Математические структуры и моделирование 2019. № 4(52). С. 97–110

УДК 004.89
DOI 10.24147/2222-8772.2019.4.97-110

СРАВНЕНИЕ ЭФФЕКТИВНОСТИ МЕТОДОВ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ СЛОВ ДЛЯ ОПРЕДЕЛЕНИЯ ТОНАЛЬНОСТИ ТЕКСТОВ

Н.М. Лыченко
д.т.н., доцент, e-mail: nlychenko@mail.ru
А.В. Сороковая
инженер-программист, e-mail: nastusha24sh-g@yandex.com

Институт машиноведения и автоматики Национальной академии наук Кыргызской республики, Бишкек, Кыргызстан

Аннотация. Векторное представление слов используется для различных задач автоматизированной обработки естественного языка. Существует множество методов векторного представления слов, в том числе нейросетевые методы Word2Vec и GloVe, и классический метод латентно-сематического анализа LSA. Настоящая работа посвящена исследованию эффективности применения методов векторного представления слов в нейросетевом классификаторе тональности русскоязычных и англоязычных текстов на базе LSTM-сети. Описаны особенности методов векторного представления слов (LSA, Word2Vec, GloVe), представлена архитектура нейросетевого классификатора тональности текста на основе LSTM-сети и рассмотренных методов векторного представления слов, приведены результаты вычислительных экспериментов и их обсуждение. Показано, что наилучшей моделью векторного представления слов с позиций скорости обучения, меньшего объёма корпуса слов для обучения, лучшей точности и скорости обучения нейросетового классификатора, является модель LSA.

Ключевые слова: векторное представление слов, LSA-метод, методы Word2Vec и GloVe, определение тональности текста, нейросетевой классификатор, LSTM-сеть, точность классификации, скорость обучения.

Введение

В настоящее время автоматический анализ информации на естественном языке приобретает всё большую актуальность и представляет огромный интерес для социологии, маркетинга, лингвистики, психологии и других сфер человеческой деятельности. В особенности этому способствует стремительное расширение сети Интернет.

Для автоматизации процесса обработки текстовой информации требуется представление текста в виде некоторой числовой модели, представляющей текст в виде вектора его признаков — признакового описания. Для получения

признакового описания (так называемой индексации текста) используются два подхода: представление текста как «мешка слов» и представление текста как последовательности слов, которые, в свою очередь, представляются векторами [1].

Представление текста в виде «мешка слов» позволяет представить текст в виде вектора размерности словаря обучающей выборки. Каждый элемент вектора представляет собой вес соответствующего слова из словаря. Весом может быть частота появления слова в тексте или другая более сложная мера. Такое представление не учитывает порядок слов в тексте, и это один из главных недостатков данного метода.

Перспективным направлением в обработке естественного языка является векторное представление слов, которое позволяет представлять текст как последовательность векторов, соответствующих словам текста.

Простейший способ преобразования слова в вектор — one-hot-кодировка [2], или кодирование каждого слова с помощью вектора длины, равной размеру словаря обучающей выборки. Каждый такой вектор состоит из нулей и одной единицы, соответствующей положению слова в словаре. Такое представление неэффективно по памяти и, самое главное, не даёт никакого объяснения семантического смысла слов, не позволяет сравнивать слова на предмет семантической близости, что сильно затрудняет процесс классификации.

В настоящее время предложены различные методы получения векторного представления слов, лишённые этих недостатков и позволяющие построить низкоразмерное векторное представление каждого слова по корпусу (набору) текстов и поддерживать контекстуальное сходство слов. Прежде всего это два хорошо известных нейросетевых метода — Word2Vec [3, 4], разработанный в Google, и Global Vectors (GloVe) [5], разработанный в Стенфордском университете. Эти два метода отлично показали себя в задачах обработки естественного языка, таких как определение похожих слов, например, и другие. Однако в [5] показано, что классические методы, в частности метод латентно-семантического анализа LSA [6], могут быть более полезными, чем Word2Vec, например. Это объясняется тем, что Word2Vec учится на низкоразмерных векторах с начала обучения и не использует всю информацию из учебного корпуса слов. Так же доказано [7], что LSA более устойчив и не сильно зависит от размера корпуса.

Настоящая работа посвящена исследованию эффективности применения указанных методов в задаче автоматического определения тональности русскоязычных и англоязычных текстов. Определение тональности текстов – частная задача классификации, процесс вычислительной идентификации и категоризации мнений, выраженных в части текста для определения того, является ли отношение автора текста к определённой теме, продукту и т. д. положительным, отрицательным или нейтральным [8]. В решении такого рода задач хорошо зарекомендовали себя искусственные нейронные сети (ИНС). При этом архитектура используемой ИНС может быть различной. ИНС прямого распространения [9] ограничены контекстом, т. е. могут учитывать только фиксированное количество слов вокруг рассматриваемого для определения его значения. Ре-

куррентные нейронные сети [10] могут использовать весь контекст, независимо от его длины (с некоторыми ограничениями). LSTM-сеть (долгая краткосрочная память) – частный вид рекуррентной нейронной сети, который позволяет обнаруживать как длинные, так и короткие шаблоны в данных, а также частично устраняет проблему исчезновения градиента [11]. В настоящей работе именно эта архитектура выбрана для реализации классификатора тональности текстов.

