Прототип

Платформа машинного обучения

Reporter: Andrey Ye Shevel
The crew: Alexey Naikov, Anatoly Oreshkin, Nikita Samokhin, Alexey Shvetsov, Anatoly Titov (young)

План

- Машинное обучение в области ФВЭ и Нобелевские премии 2024.
- Большие языковые модели (Large Language model – LLM).
- Масштабы крупнейших установок машинного обучения.
- Схема прототипа **Платформа Машинного Обучения**
- Демонстрация прототипа
- Заключение

Машинное обучение в ФВЭ

- Машинное обучение (МО) = технология включающая комбинацию алгоритмов и статистических методов для создания компьютерных систем способных самостоятельно обучаться на представленных наборах данных. Обученная система может **генерировать** ответы на вопросы или рекомендации в предметной области определённой представленными данными. Такие системы обычно реализуются в форме искусственных нейронных сетей.
- Компоненты технологии МО активно применяются в ФВЭ
  - https://www.epj-conferences.org/articles/epjconf/abs/2024/05/contents/contents.html#section_10.1051/epjconf/202429509001
  - Artificial Intelligence and Machine Learning
    - 32 доклада на CHEP 2023 (May 8 – May 12) Jefferson Lab
  - Причины – экономия времени в конкретных физических исследованиях и не только.

Higgs Boson Machine Learning Challenge 2014

- Improving statistical significance: The goal of using machine learning was to increase the statistical significance of the Higgs boson discovery. Traditional "cut-based" techniques were replaced with more advanced multivariate classification methods.
- Real-time filtering: Neural network algorithms were used to speed up online event filtering, helping to process the billions of proton collisions produced by the LHC.
- Ongoing research: Even after the initial discovery, machine learning continues to play a crucial role in making precise measurements of Higgs boson properties.
- Performance improvement: The top machine learning solutions in the Higgs Boson Challenge achieved significantly better performance than traditional physics software. The winning solutions used advanced techniques such as ensemble methods and deep neural networks.

Claire Adam-Bourdarios at al // The Higgs boson machine learning challenge // http://proceedings.mlr.press/v42/cowa14.pdf

Nobel prize winners 2024

- John J. Hopfield
  - Initial pubs: Neural networks and physical systems with emergent collective computational abilities. // Proc Natl Acad Sci U S A. 1982 Apr; 79(8): 2554–2558. doi: 10.1073/pnas.79.8.2554
- Geoffrey E. Hinton
  - Initial pubs: A Learning Algorithm for Boltzmann Machines // COGNITIVE SCIENCE 9, 147-169 (1985) https://www.cs.toronto.edu/~fritz/absps/cogscibm.pdf

Contemporary study Higgs boson with Artificial Neural Networks

- Stephen Roche, Quincy Bayer, Benjamin Carlson, William Ouligian, Pavel Serhiayenka, Joerg Stelzer, Tae Min Hong
- Nanosecond anomaly detection with decision trees and real-time application to exotic Higgs decays
- https://doi.org/10.1038/s41467-024-47704-8
  - Scenarios at the Large Hadron Collider at CERN are considered, for which the autoencoder is trained using known physical processes of the Standard Model. The design is then deployed in real-time trigger systems for anomaly detection of unknown physical processes, such as the detection of rare exotic decays of the Higgs boson.

Statistical Mechanics of Deep Learning

by Yasaman Bahri et al

- Statistical Mechanics of Deep Learning
  - Annu.Rev.Condens.Matter Phys.2020.11:501–28 // https://doi.org/10.1146/annurev-conmatphys-031119-050745
  - Keywords: neural networks, machine learning, dynamical phase transitions, chaos, spin glasses, jamming, random matrix theory, interacting particle systems, nonequilibrium statistical mechanics.

Простой способ использования моделей МО в ФВЭ

1) Генерация данных симуляции предполагаемого физического процесса.
2) Обучение (тренировка) модели МО в форме искусственной нейронной сети на данных симуляции из пункта 1).
3) Поиск похожих событий в реальных измерительных данных с помощью обученной модели МО.
    1) Отметим, что в данном случае имеем дело с цифровыми данными.

