Введение в NLP

Антон Легченко 14.07.25

Антон Легченко

Korona tech, Старший дата-аналитик

- Работаю в R&D 3 года
- Аспирант ММФ НГУ
- Работаю преймущественно с NLP, распознаванием речи
- Занимал 1ое место на AIJ 24 и Avito ML cup 25

Зачем нужно NLP?

- Автоматизация рутинных задач
- Поиск информации в больших объёмах данных
- Общение между людьми и машинами

Зачем нужно NLP?

Основные задачи NLP

Классификация текста

- Определение тональности (sentiment analysis)
- Тематическая классификация (topic classification)
- Обнаружение спама

Основные задачи NLP

Извлечение информации

- Распознавание именованных сущностей (NER)
- Извлечение фактов и событий
- Разметка частей речи (POS-tagging)
- Определение отношений (relation extraction)

Основные задачи NLP

Текст2Текст

- Машинный перевод
- Саммаризация
- Вопрос-ответные системы

Основные задачи NLP

Генерация текста

- Авто-дополнение кода
- Авто дополнение сообщений и тд.

Диалоговые системы

Генерация текста

- Чат-боты
- Агентные системы
- Интеллектуальный поиск и тд.

История NLP: этапы развития

- 1950–1980: Символьные и лингвистические подходы
- 1980–2010: Статистические методы и n-граммы
- 2000–2010: Векторные представления и увеличение данных
- 2010–2020: Глубокое обучение и нейросетевые представления
- 2018–н.в.: Трансформеры
- 2021-н.в Instruct модели

Подготовка данных в классических NLP-моделях

- Очистка текста
- Токенизация
- Стемминг / лемматизация
- Удаление стоп-слов
- Преобразование текста в числовые признаки

Языки с богатым словообразованием

- Примеры языков: русский, финский, турецкий, чешский и др.

- Однокоренные слова могут существенно различаться по написанию: "кот", "кота", "котам", "котов", "котовский", "подкотовый" и т.д.
- Одна и та же лексическая единица → десятки/сотни разных форм.

- Для векторизации (bag-of-words, TF-IDF) это означает:
  - Взрыв размерности (слишком много уникальных токенов).
  - Разреженность признакового пространства (очень много нулей).
  - Трудности при подсчете схожести/частоты для смыслово одинаковых слов.

Стемминг

- Стемминг — приведение слова к базовой, неполной форме (стему)
- Например, для английских слов "connect", "connected", "connecting" → stem = "connect".
- Стеммер Портера 1980 г. - один из наиболее ранних и популярных стеммеров для английского языка.

Стеммер Портера

Каждый шаг анализирует слово на наличие определённых суффиксов и, если находит их, обрезает, изменяет или заменяет их на минимально необходимую форму.

Примеры этапов:

- Удаление множественного числа
  cats → cat
  ponies → poni
- Приведение глагольных форм
  relational → relate
  conditional → condition
- Сокращение окончаний, характерных для деепричастий и времён
  caresses → caress
  running → run
  agreed → agre

Почему стеммер Портера хорошо работает для английского?

- **Аналитический язык с бедной морфологией:**
  Основная нагрузка значений переносится на порядок слов, аффиксы выражают немногочисленные грамматические категории.
- **Много регулярных и часто встречающихся суффиксов:**
  Окончания -ing, -ed, -ly, -es, -s довольно однотипны, и их обрезание редко приводит к неоднозначностям.
- **Алгоритм разрабатывался именно под английский:**
  Правила учитывают специфику словообразования, исключительные формы и т.п.

Почему стеммер Портера плохо работает для русского?

- **Русский — синтетический язык с очень развитой морфологией:**
  Масса падежных, родовых, временных и других окончаний, которые переплетаются с префиксами и суффиксами.
- **Основы могут меняться не только на конце, но и внутри слова.**
- **Огромное количество суффиксов и сложная морфемная структура:**
  Общее правило «отрежь окончание» работает очень плохо, часто от корней отрезается полезная часть.
- **Стемминг в русской морфологии часто не уменьшает количество уникальных слов:**
  Словизменительное разнообразие и продуктивность суффиксов приводит к тому, что слова вроде бы не похожие превращаются в похожие, и теряется различие смыслов.

Почему стеммер Портера плохо работает для русского?

- **Русский — синтетический язык с очень развитой морфологией:**
  Масса падежных, родовых, временных и других окончаний, которые переплетаются с префиксами и суффиксами.
- **Основы могут меняться не только на конце, но и внутри слова.**
- **Огромное количество суффиксов и сложная морфемная структура:**
  Общее правило «отрежь окончание» работает очень плохо, часто от корней отрезается полезная часть.
- **Стемминг в русской морфологии часто не уменьшает количество уникальных слов:**
  Словизменительное разнообразие и продуктивность суффиксов приводит к тому, что слова вроде бы не похожие превращаются в похожие, и теряется различие смыслов.

Лемматизация

Приведение слова к словарной (нормальной) форме — лемме, с учетом морфологии и полной структуры слова.

