УДК 533.9.02

*А. С. Забавин (гр. ПИИ(м)-31, Тихоокеанский государственный университет)[[1]](#footnote-1)*

*Научный руководитель Э. М. Вихтенко (канд. физ. мат. наук, доцент высшей школы кибернетики и цифровых технологий Тихоокеанского государственного университета)*

**Исследование и анализ технологий морфологического и семантического анализа и их применение в задаче информационного поиска**

**Аннотация**

Предметом статьи является анализ и разбор технологий анализа естественного языка. Истоки и устоявшиеся подходы к решению задач компьютерной лингвистики как морфемный, морфологический и семантический разбор слов. Использование традиционных подходов, а также применение машинного обучения. Детально рассмотрена архитектура моделей семантического анализа семейства Word2Vec, их применение, преимущества и недостатки.

**Введение**

В настоящее время все шире используются интерфейсы взаимодействия человека с компьютером не требующий от человека специальных знаний о языке программирования, предметной области и даже навыков управления. В 2025 году никого не удивишь голосовыми помощниками и системами распознавания движений. В числе подобных интерфейсов и общение с компьютером на естественном языке или через «естественно-языковой интерфейс». Что под этим подразумевается? Подразумевается то что слова могут иметь определяемые компьютером значения близкого уровня синонимичности в определенном контексте взаимодействия программы и человека, и высказывается некоторое предположение что меру этой связи между различными словами и определенной по смыслу командой программы можно уловить. Однако для решения подобной задачи общения с машиной на естественном языке требует подробной деконструкции такого понятия как «язык». Если не вдаваться в подробности, то этим занимается лингвистика, и уже «Компьютерная лингвистика». «Компьютерная лингвистика» или computational linguistics, это сплав лингвистики, информатики, искусственный интеллекта и математики для моделирования языка с помощью вычислительных методов.

**Основная часть**

Одной из фундаментальных задач компьютерной лингвистики является декомпозиция слов на уровни их структурной и смысловой организации, что включает морфемный, морфологический, семантический и синтаксический анализ. Эти этапы позволяют не только разобрать слово как формальную единицу, но и передать его функциональную роль в тексте, а также содержательную нагрузку.

Морфемный разбор направлен на выделение минимальных значимых частей слова — морфем, таких как корень, приставка, суффикс, постфикс и интерфикс. В NLP (Анализе Естественного языка), как правило используется понятие «лексема» и «лемма». Лексема – это часть слова без изменяемых морфем, это как правило базовая форма слова представляющая его как смысловую единицу. Лемма часто синоним лексемы, но не всегда и для разных языков это несоответствие может отличаться. Лемма – это именно «базовая, нормальная, форма», корень слова, а лексема все же может включать грамматические варианты, а также сама по себе не быть употребимой.

Морфологический разбор сосредоточен на определении грамматических характеристик слова в конкретном контексте: части речи, числа, падежа, времени, лица и т.д. Этот процесс требует учета парадигм словоизменения, и по сути является продолжением анализа морфем слова. Традиционные способы выделения лексемы из слова основаны на словарях и правилах. Процесс отбрасывания изменяемых частей от слова и оставления лексемы – называется «Стемминг» [7]. Самым известным стеммером на текущий момент является написанный Мартином Портером Porter2 Stemmer или «Snowball Stemmer». Он представляет собой набор предопределенных алгоритмов (правил) для каждого языка которые отбрасывают окончания от слова. Такой традиционный подход работает довольно быстро и позволяет сократить размерности в задачах классификации текста, а также уменьшить словарь модели при обучении. Как было сказано выше лемматизация требует возвратить словарную (нормальную) форму что требует словарей и соответственно работает медленнее.

