

# Исследование методов векторного представления текстовой информации для решения задачи анализа тональности

И.В. Бондарева  
Факультет информационных технологий  
Брянский государственный  
технический университет  
Брянск, Россия  
e-mail: innagorda@ya.ru

Д.Г. Лагерев  
Факультет информационных технологий  
Брянский государственный  
технический университет  
Брянск, Россия  
e-mail: LagerevDG@yandex.ru

## Аннотация<sup>1</sup>

В данной статье рассматриваются методы векторного представления слов для решения задачи анализа тональности текстов. Описываются общие положения, актуальность, а также особенности данной задачи в контексте гранулярности текстов. Приводится формулировка понятия «word embedding» и его объяснение. Проведен обзор существующих современных методов анализа тональности текстовой информации на английском языке. Обращается внимание на проблемы анализа тональности, работа над которыми представляет собой перспективное направление в обработке текстовой информации. Особое внимание уделяется исследованию моделей векторного представления слов, по итогам которого сделан вывод об эффективности их применения в задаче анализа тональности.

## 1. Введение

Мнения занимают основное место почти во всех областях человеческой деятельности. Наши убеждения и представления о реальности и выбор, который мы делаем, в значительной мере зависит от того, как другие видят и оценивают мир. По этой причине, когда нам нужно принять решение, мы часто ищем чужие мнения. Это справедливо не только для людей, но и для организаций. Мнения и связанные с ними понятия, такие как чувства, оценки, отношения и эмоции являются предметом изучения анализа настроений (sentiment analysis).

Коммерческим фирмам, производящим какой-либо продукт, интересно знать мнение покупателей об этом продукте. Эти данные могут быть использованы

для повышения качества продукта, определения целевой аудитории, а также для определения главных достоинств и недостатков конкурентов. Анализ мнений (opinion mining) позволяет узнать мнение автора и определить эмоциональную окраску текстового сообщения. Он состоит из двух этапов:

- морфологический анализ для выделения сущностей, которые будут оцениваться,
- анализ эмоциональной окраски выражений, относящихся к этой сущности.

Анализ тональности является неотъемлемой частью многих задач искусственного интеллекта. Он используется в специальных автоматизированных программах для анализа историй сообщений и других приложениях, в которых человеко-машинное взаимодействие происходит при помощи естественного языка.

Данное исследование проводится с целью выбора эффективного метода векторного представления, который позволит решить задачу анализа тональности коротких текстовых сообщений на русском языке, полученных из социальных сетей.

## 2. Общие положения

Задача определения тональности текста – одна из задач обработки естественного языка, которая заключается в определении эмоциональной оценки, выражаемой автором высказывания относительно некоторого объекта (объекта реального мира, события, процесса или их свойств/атрибутов). Анализ тональности успешно применяется в таких областях, как социология [1] (анализ общественного мнения относительно событий), маркетинг [2] (оценка покупателями товаров и услуг), психология [3] (определение психоэмоционального состояния человека). Задача анализа тональности сводится к задаче классификации. Текст нужно отнести к одному из конечного набора классов эмоциональной окраски, например, «положительный», «отрицательный» или «нейтральный».

---

Труды Шестой всероссийской научной конференции "Информационные технологии интеллектуальной поддержки принятия решений", 28-31 мая, Уфа-Ставрополь, Россия, 2018

Анализ тональности может быть выполнен на трех уровнях детализации [4]:

1. **Уровень документа.** В задаче определения тональности на уровне документа рассматривается выражение общего положительного или отрицательного мнения относительно одного объекта. В такой постановке это задача бинарной классификации, но также возможно сформулировать задачу регрессии, например, если необходимо оценивать рейтинг документа или отзыва от 1 до 5.
2. **Уровень предложения.** Анализ тональности на уровне предложения заключается в классификации каждого предложения в документе. Общепринятый подход к решению данной задачи – выполнить сначала классификацию субъективности предложений, а затем уже классификацию тональности тех предложений, которые являются субъективными, на два (положительное, негативное) или три (положительное, нейтральное, негативное) класса. Лиу в своей работе 2012 года [5] утверждает, что нет никакого различия между уровнем документа и уровнем предложения, и предлагает рассматривать предложения как короткие документы.
3. **Уровень аспекта.** По сравнению с анализом тональности на предыдущих двух уровнях, аспектный анализ является более тонким и точным. Он позволяет определить мнение автора текста о различных признаках описываемых сущностей, называемых также целевыми. Для упрощения в большинстве алгоритмов объединяют извлечение аспектов и сущностей, это может называться либо извлечением аспектов, либо извлечением целей тональности.