Далее статья построена следующим образом. Первый раздел кратко описывает методы и модели векторного представления слов (LSA, Word2Vec, GloVe). Второй раздел описывает основные этапы построения классификатора тональности текстов и его архитектуру на основе LSTM-нейронных сетей и рассмотренных моделей векторного представления слов. Третий раздел представляет результаты вычислительных экспериментов для определения тональности русскоязычных и англоязычных текстов и их обсуждение. Заключение и перспективы исследования представлены в четвёртом разделе.

1. Методы и модели векторного представления слов

В настоящей работе рассматриваются три метода с целью их дальнейшего сравнения: LSA как классический метод, Word2Vec как наиболее популярный в русскоязычном сообществе программистов, GloVe как популярная альтернатива, мало описанная в русскоязычных источниках. При дальнейшем изложении термин «слово» заменяется на «терм», подчёркивая, что это часть текста, несущая некоторую смысловую нагрузку.

Латентно-семантический анализ. В основе метода латентно-семантического анализа (LSA) лежат принципы факторного анализа [6], позволяющие, в частности, выявлять латентные (скрытые) связи между термами и текстами (документами), определяющие характерные тематики, присущие документам и термам. Каждая тематика характеризуется весом в формировании семантического смысла документов и термов. Сокращение размерности векторного представления происходит за счёт удаления из модели языка тематик с наименьшей смысловой нагрузкой.

Первым шагом необходимо преобразовать корпус текстов (документов) в матрицу термы-на-документы. Элементами этой матрицы обычно являются веса TF-IDF [6].

TF (term frequency — частота слова) — отношение n_t (числа вхождений в документ d некоторого слова t) к общему числу слов документа:

$$TF(t,d) = \frac{n_t}{\sum_k n_k}.$$

IDF (inverse document frequency — обратная частота документа) — инверсия частоты, с которой некоторое слово встречается в документах корпуса. Основоположником данной концепции является Карен Спарк Джонс [12]. Учёт IDF

уменьшает вес широкоупотребительных слов:

$$IDF(t, D) = \log \frac{|D|}{\{d_i \in D | t \in d_i\}},$$

где $|D|$ — число документов в корпусе; $|\{d_i \in D | t \in d_i\}|$ — число документов в корпусе, в которых встречается слово t.

Таким образом, вес TF-IDF является произведением двух сомножителей: $TF(t, d) * IDF(t, D)$.

Слова с высокой частотой в пределах конкретного документа и с низкой частотой употреблений в других документах получают большой вес TF-IDF.

Далее, согласно теореме о сингулярном разложении [13] полученная вещественная прямоугольная матрица может быть разложена на произведение трёх матриц:

$$A = USV^T,$$

где матрицы U и V — ортогональные, а S — диагональная матрица, на диагонали которой находятся сингулярные значения матрицы A. Если в матрице S оставить только k наибольших сингулярных значений, а в матрицах U и V — только соответствующие этим значениям столбцы, то произведение получившихся матриц $\tilde{U}$, $\tilde{S}$ и $\tilde{V}^T$ будет наилучшим приближением исходной матрицы A к матрице $\tilde{A}$ ранга k. Эта матрица будет отражать основную структуру ассоциативных зависимостей, присутствующих в исходной матрице, при этом каждый терм (строка матрицы $\tilde{U}$) и документ (строка матрицы $\tilde{V}$) представляются в виде векторов в общем пространстве размерности k (пространстве гипотез) (см. рис. 1). k подбирается эмпирически и зависит от количества исходных документов. Векторы термов можно использовать в качестве векторного представления слов.

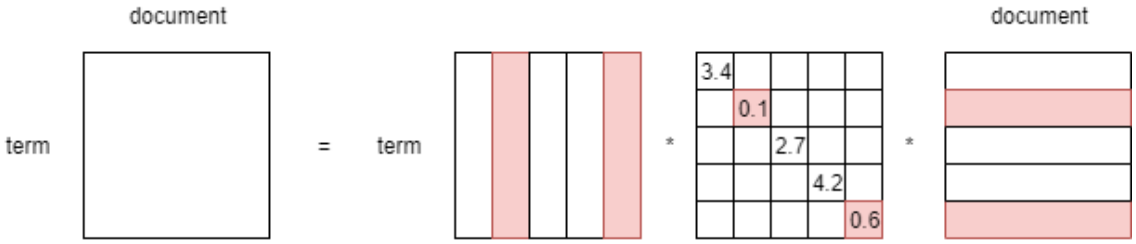

Рис. 1. Модель векторного представления слов LSA

Word2Vec. Word2Vec [3] — нейросетевой метод, позволяющий предугадывать контекст слова по заданному слову (метод Skip-Gram) или, наоборот, предугадывать слово по заданному контексту (метод CBOW). При этом скрытый слой нейросети проецирует слова в низкоразмерный вектор.