- **Цель: Преобразовать все формы слова к одной нормальной форме (“was”, “were”, “is”, “am” → “be”; “медведями”, “медведь” → “медведь”).**
- **Алгоритмы: С помощью морфологического анализа, словарей и POS-тегов.**
- **Результат: Реальное слово — лемма.**

Лемматизация

- **Плюсы:**
  - Точное объединение всех форм одной лексемы.
  - Сохраняет правильный смысл.

- **Минусы:**
  - Работает медленнее (особенно на больших текстах).
  - Требует ресурсов (словарей, моделей, морфологического анализа).

Очистка текста

"Прочитайте статью на https://example.com ! Эта статья очень интересная. Интересная статья — это всегда здорово: статья помогает понять многое."

"прочитайте статью эта статья очень интересная интересная статья это всегда здорово статья помогает понять многое"

Токенизация

"прочитайте статью эта статья очень интересная интересная статья это всегда здорово статья помогает понять многое"

[прочитайте, статью, эта, статья, очень, интересная, интересная, статья, это, всегда, здорово, статья, помогает, понять, многое]

Лемматизация (или стемминг)

[прочитайте, статью, эта, статья, очень, интересная, интересная, статья, это, всегда, здорово, статья, помогает, понять, многое]

[прочитать, статья, этот, статья, очень, интересный, интересный, статья, это, всегда, здорово, статья, помогать, понять, много]

Удаление стоп-слов

[прочитать, статья, этот, статья, очень, интересный, интересный, статья, это, всегда, здорово, статья, помогать, понять, много]

[прочитать, статья, статья, интересный, интересный, статья, здорово, статья, помогать, понять, много]

Формируем словарь

[прочитать, статья, статья, интересный, интересный, статья, здорово, статья, помогать, понять, много]

[прочитать, статья, интересный, здорово, помогать, понять, много]

Формируем Bag of Words-вектор

[прочитать, статья, статья, интересный, интересный, статья, здорово, статья, помогать, понять, много]

‘прочитать’: 1 ‘статья’: 4 ‘интересный’: 2 ‘здорово’: 1 ‘помогать’: 1 ‘понять’: 1 ‘много’: 1

[1, 4, 2, 1, 1, 1, 1]

N-граммы

- N-грамма — это последовательность из N последовательных слов в тексте.
- Распространённые варианты: униграммы (N=1), биграммы (N=2), триграммы (N=3).

Текст: "Машинное обучение изменяет мир"

- N=1 [Машинное], [обучение], [изменяет], [мир]
- N=2 [Машинное обучение], [обучение изменяет], [изменяет мир]
- N=3 [Машинное обучение изменяет], [обучение изменяет мир]

TF-IDF (Term Frequency-Inverse Document Frequency)

В чём идея?

- Частое слово в этом документе увеличивает вес (TF).
- Но если слово частое во всех документах — его вес понижается (IDF)

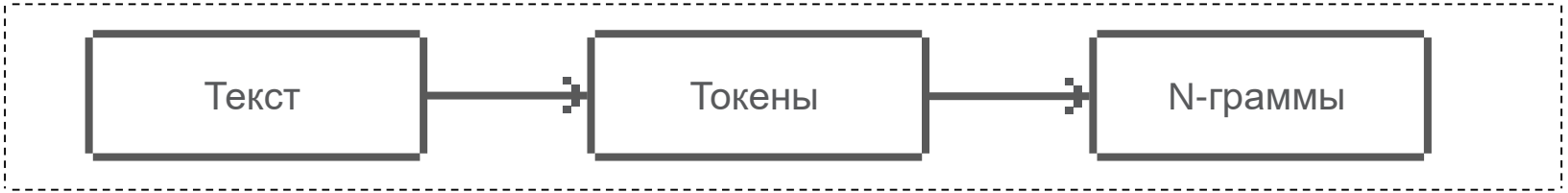
Формула:

- TF - Число вхождений слова в документе / общее число слов в документе
- IDF - log(Общее число документов / Число документов, где встречается слово)

