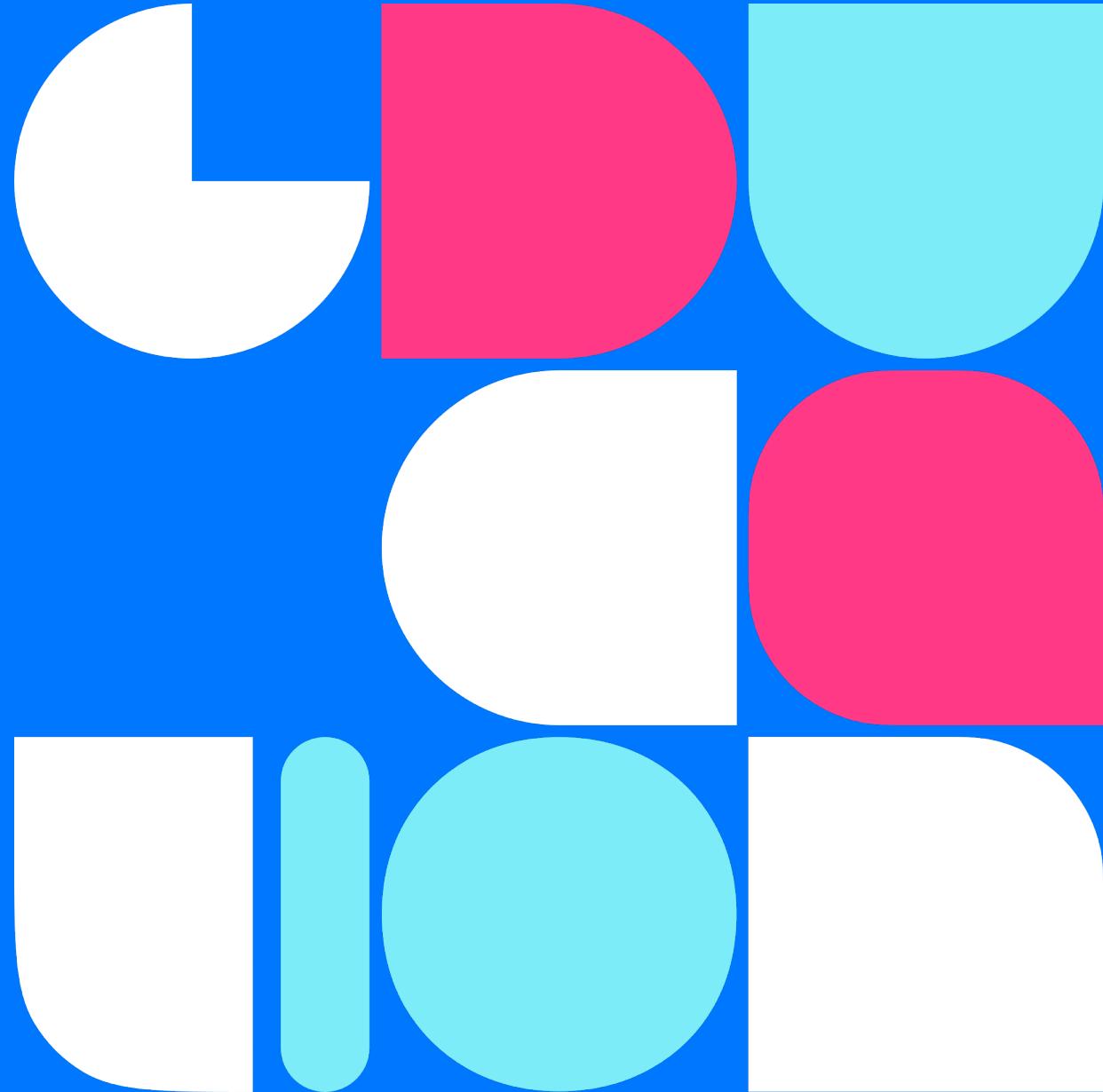




Современные технологии обработки информации

RAG, семантические энкодеры и векторный поиск



План лекции

1. Введение: современные вызовы в работе с информацией
2. Эмбеддинги: представление текста в виде векторов
3. Семантические энкодеры: создание векторных представлений
4. Векторный поиск: поиск похожих документов
5. RAG: объединяем поиск и генерацию
6. Практические примеры и применения

Почему это важно?

1. Потребность в точных и релевантных ответах
 - Традиционный поиск не справляется
 - Нужно понимание смысла, а не просто слов

2. Необходимость структурированного представления знаний
 - Связи между информацией
 - Контекстное понимание



Что такое эмбеддинги?



Что такое эмбеддинги?

Эмбеддинги - это способ представить текст в виде чисел:
"Кошка" → [0.2, 0.8, -0.3, ...]