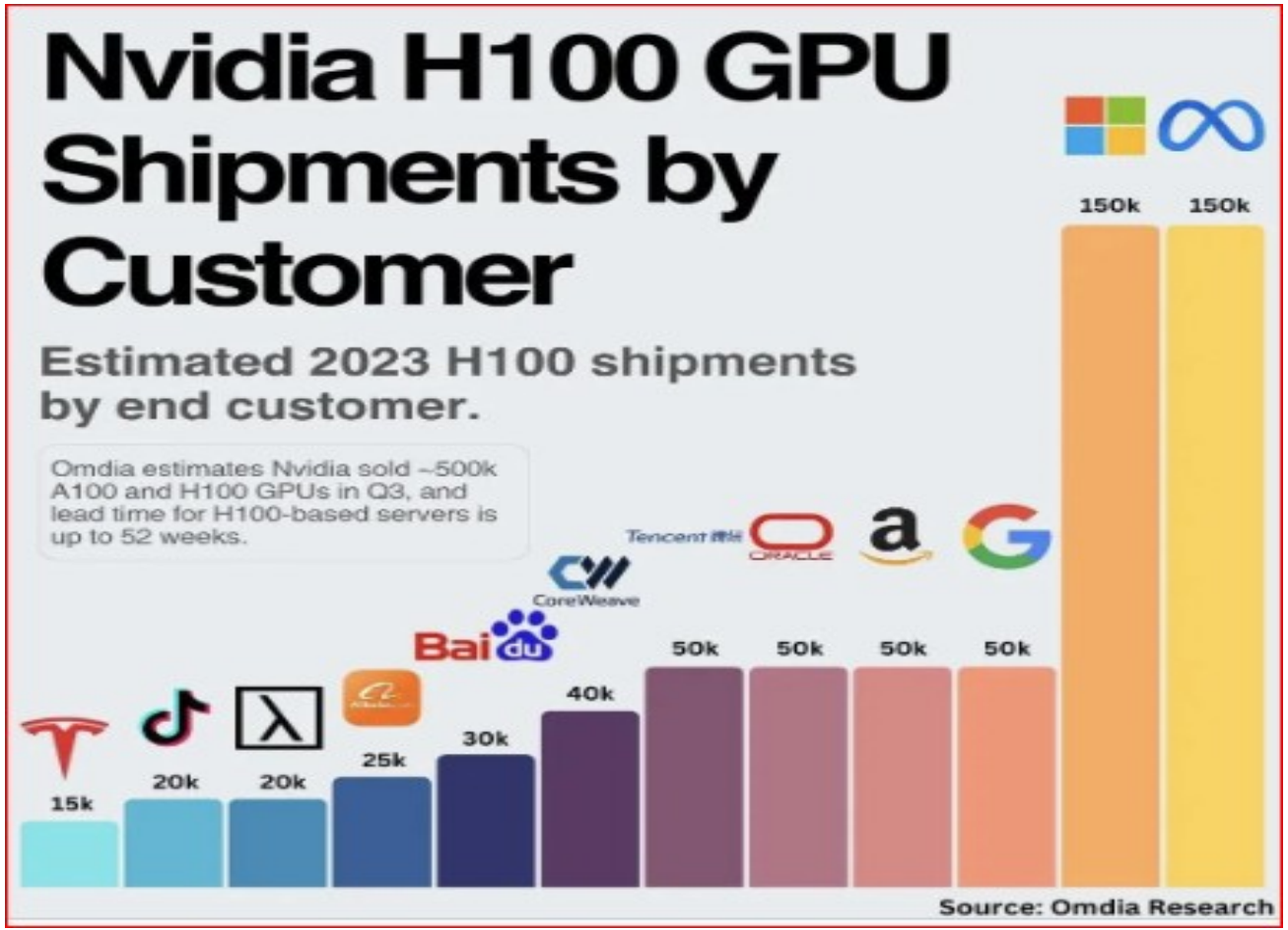
Языковые модели МО

- Использование больших языковых моделей: Large Language Model – LLM (в форме *искусственная нейронная сеть*) для перевода с одного языка на другой.
  - Русский – Китайский и обратно
  - Русский – математические формулы и обратно
  - Русский – химические формулы и обратно
  - Русский – диаграммы Фейнмана и обратно
  - Русский – изображения и обратно
  - Русский текст - описание текста и обратно
- Заметное отличие состоит в том, что здесь надо преобразовать нечисловые данные (буквы, слова, фразы, абзацы, статьи, изображения) в цифровые данные.