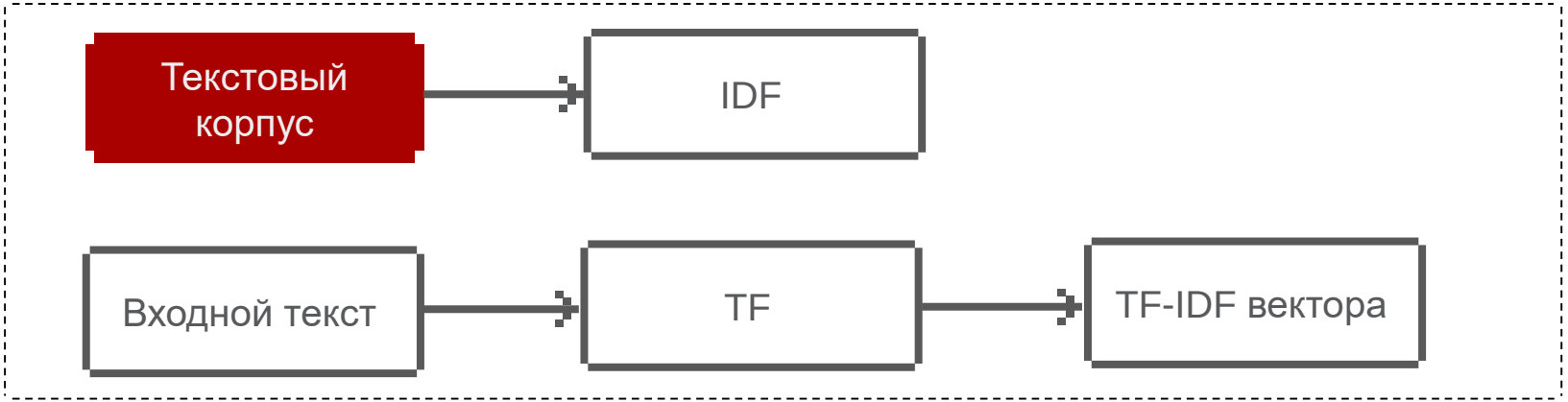
TF-IDF = TF × IDF

Алгоритм

Общее преобразование текстов



Роль корпуса текстов



Моделирование языка

Как понимать язык ?

- отношения между словами / предложениями / текстами
- отношение слова к остальному тексту

Как моделировать смысл ?

- базы знаний – составление правил и алгоритмов экспертами
- дистрибутивный подход – “Значение – сумма употреблений”

Дистрибутивный подход

Как представить смысл слова ?

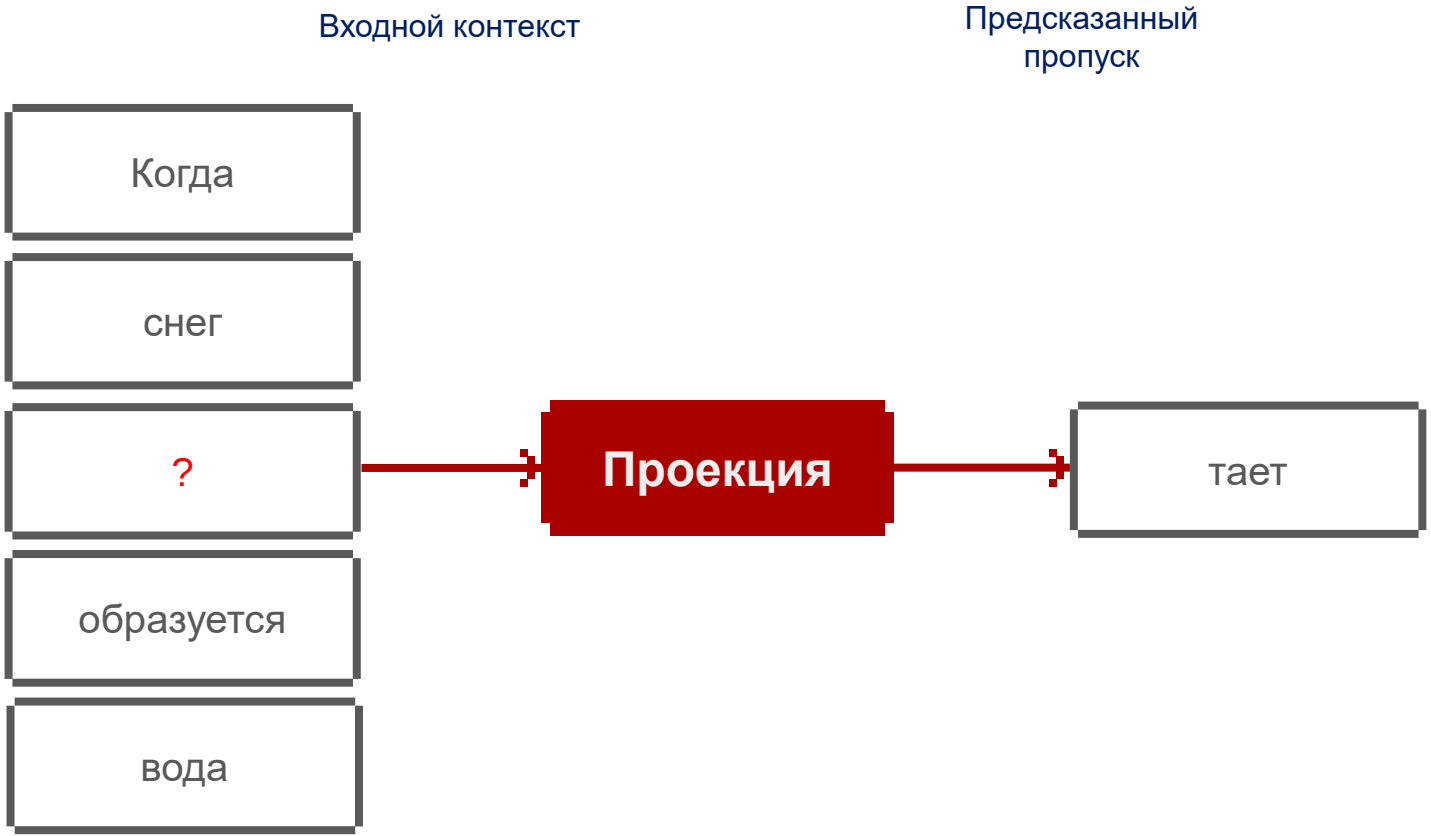
- строятся на основе текстовых корпусов
- слово = частота соседних слов

	человек	дом	город	книга	вода	...
человек	-	120	85	45	30	...
дом	120	-	110	25	40	...
город	85	110	-	20	35	...
книга	45	25	20	-	5	...
вода	30	40	35	5	-	...
...	...	...	...	...	...	...

