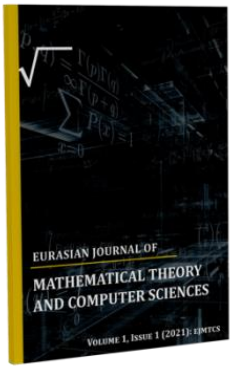
EURASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
Innovative Academy Research Support Center
IF = 7.906 www.in-academy.uz



WORD REPRESENTATION IN VECTOR SPACE USING WORD2VEC MODEL

Rakhmanov Askar
Department of "System and Applied Programming" Tashkent University of Information Technologies named after Muhammad al-Khwarizmi.
asqartr1.2.3dipu@gmail.com
Iskhakova Nargiza
Department of "System and Applied Programming"
Tashkent University of Information Technologies named after Muhammad al-Khwarizmi
Nargiza89@tuit.uz
Abduvalieva Zebiniso
Department of "System and Applied Programming"
Tashkent University of Information Technologies named after Muhammad al-Khwarizmi zebinisoabduvaliyeva@gmail.com
https://doi.org/10.5281/zenodo.15524744

ARTICLE INFO

Received: 20th May 2025
Accepted: 26th May 2025
Online: 27th May 2025

KEYWORDS

Word2Vec, Skip-gram and CBOW, vector representation of data, natural language processing, text data classification.

ABSTRACT

The article discusses the word2vec model, which is an effective method for learning vector representations of words, widely used in natural language processing tasks. The main architectures of the model - Skip-gram and CBOW, as well as key parameters that affect the quality of the resulting vector representation are described. It is shown that the use of word2vec allows transforming words into dense vectors that reflect their semantic and syntactic properties, which significantly improves the results compared to traditional text representation methods.

ПРЕДСТАВЛЕНИЕ СЛОВ В ВЕКТОРНОМ ПРОСТРАНСТВЕ ПРИМЕНЯЯ МОДЕЛЬ WORD2VEC

Рахманов Аскар
Кафедра «Системное и прикладное программирование» Ташкентский университет информационных технологий имени Мухаммада аль-Хорезми.
asqartr1.2.3dipu@gmail.com
Исхакова Наргиза
Кафедра «Системное и прикладное программирование»
Ташкентский университет информационных технологий имени Мухаммада аль-Хорезми
Nargiza89@tuit.uz
Абдувалиева Зебинисо
Кафедра «Системное и прикладное программирование»
Ташкентский университет информационных технологий имени Мухаммада аль-Хорезми zebinisoabduvaliyeva@gmail.com
https://doi.org/10.5281/zenodo.15524744

ARTICLE INFO

ABSTRACT

EURASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
Innovative Academy Research Support Center
IF = 7.906 www.in-academy.uz

Received: 20th May 2025
Accepted: 26th May 2025
Online: 27th May 2025
KEYWORDS
Word2Vec, Skip-gram и CBOW, векторное представление данных, обработка естественного языка, классификация текстовых данных.

В статье рассматривается модель word2vec -который является эффективным методом обучения векторных представлений слов, широко применяемый в задачах обработки естественного языка. Описаны основные архитектуры модели - Skip-gram и CBOW, а также ключевые параметры, влияющие на качество получаемых векторное отображение. Показано, что использование word2vec позволяет преобразовывать слова в плотные векторы, отражающие их семантические и синтаксические свойства, что значительно улучшает результаты по сравнению с традиционными методами представления текста.

1. Введение

Модели обработки естественного языка позволяют компьютерам понимать, объяснять и формировать человеческий язык, в который входит большой спектр задач, таких как работа с чат-ботами, перевод языков, а также классификация текста (например, анализ эмоционального состояния, классификация по темам). Эти задачи позволяют оптимально упорядочивать и исследовать большие объёмы текстовой информации, а также внедрять автоматизированные системы поддержки принятия решений в разных сферах деятельности.

Одним из направлений обработки текстовых данных является классификация текстовых данных. Текстовые данные представляют собой информацию, которую могут быть в разном формате и носить в себе различные данные, включая документы организации, электронную почту, форумы для обсуждений, социальные сети, заявки, публичные записи, отзывы пользователей, а также вопросы и ответы от сотрудников службы поддержки клиентов. Текст имеет очень обширный источник знаний [1]. По причине не структурированности текста получение глубоких выводов из текстовых данных нередко сопряжено с трудностями и требует много времени.

