Лекция 4. Архитектура больших языковых моделей

Основные вопросы:

1. Обзор существующих LLMs
2. Языковое моделирование
3. Базовые модели LLMs
4. Механизм внутреннего внимания

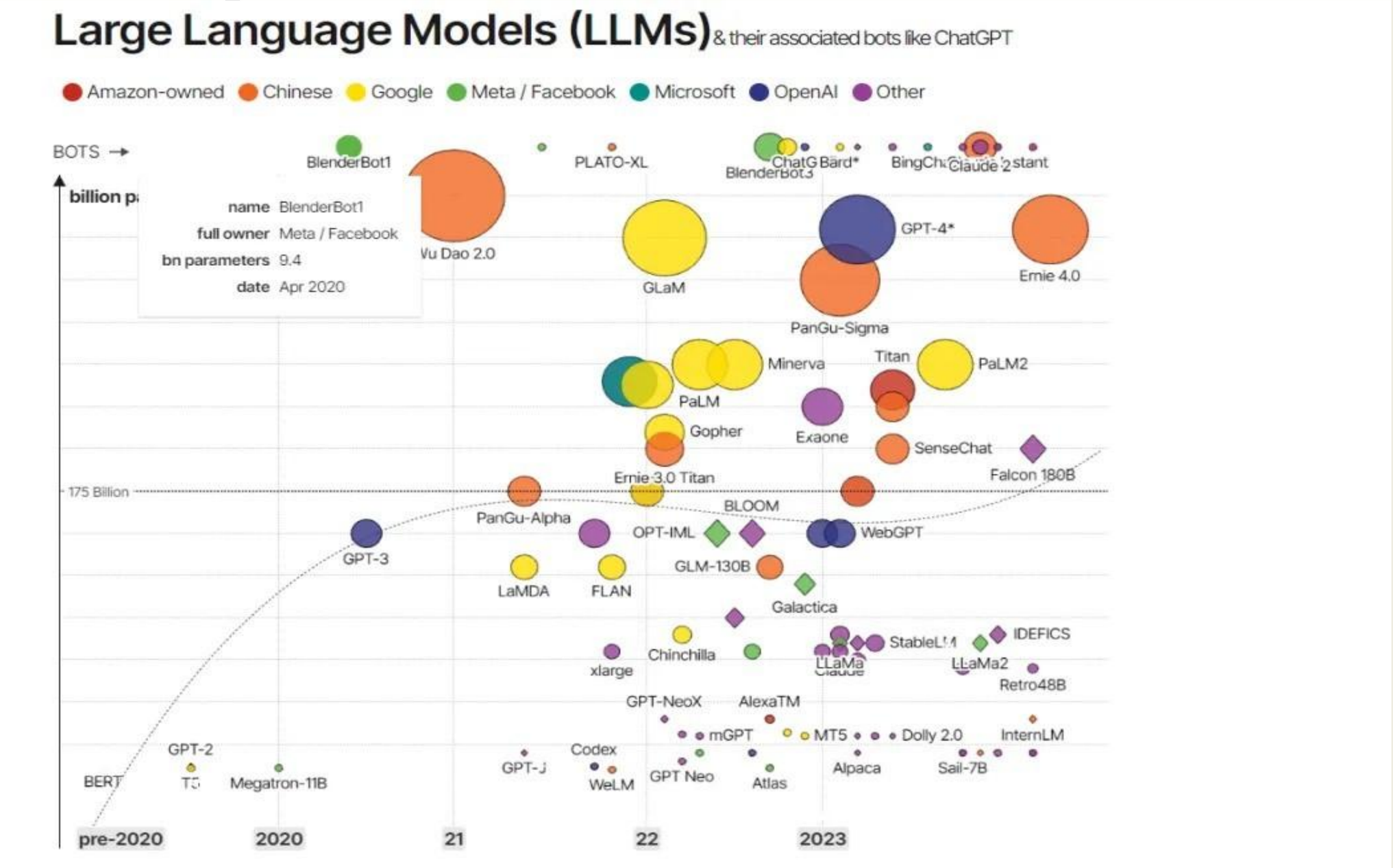
Большие языковые модели (или LLM) являются видом искусственного интеллекта, спроектированные для анализа и создания больших объемов текстовой информации. Эти модели AI, основанные на методиках глубокого обучения, используют подкатегорию нейронных сетей, называемую трансформерами.

Введение в большие языковые модели (LLMs)

Что такое LLMs?

- **LLM** - Large Language Model (Большая Языковая модель)
- **LLMs** — это продвинутые модели машинного обучения, обученные на огромных объемах текстовых данных для понимания, генерации и обработки человеческого языка.
- **Примеры:** GPT (Generative Pre-trained Transformer), BERT (Bidirectional Encoder Representations from Transformers), LLaMA (Large Language Model Meta AI).