Семантический разбор (или семантический анализ) выходит за рамки формальной структуры слова в отличии от морфемного, фокусируясь на его смысле. Он включает определение лексического значения, выявление синонимии, антонимии, полисемии, а также контекстуальной интерпретации. Он включает устранение лексической неоднозначности (disambiguation), например, различение значений слова «ключ» («ключ от двери» vs. «ключ реки»), а также построение векторных представлений слов (word embeddings), таких как Word2Vec, GloVe или модели на основе трансформеров (BERT, RoBERTa), которые кодируют слова в многомерные пространства, отражающие их смысловые отношения. Эти подходы позволяют системам оценивать семантическую близость слов и учитывать их роль в предложении, что особенно важно для задач анализа настроений, вопросно-ответных систем и машинного понимания текста.

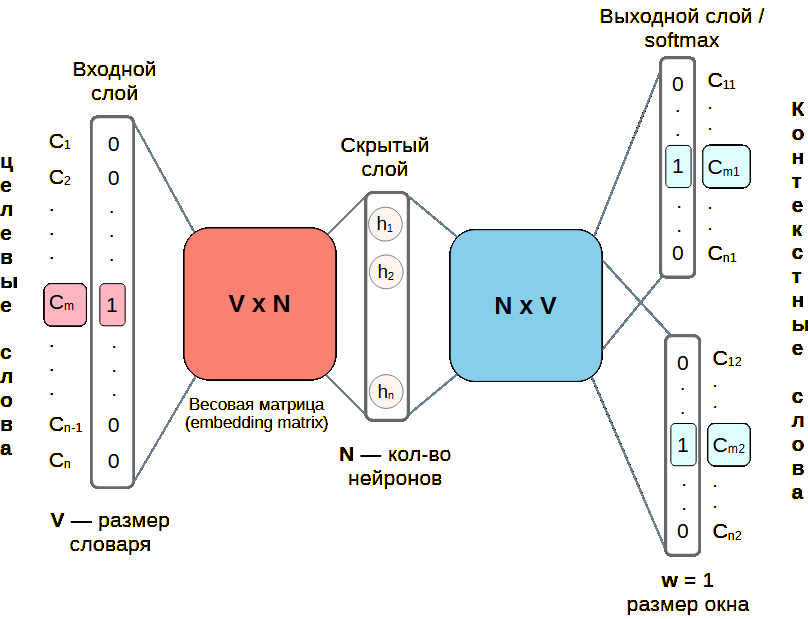
Синтаксический разбор неразрывно связан с семантическим анализом и исследует структуру предложения, выявляя иерархические связи между его компонентами. Современные методы, такие как dependency parsing, позволяют строить графы зависимостей, которые используются в машинном переводе, извлечении информации и генерации ответов.

Все четыре уровня анализа тесно взаимосвязаны: морфемный и морфологический разборы формируют основу для синтаксического анализа, который, в свою очередь, служит предпосылкой для семантической интерпретации, морфемный состав слова влияет на его морфологические свойства, а грамматические категории, в свою очередь, могут модулировать семантику (например, временные формы глагола). Современные системы NLP всё чаще интегрируют эти этапы в конвейерные архитектуры, используя глубокие нейронные сети (например, LSTM, трансформеры) для одновременного обучения на всех уровнях. Это позволяет достигать высокой точности в задачах, где важна контекстуальная и структурная информация, таких как суммаризация текстов, диалоговые системы и автоматическое аннотирование данных.

**Word2Vec**

Как было сказано выше на сегодняшний день основной подход к семантическому анализу слов заключается в создании векторных пространств определенных языков, полученных ИНС моделями на обучении огромного количество «смыслового» текста (корпусов текста). Свои истоки архитектуры моделей для обработки естественного языка ведут из рекуррентных нейронных сетей (RNN) и со временем эволюционировали в огромные ансамбли трансформерных моделей по типу ChatGPT, Deepseek, BERT и т.д. Остановимся на некоторой промежуточной архитектуре моделей со «скользящим контекстным окном»: Word2Vec, которые до сих пор вполне применимы для решения определенных задач обработки естественного языка.