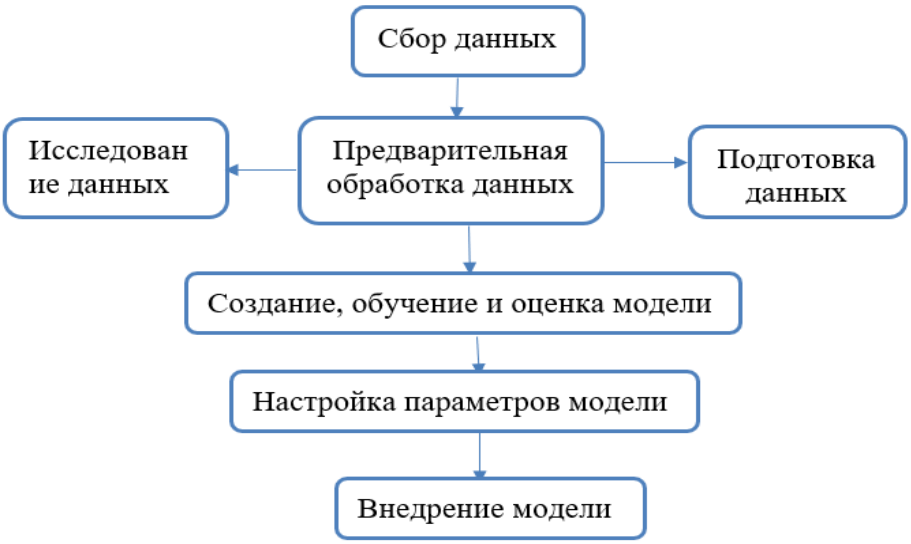


Схема 1.1 Этапы классификации текстовых данных

Первый этап, сбор данных- собрать значительный объём текстовых данных, для предметной области задачи. Данные могут поступать из разных источников, включая веб-сайты, новостные статьи, платформы социальных сетей, данные собранные из документов организации. Далее следует предварительная обработка данных. После сбора данные следует очистить и редактировать в формат, наиболее подходящий для дальнейшего преобразования и классификации [2]. Данный этап включает удаление лишней информации из текста, таких как стоп-слова, знаки препинания и специальные символы, преобразование текста в строчные буквы, Получение базовой формы слова, Сведение слов к их исходной форме и другие методы нормализации текста. Следующим этапом считается подготовка текстовых данных – из предварительно обработанного текста выделить ключевые свойства, которые могут быть использованы для обучения модели машинного обучения. Наиболее популярные методы извлечения признаков -это векторные представления слов (word embedding), мешок слов (bag-of-words), TF-IDF, Word2Vec и другие продвинутые методы глубокого обучения. После проведения вышеперечисленных операций необходимо построить, обучить и проверить модель, выбрать подходящий алгоритм машинного обучения для построения модели на основе полученных признаков[3]. Для обучения модели используется набор выбранных обучающих данных, представляющих нужную область. После обучения модель тестируют на наборе входных тестовых текстовых данных, для оценки качества. Наиболее распространённые критерии оценки для классификации текста - точность, полнота, насыщенность и матрица ошибок. По результатам нескольких испытаний, параметры системы можно скорректировать, чтобы она работала для всех вариантов [4]. Когда проведены все операции над текстовыми данными для их классификации по определенным критериям наступает этап внедрения разработанной системы. Разработанная система должна классифицировать входные данные по категориям.

2. Постановка задачи

Рассмотрим модель Word2Vec для классификации текста. Вектор слова - это строка чисел с реальными значениями (в отличие от фиктивных чисел), где каждая координата отражает измерение значения слова, а семантически близкие слова имеют похожие векторы. Это значит, что слова, такие как "mamalakat" и "boshqarish", должны иметь похожие векторы со словом "davlat" (из-за сходства значений), тогда как слово "xonadon" будет находиться далеко. Иными словами, слова, которые используются в похожем контексте, будут отображены близко друг к другу в векторном пространстве. Таким образом, модель word2vec принимает на вход одно слово (представленное в виде one-hot кодирования среди всех слов корпуса) и пытается предсказать вероятность того, что случайно выбранное слово из корпуса находится в близком контексте к входному слову. Это означает, что для каждого входного слова существует n выходных вероятностей, где n -это общий размер словаря корпуса. В процессе обучения учитывается только контекст слова, а не все слова корпуса. Это значит, что, если на вход подаётся слово "mamlakat", слово "boshqaruv" будет иметь более высокую оценку вероятности, чем "xonadon", поскольку оно ближе по контексту, то есть это учитывается в процессе обучения [5-7].

Иными словами, модель пытается предсказать вероятность того, что другие слова корпуса принадлежат к контексту входного слова.