Существующие LLM



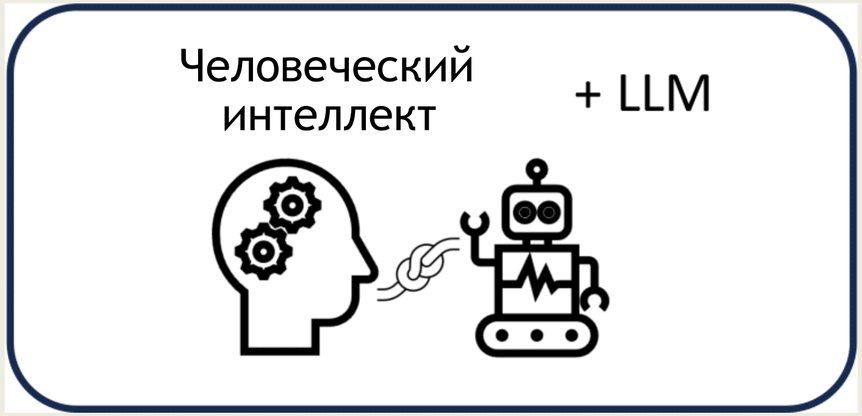
Примерные данные по размерам моделей и размерам данных на которых они обучались

Дата	Модель	Размер модели	Размер данных (Tokens)	Оценка работы (Pass@1)	MBPP (Pass@1)
2021 Jul	Codex-300M [CTJ+21]	300M	100B	13.2%	-
2021 Jul	Codex-12B [CTJ+21]	12B	100B	28.8%	-
2022 Mar	CodeGen-Mono-350M [NPH+23]	350M	577B	12.8%	-
2022 Mar	CodeGen-Mono-16.1B [NPH+23]	16.1B	577B	29.3%	35.3%
2022 Apr	PaLM-Coder [CND+22]	540B	780B	35.9%	47.0%
2022 Sep	CodeGeeX [ZXZ+23]	13B	850B	22.9%	24.4%
2022 Nov	GPT-3.5 [Ope23]	175B	N.A.	47%	-
2022 Dec	SantaCoder [ALK+23]	1.1B	236B	14.0%	35.0%
2023 Mar	GPT-4 [Ope23]	N.A.	N.A.	67%	-
2023 Apr	Replit [Rep23]	2.7B	525B	21.9%	-
2023 Apr	Replit-Finetuned [Rep23]	2.7B	525B	30.5%	-
2023 May	CodeGen2-1B [NHX+23]	1B	N.A.	10.3%	-
2023 May	CodeGen2-7B [NHX+23]	7B	N.A.	19.1%	-
2023 May	StarCoder [LAZ+23]	15.5B	1T	33.6%	52.7%
2023 May	StarCoder-Prompted [LAZ+23]	15.5B	1T	40.8%	49.5%
2023 May	PaLM 2-S [ADF+23]	N.A.	N.A.	37.6%	50.0%
2023 May	CodeT5+ [WLG+23]	2B	52B	24.2%	-
2023 May	CodeT5+ [WLG+23]	16B	52B	30.9%	-
2023 May	InstructCodeT5+ [WLG+23]	16B	52B	35.0%	-
2023 Jun	WizardCoder [LXZ+23]	16B	1T	57.3%	51.8%
2023 Jun	**phi-1**	1.3B	7B	50.6%	55.5%

Введение в большие языковые модели (LLMs)

Важность LLMs и Основная цель

- Преобразующие приложения в области обработки естественного языка (NLP): чат-боты, перевод, суммаризация, генерация контента и многое другое.
- **Мост между человеком и машиной в общении.**
- **Цель:** Научить машины обрабатывать и генерировать текст так, чтобы это имитировало человеческое понимание и креативность.



Исторический контекст и эволюция

Ранние этапы

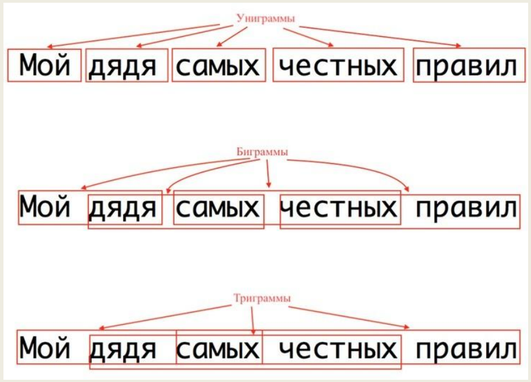
- Системы на основе правил в 1960-х годах (например, ELIZA).

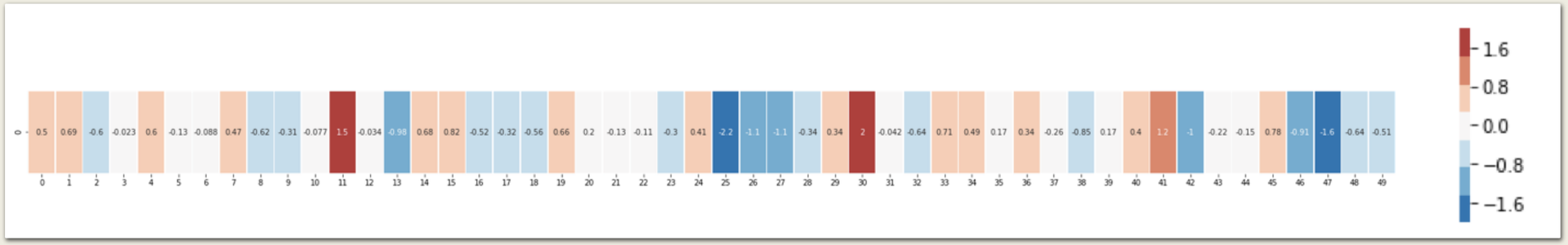
- Статистические модели в 1990-х годах (например, n-граммы).

