Ахметгалиев А.И., Гафаров Ф.М., Ситдикова Ф.Б.
Анализ тональности текстов с использованием нейросетевых моделей

*Казанский федеральный университет
(Россия, Казань)*

doi: 10.18411/sr-10-04-2019-69

idsp: sciencerrussia-10-04-2019-69

Аннотация

В статье рассматриваются методы решения задачи анализа тональности текстов на основе нейросетевых моделей обработки естественного языка. В статье рассмотрены методы, создающие векторное представление слов в n -мерном векторном пространстве, в основе которых лежат технологии «Word2Vec», «GloVe», «FastText». Подходы используются в задачах классификации, анализа тональности, исправления опечаток, рекомендательных системах. Представлены результаты сравнения классификаций в задаче анализа тональности многослойного персептрона, сверточной и рекуррентной нейронной сети, деревьев решений (случайный лес), метода опорных векторов (SVM), наивного байесовского классификатора (NB), метода k -ближних соседей (K-NN). Результаты классификации представлены для трех наборов данных: сообщения в твиттере, отзывы на различные товары и услуги, русскоязычные новости.

Ключевые слова: анализ тональности, Word2Vec, GloVe, FastText, векторное представление слов, рекуррентные нейронные сети, сверточные нейронные сети.

Введение

Анализ тональности текста представляет из себя класс методов обработки естественного языка, предназначенных для автоматизированного распознавания в текстах эмоционально окрашенной лексики и эмоциональной оценки (положительной, негативной, нейтральной) [1]. В данной статье анализ тональности рассматривается как частный случай классификации документов. Классификация документов — одна из задач информационного поиска, заключающаяся в отнесении документа к одной из нескольких категорий на основании содержания документа [2].

Определение анализа тональности текстов имеет важное практическое применение в различных направлениях человеческой деятельности: ведении бизнеса, определении политических и экономических стратегий, в оценке качества выпускаемой продукции и услуг в социальных сетях, блогах целевых сайтов. Понимание того, насколько негативно или позитивно люди реагируют на произведенный товар или услугу, позволяет оценивать будущую успешность продаж на рынке, а также анализировать качество проведенных PR-компаний. Многим крупным производителям необходимо бывает оценивать, как их бренд воспринимается потребителями.

Анализ тональности текста также можно активно использовать при определении отношения населения к различным проводимым правительством реформам или мнения электората по поводу того или иного кандидата в руководящую должность. Таким образом, анализ тональности текстов — мощнейший инструмент в руках ученых, бизнесменов и политиков.

Методы

Методы машинного обучения рассматривают анализ тональности как задачу классификации текста. В этих методах извлекаются такие функции, как

униграммы, биграммы, вложения слов сначала из текста, или векторизуются слова, а затем документы. Элементы передаются в такие модели классификации, как SVM, деревья решений или глубокие нейронные сети (CNN, RNN). Точность классификации сильно зависит от размера и «качества» обучающей выборки данных. В данной статье упор делается на методы машинного обучения с учителем [3].

Векторные представления слов.

Word2Vec. Технология (набор алгоритмов) Word2Vec от Google была использована для решения проблемы классификации документов, созданной Томасом Миколовым. На вход Word2Vec подается корпус текста, на выходе векторное представление слов. Векторы похожих по значению слов имеют скалярное произведение близкое к единице. Таким образом, Word2Vec позволяет находить слова, близкие по смыслу. В качестве корпуса в работе использовались следующие корпуса: "национальный корпуса русского языка" (788 миллионов слов), русскоязычные новости за декабрь 2018 года (5 миллиардов слов), Википедии (2.6 миллиарда слов). Технология основана на том принципе, что слова, которые часто встречаются рядом (в одном и том же контексте), являются семантически близкими друг к другу. Есть две модели обучения нейронной сети для CBOW и Skip-грамм. Алгоритм проходит через весь текст «скользящим окном» (указана ширина окна) и согласно первой модели прогнозирует центральное слово в окне на основе контекстных, согласно второй модели центральное слово прогнозирует центральное слово [4].

Рис. 1 Архитектуры Word2Vec CBOW и Skip-gram

Для задачи определения анализа тональности текстов были обучены вектора Word2Vec на базе неразменных твитов объемом 17 639 674 сообщений. Были найдены похожие слова и добавлены вектора-слов в матрицу. С помощью алгоритма t-SNE - (t-distributed stochastic neighbor embedding) был понижена размерность векторов до 2.