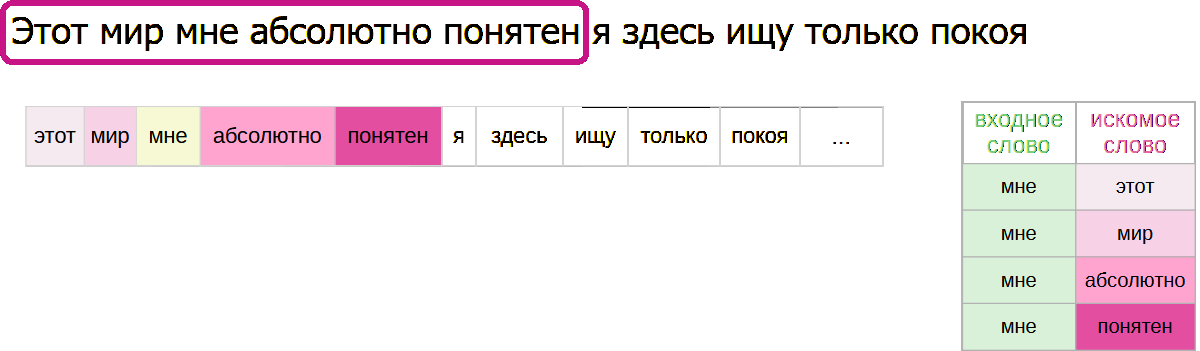
Архитектура модели была одним из прорывных достижений в этой области, предложенный Томашем Миколовым и его коллегами алгоритм Word2Vec, который преобразовал подход к моделированию семантики слов, представлена на рисунке 1. V - количество слов в словаре после обучения, каждое слово в словаре описывается как вектор с однократным кодированием (двоичный вектор, в котором только позиция соответствующего слова имеет значение 1). Словосочетание с пропуском или skip-gram – это подтип модели Word2Vec позволяющий предсказывать семантические близкие слова к целевому.



*Рис. 1. Архитектура Word2vec ИНС (skip-gram), 1 скрытый слой, окно = 1*

В ходе исследования данной статьи с помощью пакета gensim была обучена модель 4corpora\_3,5Msentences.gz.bin на основе 3 корпусов текста русского языка Лейпцигского университета [8] и архива статей википедии за 2020 год проекта Corus [4].

Составление фактического датасета для обучения модели представлено на рисунке 2.