Ключевые особенности:

- Сохраняют смысловые связи
- Похожие слова → похожие векторы
- Позволяют измерять семантическую близость

Примеры семантической близости

Примеры:

"кошка" \leftrightarrow "котенок" = 0.8

"кошка" \leftrightarrow "собака" = 0.6

"кошка" \leftrightarrow "автомобиль" = 0.1

Как это работает:

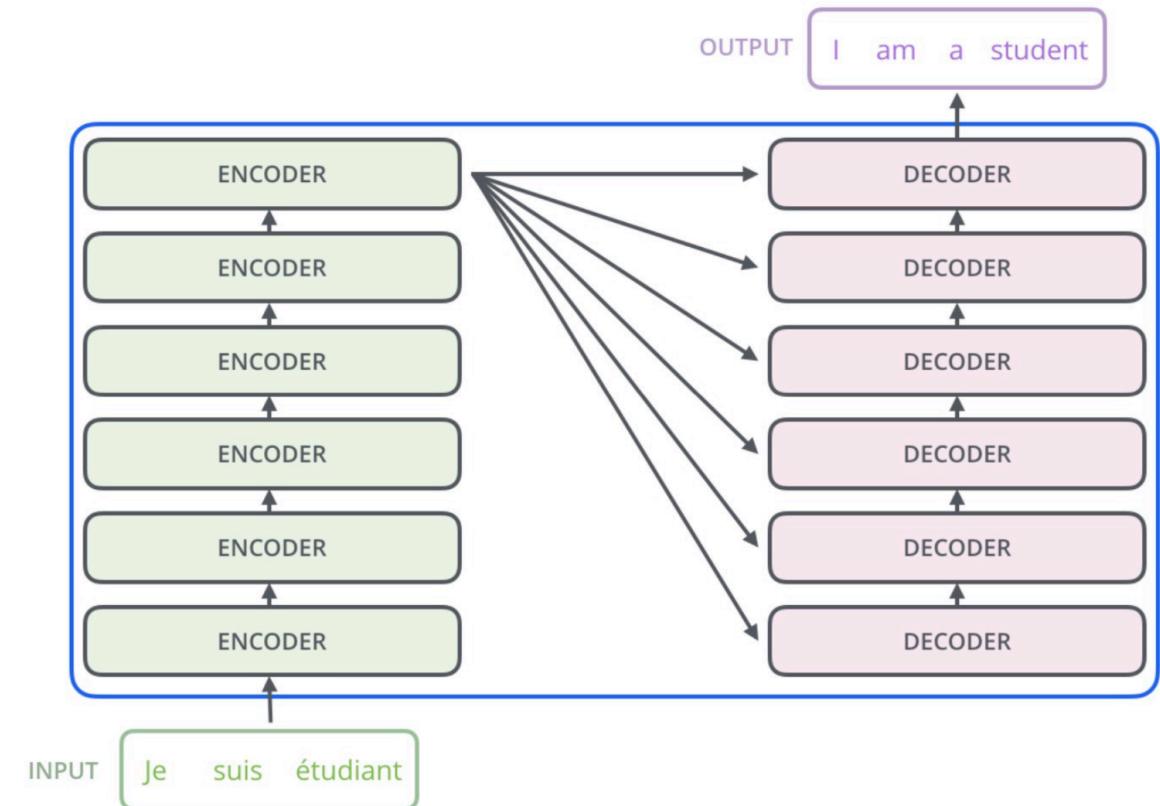
- Каждое измерение вектора отражает определенную характеристику
- Близость векторов = близость смыслов
- Можно выполнять векторные операции
- Косинусное сходство: Измеряет угол между векторами
- Евклидово расстояние: Линейное расстояние между векторами

Семантические энкодеры

- Что это?
 - Нейронные сети для создания эмбеддингов
 - Понимают контекст и связи между словами
 - Преобразуют текст в векторы
- Популярные модели:
 - BERT
 - Sentence-BERT
 - text-embedding-ada-002