- Переход к моделям глубокого обучения (например, word2vec, GloVe)

```
Welcome to
                EEEEEE  LL      IIII   ZZZZZZ   AAAAA
                EE      LL       II       ZZ   AA   AA
                EEEEE   LL       II     ZZZ    AAAAAAA
                EE      LL       II    ZZ      AA   AA
                EEEEEE  LLLLLL  IIII ZZZZZZ    AA   AA

Eliza is a mock Rogerian psychotherapist.
The original program was described by Joseph Weizenbaum in 1966.
This implementation by Norbert Landsteiner 2005.
```

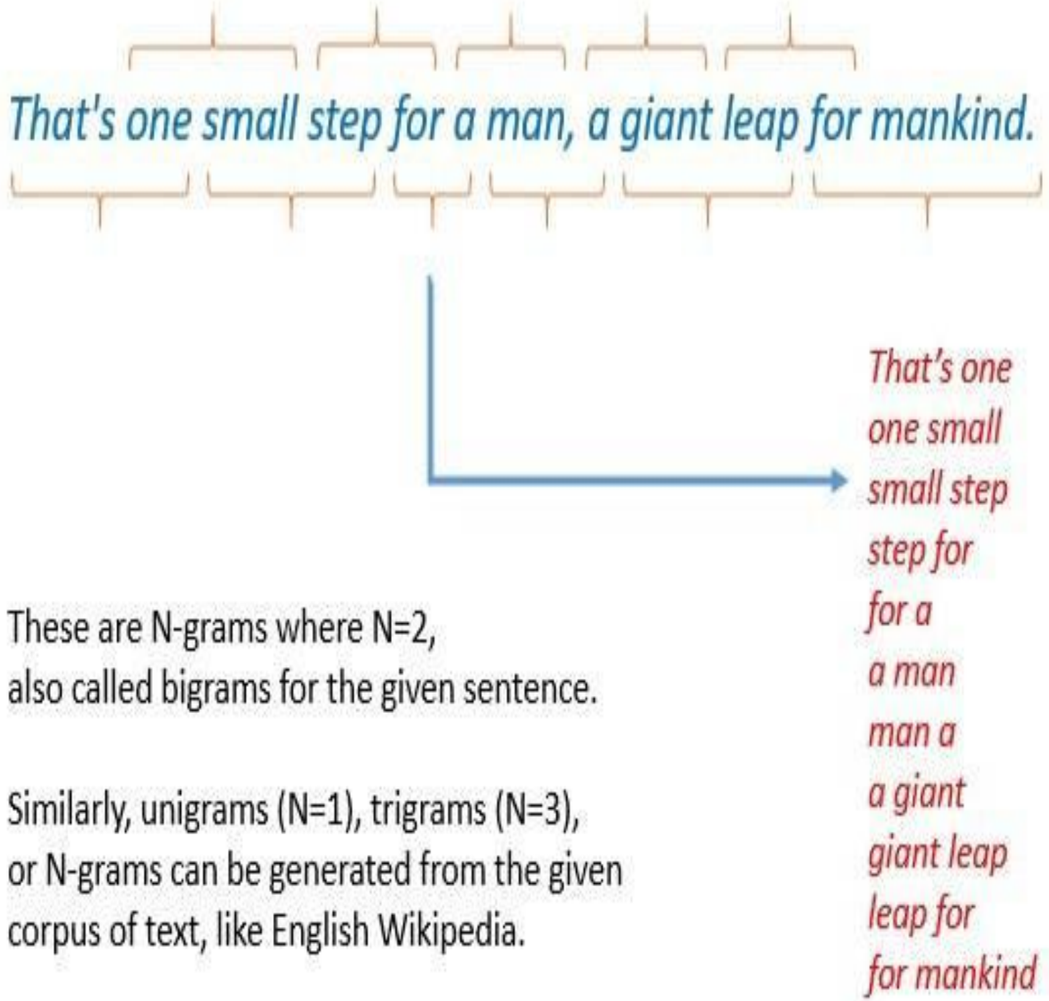


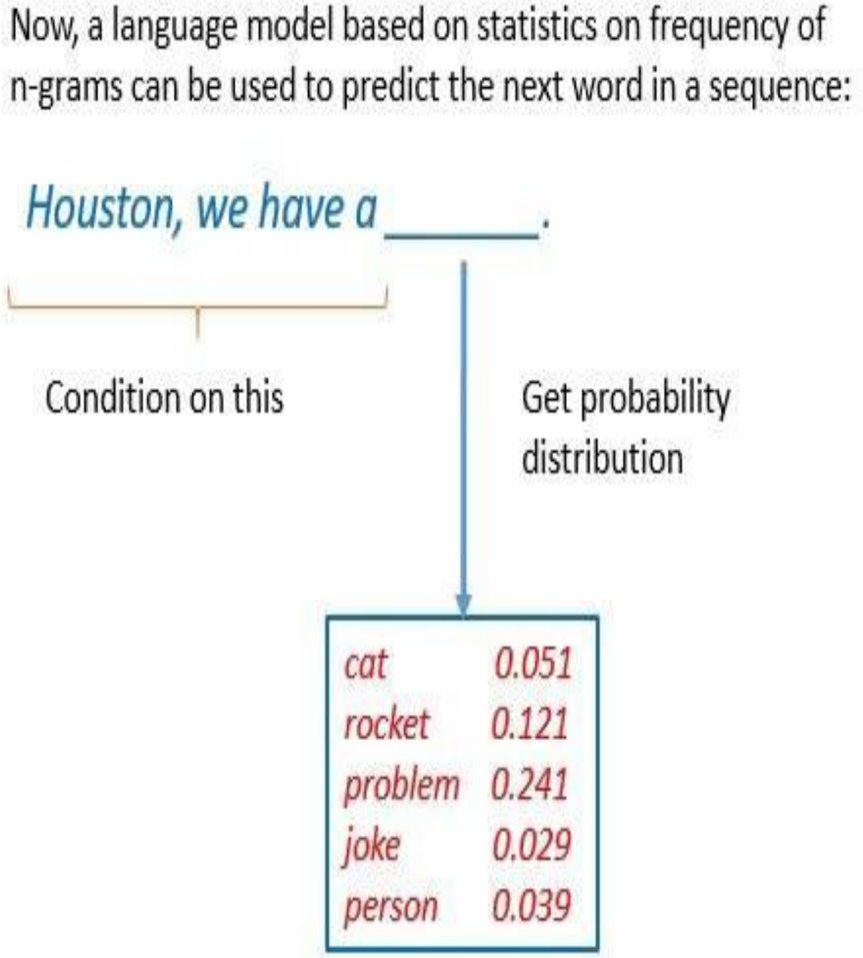


Языковое моделирование (LM)

используует **статистические и вероятностные методы для определения вероятности того, что данная последовательность слов встретится в предложении.** Следовательно, языковая модель - это, по сути, распределение вероятностей по последовательностям слов:
Здесь выражение вычисляет условное распределение вероятностей, где может быть любым словом из словаря.