Базовая идея эмбеддингов

- Эмбеддинг — способ представить слова (токены) в виде векторов
- Схожие по смыслу слова — близки в пространстве
- Основа современных NLP-моделей

CBOW


Входной контекст
Предсказанный пропуск
Когда
снег
?
образуется
вода
Проекция
тает

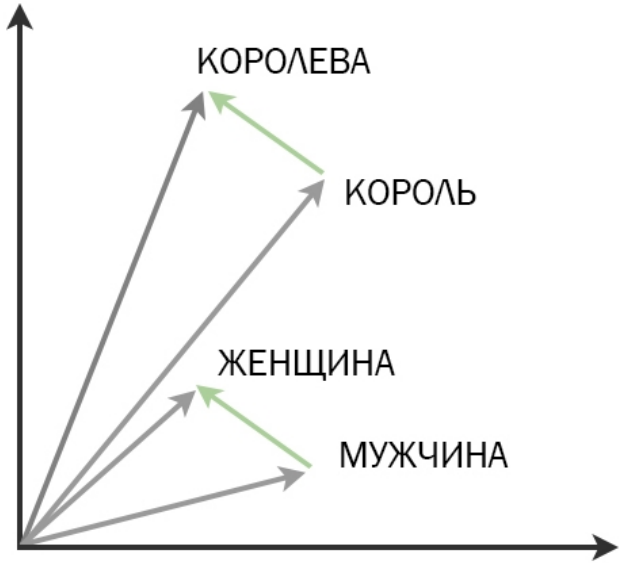
Skip-gram


Предсказанный контекст
Входное слово
тает
Проекция
Когда
снег
?
образуется
вода

Word2Vec

Что получилось ?

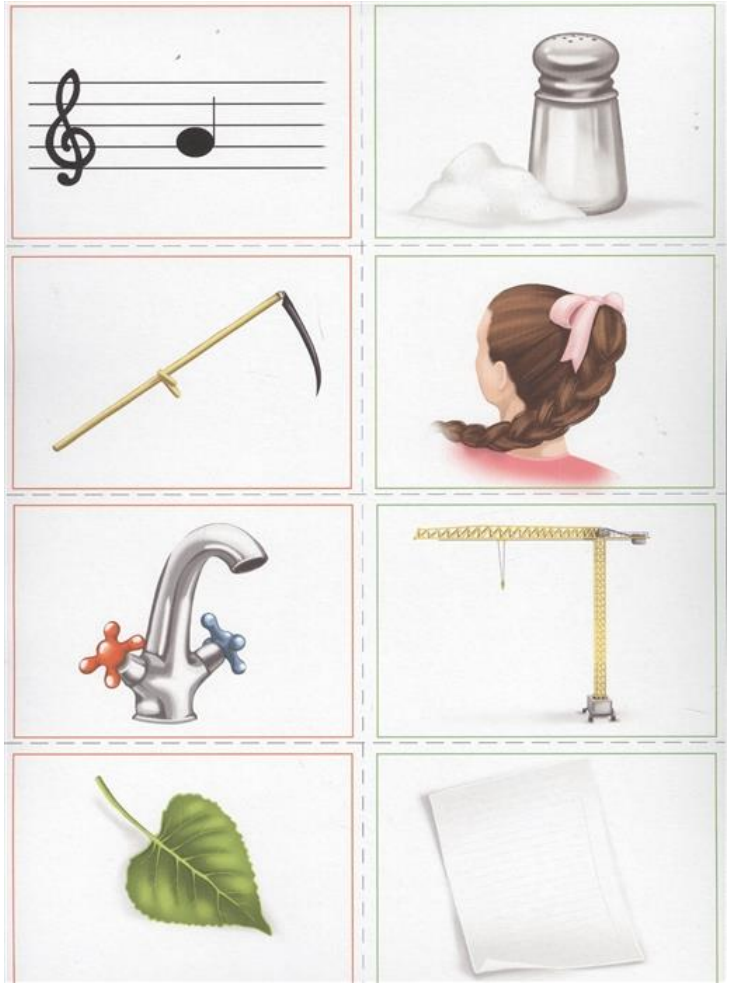
2d проекция полученных представлений



- Представления фиксированной размерности (например dim 100)
- Репрезентативные вектора отражающие отношения между словами