Помимо основной задачи определения отношения автора текста к предмету, описанному в тексте, к анализу тональности относятся [6]:

- анализ эмоций;

- обнаружение сарказма;
- мультилингвальный анализ тональности.

### 3. Word embeddings

Использование слов в качестве подготовленных признаков в задачах обработки естественного языка – интуитивное и естественное решение, потому что слова являются основными значащими единицами для естественных языков. И не удивительно, что представлению слов в виде числовых векторов для последующего применения методов машинного обучения посвящено большое количество исследований. Числовое представление слов обозначается термином word embedding [7] и вытекает из идеи «вложения» дополнительной информации о слове в векторное представление.

На рис. 1. представлен примитивный вариант векторного представления слов с помощью one-hot кодирования [8].

История word embedding начинается около 30 лет назад. Первые эксперименты с ними проводили Хинтон и МакКлелланд в 1990 году, Хинтон и Румельзарт в 1985 и Эльман в 1990. С тех пор создание представлений слов установилось в двух семействах моделей: глобальных матричных методах факторизации и методах локального контекстного окна. Глобальная матричная факторизация работает с глобальным уровнем корпуса и основана на подсчете частоты употребления слов в корпусе, она хорошо представляет статистические характеристики слов [9]. Методы локального окна работают на более детальном уровне корпуса, в пределах небольшого окна слов, расположенных рядом. Такие методы успешно применяются в решении задач поиска аналогичных слов, но они не могут отображать статистические характеристики. Методы локального окна создают векторное пространство для слов исходя из такого предположения, что семантически близкие слова располагаются близко друг к другу [7]. Семантическое представление объясняет их эффективную применимость в задачах поиска аналогий.



Рис. 1. Представление слов с помощью one hot кодирования

Все методы векторного представления слов плохо обобщаются на морфологически богатые языки. Скорее всего, это происходит по причине проблемы «словарного взрыва». Такала [10] в своей работе предлагает метод векторного представления слов для морфологически богатого языка, который основан на разбиении слова на части: основу и окончание. Такала обнаружил, что word embedding на уровне части слова просты в реализации и могут превзойти методы word embedding на уровне целого слова в решении нескольких задач обработки естественного языка. Но в его работе не проводилось сравнение отличий подхода с разделением основы и окончания слова от подхода с использованием слова целиком.

Одна из проблем, возникающих в связи с использованием методов word embedding для задач обработки естественного языка в высоко морфологических языках, заключается в том, что даже если word embedding словаря большой, некоторые слова по-прежнему будут в нем отсутствовать. Эту проблему можно частично решить с помощью добавления этапа лемматизации слов перед поиском словаря векторных представлений. Лемматизация, выполненная с достаточной точностью, может полностью устранить данную проблему для некоторых морфологически богатых языков, но существуют и такие языки, в которых структура слов слишком сложная. Обработка таких слов, не вошедших в словарь, вызывает затруднения, поскольку представление слов не может быть аппроксимировано из письменной формы. Письменная форма слова содержит лишь малую часть семантического значения слова, поэтому методы локальных окон обрабатывают семантическую информацию исключительно из контекста, в котором появляется слово.

#### **4. Анализ тональности с использованием word embeddings**

Очевидно, что word embedding играют важную роль в моделях анализа тональности, основанных на глубоком обучении. Стоит отметить, что word embedding могут использоваться как дополнительные входные признаки для различных задач, в которых используются модели без нейронных сетей и глубоких моделей обучения.

Рассмотрим работы, в которых используются word embedding с размеченной тональностью. Применение для анализа тональности обычных текстовых методов, таких как CBOW или Skip-gram с целью обучения word embedding по контексту, может вызвать трудности, потому что слова с похожими контекстами, но противоположными полярностями тональности (например, «хороший» или «плохой») могут отображаться в векторы, расположенные в пространстве близко друг к другу. Поэтому были предложены методы word embedding с размеченной тональностью. В работе Масса [11] представлено

исследование таких word embedding, которые могут фиксировать как семантическую, так и тональную информацию. В работе Беспалова [11] показано, что n-граммная модель в сочетании с латентным представлением дает word embedding более подходящий для классификации тональности. Лабутов и Липсон [12] повторно переобучили существующие word embedding с использованием логистической регрессии и данных с размеченной тональностью.