Общая архитектура нейронной сети (см. рис. 2): входной слой, который принимает векторы слов контекста для CBOW или вектор слова для Skip-Gram в one-hot кодировке размерности v; скрытый слой с количеством нейронов k;

выходной слой с функцией активации иерархический софтмакс (Hierarchical Softmax) [4] или негативное семплирование (Negative Sampling) [4], выход которого приближается в процессе обучения к вектору слова для CBOW или векторам слов контекста для Skip-Gram в one-hot кодировке размерности v. Выход скрытого слоя после обучения используется для получения векторного представления слов размерности k.

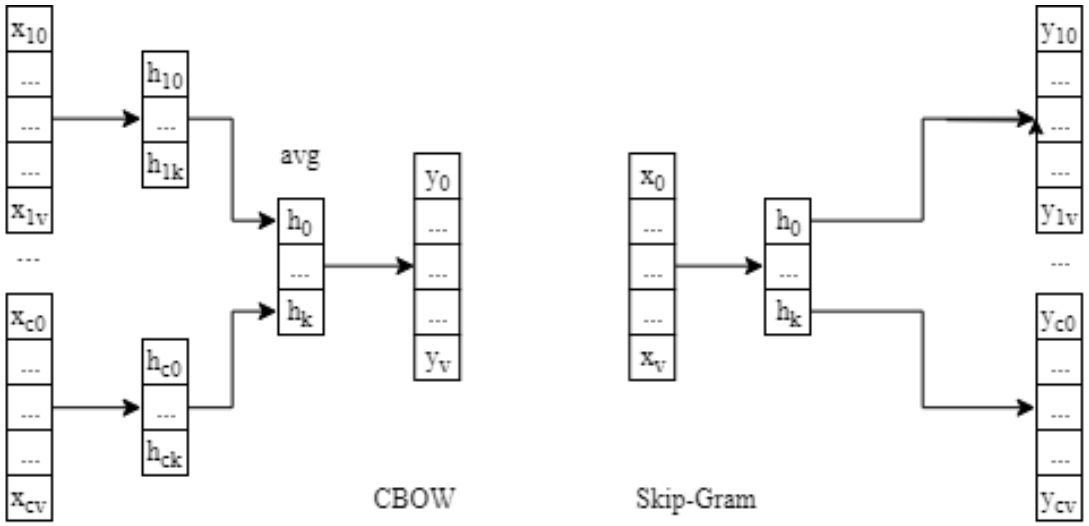

Рис. 2. Архитектура нейронной сети для метода Word2Vec

Хотя в метод заложены только статистические свойства текстов, оказывается, что натренированная модель Word2Vec улавливает некоторые семантические свойства слов.

GloVe. GloVe [5] — модель, сочетающая в себе особенности сингулярного разложения и методов Word2Vec.

Первым шагом строится матрица совместной встречаемости X из учебного корпуса. Значение элемента X_{ij} отображает, как часто встречается слово j в контексте слова i. Для оценки семантической близости между словами i и j используется отношение вероятностей их совместного появления в контексте k:

$$F(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}} = \frac{X_{ik}/\sum_m X_{im}}{X_{jk}/\sum_n X_{jn}},$$

где w_i, w_j — векторы слов, $\tilde{w}_k$ — вектор контекста.

Семантическая близость получаемых векторов определяется их скалярным произведением. Путём преобразований и предположений показано [5], что целью обучения модели GloVe является изучение векторов таким образом, чтобы их скалярное произведение приближалось к логарифму вероятности совместного появления слов в обучающей выборке. Чтобы снизить вес совместных появлений слов, которые редко встречаются (несут меньше информации, шумные и ненадёжные) или не встречаются вообще, а также снизить вес слишком частых совместных появлений (например, таких как “it is”), Пеннингтон и др.

[5] используют регрессионную модель взвешенных наименьших квадратов как целевую функцию обучения (функцию потерь) (1), где w_i — вектор основного слова, $\tilde{w}_j$ — вектор контекста, b_i и $\tilde{b}_j$ — скалярные величины отклонений для основного слова и слова контекста соответственно, V — размер словаря:

$$J = \sum_{i,j=1}^{V} f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij})^2. \tag{1}$$

Весовая функция $f(X_{ij})$ выбирается из класса весовых функций (2), где α и x_{max} подбираются эмпирически:

$$f(x) = \begin{cases} \left(\frac{x}{x_{max}}\right)^{\alpha}, \text{ если } x < x_{max}; \\ 1, \text{ если } x \geqslant x_{max}. \end{cases} \tag{2}$$

Модель генерирует два вектора каждого слова: вектор слова как основного и вектор слова как контекста. В качестве векторного представления можно использовать один из них или, например, их сумму.