Языковые модели **генерируют вероятности путем изучения одного или нескольких текстовых корпусов.** Текстовый корпус - это языковой ресурс, состоящий из большого и структурированного набора текстов на одном или нескольких языках. Текстовый корпус может содержать текст на одном или нескольких языках и часто снабжен комментариями.
Один из **самых ранних подходов к построению языковой модели основан на n-грамме**. N-грамм - это непрерывная последовательность из n элементов из заданного образца текста. Здесь модель предполагает, что вероятность следующего слова в последовательности зависит только от окна фиксированного размера предыдущих слов:

Языковое моделирование (LM)

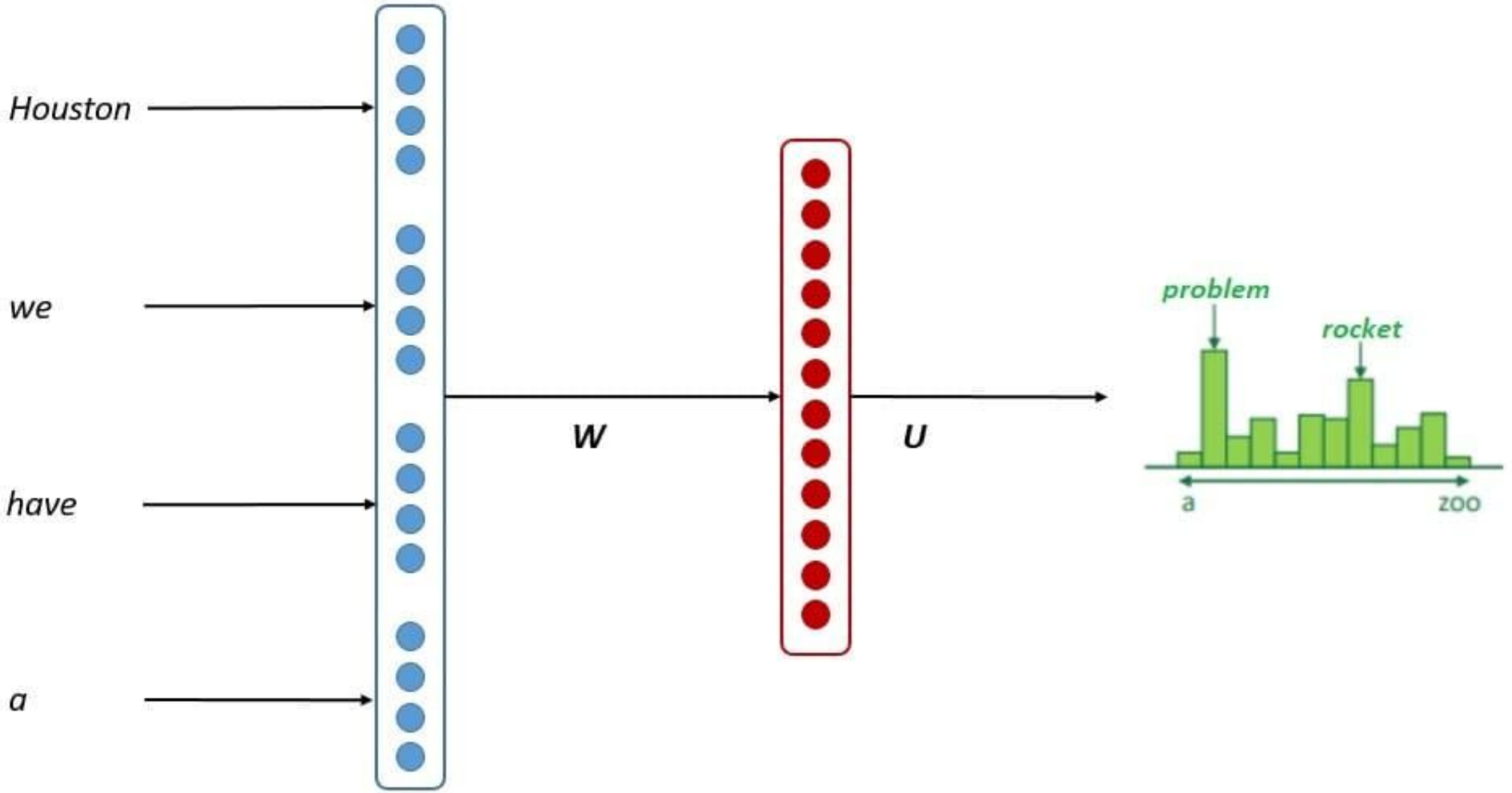




Языковое моделирование (LM)

Однако языковые модели на n-граммах были **в значительной степени вытеснены нейронными языковыми моделями.** Он основан на нейронных сетях, вычислительной системе, вдохновленной биологическими нейронными сетями. Эти модели используют непрерывные представления или вложения слов для составления своих прогнозов.