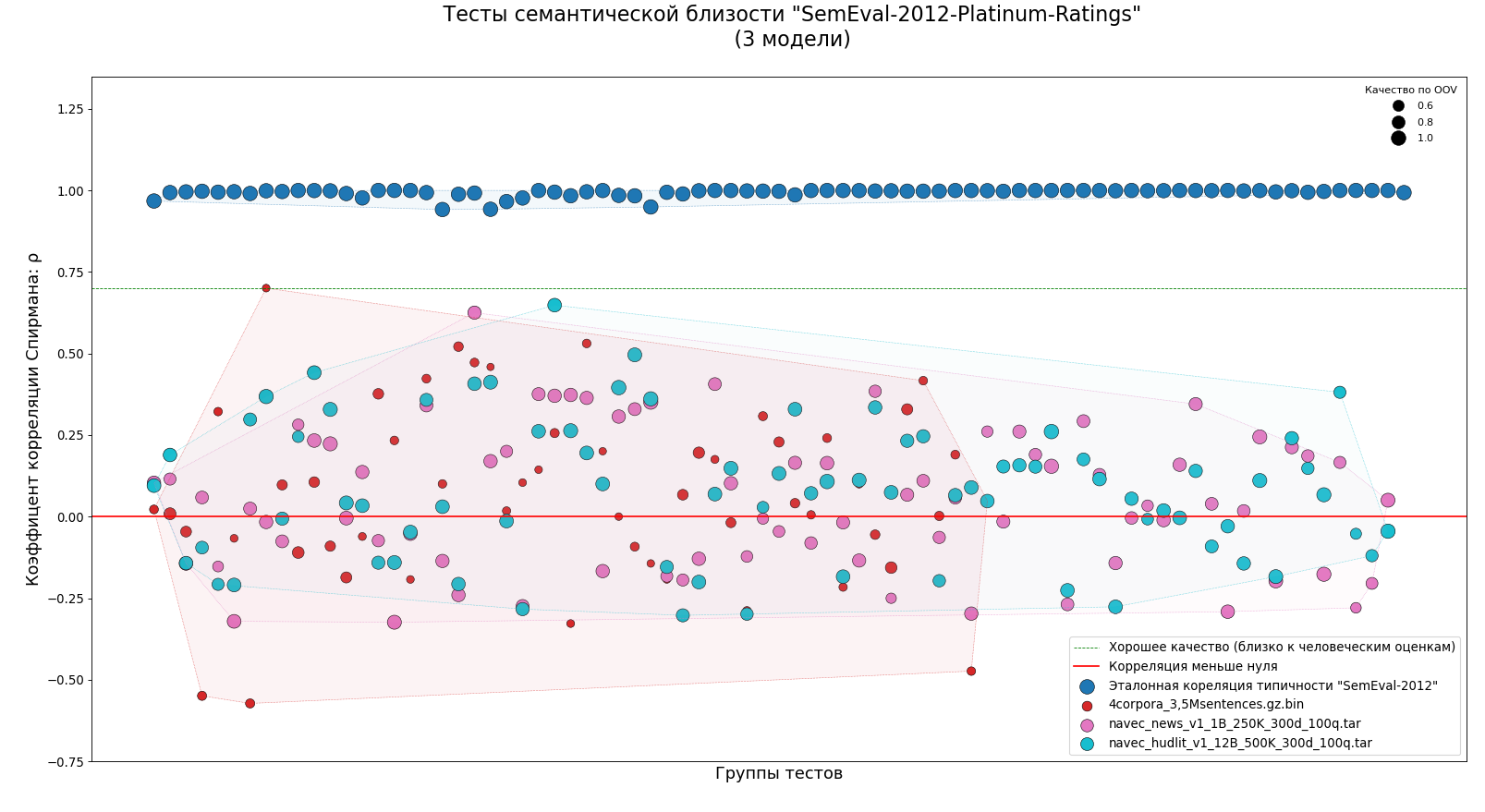
*Рис. 2.* Принцип составления датасета для получения Skip-gram-модели Word2Vec для корпуса текста (рамка — контекстное окно, градации цвета - вероятности)

Для Skip-gram модели объективная функция (функция потерь) — это логарифмическая вероятность окружающих n-искомых слов вокруг целевого слова [2]:

|  |  |
| --- | --- |
|  | (1) |

Обучение производилось с такими параметрами: количество воркеров 8, размер вектора слова 300, максимальное количество токенов в словаре 500 000, контекстное окно 5, эпох 5. В алгоритме стохастического градиентного спуска для настройки модели эффективная альфа влияет на то, насколько сильно корректируется модель после оценки каждого обучающего примера, по умолчанию при обучении использовалось α= 0.025.

Для оценки насколько качественно модель улавливает семантическую связь между словами использовался тест SemEval-2012 [9]. Тест представляет собой собранные ответы людей на различные семантические соотношения пар слов и выставлены оценки людьми. Для меры качества модели используется корреляция Спирмана между оценками связи пар слов людьми в тесте и полученные от обученной модели, связь определяется через косинное расстояние. Для сравнения использовались две дистиллированные модели проекта Natasha: news\_v1\_1B\_250K\_300d\_100q, hudlit\_v1\_12B\_500K\_300d\_100q [5]. Результаты представлены на рисунке 3.