2. Классификатор тональности текста на основе LSTM–сети

Основные этапы построения и обучения классификатора текста следующие: предобработка текстов, индексация текстов, построение и обучение классификатора, оценка качества классификации.

Предварительная обработка текста обязательно включает в себя токенизацию — выделение единиц языка в тексте, так называемых лексем, т. е. слов или термов.

Также предварительная обработка текста может включать: приведение слов к одному регистру для исключения семантического различия между одинаковыми словами в разном регистре; удаление таких слов, как союзы, предлоги, артикли, т. е. семантически нейтральных слов (стоп-слов); удаление чисел или их замена на текстовый эквивалент; удаление пунктуации и излишних пробельных символов; нормализацию слов — выделение основы или корня слова (стемминг), или получение словарной формы слова (лемматизация), замена всех слов на нормализованную форму.

Возможны и другие способы предобработки текста. Выбор сценария предобработки зависит от рода решаемой задачи и свойств выборки. В результате из текста выделяются все значимые слова, действительно определяющие его семантический смысл.

Далее для классификации текстов необходима их индексация, т. е. требуется получение признакового описания каждого классифицируемого текста. Для оценки качества классификации можно использовать метрики полноты и точности, специальные тестовые выборки.

Для сравнения эффективности рассмотренных в предыдущем разделе методов векторного представления слов в задаче определения тональности текстов

построена нейронная сеть, представляющая собой гибрид свёрточной и LSTM сетей.

Архитектура сети представлена на рис. 3 и содержит следующие слои.

1. Входной слой, принимающий документ как последовательность индексов лексем — термов (токенов).

2. Слой векторного преобразования слов с фиксированными весами, где веса слоя преобразования задаются матрицей, i-ая строка которой представляет собой векторное представление i-го терма; служит для выбора вектора соответствующего терма; матрица строится на основе векторов, сгенерированных моделью векторного представления слов.

3. Слой регуляризации, который меняет определённый процент значений выхода предыдущего слоя для предотвращения переобучения.

4. Свеёрточная сеть — последовательность свёрточного и субдискретизирующего слоя с функцией максимума; используется для уменьшения количества входных параметров следующего слоя, повышает скорость обучения.

5. Слой LSTM как основной классификатор.

6. Выходной слой с сигмоидальной функцией активации.

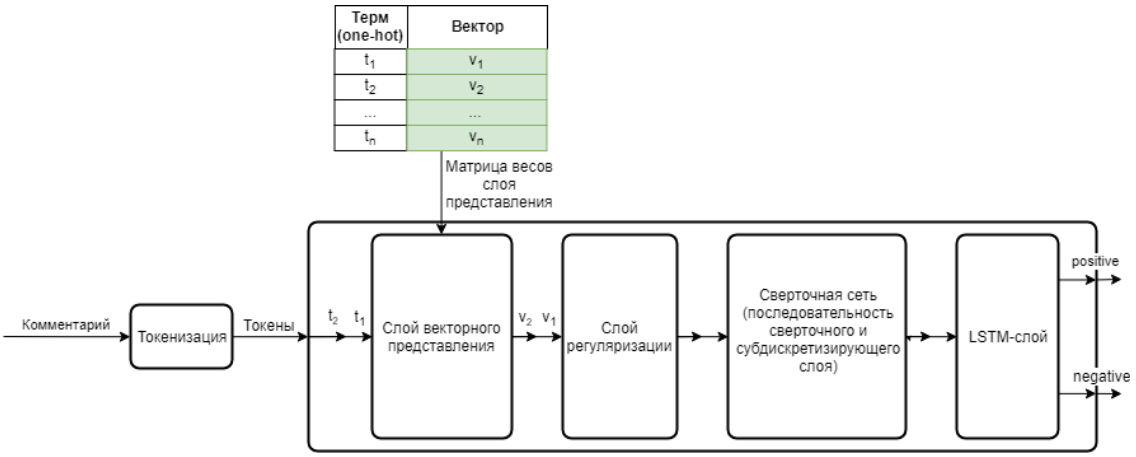

Рис. 3. Архитектура нейронной сети для определения тональности текста

При реализации нейронной сети и алгоритмов обучения моделей векторного представления слов использовались научные библиотеки: Keras, NumPy, GenSim, Glove-Python.

3. Результаты вычислительных экспериментов и обсуждение

В работе использовался русскоязычный корпус комментариев в социальных сетях [14] и англоязычный корпус рецензий на фильмы с сайта IMDB [15].