По сути, нейронные сети представляют слова распределенно как нелинейную комбинацию весов. Следовательно, это позволяет избежать проблему размерности в языковом моделировании.



x^1, x^2, x^3, x^4

Input text sequence

$e = [e^1; e^2; e^3; e^4]$

Concatenated word embeddings

$h = f(We + b_1)$

Hidden layer of the neural network

$\hat{y} = softmax(Uh + b_2)$

Output probability distribution

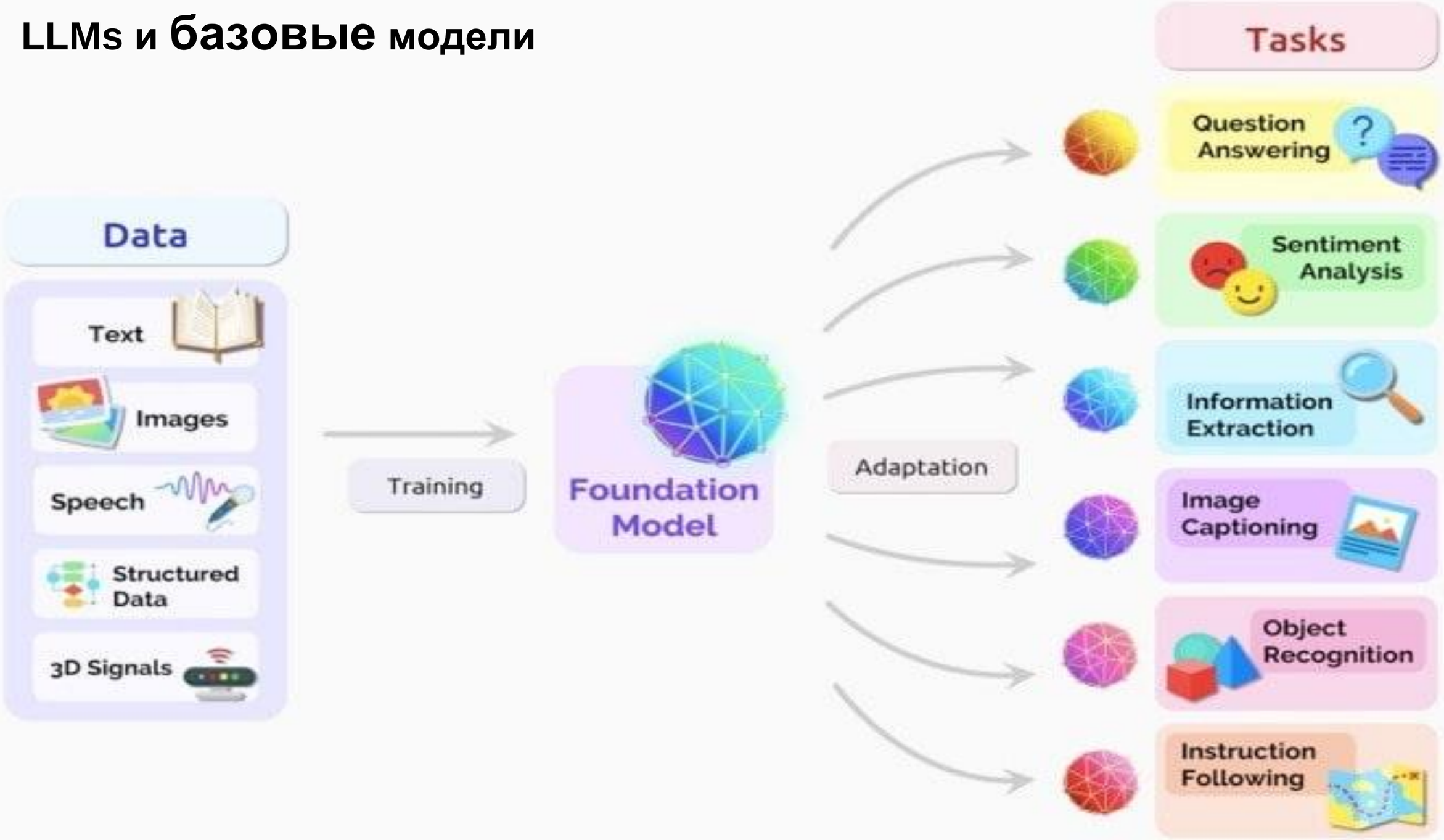
LLMs и базовые модели

Базовая модель обычно относится к **любой модели, обученной на обширных данных, которая может быть адаптирована к широкому кругу последующих задач.** Эти модели обычно создаются с использованием глубоких нейронных сетей и обучаются с использованием самостоятельного обучения на множестве немаркированных данных.

Этот термин был введен не так давно Стэнфордским институтом искусственного интеллекта, ориентированного на человека (HAI). Однако нет четкого различия между тем, что мы называем базовой моделью, и тем, что квалифицируется как модель большого языка (LLM).