*Рис. 3. Коэффицент Спирмана для моделей по тесту SemEval-2012*

Как видно из пузырьковой диаграммы, лучше всего улавливает семантическую близость модель navec обученная на корпусах из 12 миллиардов слов художественной литературы. Немаловажен размер точек он представляет насколько часто при инференсе слов у модели случалась ошибка Out Of Vocabulary (OOV), хотя обученная в ходе статьи модель 4corpora\_3,5Msentences иногда и давала результат порядка 0.75, что считается хорошим качеством, однако в данном наборе слов модели удалось найти векторное представление слов в лучшем случае в 50% случаев. Результаты с OOV больше чем 75% отбрасывались поэтому мы видим что модель 4corpora\_3,5Msentences в конечных тестах вообще не смогла добиться значимых оценок семантической близости на этих тестовых наборах.

**Заключение**

Многоуровневый анализ текста — от морфемного и морфологического до синтаксического и семантического — обеспечивает фундамент для решения сложных задач NLP, анализ настроений, выделение основной мысли (суммаризация) и генерация текста. В этом контексте особое значение приобретают методы представления слов в числовом виде, позволяющие моделям учитывать, как лексические, так и контекстуальные особенности языка. Эти аспекты имеют потенциал для улучшения взаимодействия интерфейса программ с пользователем, а также улучшения качества поиска в базах знаний. Word2Vec, основанный на обучении плотных векторных представлений слов (word embeddings) с использованием нейронных сетей, позволил зафиксировать лексическую и семантическую близость слов в многомерном пространстве. Это значительно повысило эффективность таких задач, как классификация текста, рекомендательные системы, а также поиска по контексту, вопросно-ответные системы и нечеткий поиск. Однако подход имеет ограничения: он не учитывает полисемию (многозначность слов), а качество векторов зависит от объема и разнообразия обучающего корпуса. Однако Word2Vec остаётся актуальным благодаря своей простоте, скорости обучения и возможности использования в ресурсоограниченных средах. Его применение особенно ценно в задачах, где контекстуальная гибкость не критична, а набор ответов на поставленные человеком вопросы программа выбирает из заранее обозначенных.

**Список источников**

1. Крылов В. “Что такое эмбеддинги и как они помогают искусственному интеллекту понять мир людей” // Текст: непосредственный // Наука и Жизнь. 2023. № 9 (399). C. 10–11.
2. Tomas Mikolov Kai Chen G. C. “Efficient Estimation of Word Representations in Vector Space” // Текст: электронный // arXiv [Электронный ресурс]. URL: https://arxiv.org/abs/1301.3781.
3. Mikolov T. “Distributed Representations of Words and Phrases and their Compositionality” // Текст: электронный // arXiv [Электронный ресурс]. URL: https://arxiv.org/pdf/1310.4546.pdf.
4. Кукушкин А. “Corus - набор корпусов текста русского языка” // Текст: электронный // natasha.github.io [Электронный ресурс]. URL: https://github.com/natasha/corus.
5. Кукушкин А. “Проект Natasha. Набор качественных открытых инструментов для обработки естественного русского языка (NLP)” [Электронный ресурс]. URL: https://habr.com/ru/articles/516098/.
6. Кукушкин А. “SlovNet - библиотека Python для моделирования NLP на основе глубокого обучения для русского языка” // Текст: электронный // natasha.github.io [Электронный ресурс]. URL: https://github.com/natasha/slovnet.
7. Портер М. Russian stemming algorithm // Текст: электронный // snowball.tartarus.org [Электронный ресурс]. URL: http://snowball.tartarus.org/algorithms/russian/stemmer.html.
8. Корпус русского текста Лейпцигского университета // Текст: электронный // wortschatz.uni-leipzig.de [Электронный ресурс]. URL: [hhttps://wortschatz.uni-leipzig.de/en/download/Russian](https://habr.com/ru/companies/econtenta/articles/303458/).
9. SemEval-2012: Semantic Evaluation Exercises. Task 2: Measuring Degrees of Relational Similarity // Текст: электронный // sites.google.com [Электронный ресурс]. URL: https://sites.google.com/site/semeval2012task2/home?authuser=0.

1. Забавин А. С., 2025 [↑](#footnote-ref-1)