Для каждого корпуса проведены эксперименты с разным количеством текстов для обучения и разным сочетанием параметров моделей векторного представления слов:

— число текстов для обучения — 1000, 10000, 25000;

— размерность векторного представления — 50, 150, 300;

— число эпох обучения нейросетевых моделей векторного представления или итераций для LSA — 2, 8, 15.

Для моделей Word2Vec и GloVe использовался размер окна контекста: 2 слова слева и 2 слова справа от основного слова. Для моделей Word2Vec использовалось негативное семплирование. В результате каждого эксперимента было зафиксировано время обучения модели векторного представления слов emb time (ось абсцисс на рис. 4, 5). Для проверки эффективности LSTM-сеть для определения тональности текста обучалась на 13000 комментариев. Проверка точности проводилась на 3250 комментариев. В результате была зафиксирована максимальная достигнутая точность классификации на тестовой выборке за 10 эпох обучения классификатора max valid accurancy (ось ординат на рис. 4, 5).

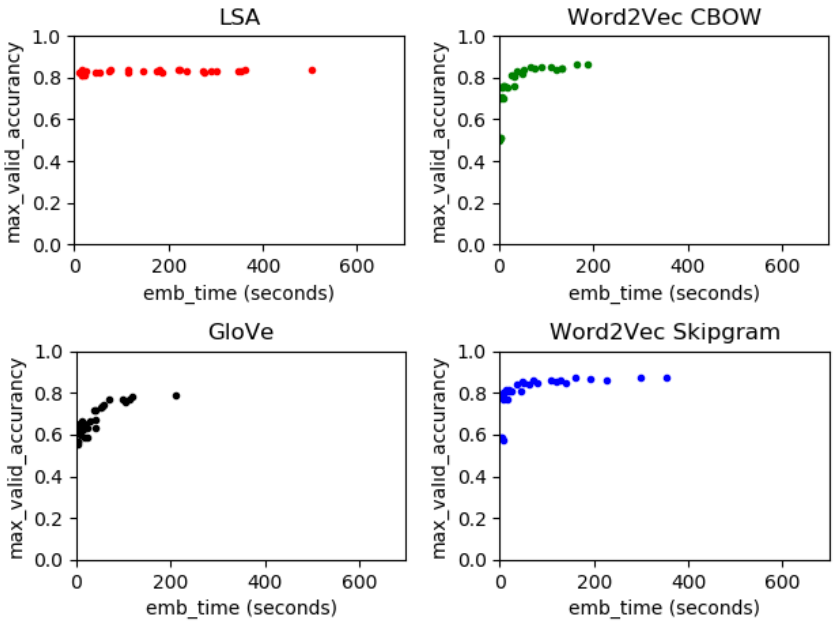

Рис. 4. Диаграммы рассеивания для экспериментов (англоязычные тексты)

Большая длительность обучения моделей на англоязычном корпусе текстов связана с большей длиной самих текстов в корпусе (в англоязычном корпусе — рецензии на фильмы, в русскоязычном — короткие посты в соцсети Twitter).

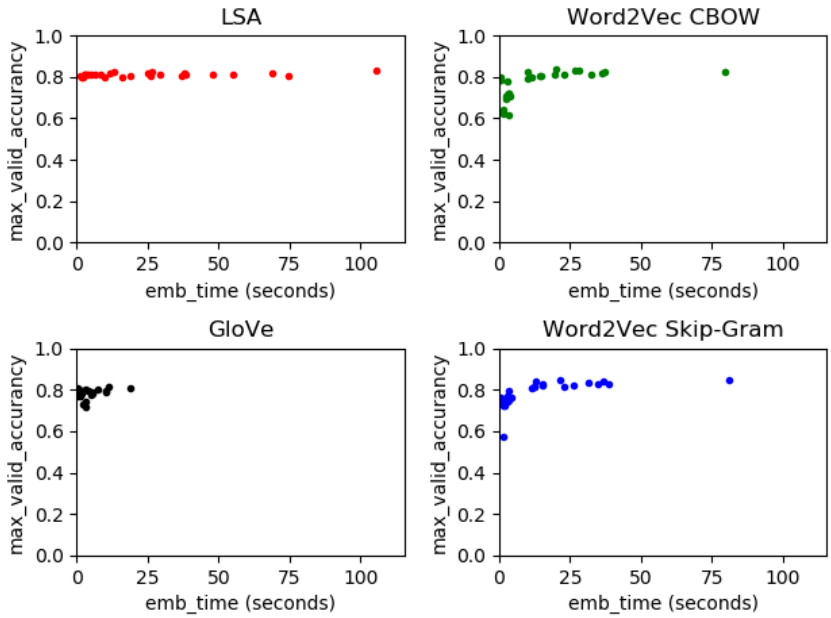

Рис. 5. Диаграммы рассеивания для экспериментов (русскоязычные тексты)

Наибольшие средние значения точности показали классификаторы англоязычных текстов (за исключением экспериментов с использованием GloVe), однако это также может быть связано с их большей длиной — данных для обучения было больше.