Examples of Large Language Models

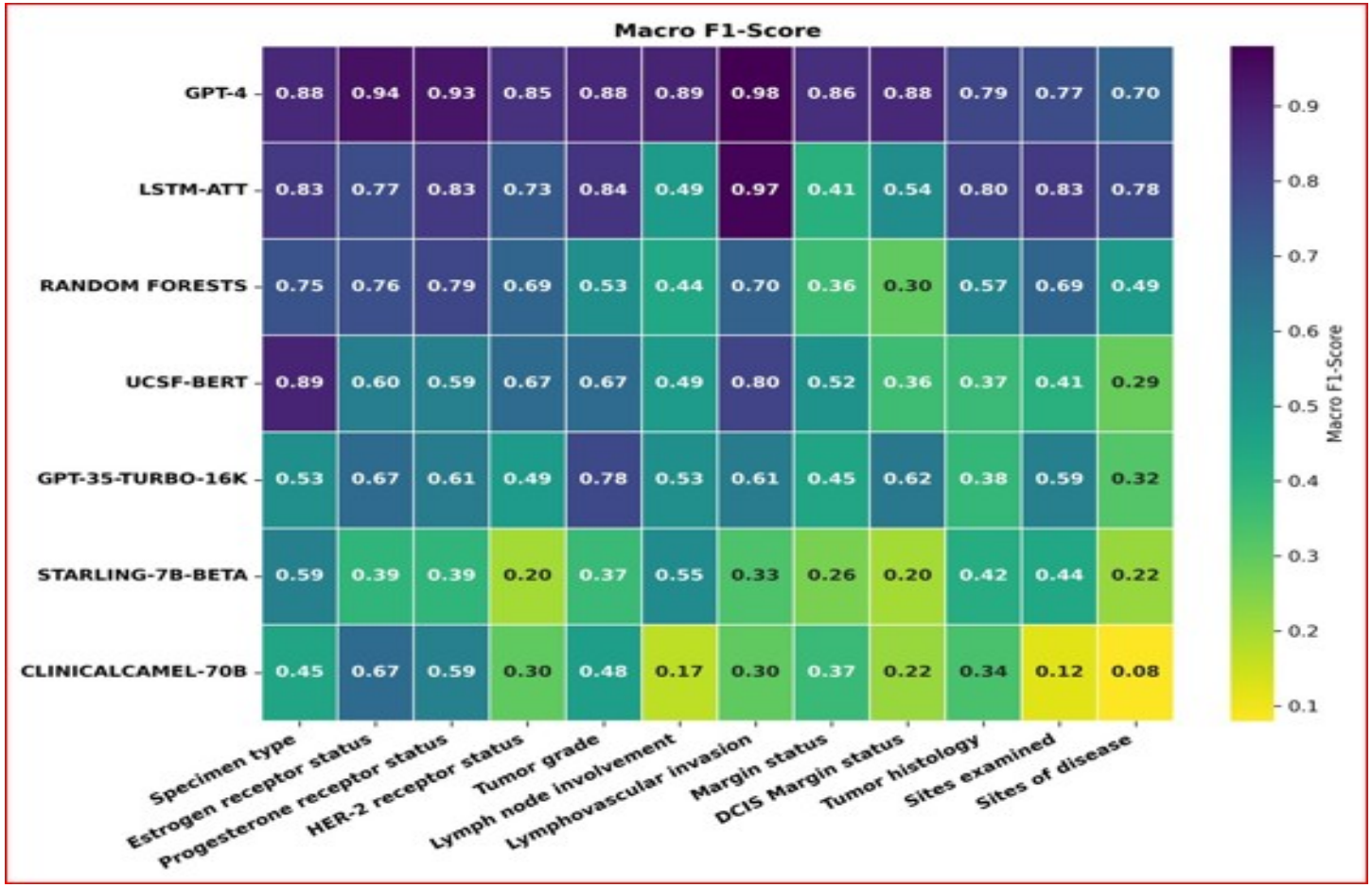
- A large language model (LLM) is a type of artificial intelligence program that can be used to understand, interpret and generate human language. Most of them are *foundation models*.
  - GPT series (GPT, GPT-2, GPT-3, GPT-4) by OpenAI
  - BERT by Google
  - T5 by Google
  - LLaMA and LLaMA 2 by Meta
  - Claude series by Anthropic
  - Gemini by Google
  - PaLM by Google
  - BLOOM by BigScience
  - Galactica by Meta
  - Jurassic-1 by AI21 Labs
  - Megatron-Turing NLG by NVIDIA and Microsoft
  - ERNIE by Baidu