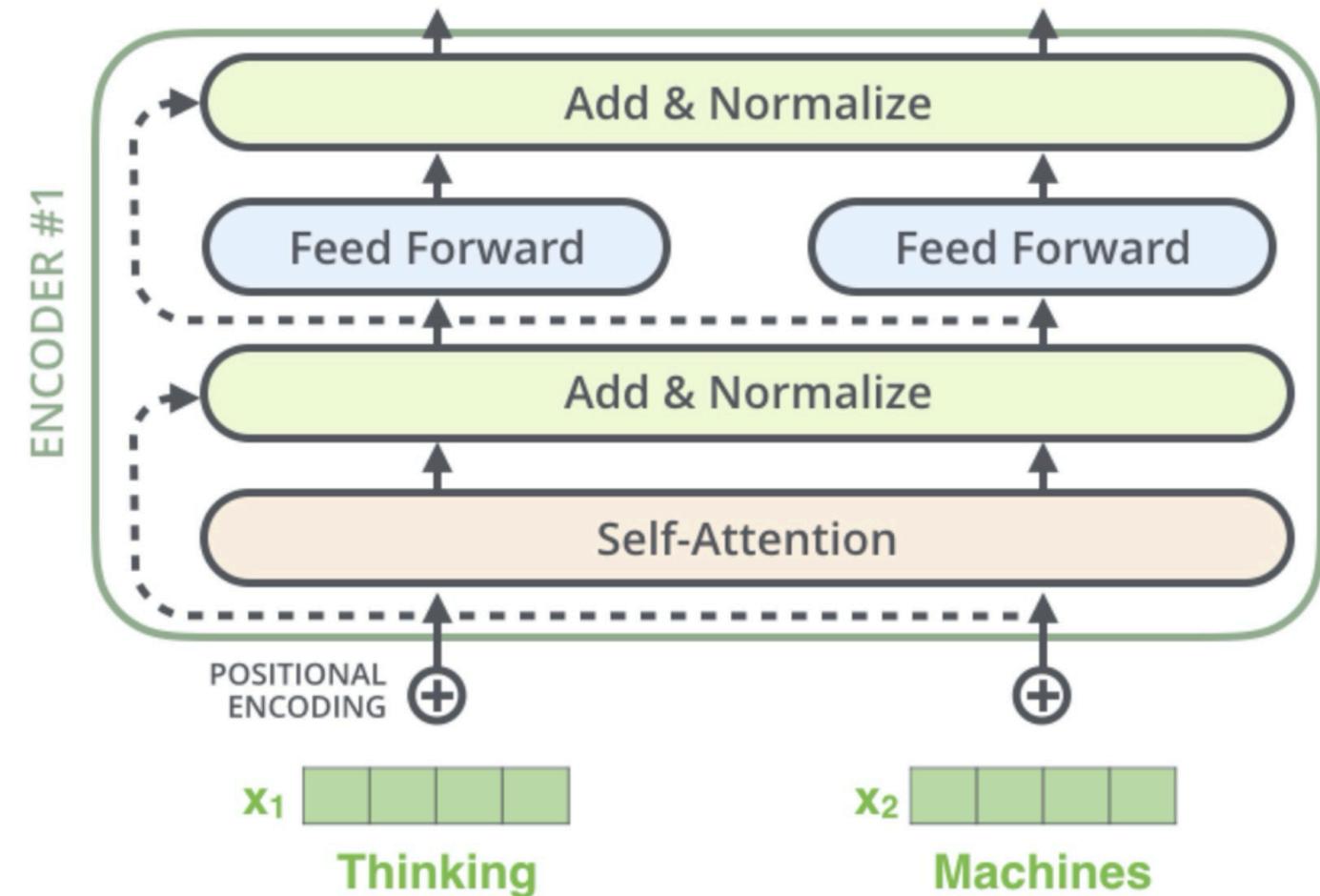
Encoder и decoder

- Хотим качественные эмбеддинги токенов для разных задач
 - классификация текста
 - NER на токенах
- Токены должны узнать всю информацию о контексте
- Какая часть трансформера может приготовить эмбеддинги токенов?



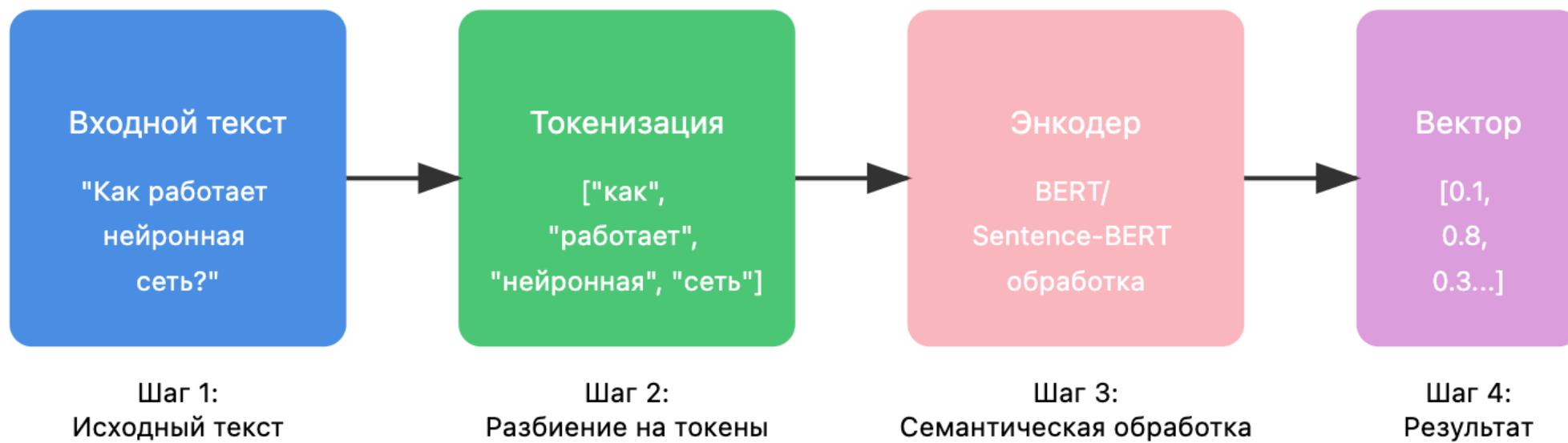
БЛОК ЭНКОДЕРА

- Каждый слой энкодера формирует более сложное представление исходного токена с учетом всего контекста
- Энкодер видит **всю** последовательность
- Больше энкодеров – выше уровень абстракции представлений (представления “умнее”)