Тем не менее, магистры права обычно обучаются на данных, связанных с языком, таких как текст. Но **базовая модель обычно обучается на мультимодальных данных**, сочетании текста, изображений, аудио и т.д. Что еще более важно, базовая модель предназначена для того, чтобы служить основой для более конкретных задач:

LLMs и базовые модели



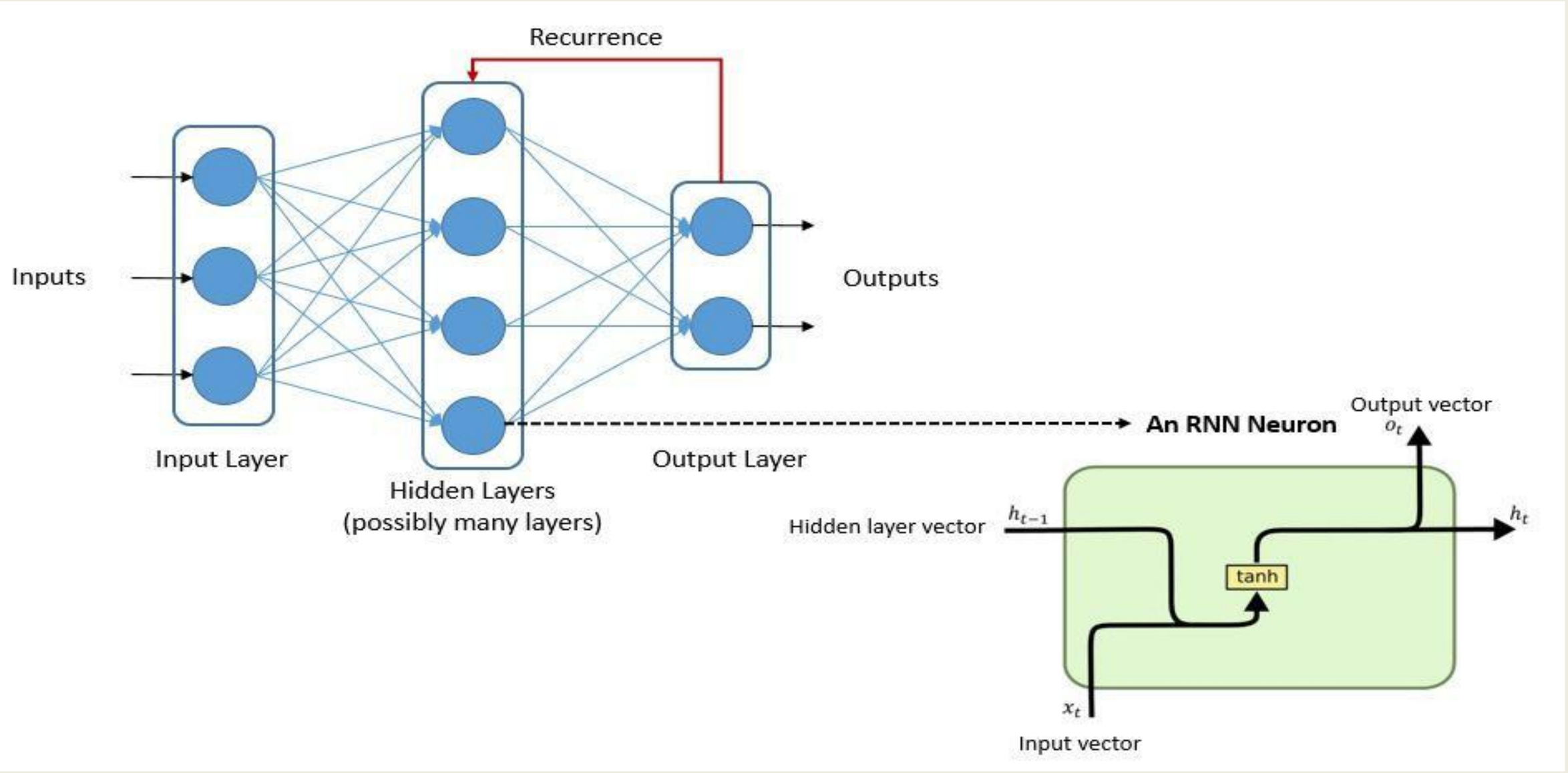
LLMs и базовые модели

Базовые модели **обычно дорабатываются с дальнейшим обучением различным последующим когнитивным задачам.** Точная настройка относится к процессу использования предварительно подготовленной языковой модели и ее подготовки к другой, но связанной задаче с использованием конкретных данных. Этот процесс также известен как трансферное обучение.

Более ранняя архитектура LLMS

Когда все начиналось, LLM в основном создавались с использованием алгоритмов самостоятельного обучения. Самостоятельное обучение относится к **обработке немаркированных данных для получения полезных представлений,** которые могут помочь в последующих задачах обучения.

Наиболее широко используемой архитектурой для LLM была рекуррентная нейронная сеть (RNN)



Исторический контекст и эволюция

Архитектура Трансформеров

- Представлена Vaswani и др. (2017) в статье *«Attention is All You Need».*
- **Ключевая инновация:** механизм внутреннего внимания, позволяющий модели эффективно сосредотачиваться на релевантных частях текста.
- Основа современных LLM, таких как GPT и BERT.
- **Законы масштабирования:** Увеличение размера модели и объема данных для обучения улучшает производительность, как показано в исследованиях OpenAI.