Ле и Миколов [13] предложили концепцию вектора абзаца, чтобы сначала обучить представление фиксированной длины для частей текста переменной длины, включая предложения, абзацы и документы. Они экспериментировали с задачами классификации как предложений, так и документов и добились повышения производительности, что демонстрирует достоинства векторов абзацев в выявлении семантических особенностей, которые улучшают качество определения тональности текста.

В работах Д. Танга [14, 15] представлена модель обучения Sentiment-specific Word Embeddings (SSWE), в которой в словарных векторах отображается не только семантическая информация, но и информация о эмоциональной окраске. Ванг и Ксин [15] разработали нейронную архитектуру для обучения word embedding с эмоциональной окраской, объединив алгоритм обучения с учителем (размеченные оценки тональности) как на уровне документов, так и на уровне слов. Юй в своей работе [16] выбрал подход с использованием уточнения для получения векторов слов, в которых объединены семантические эмоциональные окраски.

В рамках задачи анализа тональности также изучается усовершенствование признаков и многозначные word embedding. Во и Жанг [17] изучали аспектный анализ тональности с использованием автоматических признаков, которые представляют собой дополнительные функции, полученные с использованием методов обучения без учителя. Ли и Джурафский [18] экспериментировали с применением многозначных word embedding в различных задачах обработки естественного языка. В результате проведенной работы показано, что, хотя такие векторные представления хоть и улучшают производительность в решении некоторых задач, но при этом мало влияют на решение задачи классификации. Пен [19] предложил методы обучения многозначных word embedding с заданными темами в задаче анализа тональности сообщений в Twitter.

Многоязычные word embeddings также могут применяться в задаче анализа тональности. Жоу [20] представил двуязычную модель Bilingual Sentiment Word Embedding (BSWE). В ней содержится информация о эмоциональной окраске в англо-китайских векторных представлениях, составленных из переводов вместо огромных параллельных

корпусов. Барнес [21] сравнил несколько типов двуязычных word embeddings и техник машинного перевода на основе нейронных сетей в задаче многоязычного аспектного анализа тональности.

Жанг [22] предложил подход выборочно-аддитивного обучения (SAL) в мультимодальном анализе тональности для решения проблемы факторов, которые вносят искажение. Такой подход позволяет устранить отдельные особые скрытые векторные представления, полученные с помощью нейронных сетей (например, сверточных). Чтобы достичь этого, реализованы две фазы обучения, а именно: фаза выбора идентификаторов искажающих факторов и их удаления.

## 5. Обзор методов LSA, Word2vec и Glove

Слова можно точно описать, используя их зависимость от контекста. Такой подход – главная идея таких методов, как LSA, Word2vec и Glove. Тем не менее, у этих алгоритмов есть различные особенности.

### 5.1. Latent semantic analysis

Латентно-семантический анализ (Latent semantic analysis, LSA) [23] – это эффективный статистический алгоритм, состоящий из двух основных шагов. Первый шаг – построение терм-документной матрицы [24]  $M$ . Размер матрицы  $M$  равен  $n*m$ , где строки определяют  $m$  термов, столбцы соответствуют  $n$  документам, а  $M[i, j]$  – частота появления термина  $i$  в документе  $j$ . Второй шаг – сингулярное разложение (Singular Value Decomposition, SVD), в котором  $M$  разлагается на три матрицы согласно уравнению (1): две из них  $U$ ,  $V^T$  являются ортогональными матрицами, а  $S$  – диагональной матрицей. Наконец, исходя из уравнения 2 для уменьшения семантического пространства, соответствующего  $M_k$ , будут использоваться только  $k$  наибольших сингулярных значений и их соответствующие сингулярные векторы из матриц  $U$  и  $V^T$ .

$$M = U * S * V^T. \quad (1)$$

$$M_k = U_k * S_k * V_k^T. \quad (2)$$

Метод LSA основывается на нескольких параметрах, таких как: локальные и глобальные частоты встречаемости слов, функции локального и глобального взвешивания и размерность семантического пространства.