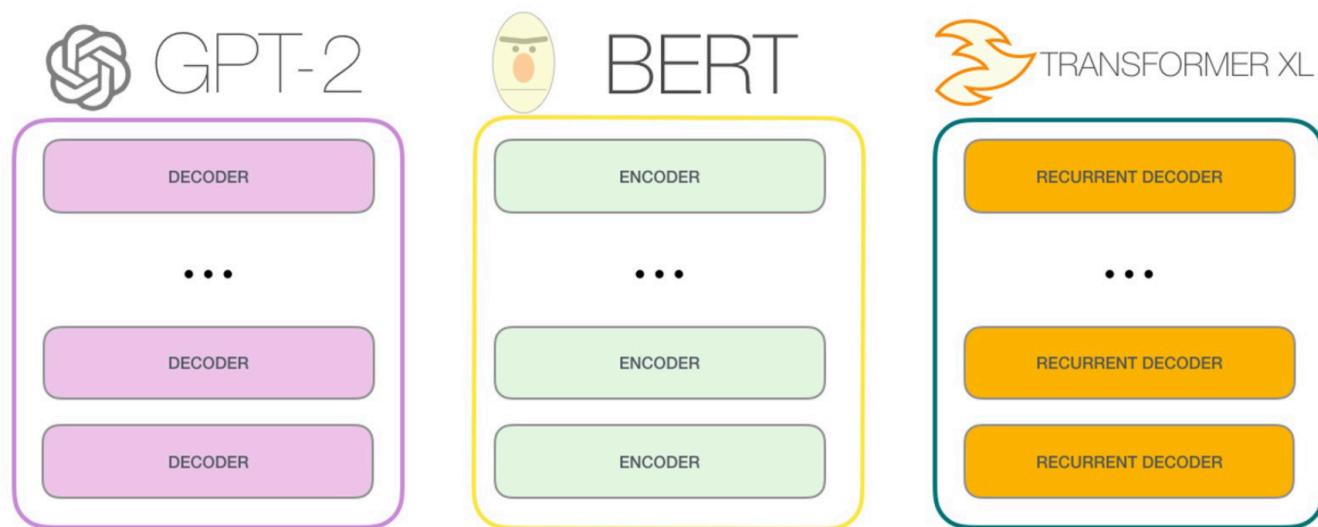
Как работает семантический энкодер

1. Токенизация текста
2. Обработка в нейронной сети
3. Создание векторного представления
4. Получение финального эмбеддинга



Энкодеры – для эмбеддингов

- Можем не использовать стек декодеров вообще
- Выкидываем декодеры, обучаем только энкодеры
- BERT – текстовый энкодер



Кто такой этот ваш BERT

- Bidirectional Encoder Representations from Transformers
- Используется для получения представлений (эмбеддингов) токенов
- Из представлений токенов можно получить эмбеддинг текста
- Для чего используются эмбеддинги:
 - Классификация текста / токенов
 - Поиск похожих текстов
 - Экстрактивная суммаризация текста
 - NER

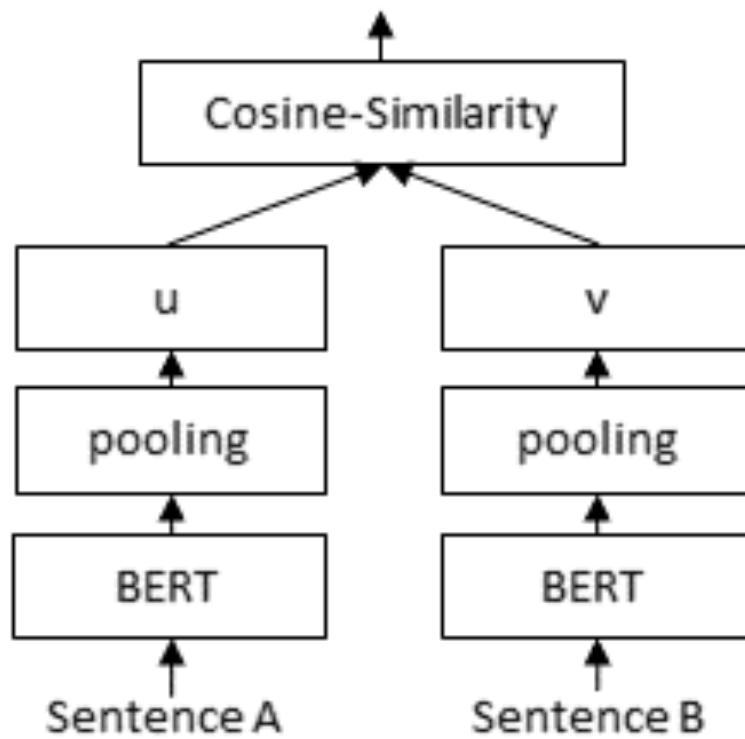


Типы семантических энкодеров

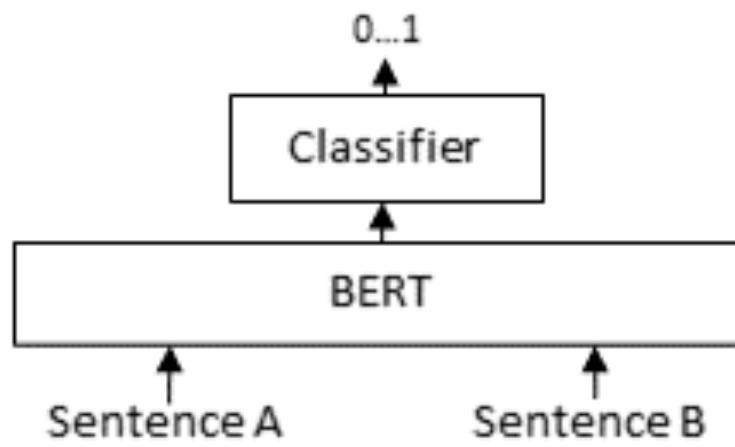
Характеристика	Bi-encoders	Cross-encoders	Гибридные модели
Скорость работы	Высокая	Низкая	Средняя
Точность	Средняя	Высокая	Высокая
Масштабируемость	Отличная	Плохая	Хорошая
Предварительная индексация	Возможна	Невозможна	Частично возможна
Использование памяти	Низкое	Высокое	Среднее
Типичные применения	Поиск, рекомендации	Ранжирование, классификация	Универсальные системы

Типы семантических энкодеров

Bi-Encoder



Cross-Encoder



Векторный поиск

Принцип работы:

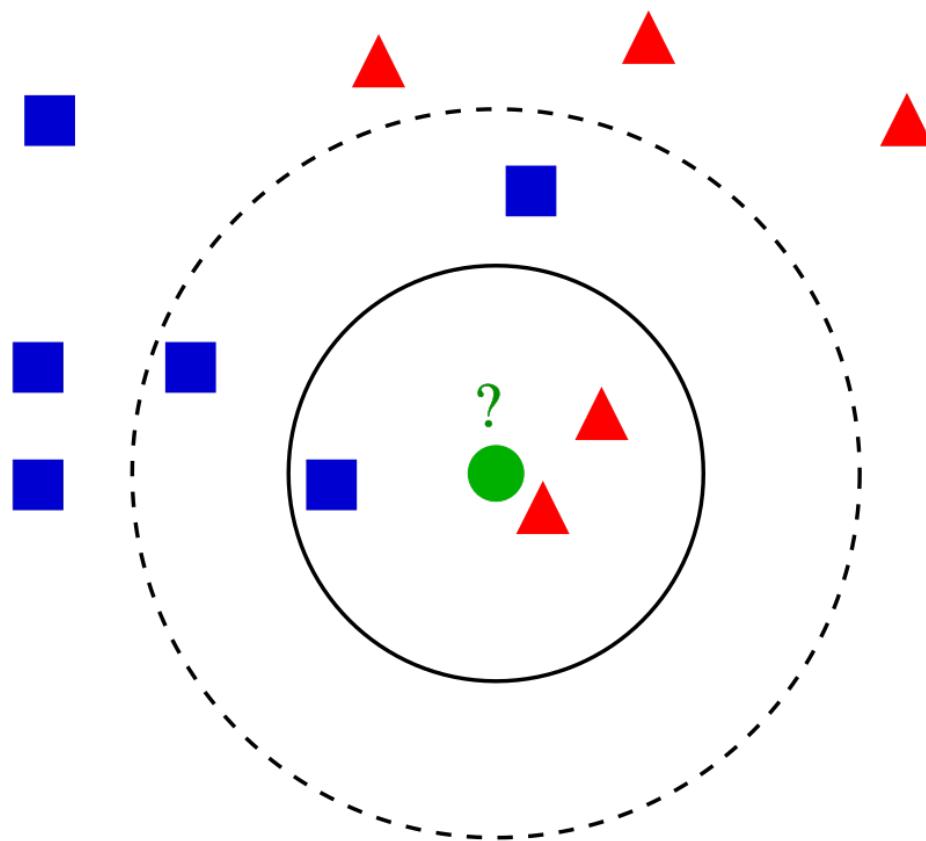
1. Преобразование документов в векторы
2. Индексация векторов
3. Поиск ближайших соседей

Преимущества:

- Семантический поиск вместо текстового
- Понимание контекста
- Масштабируемость



Алгоритмы поиска ближайших соседей



1. Точный поиск

- Полный перебор
- Гарантированная точность
- Медленная работа

2. Приближенный поиск (ANN)

- HNSW
- LSH
- Product Quantization
- Быстрая работа с небольшой потерей точности

Перейдем к
самому вкусному?

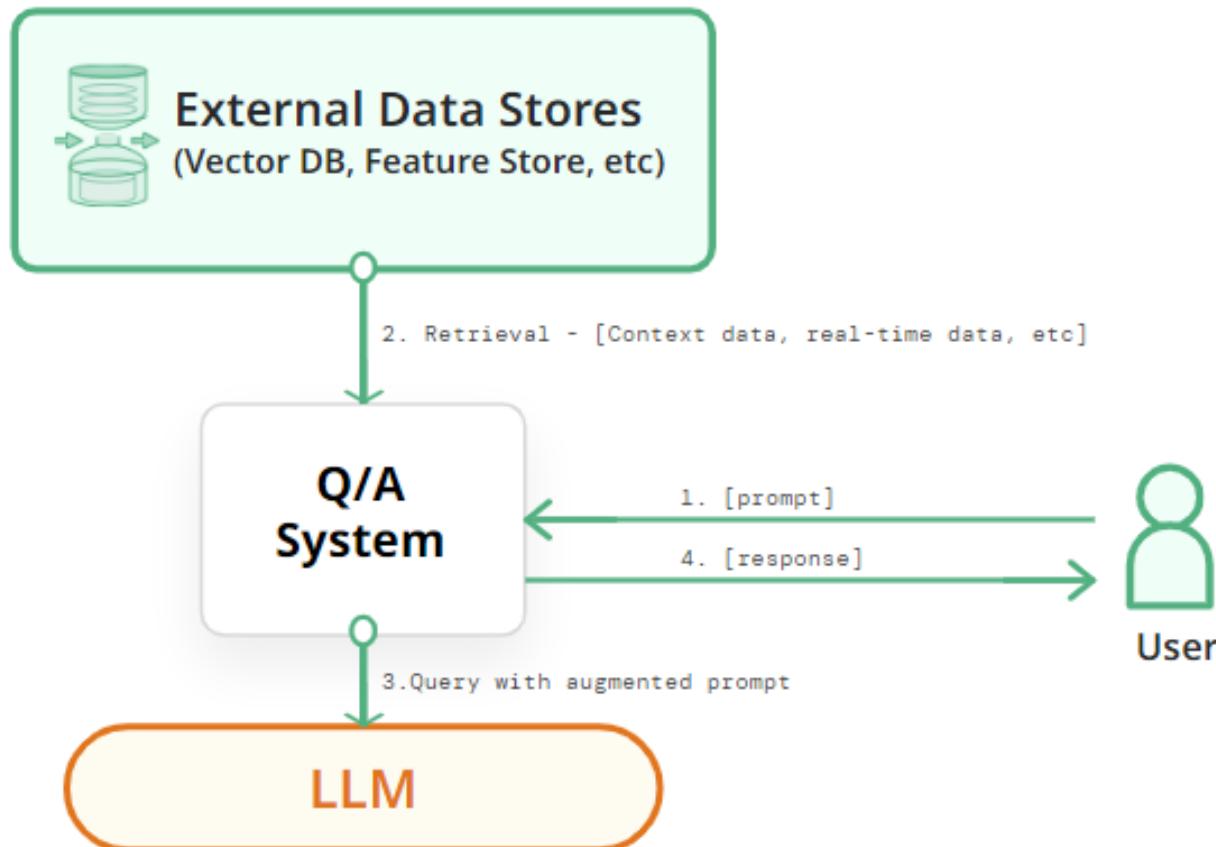


RAG: Retrieval-Augmented Generation

Компоненты:

1. Поисковый модуль
2. База знаний
3. Генеративная модель

Как это работает вместе ->



Все любят пингвинов?

Какой средний рост императорского пингвина?



Поиск релевантной информации в энциклопедии



Энциклопедия



Формирование запроса языковой модели

Вопрос: Какой средний рост императорского пингвина?
Релевантная информация:
Императорский пингвин — самый крупный среди пингвинов... Средний рост – 120см



Запрос языковой модели



Языковая модель



Генерация ответа языковой моделью

Средний рост императорского пингвина составляет 120см

А давайте сравним

Характеристика	Базовая LLM	RAG-система
Актуальность информации	Ограничена данными обучения	Всегда актуальная через внешние источники
Достоверность	Возможны галлюцинации	Опирается на проверенные источники
Источники информации	Неявные, из обучающих данных	Прослеживаемые, с указанием конкретных документов
Обновляемость	Требует переобучения	Простое обновление базы знаний
Контроль контента	Ограниченный	Полный контроль через базу знаний
Вычислительные затраты	Средние	Низкие (поиск + генерация)

Это все конечно
хорошо, но мы не
закончили

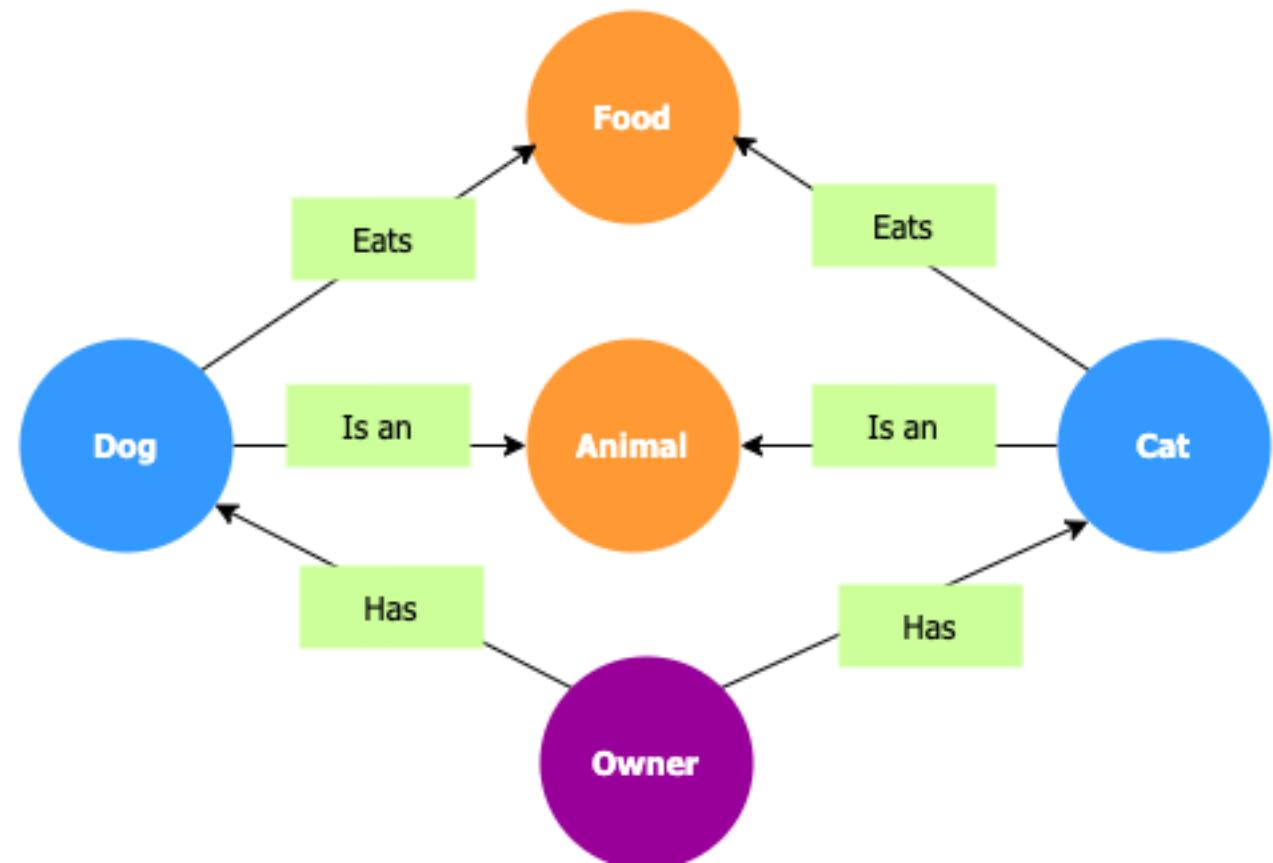
Knowledge Graphs: структурируем знания

Что такое граф знаний?