Рис.2 Визуализация векторов слов Word2Vec схожих со словами «отрицание», «продукт», «популярный», «нравится».

Из рис. 2 визуализации видно, что слово «нравится» и «отрицание» имеют большее косинусное расстояние чем, «популярный» и «нравится». Топ 5 слов, близких к слову «интересный»: [('неплохой', 0.7640383243560791), ('забавный', 0.7558714151382446), ('любопытный', 0.7503457069396973), ('познавательный', 0.7198209762573242), ('прикольный', 0.7184607982635498)]

FastText. FastText – способ построения векторного представления от компании facebook. В FastText, как и в модели Word2Vec, используются подходы skip-gram или cbow. Главная идея такая же, как и в Word2Vec: максимизировать вероятность (softmax) встретить центральное слово с контекстным или наоборот, пробега по корпусу с заданной шириной окна. Основное отличие FastText от word2vec или Glove — это то, что каждое слово представляется в виде набора символов n-грамм. Приведем пример при $n=3$, слово «школа» разбивается на следующие 3-граммы <шк, шко, кол, ола, ла>. Авторы FastText рекомендуют выбирать n от 3 до 6, оптимальное n зависит от языка, корпуса и задачи. Пусть у нас имеется словарь $G_w \subset \{1, \dots, G\}$ n-грамм, тогда мы связываем векторное представление z_g с каждым n-граммом g . И представляем слово как сумму векторных представлений его n-грамм [5]. Тогда близость двух слов в FastText определяется как

$$s(w, c) = \sum_{g \in G_w} z_g^T v_c$$

Обучение в FastText происходит быстрее, чем в модели word2vec.

Рис.3 Визуализация векторов слов FastText схожих со словами «технологии», «плохой», «средний», «нравится», «мрак».

Топ 5 слов, близких к слову «хороший» в FastText, обученном на новостном корпусе: [('отличный', 0.7849140167236328), ('неплохой', 0.7302890419960022), ('плохой', 0.7082219123840332), ('превосходный', 0.6944824457168579), ('прекрасный', 0.6733499765396118)]

GloVe. Global Vectors, представляет собой модель для распределенного векторного представления слов. Является алгоритмом обучения без учителя. Обучение выполняется на агрегированной глобальной частотной матрице слов (co-occurrence matrix), которая строится из корпуса. GloVe так же, как и Word2Vec позволяет находить слова близкие по смыслу, т.е вектора-слов близких по значению будут находиться рядом в векторном пространстве. На первом этапе строится так называемая co-occurrence матрица — матрица, показывающая насколько часто слова, встречаются друг с другом в данном корпусе. Где элемент матрицы X_{ij} количество появлений слова j со словом i в контексте. Модель GloVe обучается на ненулевых элементах матрицы. Чтобы заполнить данную матрицу необходим один проход по всему корпусу. Пусть $X_i = \sum_k X_{ik}$ - число раз когда любое слово встречалось рядом со словом i .

$P_{ij} = P(j | i) = \frac{x_{ij}}{x_i}$ — вероятность, что слово j появится в контексте слова i .

Нахождение векторов-слов u_i для $i = 1, W$, где W — размер словаря, заключается в решении задачи минимизации функции, представленной ниже [7].

$$J = \frac{1}{2} \sum_{i,j=1}^W f(P_{ij}) (u_i^T u_j - \log P_{ij})^2$$

В качестве алгоритма решения задачи минимизации используют метод стохастического градиентного спуска. По сравнению с Word2Vec обучается быстрее.

Рис.5 Визуализация векторов слов GloVe схожих со словами «красивый», «плохой», «большой», «нравится», «мебель».

Топ 5 слов, близких к слову «мебель» в GloVe, обученном на корпусе отзывов:
 [('мебели', 0.5776834487915039), ('интерьеры', 0.4908282160758972),
 ('гламурные', 0.4803837537765503), ('матрасы', 0.4794725179672241),
 ('дизайна', 0.4584319591522217)]

Классификация текстов по тональности.