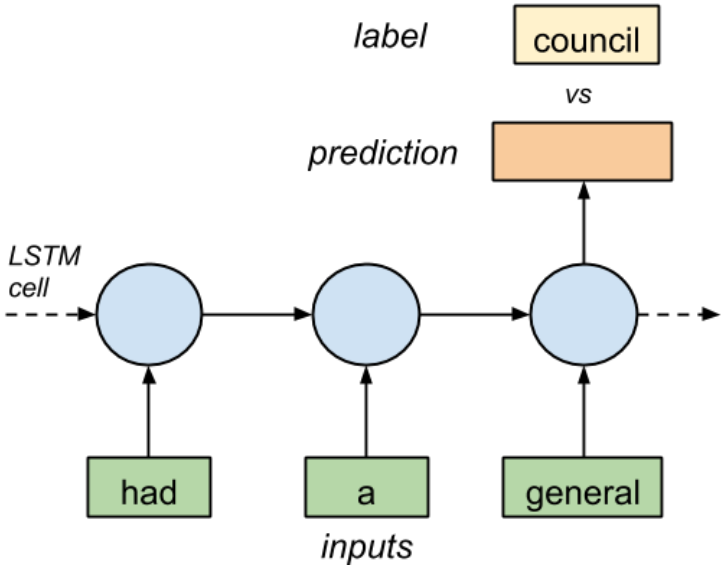
Word2Vec

- Не учитывает контекст
- Омонимы
- Полисемия



Рекуррентные нейронные сети

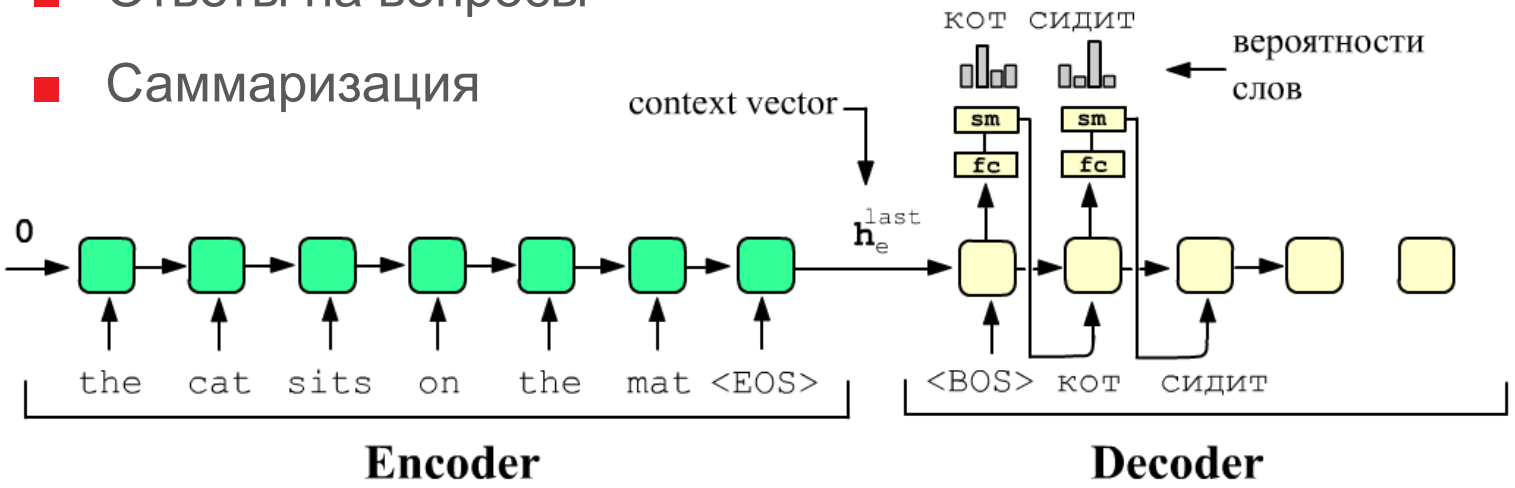
Общим блоком сети последовательно проходим текст, формирую вектор контекста, передаем на следующий шаг



Рекуррентные нейронные сети

Text2text

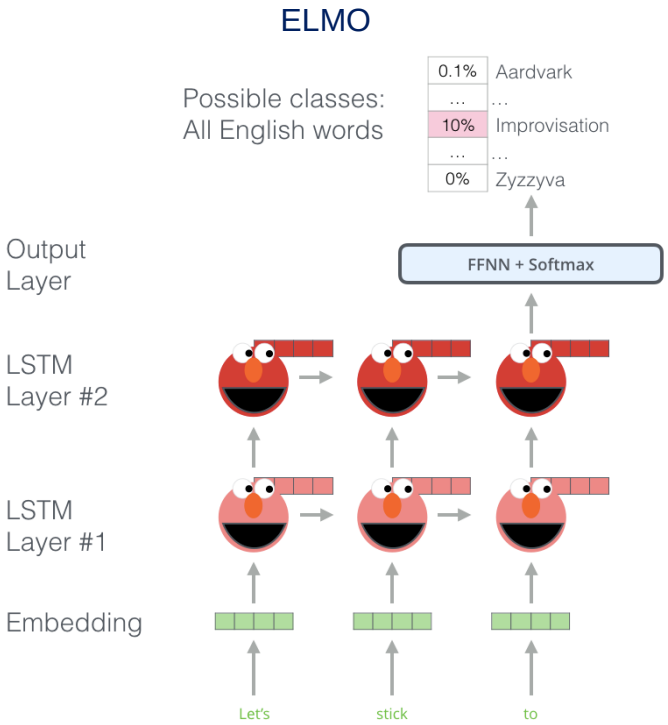
- Машинный перевод
- Ответы на вопросы
- Саммаризация