### 5.2. Word2Vectors

Word2Vec – это инструмент (набор алгоритмов) для расчета векторных представлений слов, предложенный Миколовым в 2013 году [25]. Он реализует две основные архитектуры – Continuous Bag of Words (CBOW) и Skip-gram.

Непрерывный мешок слов (CBOW) позволяет предсказывать текущее слово, основываясь на его

контексте, который определяется соседними словами в окне. В CBOW используются три слоя. Входной слой соответствует контексту. Скрытый слой – проекции каждого слова из входного слоя в весовую матрицу, которая проецируется в третий выходной уровень. Последним этапом модели является сравнение ее вывода с самим словом, чтобы скорректировать его представление, основанное на обратном распространении градиента ошибки. Таким образом, целью нейронной сети CBOW является максимизация следующего уравнения (3):

$$\frac{1}{V} \sum_{t=1}^V \log p(m_t | m_{t-\frac{c}{2}} \dots m_{t+\frac{c}{2}}), \quad (3)$$

где  $V$  – размер словаря,  $c$  – размер окна для каждого слова.

Метод skip-gram решает обратную задачу: на основании одного слова предсказывается контекст. Последний шаг алгоритма – сравнение вывода с каждым словом в контексте с целью корректирования представления, основанного на обратном распространении градиента ошибки. Данный метод выполняет максимизацию следующего уравнения:

$$\frac{1}{V} \sum_{t=1}^V \sum_{j=t-c, j \neq t}^{t+c} \log p(m_j | m_t). \quad (4)$$

где  $V$  – размер словаря,  $c$  – размер окна для каждого слова.

Согласно работе Миколова [25], оба алгоритма обладают собственными преимуществами. Например, skip-gram наиболее эффективен при обработке небольшого корпуса обучающих данных. Более того, с помощью него хорошо описываются редко встречающиеся слова. С другой стороны, CBOW работает быстрее и лучше обрабатывает часто встречающиеся слова.

Однако обучение выходного вектора алгоритмов CBOW и skip-gram представляет собой одно из самых больших ограничений этих моделей, так как оно может быть тяжелой задачей, требующей больших вычислительных затрат. Для решения этой проблемы можно использовать два алгоритма. Первый – негативное семплирование. Основная идея алгоритма заключается в ограничении количества выходных векторов, которые требуется обновлять. Таким образом, только некоторые векторы обновляются случайным образом. Это распределение шума является вероятностным и используется в процессе семплирования. Второй алгоритм – иерархический softmax. Он основан на дереве Хаффмана. Фактически, это двоичное дерево, которое представляет все термины на основе их частоты появления в корпусе текста. Затем каждый шаг от корня до цели нормализуется. По мнению Миколова, каждый алгоритм обладает преимуществами по сравнению с другим в зависимости от обучающих данных. Например, негативное семплирование более

эффективно работает с векторами малой размерности и часто употребляемыми словами. Тем не менее, иерархический softmax показывает себя лучше в работе с редко употребляемыми словами.

### 5.3. Global Vectors

GloVe – один из наиболее популярных методов векторного представления слов, был предложен Пеннингтоном в 2014 году [9]. В его основе лежит способ подсчета частоты появления слов в текстовом корпусе. Фактически он состоит из двух основных этапов. Первый этап – построение матрицы смежности  $X$  из обучающего корпуса, где  $X_{ij}$  – частота появления слова  $i$  вблизи слова  $j$ . Тогда общее количество слов  $i$  в корпусе ( $V$  соответствует размеру корпуса) можно посчитать по формуле:

$$X_{ij} = \sum_k^V X_{ik} \cdot X_{jk} \quad (5)$$

Второй шаг – факторизация матрицы  $X$  для получения векторов. В работе Пеннингтона [9] показано, что информативны отношения вероятностей совпадения двух слов (уравнение (6)), а не сами вероятности совпадения, и они используются для кодирования этой информации в виде разности векторов.

$$F(w_i - w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}} \quad (6)$$

В уравнении (6)  $w_i$ ,  $w_j$  и  $w_k$  – три вектора слов,  $P_{ik} = X_{ik}/X_i$  – вероятность появления слова  $k$  в контексте слова  $i$ ,  $w$  – векторы слов и  $\tilde{w}_k$  – вектор контекстного слова.