Сначала необходимо получить векторное представление слов, используя какую-либо из технологий Word2vec, GloVe, FastText для каждого слова из словаря корпуса. Далее найти для каждого документа из выборки его средний вектор фиксированной размерности (т.е. просуммировав все вектора слов документа и разделив на количество слов документа). Полученные агрегированные вектора документов с соответствующими метками принадлежности к определенному классу подаются на обучение в классификатор. Далее на тестовой выборке определяется точность классификации. Но при нахождении среднего вектора не учитывается порядок слов в документе. Поэтому при суммировании векторов слов можно умножать их на какой-либо вес, например, TF-IDF термина. При классификации текста по тональности проводились эксперименты с обычным многослойным персептроном с двумя скрытыми слоями по 6 нейронов в слое. Размерность входного вектора равнялась размерности вектора документа. Число нейронов в выходном слое равнялось 2 или 3 в зависимости от того, на сколько классов тональности происходило разделение. В качестве метода обучения использовался метод стохастического градиентного спуска. Также использовались классические методы Data-mining, такие как метод опорных векторов SVM [8], Метод k-ближайших соседей (k-NN), Случайный лес (Random Forest), наивный байсовский классификатор (Naive Bayes). В качестве классификаторов использовались сверточные и рекуррентные нейронные сети. Сверточная сеть показывает хорошие результаты в распознавании изображений и на данный момент активно используется в задачах обработки естественного языка.

Рассмотрим, как используются CNN в задаче классификации документов. Пусть мы построили векторное представление слов для каждого термина из словаря используя одну из моделей Word2Vec, GloVe или FastText. Составляем из этих векторов-слов матрицу — так называемый embedding слой, для каждого документа из обучающей выборки. Так как CNN принимает на вход матрицу фиксированной длины, мы обрезаем наши документы до фиксированной длины. Если при анализе количество слов превышало высоту матрицы, оставшиеся слова отбрасывались. Длина документа подбирается так, чтобы покрывался большой процент всех слов текстов в сформированном корпусе. Таким образом, размерности матрицы $n \times k$, n — фиксированное количество слов в документе, k — размерность вектора. В работе были использованы различные архитектуры сверточных нейронных сетей. Одна из наиболее точных оказалась сеть, в которой использовались фильтры размерности от 2-5 по 10 сверточных слоев для каждого фильтра. После сверточных слоев применялась операция max-pooling. Далее осуществлялась конкатенация слоев в общий вектор, который подавался на вход в полносвязный слой из 40 нейронов [9,10].

Рис.5. CNN. При классификации используются свертки размерности 1

Использовалось 10 эпох обучения и дополнительные 5 с весами эпохи, в которой сеть получила наибольший f1-score. Объем обучающей и тестовой выборки был 80 и 20 процентов соответственно. И размер валидационной выборки брался 25% от обучающей. В качестве классификатора так же использовалась сеть долгой краткосрочной памяти (Long Short Term Memory network LSTM) - разновидность рекуррентной нейронной сети. Она сохраняет и учитывает предшествующую информацию. На вход сети RNN передается некоторая последовательность. Запоминание означает, что RNN выполняет одну и ту же задачу для каждого элемента последовательности. Где каждый выход зависит от предыдущих вычислений. Длина временных шагов определяется длиной ввода. Например, если обрабатываемая последовательность слов является предложением из 4 слов RNN будет развернут в нейронную сеть с 4 временными слоями. Один слой соответствует слову. Последовательность векторов слов документа подавалась на вход LSTM, где число временных слоев совпадало с фиксированным размером терминов в документе. Для обучения сетей RNN необходимо большое количество данных. Сеть RNN склонна к переобучению, поэтому количество эпох обучения выбирается небольшое [11].

Результаты

Эксперименты в классификации анализа тональности проводились на 3-х различных наборах данных, которые описаны ниже.

Выборка отзывов. Для сбора отзывов был написан бот, который собирал с сайта [irecommend.ru](https://www.irecommend.ru) текстовые отзывы, проходя по различным категориям: кино, книги, автомобили, красота и здоровье, туризм, техника, животные. У каждого отзыва был рейтинг от 1 до 5. Отзывы, имеющие рейтинг меньше 4, относились к классу негативных, остальные относились к положительным. Таким образом была собрана выборка из 47847 отзывов. Из отзывов отбиралась наиболее значимая часть (выводы и резюме). Обучающая выборка составила 27759 примера, валидационная выборка составила 9253 примеров. Перед обучением из выборки удаляются лишние символы, слова не несущие смысловой нагрузки (стоп-слова), например, частицы, предлоги, знаки препинания и символы из других языков. Все слова приводились к нижнему регистру, к нормальной форме с помощью специальных морфологических анализаторов. При тестировании данные проходят, тот же алгоритм предобработки, что и обучающие данные. Для каждого отзыва был найден агрегированный вектор для всех типов векторизации слов, описанных в работе W2V, FastText, GloVe. Так же использовались статистические методы Bag of Words, TF-IDF, n-grams. Результаты классификации отзывов представлены в таблице. В качестве метрики качества классификации использовался F1-score.