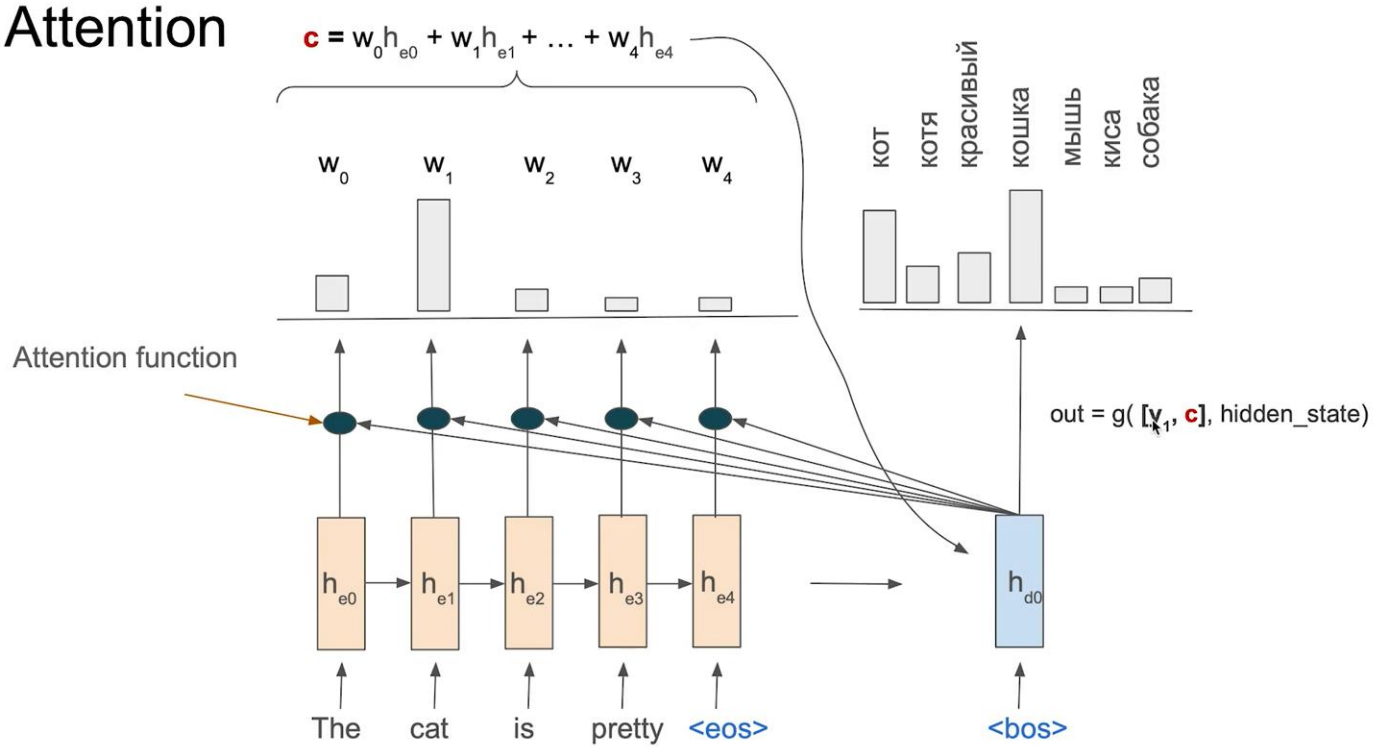
Рекуррентные нейронные сети

Недостатки

- Забывчивость
- Представление контекста вектором фиксированного размера
- Отсутствие параллелизма по последовательности



Механизм внимания



Transformer

Достоинтсва

- Прямой доступ к контексту
- Нет рекуррентности – обрабатываем сразу всю последовательность