Механизм внутреннего внимания

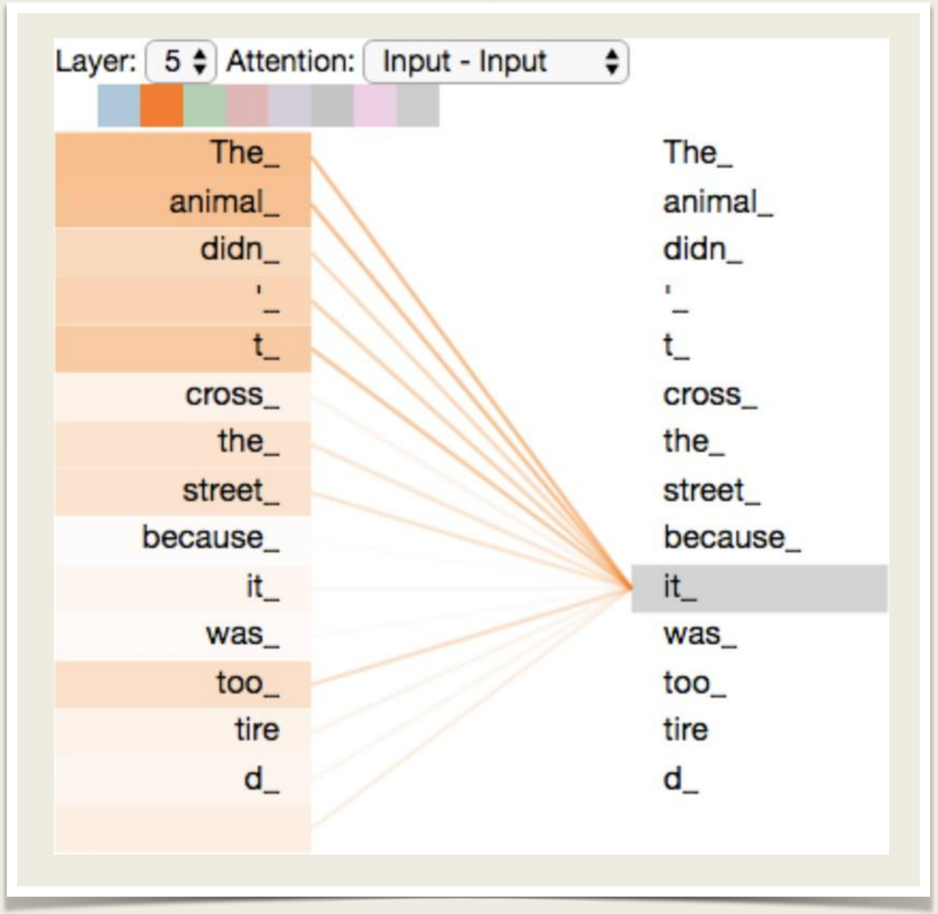
Внимание - это **метод усиления некоторых частей входных данных при одновременном уменьшении других частей**. Мотивация, стоящая за этим, заключается в том, что сети следует уделять больше внимания важным частям данных.

Механизм внимания относится к способности обращать внимание на разные части другой последовательности, самонаблюдение относится к способности обращать внимание на разные части текущей последовательности.

Механизм внутреннего внимания

Пример

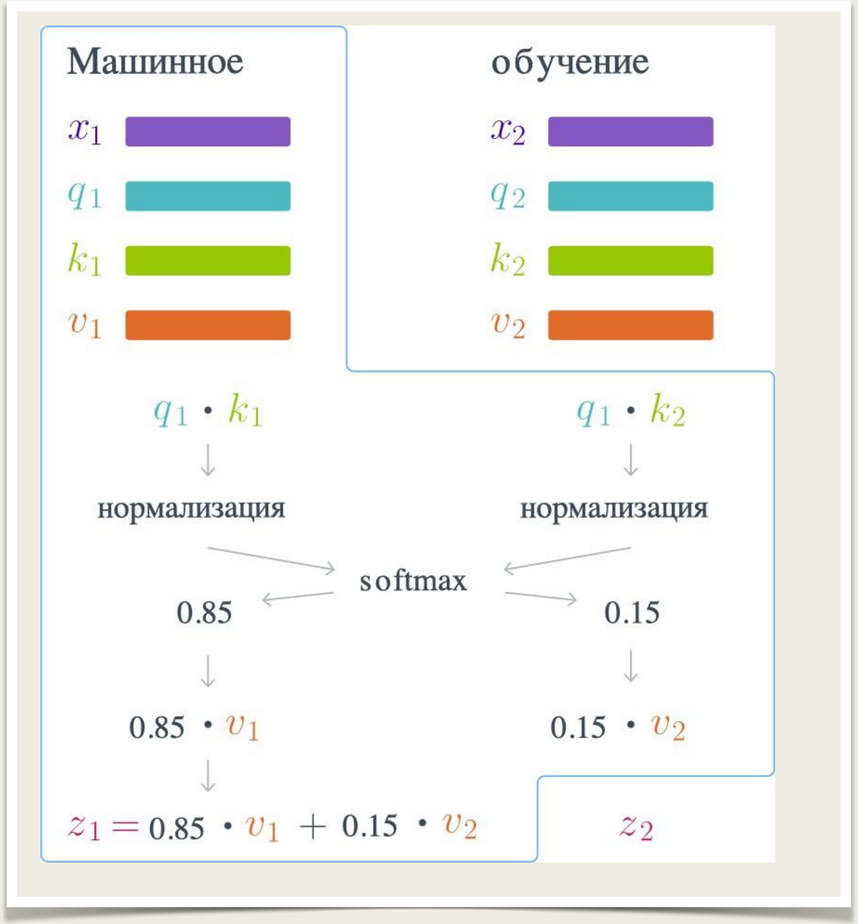
- *«Мама мыла раму. Она держала в руках тряпку».* Местоимение «она» относится к маме или к раме?
- Для человека это очень простой вопрос, но для модели машинного обучения — нет.
- **Механизм внутреннего внимания** помогает выучить взаимосвязи между токенами в последовательности, моделируя «смысл» других релевантных слов в последовательности при обработке текущего токена.



