразными полезными свойствами, и могут быть использованы следующим образом:

1. Для поиска синонимов или опечаток в поисковых запросах.
2. Отражения семантической близости между словами.
3. В качестве признаков для решения следующих задач:
   - 3.1. Выявление именованных сущностей;
   - 3.2. Тэгирование частей речи.
   - 3.3. Машинный перевод.
   - 3.4. Кластеризация документов.
   - 3.5. Ранжирование документов.
   - 3.6. Анализ тональности текста.

Ниже представлены алгоритмы получения векторных представлений слов:

1. One-hot encoding.
2. SVD.
3. Topic modeling.
4. Word2vec, GloVe, FastText, StarSpace.

Рассмотрим подробнее технику векторного представления GloVe [9] от Стэнфордского университета, которая часто используются для задач NLP.

GloVe – предназначена для статистической обработки больших массивов текстовой информации. GloVe собирает статистику по совместному появлению слов в фразах, после чего методами нейронных сетей решает задачу снижения размерности и выдает на выходе компактные векторные представления слов, в максимальной степени отражающие отношения этих слов в обрабатываемых текстах.

Для более удобной работы с векторными представлениями в данной работе была использована предварительно обученная модель GloVe, которая представляет собой файл, содержащий токены и связанные с ними вектора слов. В частности, будет использована 100-мерная версия модели GloVe состоящей из 400 тыс. слов, рассчитанная на данных английской Википедии за 2014 год С 6 миллиардами токенов.

5. Сверточные нейронные сети для задачи классификации текстов

Существует несколько подходов с использованием сверточных нейронных сетей для задачи классификации текстов. В данной работе был применен подход, основанный на кодировании слов, описанный в статье [3].

В этом подходе каждому слову в тексте сопоставляется вектор фиксированной длины. Затем из полученных векторов для каждого фрагмента выборки составляется двумерная матрица, которая подается на вход сверточной нейронной сети.

На рисунке 1 приведен пример сверточной нейронной сети с использованием кодирования слов.

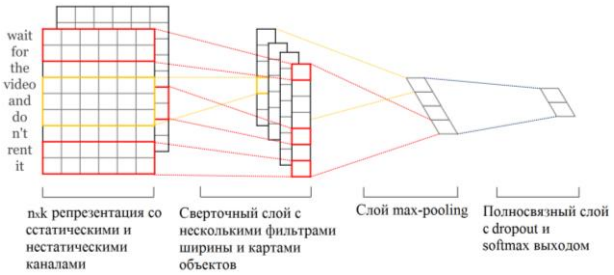

Рис. 1. Пример CNN с использованием кодирования слов.

В таблице 1 представлена конфигурация сверточной нейронной сети с использованием алгоритма GloVe, описанного в разделе 4, для классификации данных.

Получившаяся конфигурация состоит из 3 сверточных слоев, каждый из которых содержит 128 карт признаков и имеет окно свертки 5×5 и 3 подвыборочных слоев с размером окна подвыборки 5×5 и 35×35 на последнем слое.

Сеть также включает в себя слой решейпинга и 2 полносвязных слоя. Функция активации на всех слоях, кроме последнего, – Relu, на последнем – Softmax.

Таблица 1. Конфигурация сверточной нейронной сети

Тип слоя	Функция активации	Кол-во настраиваемых параметров
Входной слой	–	0
Слой векторизации	–	8420100
Слой свертки, кол-во карт признаков:128, ядро свертки: 5x5	ReLU	64128
Слой подвыборки, кол-во карт признаков: 128, окно подвыборки: 5x5	–	0
Слой свертки, кол-во карт признаков:128, ядро свертки: 5x5	ReLU	82048
Слой подвыборки, кол-во карт признаков: 128, окно подвыборки: 5x5	–	0
Слой свертки, кол-во карт признаков:128, ядро свертки: 5x5	ReLU	82048
Слой подвыборки, кол-во карт признаков: 128, окно подвыборки: 35x35	–	0
Вспомогательный слой решейпинга	–	0
Полносвязаный слой, кол-во нейронов:128	ReLU	16512
Полносвязаный слой, кол-во нейронов: 2	Softmax	258
Общее кол-во настраиваемых параметров		8 665 094

6. Результаты тестирования

Ходе экспериментов были получены следующие результаты. На рисунках 2 и 3 показаны графики изменения точности распознавания и ошибки в ходе обучения сверточной нейронной сети с использованием алгоритма GloVe, для 10 эпох обучения и размером мини-выборки равном 128. Из рисунка видно, что максимальная точность распознавания на тестовой выборке достигается на 7 эпохе обучения и равна 88.88%.

Исходя из полученных результатов, можно сделать вывод, что выбранное количество эпох обучения является избыточным, и дальнейшее увеличение точности распознавания может быть достигнуто за счет более точной настройки гиперпараметров сети, таких как размер мини-выборки, размер окна свертки, количество карт признаков.