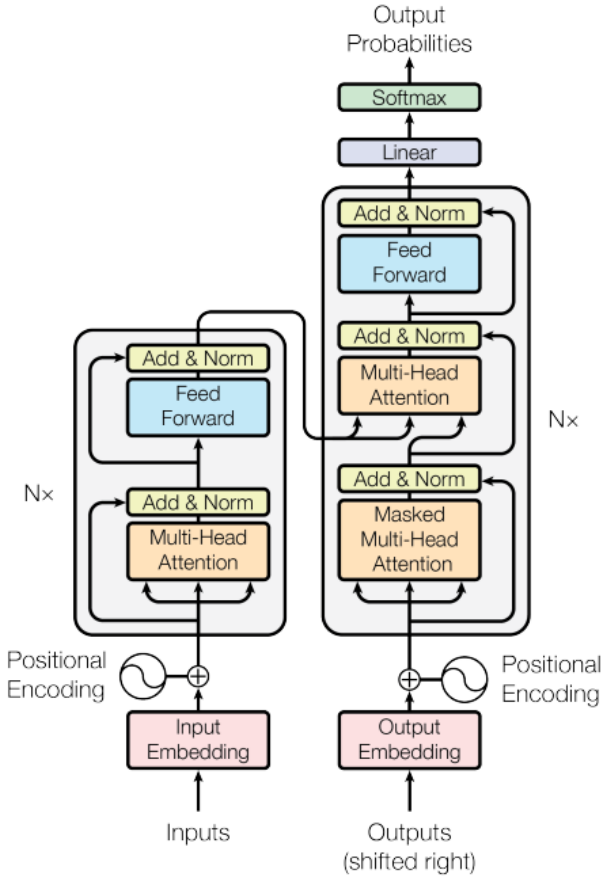
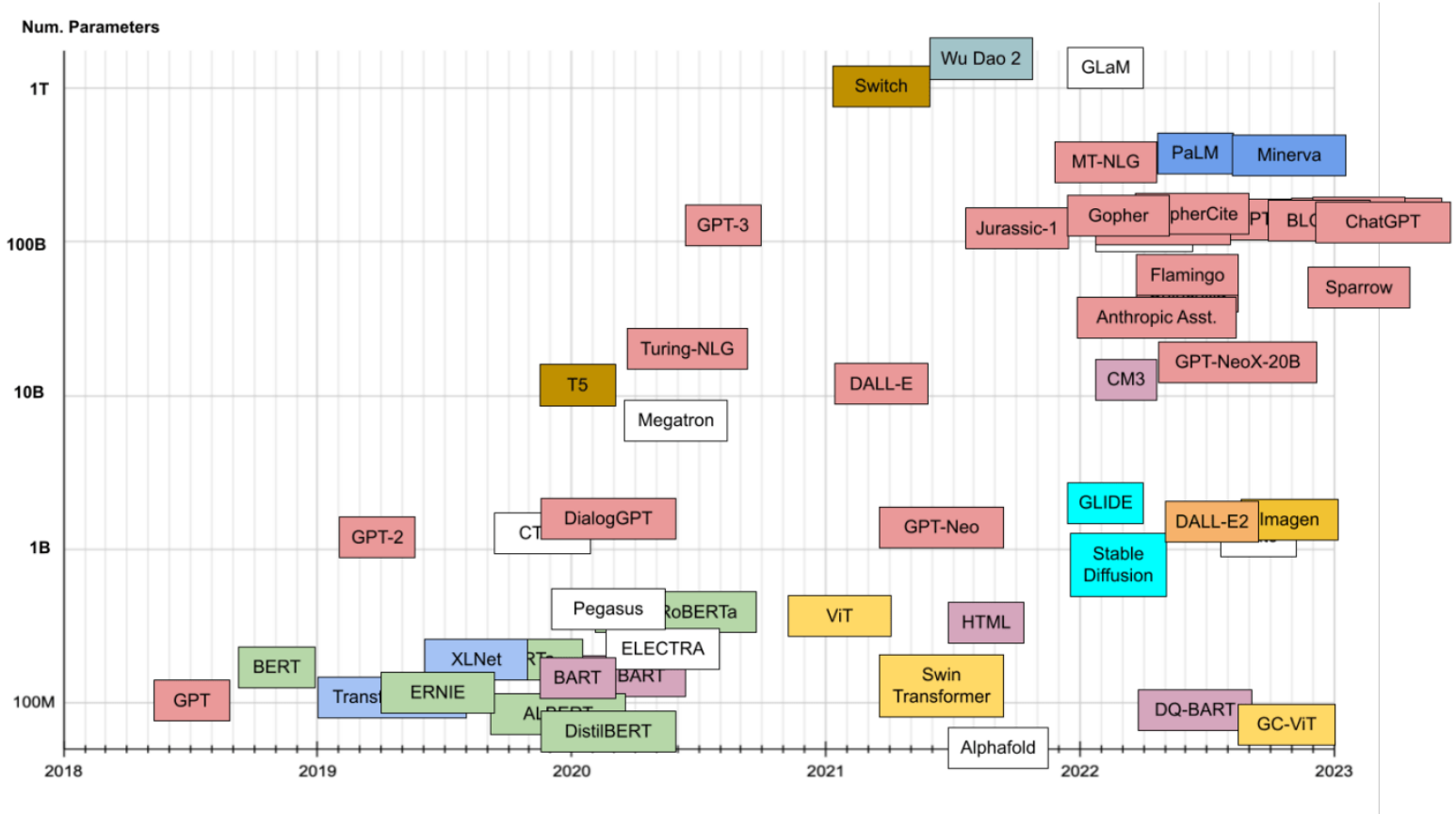


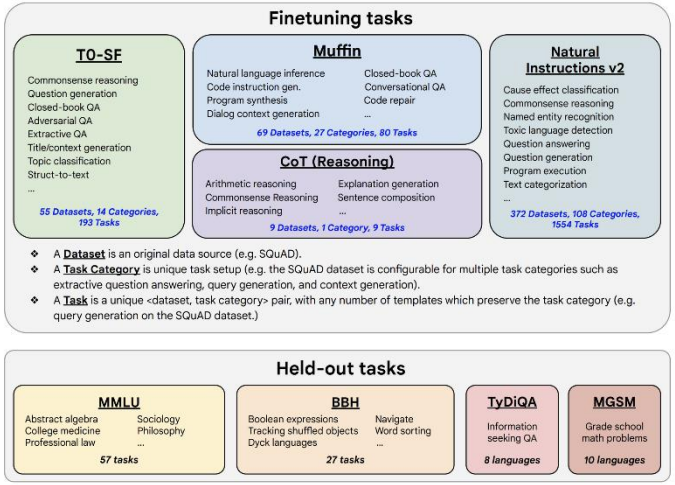
Figure 1: The Transformer - model architecture.