Структурированное представление информации

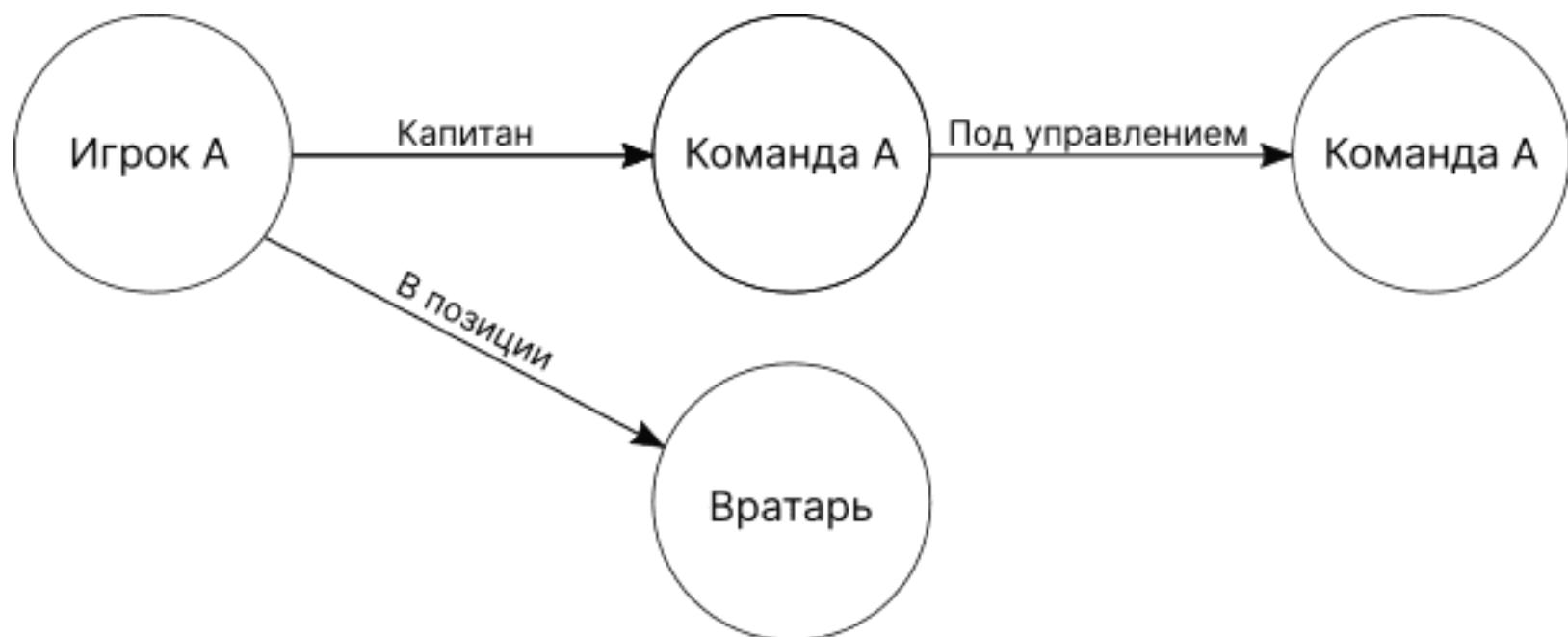
Связи между сущностями

Семантические отношения

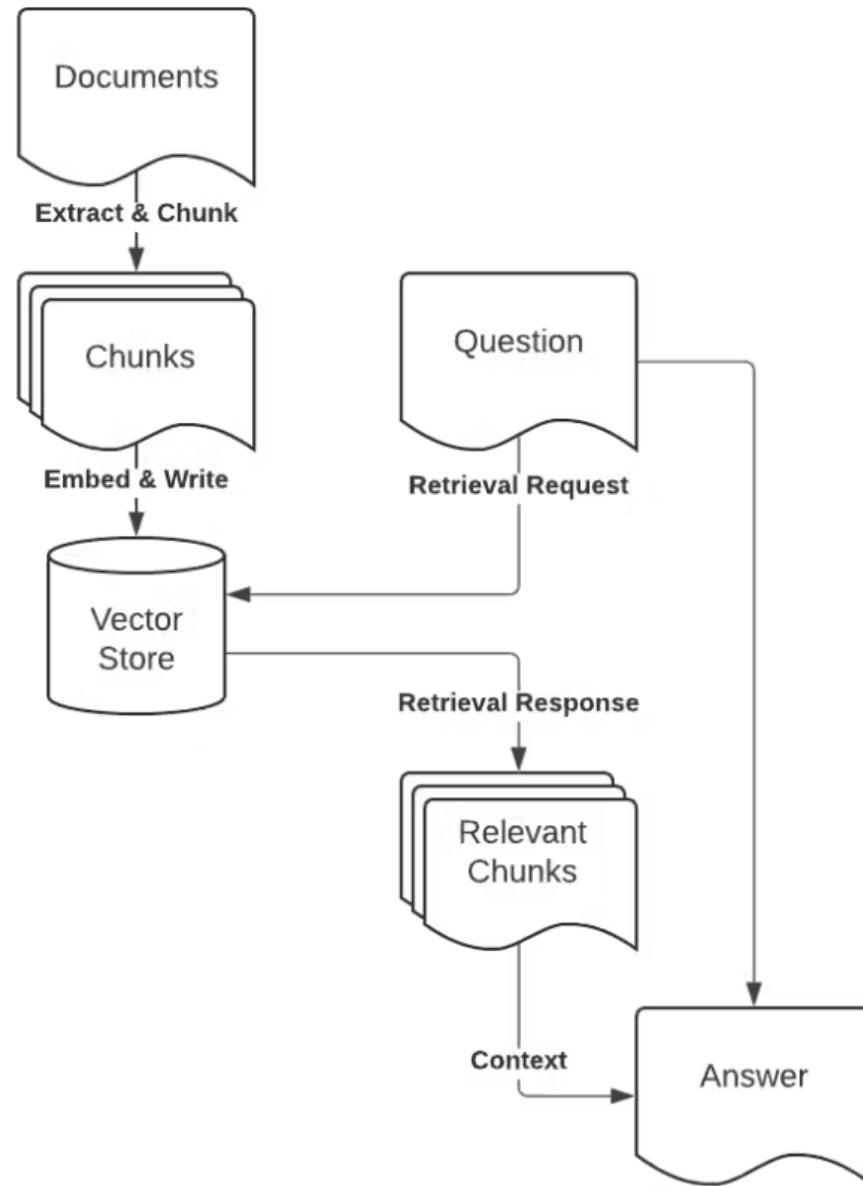


Компоненты графа знаний

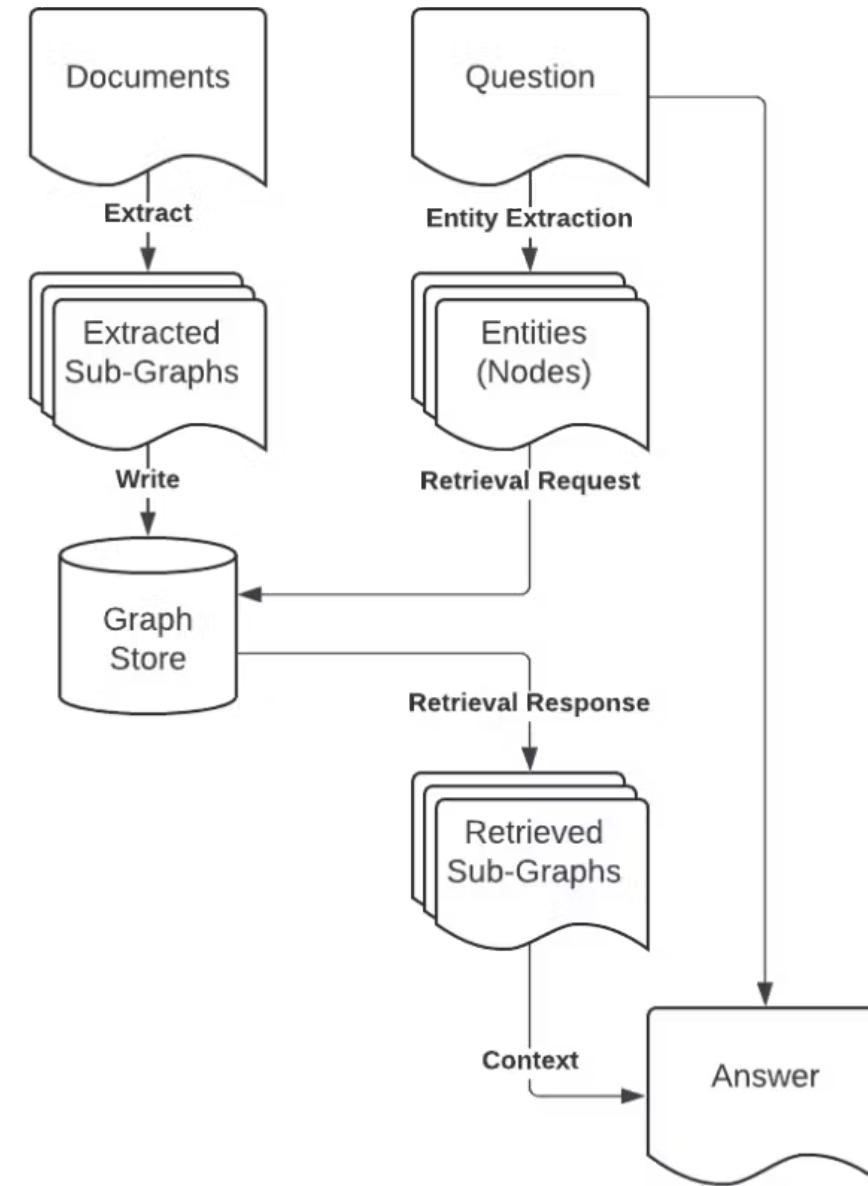
1. Узлы (сущности)
2. Рёбра (отношения)
3. Атрибуты (свойства)



Similarity-Based RAG



Knowledge Graph RAG



Vector Store и Graph Store

Вектор стор (Vector Store) - это специальная база данных, которая хранит и работает с векторами (числовыми представлениями) данных. Попробую объяснить на простом примере:

Представьте, что:

- У вас есть текст, изображение или любой другой тип данных
- Эти данные преобразуются в длинный список чисел (вектор)
- Вектор стор хранит эти числа таким образом, чтобы можно было быстро найти похожие векторы

Граф стор (Graph Store):

- База данных, которая хранит информацию в виде графа
- Граф состоит из узлов (точек) и связей между ними
- Позволяет хранить сложные взаимосвязи между данными

Пример граф стора:

- Узлы: "Иван", "Мария", "Компания А"
- Связи: "Иван работает в Компании А", "Мария - начальник Ивана"



Спасибо за внимание!

Киков Джантемир, Старший разработчик-исследователь ML mail.ru
TG - @kikovj