Estimated shipments of Nvidia H100 GPU by customers



From: https://www.reddit.com/r/singularity/comments/1890o9y/nvidia_gpu_shipments_by_customer/

Стохастический характер генерации ответов моделей



From: https://doi.org/10.1093/jamia/ocae146

Развитие проекта: прототип

- На семинаре ОФВЭ (14 ноября 2023 - https://hepd.pnpi.spb.ru/hepd/events/abstract/2023/Monitoring-2023-11-13-1928-new.pdf) под названием “Monitoring of large scale technical systems” я представил проект “Automate assistance for administrators”, который предполагал сценарий *“вопрос человека – ответ модели”*.
- Сегодня рассматривается прототип машинного обучения в качестве помощника разработчик[а/ов] некоторой технической системы в сценарии “**вопрос человека – ответ прототипа**”, т.е. ***чатбот на основе LLM для организации диалога с текстом описания*** разработанной (или разрабатываемой) системы.
- В качестве основы прототипа предложено использование свободно распространяемых LLM, т.е. использовать технологию “передача знаний (knowledge transfer)”.
- На основе упомянутых выше соображений был разработан прототип “**Платформа машинного обучения (МО)**”.

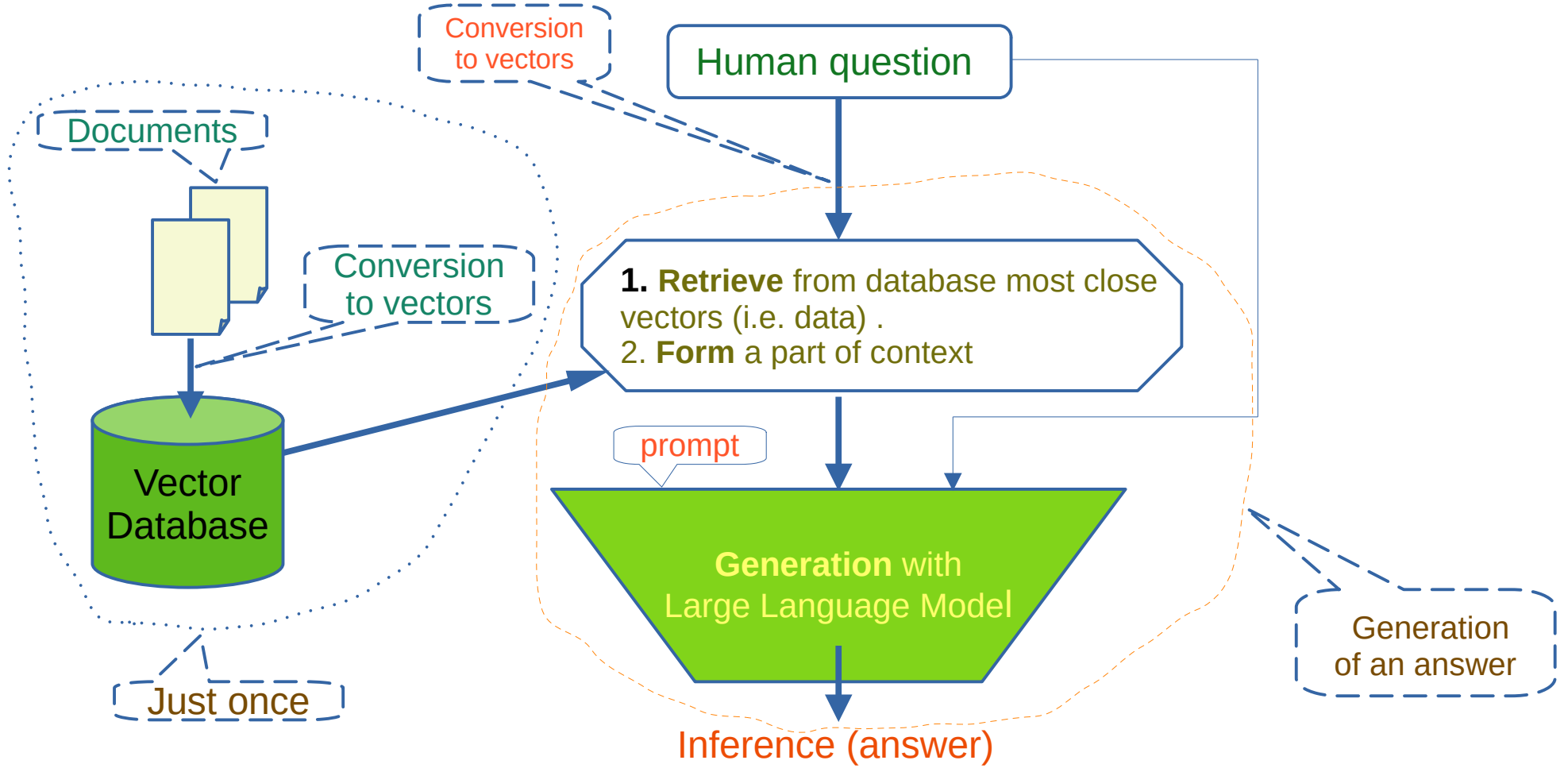
Выбор архитектуры

- Для прототипа платформы МО был выбран подход Retrieval Augmented Generation (RAG) - Извлечение (данных) Дополненное Генерацией.
  - Извлечение означает поиск по базе(ам) данных.
  - Генерация – использование одной или более моделей LLM.
- Основные достоинства:
  - Всё выполняется локально с использованием локальных документов, т.е. за пределы системы не выдаются никакие данные.
  - Для локальных задач не требуются громадные вычислительные мощности.

Пребразование документов в векторную базу данных

- Преобразование текста описаний (в одном из популярных форматов, например PDF) в векторную базу данных называется embedding.
- Введённый вопрос (или утверждение) также преобразуется в векторную форму, т.е. также выполняется embedding.
- Embedding обычно выполняется специально тренированной нейронной сетью.

Basic (naive) Retrieval Augmented Generation (RAG) – emerged in 2021



Формирование запроса к LLM

- Поиск подходящих данных по векторной базе данных и другим базам может выполняться в соответствии с мерой близости с использованием разных алгоритмов или комбинацией алгоритмов.
- Результат поиска представляет собой некоторый текстовый отрезок, который будет использоваться как часть контекста для LLM.
- После сего полученный контекст, который включает *введённый вопрос*, *найденные текстовые отрезки*, *инструкция (prompt)* для LLM подаются на вход LLM для генерации ответа.
  - Ключевой термин – генерация, который носит стохастический характер.