Однако для сохранения линейности и предотвращения смешивания измерений, Пеннингтон использовал разности векторов и точечный результат аргументов в уравнении (6):

$$F((w_i - w_j)^E, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}} \quad (7)$$

Но окончательная модель не должна изменяться после замены  $w \rightarrow \tilde{w}$  и  $X \rightarrow X^T$ . Чтобы работало свойство симметрии, уравнение (7) следует преобразовать следующим образом:

$$w_i^T \tilde{w}_k + b_i + \tilde{b}_k = \log(X_{ik}) \quad (8)$$

В итоге Пеннингтон предложил использовать метод наименьших квадратов, как показано в уравнении (8), где  $f(x)$  – весовая функция.

$$J = \sum_{i,j=1}^V f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log(X_{ij}))^2 \quad (9)$$

### 6. Перспективы исследования векторного представления слов

Помимо работ, в которых сформировалось понимание того, что word2vec с негативным

семплированием и skip-gram неявно факторизуют матрицу поточечной взаимной информации (pointwise mutual information, PMI) [26], было проведено сравнительно мало работ, которые помогли бы достичь лучшего теоретического понимания векторного представления слов и его свойств (например, такого свойства, которое формулируется следующим образом: суммирование улавливает отношение аналогии). Апра в своей работе [27] представляет новую генеративную модель для word embedding, которая генерирует корпус случайным прохождением вектора аспекта и устанавливает некоторые теоретические обоснования относительно поведения аналогий. Гиттенс [28] дает более полное теоретическое обоснование аддитивной композиции и показывает, что векторы слов на базе skip-gram являются оптимальными в теоретико-информационном смысле. Кроме того, Мимно и Томпсон [29], показывают интересное соотношение между представлениями слов и представлениями контекстных слов, а именно то, что они не равномерно распределены по векторному пространству, а занимают узкий конус, диаметрально противоположный представлениям контекстных слов. Несмотря на перечисленные выше дополнительные наблюдения, у нас все еще недостаточно понимания относительно расположения и свойств word embedding, и необходимо проводить теоретические исследования по данной теме.

### 7. Выводы

В ходе проведения исследования было установлено, что на данный момент разработано большое количество методов векторного представления слов в задаче анализа тональности, в особенности на английском языке. А поскольку суть word embeddings во многом является черным ящиком, исследователи проверяют различные гипотезы относительно факторов, влияющих на конечный результат работы метода.

В итоге сделан следующий вывод: для анализа тональности коротких русскоязычных текстовых сообщений, полученных из социальных сетей, следует реализовать стемминг, лемматизацию, алгоритм Sentiment-specific Word Embeddings, использовать размеченные оценки тональности и обучать word embeddings на двух уровнях: целого сообщения и отдельного слова.

### Список используемых источников

1. Averchenkov V., Budylskii D., Podvesovskii A. (et. al.) Hierarchical Deep Learning: A Promising Technique for Opinion Monitoring and Sentiment Analysis in Russian-Language Social Networks // A. Kravets et al. (Eds.): CIT&DS 2015, Communications in Computer and Information Science, 2015. Vol. 535. P. 583-592.