Видно, что LSA является самой устойчивой моделью относительно изменяемых параметров по точности классификатора на её основе, т. к. все эксперименты привели к примерно одинаковой точности классификации. LSA показала лучшие результаты по точности классификации (в среднем 81,2 % для русскоязычных текстов и 82,9 % — для англоязычных), но при этом худшие по времени обучения как для русскоязычного, так и для англоязычного корпусов.

Широко используемые методы Word2Vec Skip-Gram и Word2Vec CBOW вели себя примерно одинаково на обоих корпусах текстов: точность классификации падает при небольшом количестве обучающих текстов. Скорость их обучения меньше, чем скорость обучения LSA-моделей, однако значительно проигрывает скорости обучения GloVe-моделей.

Интерес вызывает метод GloVe, который, несмотря на меньшую длину текстов в русскоязычном корпусе, показал более высокую среднюю точность классификации и небольшую чувствительность к размеру обучающей выборки. Эксперимент, показавший максимальную точность классификации на русскоязычном корпусе, был проведён именно с помощью этого метода. При этом на англоязычном корпусе текстов этот метод показал худшие результаты по точности классификации и наибольшую чувствительность к размеру обучающей выборки.

Согласно исследованию [5], эксперты обычно соглашаются в оценках тональности конкретного текста в 79 % случаев. Классификатор, который опреде-

ляет тональность текста с точностью более 70 %, считают эффективным. Для русскоязычного корпуса с точки зрения этого критерия все классификаторы, построенные на базе LSA и GloVe, можно применять на практике (см. рис. 5). Как показали вычислительные эксперименты, Word2Vec плохо применим на корпусах небольшого размера, однако в остальных случаях показывает хорошие результаты. Для англоязычного корпуса все классификаторы, построенные на базе LSA, можно применять на практике (см. рис. 4). Word2Vec также плохо применим на корпусах небольшого размера, однако в большинстве случаев показывает хорошие результаты. Большинство экспериментов с использованием модели GloVe показали неприемлемые результаты.

Выберем в качестве лучшей модель LSA, обученную на 1000 русскоязычных комментариев в течение 8 эпох, проектирующую терм в вектор размерности 150. Точность модели (acc) и значения функции потери (loss) на каждой эпохе обучения (epoch) показаны на рис. 6.