Таблица 1

F1-score для набора данных отзывов

	Bag of Words	N - grams	TF-IDF	W2V (avg)	FastText (avg)	GloVe (avg)
MLP	0.65	0.73	0.81	0.78	0.77	0.76
RF	0.62	0.72	0.79	0.76	0.74	0.75
NB	0.62	0.72	0.74	0.74	0.73	0.73
KNN	0.64	0.73	0.77	0.75	0.73	0.74
SVC	0.65	0.74	0.81	0.78	0.77	0.77
CNN	0.64	0.72	0.78	0.81	0.80	0.78
RNN (LSTM)	0.62	0.70	0.75	0.76	0.75	0.75

Таким образом, наиболее точной оказалась классификация с методами TF-IDF, с классификатором SVC (метод опорных векторов) и векторным представлением Word2Vec в связке со сверточной нейронной сетью. Они выдали лучший F1-score. Рекуррентная нейронная сеть выдавала не самый лучший результат, поскольку ей для обучения необходимо намного больше данных для обучения, чем другим классификаторам.

Выборка новостей. Выборка новостей был найдена на соревновании по машинному обучению [kaggle.com](https://www.kaggle.com) [12], датасет был разделен на 3 класса: негативная, позитивная, положительная новость. В выборке преобладали данные с нейтральными новостями, поэтому было добавлена часть негативных новостей из рубрики происшествия из новостного портала Риа-новости. Обучающая выборка составляла 18798 новости. Лучший результат дал статистический метод TF-IDF с вектором опорных векторов SVM.

Таблица 2

F1-score для набора данных новостей

	Bag of Words	N - grams	TF-IDF	W2V (avg)	FastText (avg)	GloVe (avg)
MLP	0.55	0.63	0.72	0.73	0.71	0.67

RF	0.53	0.62	0.70	0.71	0.70	0.68
NB	0.52	0.62	0.68	0.72	0.69	0.69
KNN	0.54	0.63	0.69	0.70	0.70	0.69
SVC	0.55	0.64	0.77	0.73	0.73	0.70
CNN	0.52	0.62	0.70	0.73	0.73	0.72
RNN (LSTM)	0.52	0.60	0.65	0.66	0.63	0.62

Рис. 6 Матрица ошибок для выборки новостей классификатора SVM

Как можно видеть из матрицы ошибок, чаще всего ошибки случаются при классификации нейтральных новостей. Классификатор относит нейтральную новость к положительной или отрицательной, поскольку определение анализа тональности является относительно субъективной классификацией, так как даже человек не всегда корректно справляется с этой задачей. Но классификатор с минимальной ошибкой отличает негативную новость от положительной.

Выборка сообщений твиттера. Выборка русскоязычных сообщений социальной сети твиттер была найдена в [13]. Она содержит 15 миллионов записей, полученных с помощью API Twitter. На нем были обучены векторные представления слов для W2V, FastText, Glove. Размер обучающей выборки 183 521 твит.

Таблица 3

F1-score для набора данных твиттера

	Bag of Words	N - grams	TF-IDF	W2V (avg)	FastText (avg)	GloVe (avg)
MLP	0.63	0.65	0.75	0.75	0.75	0.74
RF	0.62	0.64	0.74	0.74	0.74	0.73
NB	0.62	0.64	0.74	0.73	0.72	0.74
KNN	0.60	0.63	0.73	0.74	0.73	0.73
SVC	0.63	0.65	0.76	0.75	0.75	0.74
CNN	0.62	0.62	0.74	0.77	0.77	0.74
RNN (LSTM)	0.61	0.60	0.72	0.76	0.75	0.74

Таким образом, наилучший F1-score 0.77 показала сверточная нейронная сеть с векторным представлением W2V. Также близкий результат показала SVC со статическим TF-IDF. Наборы данных могут содержать ошибки аннотации, то есть примеры из выборки содержат экземпляры, размеченные некорректно, что влияет на точность классификации.