Примеры мер близости между векторами

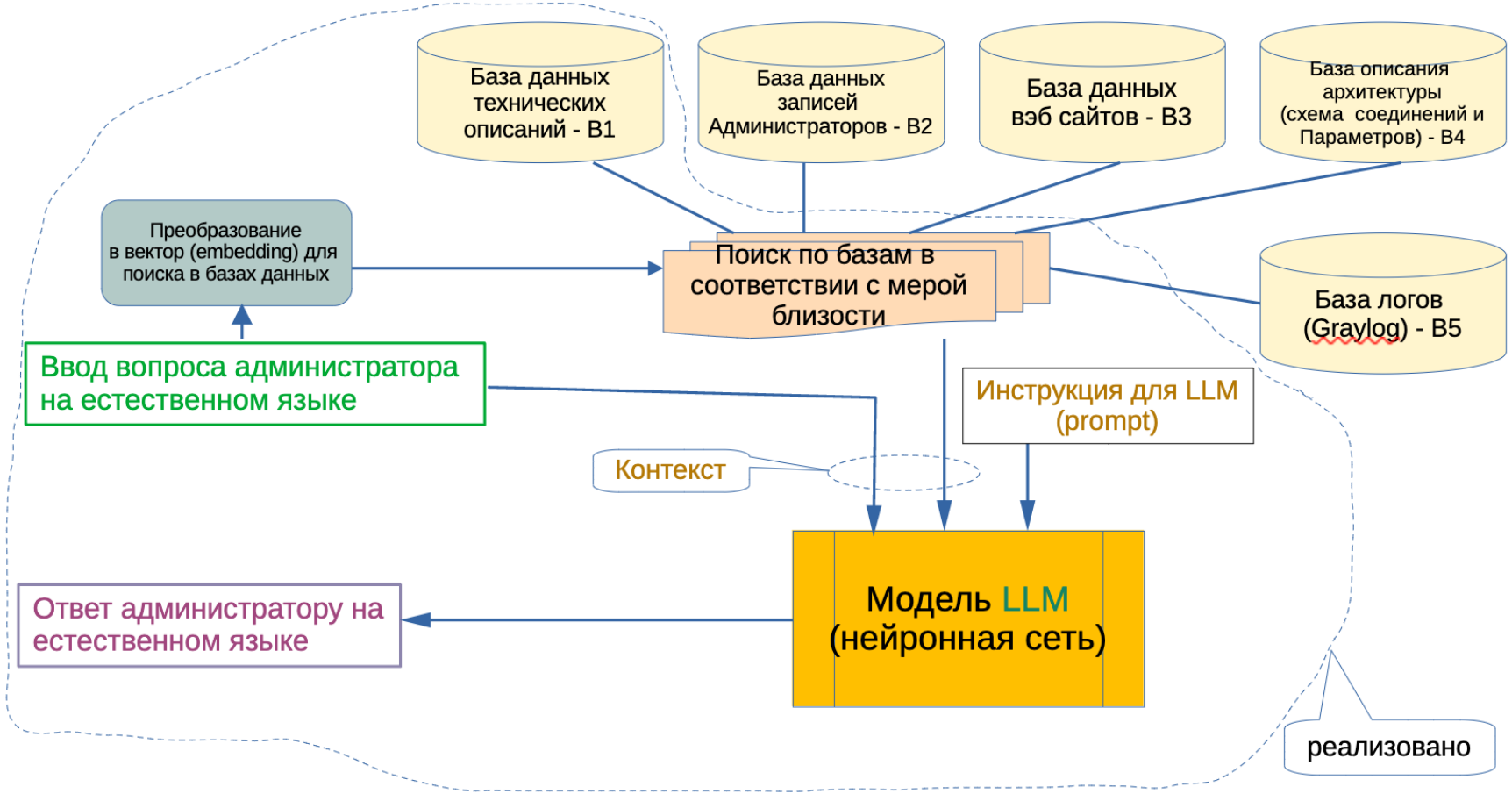
The Euclidean distance between two vectors u = (u_1, u_2, ..., u_n) and v = (v_1, v_2, ..., v_n) is given by:

$$d = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2 + ... + (u_n - v_n)^2}$$

$$\cos(\theta) = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

1. **Range**: Cosine similarity values range from -1 to 1.
   - 1 indicates perfect similarity (vectors point in the same direction)
   - 0 indicates no similarity (vectors are perpendicular)
   - -1 indicates perfect dissimilarity (vectors point in opposite directions)

Реализация платформы МО



Прототип реализован в нескольких инстансах

1) RAG архитектура запущена на сервере **cryoem** (ОМРБ) – (документы с описанием на русском сегмента серверной инфраструктуры ОФВЭ)
   - Вопросы на естественном русском задаются посредством специального скрипта на pcfarm передаются в RAG архитектуру на сервере **cryoem**.
   - Полученный от RAG на **cryoem** ответ можно прочесть на pcfarm.
2) RAG архитектура (документы описания на английском RedHat Linux) запущена на лаптоп с вэб интерфейсом.
3) RAG архитектра запущена на сервере pnpi-itmo (выполняется очень медленно!).