2. García-Moya L., Anaya-Sanchez H., Berlanga-Llavori R. Retrieving product features and opinions from customer reviews // *IEEE Intelligent Systems*. 2013. Vol. 28. № 3, P. 19–27.
3. Cha M., Haddadi H., Benevenuto F., Gummadi, K.P. Measuring User Influence in Twitter: The Million Follower Fallacy. // *Proc. of the 4th International AAAI Conference on Weblogs and Social Media, USA, Washington, 2010*.
4. Kolkur S., Dantal G. Mahe R. Study of Different Levels for Sentiment Analysis // *International Journal of Current Engineering and Technology*. 2015. Vol. 5, №. 2.
5. Liu B. Sentiment Analysis and Opinion Mining // *Synthesis Lectures on Human Language Technologies*. 2012. Vol. 5. №. 1.
6. Liu B. Sentiment analysis: mining opinions, sentiments, and emotions // *The Cambridge University Press*, 2015.
7. Mikolov, T., Yih W., Zweig G. Linguistic regularities in continuous space word representations. // *Proc of NAACL-HLT 2013*. P. 746–751.
8. McGinnis W. Beyond one-hot: an exploration of categorical variables // *Data science, technology, Atlanta*. – 2015; URL: <http://www.willmcginnis.com/2015/11/29/beyond-one-hot-an-exploration-of-categorical-variables/> (дата обращения: 20.03.2018).
9. Pennington, J., Socher R., Manning C.D. Global Vectors for Word Representation. // *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing*, P. 1532–1543.
10. Takala, P. Word Embeddings for Morphologically Rich Languages // *Computational Intelligence and Machine Learning*. Belgium. Bruges. 2016. P. 27–29.
11. Mass A. L., Daly R. E., Pham P. T., Huang D., Ng A. Y., Potts C. Learning word vectors for sentiment analysis // *Proc. of the Annual Meeting of the Association for Computational Linguistics*. 2011.
12. Beshpalov D., Bai B., Qi Y., Shokoufandeh A. Sentiment classification based on supervised latent n-gram analysis. // *Proc. of the International Conference on Information and Knowledge Management*. 2011.
13. Le Q., Mikolov T. Distributed representations of sentences and documents. // *Proc. of the International Conference on Machine Learning*. 2014.
14. Tang D., Wei F., Yang N., Zhou M., Liu T., Qin B. Learning sentiment-specific word embedding for twitter sentiment classification. // *Proc. of the Annual Meeting of the Association for Computational Linguistics*. 2014.
15. Tang D., Wei F., Qin B., Yang N., Liu T., Zhou M. Sentiment embeddings with applications to sentiment analysis // *IEEE Transactions on Knowledge and Data Engineering*. 2016. Vol. 28. № 2.
16. Yu L.C., Wang J., Lai K.R., Zhang X. Refining word embeddings for sentiment analysis // *Proc. of the Conference on Empirical Methods on Natural Language Processing*. 2017.
17. Vo D-T., Zhang Y. Target-dependent twitter sentiment classification with rich automatic features. // *Proc. of the Internal Joint Conference on Artificial Intelligence*. 2015.
18. Li J., Jurafsky D. Do multi-sense embeddings improve natural language understanding? // *Proc. of the Conference on Empirical Methods in Natural Language Processing*. USA. Colorado. Denver. 2015. P. 1287–1292
19. Zhou H., Chen L., Shi F., Huang D. Learning bilingual sentiment word embeddings for cross-language sentiment classification // *Proc. of the Annual Meeting of the Association for Computational Linguistics*. 2015.
20. Ren Y, Zhang Y, Zhang, M and Ji D. Improving Twitter sentiment classification using topic-enriched multiprototype word embeddings. // *Proc. of AAAI Conference on Artificial Intelligence*. 2016.
21. Barnes J., Lambert P., Badia T. Exploring distributional representations and machine translation for aspectbased cross-lingual sentiment classification. // *Proc. of the 27th International Conference on Computational Linguistics*. 2016.
22. Zhang W, Yuan Q, Han J, and Wang J. Collaborative multi-Level embedding learning from reviews for rating prediction. // *Proc. of the International Joint Conference on Artificial Intelligence*. 2016.
23. Deerwester S., Dumais S.T., Furnas G.W., Landauer T.K., Harshman R. Indexing by Latent Semantic Analysis // *The American Society for Information Science*. 1990. Vol. 41. P. 391-407
24. Векторная модель // *MachineLearning.ru*. 2016. url: [www.machinelearning.ru/wiki/index.php?title=Векторная\\_модель](http://www.machinelearning.ru/wiki/index.php?title=Векторная_модель)
25. Mikolov T., Chen K., Corrado G., Dean J. Efficient Estimation of Word Representations in Vector Space // *Proc. of Workshop at ICLR*. 2013. P. 1301-3781.
26. Levy, O., & Goldberg, Y. Neural Word Embedding as Implicit Matrix Factorization. // *Advances in Neural Information Processing Systems*. 2014 P. 2177–2185.
27. Arora, S., Li, Y., Liang, Y., Ma, T., & Risteski, A. A Latent Variable Model Approach to PMI-based Word Embeddings // *Transactions of the Association for Computational Linguistics*. 2016 Vol. 4, P. 385–399.
28. Gittens, A., Achlioptas, D., & Mahoney, M. W. Skip-Gram – Zipf + Uniform = Vector Additivity. // *Proc. of the 55th Annual Meeting of the Association for Computational Linguistics*. 2017 P. 69–76.
29. Mimno, D., & Thompson, L. The strange geometry of skip-gram with negative sampling. // *Proc. of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2017. P. 2863–2868.