Механизм внутреннего внимания

Пример

- Для начала, из входного вектора формируются три вектора: Query (запрос), Key (ключ) и Value (значение).
- Процесс происходит так: по очереди фиксируется каждый токен (становится query) и просчитывается степень его связанности со всеми оставшимися токенами.
- Для этого поочередно key-вектора всех токенов скалярно умножаются на query-вектор текущего токена.
- Полученные числа будут показывать, насколько важны остальные токены при кодировании query токена в конкретной позиции.



Как работает LLM

Внедрение transformers командой Google Brain в 2017 году, пожалуй, является одним из самых важных переломных моментов в истории LLMs. Transformer - это **модель глубокого обучения, которая использует механизм самоконтроля** и обрабатывает весь ввод сразу.

Еще одним существенным преимуществом использования модели transformer является то, что они **более распараллеливаемы и требуют значительно меньшего времени на обучение**. Это именно то, что нам нужно для создания LLM на большом массиве текстовых данных с доступными ресурсами.

Как работает LLM

Ключевые компоненты

1. **Токенизация**: Разбиение текста на более мелкие единицы (токены).
2. **Слой вложений**: Преобразует токены в плотные числовые векторы.
3. **Блоки трансформеров**:
   - Многоголовое внутреннее внимание (self-attention): Определяет связи между токенами.
   - Полносвязные сети: Обрабатывают и уточняют информацию.
4. **Выходной слой**: Генерирует предсказания или текст.

Как работает LLM

Процесс обучения

- **Предварительное обучение**: Модель изучает языковые шаблоны, предсказывая пропущенные слова (маскированное языковое моделирование) или следующие слова (причинное языковое моделирование). LLM часто заранее обучаются на огромном наборе данных без контроля, изучая общую структуру и контекст языка.

- **Тонкая настройка**: Адаптация модели для конкретных задач с использованием небольших специализированных наборов данных.

- **Обучение с подкреплением и обратной связью от человека (RLHF)**

  - Методика для согласования выходных данных модели с предпочтениями человека.

  - Широко используется в системах, таких как ChatGPT.

Применение LLMs

Генеративные приложения:

Чат-боты (например, ChatGPT, Bard).

Создание контента: статьи, истории, сценарии.

Генерация кода: Copilot, ChatGPT для программирования.

Аналитические приложения

Краткое содержание текста.

Анализ тональности.

Семантический поиск.

Специализированные области

Здравоохранение: Поддержка медицинской диагностики.

Образование: Персонализированное обучение.

Юриспруденция: Обзор и суммаризация документов.

Проблемы и ограничения

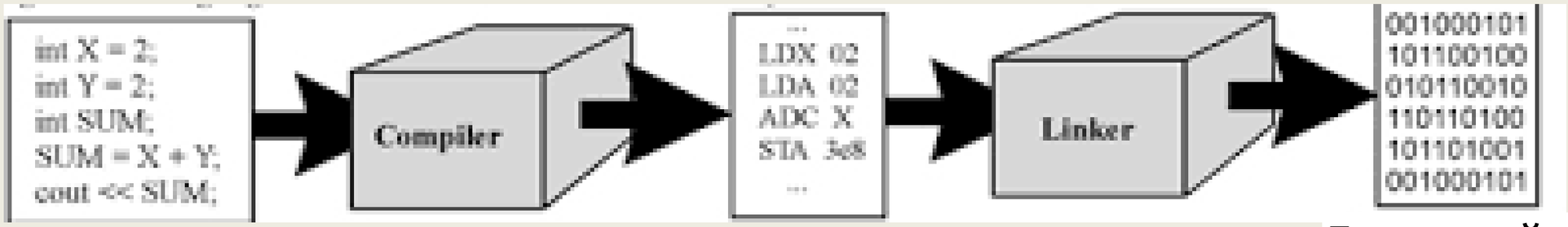
Технические проблемы & Этические вопросы

- **Технические проблемы**
  - **Затраты на вычисления и энергию**: Обучение LLMs требует значительных вычислительных ресурсов.
  - **Смещения в данных**: Модели могут наследовать смещения, присутствующие в обучающих данных.
  - **Длина контекста**: Трудности в обработке очень длинных текстов.
- **Этические вопросы**
  - **Дезинформация**: Генерация убедительного, но неверного контента.
  - **Конфиденциальность**: Риск воспроизведения конфиденциальной информации из обучающих данных.
  - **Доступность**: Ограниченный доступ из-за стоимости и инфраструктурных требований.

Заключение

- LLM являются краеугольным камнем современного ИИ, **революционизируя взаимодействие человека и машины.**
- Ответственная разработка и внедрение имеют ключевое значение для увеличение их преимуществ при уменьшении рисков.

Язык программирования как инструкция для машины vs. Команды для LLM как инструкции для машины



C++ ASCII Бинарный код

Спасибо за внимание!