Для преобразования текстовых данных в векторные представления слов с помощью Word2Vec следует проделать следующие действия. В начале текс пред обрабатывается, то есть разбивают текст на предложения. Далее разбивают слова на токены. То есть на отдельные слова. После разбиения на токены предложение следует очистить от всего лишнего, что мешает его переводу в вектор. К лишнему можно отнести пунктуацию, запятые, такие слова как и, или, иногда. Слова, которые не будут менять смысл предложения. На следующем этапе происходит обучение модели Word2Vec. Для этого следует выбрать архитектуру. Skip-gram или CBOW. После выбора архитектуры следует настроить параметры вектора[6]. Размер вектора- это количество измерений в векторном представлении каждого слова. Чем больше размерность, тем более детально модель может захватывать смысловые нюансы слов, но при этом увеличиваются требования к памяти и вычислительным ресурсам. Обычно выбирают значения от 100 до 500.

В модели word2vec существуют две основные архитектуры обучения векторных представлений слов[8]. CBOW (Continuous Bag of Words) Эта модель пытается предсказать текущее (центральное) слово на основе слов из его контекста (окружающих слов). Например, в предложении " Iqtisodiy taraqqiyot va _____ qisqartirish vazirligi модель CBOW по словам контекста ("Iqtisodiy ", " taraqqiyot ", " va ") пытается угадать пропущенное слово, например, “kambag‘allikni” CBOW работает быстрее и обычно лучше подходит для небольших наборов данных[9]. Она усредняет контекстные слова и на их основе предсказывает целевое слово. Вторая модель Skip-gram.

Эта модель работает наоборот: по текущему слову она пытается предсказать слова из его контекста. Например, если взять слово " vazirligi ", модель Skip-gram будет пытаться предсказать слова, которые часто встречаются рядом с ним, например, " Qoraqalpog‘iston Respublikasi", "O’zbekiston Respublikasi" и т.д. Skip-gram обычно обучается дольше, но лучше справляется с редкими словами и большими корпусами, так как учится предсказывать контекст по слову. Следующий шаг, получить вектор для каждого слово из словаря.

3. Метод решения

Математическая модель Word2Vec (Skip-gram) будет выполняться по формуле приведенной ниже[9-10]. Применим формулу для перевода предложения из первой категории. Ранее в статьях было приведено полное описание категорий и документов. Применим к следующему предложению “ Mamlakatimizning ijtimoiy va ishlab chiqarish infratuzilmasini izchil rivojlantirish, qishloq va mahallalarni yanada obod qilish, joylarda qulay tadbirkorlik va investitsiya muhitini shakllantirish, shuningdek, iqtisodiyot tarmoqlari va ijtimoiy sohaga investitsiyalarni keng jalb qilish orqali yangi ish o‘rinlarini yaratish, aholining turmush darajasini yaxshilash va kambag‘allikni qisqartirish maqsadi”.

Цель, максимизировать вероятность появления контекста *c* их корпуса при данном слове *S*.

EURASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
Innovative Academy Research Support Center
IF = 7.906 www.in-academy.uz

$$r(c|S;\theta) = \frac{\exp(e_c\ e_s)}{\sum_{c' \in} \exp(e_{c'}\ e_s)} \quad (1.1)$$

Где V-словарь из которого берутся слова, $S \in V$-слово, $e_S, e_c \in R^t$-векторное представление слов из контекста. Где $\theta = \{e_S, e_c\}$-параметры модели. После максимизирования логарифмической функции получаем:

$$G(\theta) = \sum_{S,c\in D} \log r(c|S;\theta) \qquad (1.2)$$

Для повышения эффективности используется отрицательное сэмплирование, где оптимизируется сдедующая функция. Вместо того чтобы сравнивать каждое слово со всеми словами в словаре, что требует очень много времени, сравнение проводится только с несколькими случайными "отрицательными" примерами. Модель учится отличать настоящие связи между словами от случайных словосочетаний.

$$\max_{\theta} \sum_{S,c\in D} \left[\log \sigma(e_S\ e_c) + \sum_{i=1}^{n} E_{c_i} \sim R_k \log\sigma(-e_S\ e_c)\right] \qquad (1.3)$$

Где $(S, c) \in D$-все пары слов из обучающего корпуса, $e_S\ e_c$-векторное представление слова из его контекста, $\sigma(y) = \frac{1}{1+e^{-y}}$-сигмоидальная функция, переводящая значение в диапазоне 0, $n -$ количество отрицательных примеров для каждого положительного, R_k-распределение по которому выбирают отрицательные контексты.