LLM

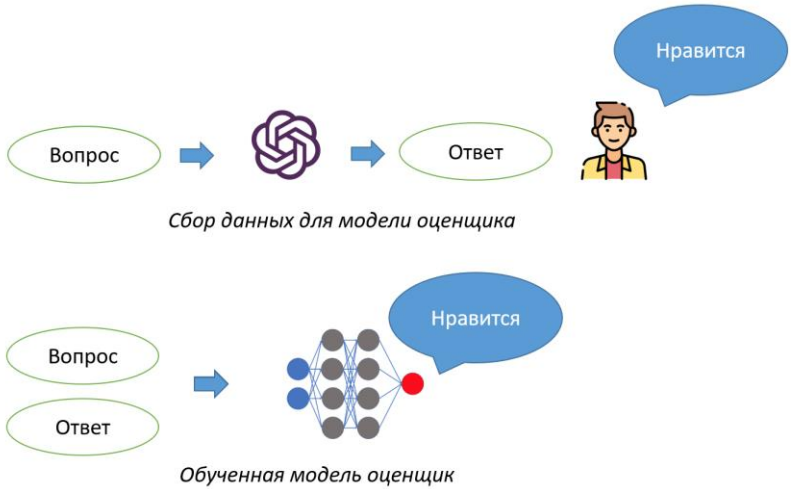

Num. Parameters
1T
100B
10B
1B
100M
2018
2019
2020
2021
2022
2023
Wu Dao 2
GLaM
Switch
MT-NLG
PaLM
Minerva
GPT-3
Jurassic-1
Gopher
ChatGPT
Flamingo
Sparrow
Anthropic Asst.
Turing-NLG
T5
DALL-E
CM3
GPT-NeoX-20B
Megatron
GLIDE
DALL-E2
Imagen
DialogGPT
GPT-2
GPT-Neo
Stable Diffusion
Pegasus
ViT
HTML
ELECTRA
BERT
XLNet
BART
Swin Transformer
ERNIE
GPT
DQ-BART
GC-ViT
DistilBERT
Alphafold

LLM

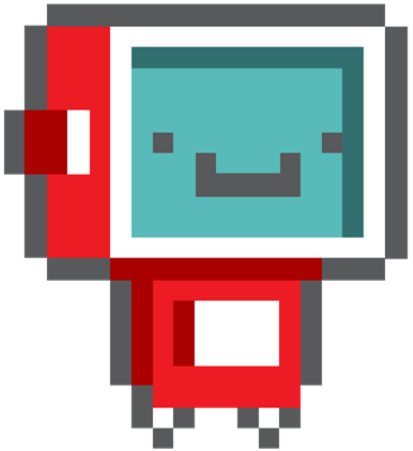
Инструкционные задачи



RLHF



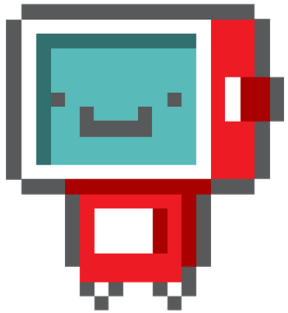
КТО ТАКИЕ СЕЙЧАС LLM?



- Общаются на естественном языке
- Выполняют (пытаются) поставленные задачи
- Существует open source

КТО ТАКИЕ СЕЙЧАС LLM?

Изучит годовой отчет



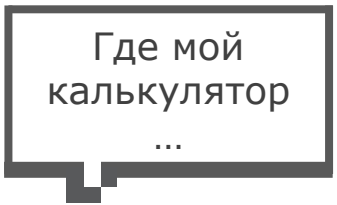
Слышит и говорит

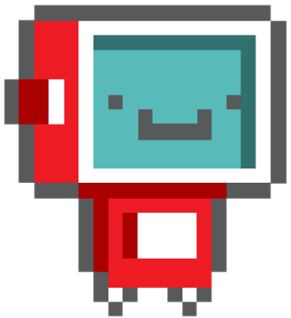
Оценит вашего любимца

Если надо, прочитает и Excel

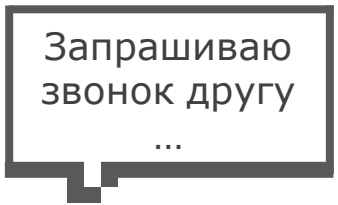
Прочтет за вас всю документацию

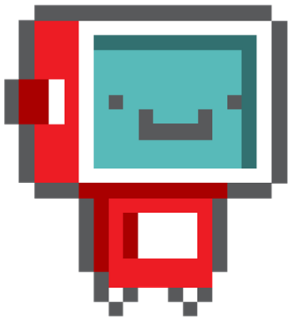
КТО ТАКИЕ СЕЙЧАС LLM?



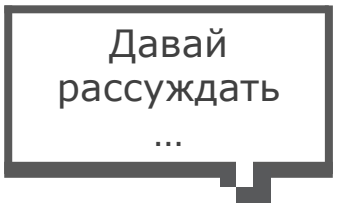


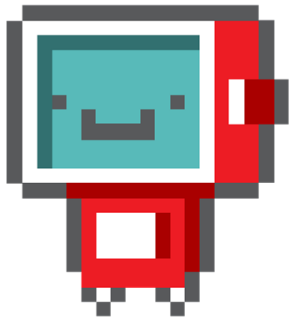
Function calling





Agents





Reasoning

КОНЕЦ