Заключение и выводы.

В работе были рассмотрены методы решения задачи классификации документов, в частности, задача определения тональности текстов. Сравнивались стандартные статистические модели с современными моделями векторных представлений слов, таких как Word2Vec, FastText, Glove. Были получены векторные представления слов на различных русскоязычных корпусах. Были проверены свойства векторов на нахождение похожих слов по значению. В работе использовались 3 обучающие выборки: выборка отзывов на различные товары и услуги, выборка сообщений твиттера, выборка новостей.

В задаче анализа тональности сравнивались различные виды классификаторов: MLP (многослойная нейронная сеть), Random Forest (Случайный лес), NB (Наивный байесовский классификатор), KNN (метод к-ближних соседей), SVM (метод опорных векторов), CNN (сверточная нейронная сеть), LSTM (рекуррентная сеть с долгой краткосрочной памятью).

Результаты показали, что сверточная нейронная сеть со связкой Word2Vec показывает наилучшую точность (F1-score) на средних по размеру выборках. Близкий по точности результат дают так же FastText и Glove. Статистический метод TF-IDF также дает хорошую точность в связке с методом опорных векторов SVM на всех наборах данных, используемых в работе. Таким образом, статистические методы не всегда уступают методам, основанным на векторном представлении слов. Заметим, что определение тональности является по своей сути субъективной задачей. Классификаторы обучаются в соответствии с интуицией пользователя, который эти данные разметил. Поэтому в задаче определения тональности текстов приемлемой точностью считается F1-score больше 0.70. Превзойти такую точность в данной работе удалось на всех представленных обучающих выборках.

1. Свободная энциклопедия Википедия. Анализ тональности текста. [Электронный ресурс] https://ru.wikipedia.org/wiki/Анализ_тональности_текста (дата обращения: 31.03.19).
 2. Свободная энциклопедия Википедия, "Классификация документов " [Электронный ресурс] https://ru.wikipedia.org/wiki/Классификация_документов (дата обращения: 31.03.19).
 3. Data Review, "Анализ тональности текста: концепции, методы, области применения" [Электронный ресурс] <http://datareview.info/article/analiz-tonalnosti-teksta-kontseptsiya-metodyi-oblasti-primeneniya/> (дата обращения: 31.03.19).
 4. Mikolov T. et al. Distributed Representations of Words and Phrases and Their Compositionality //Advances in Neural Information Processing Systems. —2013. — Pp. 3111- 3119.
 5. Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomas Mikolov. Enriching Word Vectors with Subword Information [Электронный ресурс] <https://arxiv.org/abs/1607.04606> (дата обращения: 31.03.19).
 6. Armand Joulin, Edouard Grave, Piotr Bojanowski, Tomas Mikolov, Bag of Tricks for Efficient Text Classification [Электронный ресурс] <https://arxiv.org/abs/1607.01759> (дата обращения: 31.03.19).
 7. Jeffrey Pennington, Richard Socher, Christopher D. Manning. GloVe: Global Vectors for Word Representation [Электронный ресурс] <https://www.aclweb.org/anthology/D14-1162> (дата обращения: 31.03.19).
 8. Christopher M. Bishop. Pattern recognition and machine learning, 2006.
 9. Zhang Y., Wallace B. A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification [Электронный ресурс] <https://arxiv.org/pdf/1510.03820.pdf> (дата обращения: 31.03.19).
 10. Yoon K. Convolutional Neural Networks for Sentence Classification. [Электронный ресурс] <https://www.aclweb.org/anthology/D14-1181.pdf> (дата обращения: 31.03.19).
 11. Lei Zhang, Shuai Wang, Bing Liu. Deep learning for sentiment analysis: A survey. // In Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, p. 1253. Wiley Online Library [Электронный ресурс] <https://onlinelibrary.wiley.com/doi/full/10.1002/widm.1253> (дата обращения: 31.03.19).
 12. Sentiment Analysis in Russian [Электронный ресурс] <https://www.kaggle.com/c/sentiment-analysis-in-russian/leaderboard> (дата обращения: 31.03.19).
 13. Рубцова Ю.В. Построение корпуса текстов для настройки тонового классификатора // Программные продукты и системы. – 2015. – №1(109) – С.72-78. [Электронный ресурс] <http://study.mokoron.com> (дата обращения: 31.03.19).
-