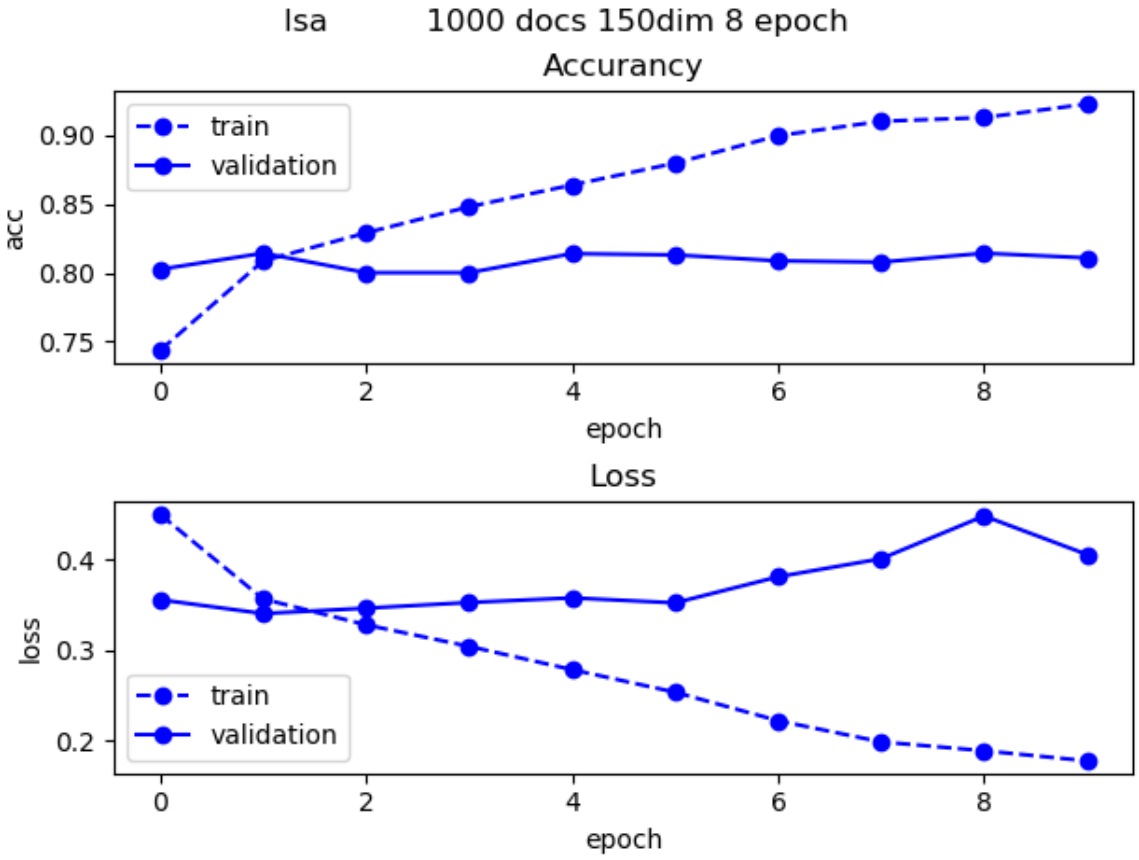

Рис. 6. Прогресс обучения ИНС-сети с использованием модели LSA

Значение функции потерь на тестовой выборке уменьшается до первой эпохи. Однако, начиная с 6 эпохи, функция потерь на тестовой выборке значительно увеличивается, при этом на обучающей выборке продолжает стремительно падать. Это говорит о начале процесса переобучения сети. Можно проследить изменение значений точности определения тональности текста, достигнутых на

каждой эпохе — точность на обучающей выборке растёт, точность на тестовой выборке меняется незначительно, а значит, увеличение количества эпох обучения нейросети для классификации не имеет смысла. Оптимальное количество эпох обучения нейронной сети для данной модели векторного представления — 1–5 эпох.

4. Заключение

Таким образом, в работе представлено описание методов векторного представления слов и сравнение эффективности полученных моделей для обучения классификатора тональности текстов на основе LSTM-нейронной сети.

Исследование показало, что наиболее эффективной в задаче определения тональности рассмотренных текстов является модель LSA. Модель LSA довольно ресурсоёмкая, но не слишком чувствительна к объёму выборки для обучения.

Стоит учитывать, что для каждого набора параметров моделей векторного представления слов был проведён лишь один эксперимент, что вносит большие ошибки в оценку измерений, в особенности, времени обучения.

Точность классификатора, основанного на LSTM-сети, также зависит от способа токенизации текста и обработки токенов (стемминг или лемматизация, удаление стоп-слов, приведение к одному регистру и т. д.), от параметров моделей векторного представления, а также от структуры и параметров самой нейронной сети. Исследование, описанное в данной работе, проводилось на ограниченном количестве значений параметров моделей векторного представления слов. Поэтому для улучшения результатов стоит рассмотреть различные виды предобработки текстов, проверить эффективность моделей векторного представления с другими параметрами и на других архитектурах LSTM-сетей.

Стоить обратить внимание и тщательно исследовать эффективность модели GloVe на других русскоязычных корпусах текстов, т. к. она показывает очень хорошие результаты в анализе тональности комментариев. Однако в русскоязычных источниках литературы не было найдено ни одного примера его применения.

Выбор лучшей модели векторного представления текста также основывается на ранжированных критериях оценки. Для различных задач критерии и их приоритет не совпадают, а поэтому модель следует выбирать относительно ограничений, которым должна удовлетворять выбираемая модель.

Литература

1. Word Bags vs Word Sequences for Text Classification. URL: `https://towardsdatascience.com/word-bags-vs-word-sequences-for-text-classification-e0222c21d2ec` (дата обращения 01.02.2019).
2. How to One Hot Encode Sequence Data in Python. URL: `https://machinelearningmastery.com/how-to-one-hot-encode-sequence-data-in-python` (дата обращения 05.02.2019).

3. Le Q., Mikolov T. Distributed Representations of Sentences and Documents // Proceedings of the 31st International Conference on Machine Learning. Beijing, China. 2014. V. 32. P. 1188–1196.
4. Mikolov T., Chen K., Corrado G., Dean J. Efficient Estimation of Word Representations in Vector Space // Proceedings of Workshop at ICLR, arXiv. 2013. P. 1301–3781.
5. Pennington J., Socher R., Manning C. GloVe: Global Vectors for Word Representation // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014. V. 14. P. 1532–1543.
6. Landauer T.K., Foltz P.W., Laham D. Introduction to Latent Semantic Analysis // Discourse Processes. 1998. V. 25. P. 259–284.
7. Altszyler E., Sigman M., Slezak D.F. Comparative study of LSA vs Word2vec embeddings in small corpora: a case study in dreams database // Consciousness and Cognition. 2017. V. 56. P. 178–187.
8. Кузнецов А.Л., Кочуров Д.Н. Анализ тональности текста в социальных медиа с использованием искусственных нейронных сетей // Материалы X Международной студенческой научной конференции «Студенческий научный форум». УрФУ, Екатеринбург, Россия, 2018. https://scienceforum.ru/2018/article/2018009449 (дата обращения: 07.03.2019).
9. Хайкин С. Нейронные сети: полный курс. М. : ООО «И.Д. Вильямс», 2006. 1104 с.
10. Williams R., Zipser D. A learning algorithm for continually running fully recurrent neural networks // Neural computation. 1989. V. 1, Issue 2. P. 270–280.
11. Understanding LSTMs. URL: http://colah.github.io/posts/2015-08-Understanding-LSTMs/ (дата обращения 16.10.18).
12. Jones K. A statistical interpretation of term specificity and its application in retrieval // Journal of Documentation. MCB University: MCB University Press. 2004. V. 60, No. 5. P. 493–502.
13. Голуб Д., Ван Лоун Ч. Матричные вычисления. М. : «Мир», 1999. 548 с.
14. Тренировочные корпусы текстов на русском языке. URL: http://study.mokoron.com/ (дата обращения 09.01.2019).
15. IMDB Dataset of 50K Movie Reviews. URL: https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews (дата обращения 12.02.2019).