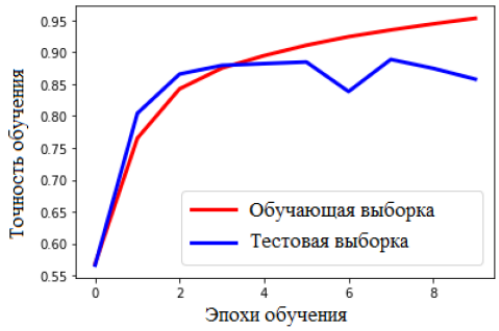

Рис. 2. Изменение точности распознавания тональности текста для CNN с размером мини-выборки 128

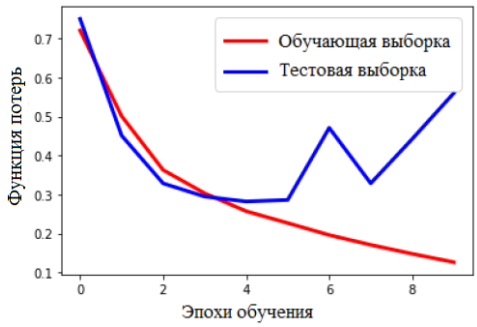

Рис. 3. Изменение ошибки распознавания тональности текста для CNN с размером мини-выборки 128

На рисунках 4 и 5 приведены результаты обучения сети для размера мини-выборки 64 и 256 для 10 эпох обучения.

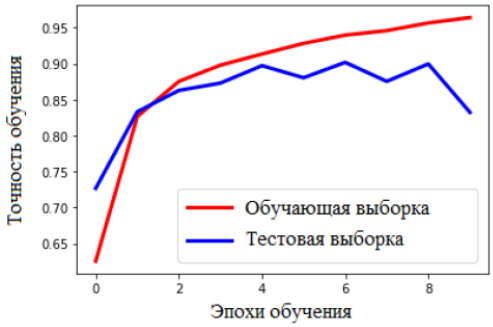

Рис. 4. Изменение точности распознавания тональности текста для CNN с размером мини-выборки 64

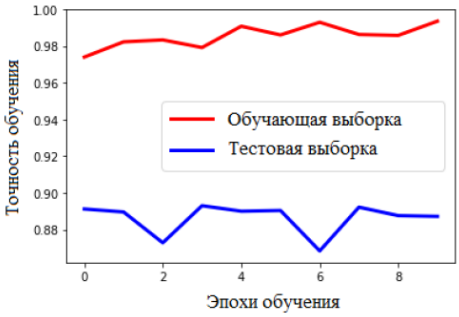

Рис. 5. Изменение точности распознавания тональности текста для CNN с размером мини-выборки 256

Из полученных данных видно, что уменьшение размера мини-выборки позволило повысить точность классификации до 90%, в то время как увеличение дало более стабильный результат на протяжении всех эпох обучения, однако не дало значительного прироста в точности.

7. Результаты тестирования

В результате проведенного исследования были выявлены основные группы задач NLP, рассмотрены методы предобработки и векторизации текстов. Также в ходе исследования была изучена возможность применения сверточных нейронных сетей для задачи классификации текстов.

В ходе обучения и тестирования сети обучающая и тестовая выборки были разделены на 20000 и 5000 образцов соответственно. Для тестовой выборки максимально достигнутая точность составила 90.16%.

Значение точности вычисляется по формуле:

$$R = \frac{n}{N} = \frac{4508}{500} = 0.9016,$$

где R – точность распознавания, n – количество правильно классифицированных текстов, N – количество элементов в выборке.

Можно сделать вывод, что сеть справилась с задачей определения эмоциональной тональности предложенных текстов, однако, исходя из полученных результатов, для достижения большей точности классификации в дальнейшем необходимо:

1. Более тонкая настройка гиперпараметров: размера мини-выборки, размера окна свертки, количества карт признаков.
2. Дополнительное улучшение предобработки текста.
3. Использование dropout-слоев.

8. Благодарности

Работа выполнена в рамках Программы повышения конкурентоспособности ТПУ при финансовой поддержке РФФИ в рамках научного проекта № 18-08-00977 А.

9. Литература

[1] Введение в анализ естественных языков / Учебно-методическое пособие / И.В. Смирнов, 2014 г.

[2] Спицын В.Г., Интеллектуальные системы: учебное пособие /В.Г. Спицын, Ю.Р. Цой; Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2012.–176 с.

[3] Федюшкин Н.А., Федосин С. А. Понятие, проблемы и разновидности интеллектуального анализа текста – Проблемы и достижения в науке и технике. Сборник научных трудов по итогам международной научно-практической конференции – № 3 – г. Омск, 2016 – 206 с.

[4] Хайкин С. Нейронные сети: полный курс. М.: Вильямс, 2006. - 1104 с.

[5] Bai, S., Kolter, J. Z., & Koltun, V. (2018). An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. arxiv.org/abs/1803.01271.

[6] Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), 1746–1751.

[7] LeCun, Y. Efficient BackProp in Neural Networks: Tricks of the trade / Y.LeCun, L. Bottou, G. Orr, K. Muller – Springer, 1998.

[8] LeCun, Y. Scaling learning algorithms towards AI / Y.LeCun, Y. Bengio – MIT Press, 2007.

[9] Pennington, J., Soche, R., D. Manning, C. GloVe: Global Vectors for Word Representation [Электронный ресурс] Точка доступа: https://nlp.stanford.edu/projects/glove.

[10] Zhang, X. Character-level convolutional networks for text classification / Xiang Zhang, Junbo Zhao, Yann LeCun // In Advances in Neural Information Processing Systems. - 2015. - Feb. - 649-657pp.

Об авторах

Спицын Владимир Григорьевич, д.т.н., профессор инженерной школы информационных технологий и робототехники Томского политехнического университета. E-mail spvg@tpu.ru.

Иванова Юлия Александровна, к.т.н., доцент инженерной школы информационных технологий и робототехники Томского политехнического университета. E-mail: jbolotova@tpu.ru.

Попова Екатерина Сергеевна, аспирант инженерной школы информационных технологий и робототехники Томского политехнического университета. E-mail: esp9@tpu.ru.