Если разобрать пример то получим, $\log \sigma(e_S\ e_c)$-повыщает вероятность, что данное слово и его контекст действительно связаны , их скалярное произведение большое, их сигмоидная близка к 1. $\sum_{i=1}^{n} E_{c_i} \sim R_k \log\sigma(-e_S\ e_c)$ данная часть формулы означает минимизацию вероятности, не правильного корпуса и связывание его с целевым словом, то есть его сигмоида близка к 0. В результате суммируются все пары по всему тексту и предложениям[10]. После обучения используется полученные векторы как векторное отображение слов.

Для векторизации предложения применяется нижеследующая формула

$$S_{predlojeniya} = \frac{1}{k} \sum_{i=1}^{k} S_{w_i} \qquad (1.4),$$

k-количество слов в предложении.

После применения выщеприведенных формул получим следующий вектор: [0.05957031 0.01489258 -0.00811768 0.09619141 -0.03759766 0.00146484 0.01708984 0.0637207 -0.07739258 0.03271484] первые 10 элементов вектора.

Приведенный способ- это метод быстрого и эффективного обучения word2vec, акцентируясь на различии между настоящими и случайными парами слов. Формула отражает этот процесс, максимизируем вероятность для правильных пар и минимизируем для неправильных.

4. Вывод

В данной работе рассмотрена модель word2vec - эффективный и широко используемый метод обучения векторных представлений слов. В статье были проанализированы основные архитектуры модели - Skip-gram и CBOW, а также ключевые параметры настройки, влияющие на качество получаемых векторных отображений. Выявлено, что word2vec позволяет преобразовывать слова в плотные векторы, отражающие их семантические и синтаксические свойства, что значительно

превосходит традиционные методы представления текста, такие как one-hot кодирование или мешок слов. В данной статье был рассмотрен перевод слов в векторы по категориям. Применяя векторное пространство слов.

Практическое применение word2vec охватывает широкий спектр задач обработки естественного языка, включая машинный перевод, распознавание речи и анализ текстов. Использование обученных моделей и возможность обучения на больших корпусах текстов делают word2vec мощным инструментом для дальнейших исследований и разработки интеллектуальных систем. Можно сделать заключение, что word2vec представляет собой важный шаг в развитии методов обработки естественного языка, открывая новые возможности для глубокого понимания и анализа текстовых данных.

References:

1. Г. Д. Валиева, Р. Б. Камаева, М. Г. Усманова, “КЛАССИФИКАЦИЯ ТЕКСТОВ”. Вестник Башкирского университета. Т. 24. №3. С 729-732, 2019.
2. Цитульский. А.М, Иванников. А.В, Рогов И. С, “ NLP - ОБРАБОТКА ЕСТЕСТВЕННЫХ ЯЗЫКОВ ”. Научно-образовательный журнал для студентов и преподавателей «StudNet» УДК-004. С 467-475, №6/2020
3. M. Goudjil, M. Koudil, M. Bedda, and N. Ghoggali, “A novel active learning method using SVM for text classification,” International Journal of Automation and Computing, vol. 15, no. 3, pp. 290–298, 2018.
4. Isaeva, M., Yoon, H., Y.: Paperless university — How we can make it work?. In: 15th International Conference on Information Technology Based Higher Education and Training (ITHET). pp. 1–8 (2016). Luo, H., Fan, Y., Wu, C.: Overview of Workflow Technology. J. Softw. 11, 78-82
5. Fan, Yusun: Base on Workflow Management Technology. Beijin:Tsinghua University Press, 32, (2001)
6. DerryJatnikaa, Moch Arif Bijaksanaa ,ArieArdiyanti Suryania. International Conferenceon,12-13September 2019 Word2Vec Model Analysis for Semantic Similarities in English Words. Computer Science and Computational Intelligence 2019 (ICCSCI)
7. Chen, Hong-na, Zu, Xu, Zhou, Feng: On the Developing Situation, Research Content and Trend of Workflow Technology. Journal of Chongqing Instiute of Technology. 20(2), 65-69 (2006)
8. Li, Zhao, Qing, Li, Farong, Zhong: A Visual Modeling Framework of Workflow Systems Based on CCS. Semantics, Knowledge and Grid. Fifth International Conference. pp. 200-207 (2009)
9. Abduvalieva Z.A , Marisheva L.T , Latipova N.X , Sheyna N.E. Structure and functional features of document management systems on the example of the department. Journal of Northeastern University, Volume 25 Issue 04, 2022.
10. Dinesh, P, Mital, Goh, Week, Leng School of Electrical & Electronic Engineering Nanyang Technology University Nanyang Avenue, Singapur 2263. Text segmentation for automatic document processing. Rosemont, IL, USA. pp. 132-133 (1995).