COMPARISON OF EFFECTIVENESS OF WORD REPRESENTATIONS METHODS IN VECTOR SPACE FOR THE TEXT SENTIMENT ANALYSIS

N.M. Lychenko
Dr.Sc. (Eng.), e-mail: nlychenko@mail.ru
A.V. Sorokovaja
Software Engineer, e-mail: nastusha24sh-g@yandex.com

Institute of Mechanical Engineering and Automation of National Academy of Sciences, Bishkek, Kyrgyzstan

Abstract. The word representations in vector space is used for various tasks of automated processing of a natural language. There are many methods of vector

representation of words, including neural network methods Word2Vec and GloVe, and the classical method of latent-sematic analysis LSA. This work is devoted to the study of the effectiveness of the application of vector representation of words in the neural network classifier for sentiment analysis of Russian and English texts based on the LSTM network. The features of word representations methods in vector space (LSA, Word2Vec, GloVe) are described, the architecture of a neural network classifier for sentiment analysis of text based on the LSTM network and the considered methods of vector representation of words are presented, the results of computational experiments and their discussion are presented. It is shown that the LSA model is the best model for the vector representation of words from the standpoint of learning speed, less corpus of words for learning, better accuracy and learning speed of the neural network classifier.

Keywords: word representations in vector space, LSA-method, Word2Vec and GloVe Methods, sentiment analysis, neural network classifier, LSTM network, classification accuracy, learning speed.

References

1. Word Bags vs Word Sequences for Text Classification. URL: `https://towardsdatascience.com/word-bags-vs-word-sequences-for-text-classification-e0222c21d2ec` (01.02.2019).
2. How to One Hot Encode Sequence Data in Python. URL: `https://machinelearningmastery.com/how-to-one-hot-encode-sequence-data-in-python` (05.02.2019).
3. Le Q. and Mikolov T. Distributed Representations of Sentences and Documents. Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 2014, vol. 32, pp. 1188–1196.
4. Mikolov T., Chen K., Corrado G., and Dean J. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, arXiv, 2013, pp. 1301–3781.
5. Pennington J., Socher R., and Manning C. GloVe: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, vol. 14, pp. 1532–1543.
6. Landauer T.K., Foltz P.W., and Laham D. Introduction to Latent Semantic Analysis. Discourse Processes, 1998, vol. 25, pp. 259–284.
7. Altszyler E., Sigman M., and Slezak D.F. Comparative study of LSA vs Word2vec embeddings in small corpora: a case study in dreams database. Consciousness and Cognition, 2017, vol. 56, pp. 178–187.
8. Kuznetsov A.L. and Kochurov D.N. Analiz tonal'nosti teksta v sotsial'nykh media s ispol'zovaniem iskusstvennykh neironnykh setei. Materialy X Mezhdunarodnoi studencheskoi nauchnoi konferentsii "Studencheskii nauchnyi forum", UrFU, Ekaterinburg, Rossiya, 2018. `https://scienceforum.ru/2018/article/2018009449` (07.03.2019). (in Russian)
9. Khaikin C. Neironnye seti: polnyi kurs. Moscow, OOO I.D. Vil'yams Publ., 2006, 1104 p. (in Russian)
10. Williams R. and Zipser D. A learning algorithm for continually running fully recurrent neural networks. Neural computation, 1989, vol. 1, issue 2, pp. 270–280.

11. Understanding LSTMs. URL: http://colah.github.io/posts/2015-08-Understanding-LSTMs/ (16.10.18). (in Russian)
12. Jones K. A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation, MCB University, MCB University Press, 2004, vol. 60, no. 5, pp. 493–502.
13. Golub D. and Van Loun Ch. Matrichnye vychisleniya. Moscow, Mir Publ., 1999, 548 p. (in Russian)
14. Trenirovochnye korpusy tekstov na russkom yazyke. URL: http://study.mokoron.com/ (09.01.2019). (in Russian)
15. IMDB Dataset of 50K Movie Reviews. URL: https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews (12.02.2019). (in Russian)

Дата поступления в редакцию: 07.09.2019