Первые тесты реализованной RAG архитектуры

- В качестве документов взяли знакомое для нас текстовое описание сегмента компьютерной архитектуры и выполнили преобразование текста описания в векторную базу данных (embedding).
- Ввод вопросов на естественном (русский, английский) языке.
- Первоначально половина ответов RAG архитектуры оказалась неточными или совсем неверными.
- Сравнивая полученные неточные/неверные ответы с текстом описания стало ясно, что ряд важных аспектов в тексте описания компьютерного сегмента не представлено с должной ясностью и полнотой.
  - Таким образом для улучшения качества ответов RAG архитектуры потребовалось уточнить формулировки в описании сегмента компьютерной архитектуры, а также уточнить некоторые запросы к режиму функционирования RAG архитектуры.
  - После уточнения текста описания и других улучшений тестирование повторялось.
  - Цикл – тестирование ↔ коррекция был повторен несколько раз, пока результаты тестирования не стали приемлемыми..

Цикл настройки архитектуры RAG

1) Преобразование текста описания сегмента компьютерной архитектуры в векторную базу данных для RAG архитектуры.
2) Подготовка вопросов, получение ответов от RAG архитектуры, анализ ответов RAG.
3) Если ответы адекватны, то завершить тестирование. В противном случае коррекция, описания и инструкций для настройки RAG архитектуры.
4) Переход к пункту 1).

Взаимодействие с разработчиком

- В процессе коррекции описания было выполнено следующее:
  - Уточнили формулировки и добавили некоторые разделы в тексте описания.
  - Пояснили друг другу что есть что и откуда.
  - Иными словами RAG архитектура использовалась в цикле взаимодействия с разработчиками как технический помощник разработчика.

Тестировалось несколько моделей

- Несколько свободно распространяемых моделей LLM (llama3.1, mixtral, aya:35b, etc) на платформе ollama
- Несколько свободно распространяемых моделей для embedding (LaBSE, paraphrase-multilingual-MiniLM-L12-v2, sbert_large_nlu_ru, etc.)
  - Небольшое число компонент
  - pip list | wc -l
  - 267

Выводы 1

- Точность/полнота ответов архитектуры RAG
  - Близость LLM к конкретной тематике документов.
  - Близость embedding к используемому языку.
  - Точность инструкций (prompt) к модели LLM.
  - **Полнота и ясность текста документов.**

Выводы 2

- Необходимо принимать во внимание процесс взаимодействия разработчик(и) ↔ RAG архитектура в период разработки технической системы, где применяется RAG.
  - Обсуждение внутри команды разработчиков для уточнения ответов RAG архитектуры продвигает общее понимание задачи.
  - Наличие разработанной RAG архитектуры в конкретном техническом проекте упрощает
    - ввод новых участников в процесс разработки;
    - эксплуатацию разработанной системы;
    - дальнейшую модернизацию разработанной системы;
    - создание цифрового двойника разрабатываемой технической системы системы.

Живая демонстрация

- Демонстрация инстанса платформы на сервере cryoem – диалог с описанием на русском серверной инфраструктуры.
- Демонстрация инстанса платформы на лаптоп – диалог с описанием на английском RedHat Enterprise Linux (RHEL) (вэб интерфейс).

Заключение

- Предполагается, что в обозримом будущем RAG архитектура будет неотъемлемой частью разработок крупных технических систем как дополнительный элемент к обычной технической документации. Так что адаптацию технологий машинного обучения, в частности RAG, следует продолжать.
- Для освоения машинного обучения следует подготовить задание на создание коммунального стенда достаточной производительности для хранения библиотек языковых моделей, хранилища документов и прочее необходимое для сопровождения стенда.
- Рассмотреть потенциальные применения технологии машинного обучения при подготовке эсперимента на НИКА (ОИЯИ), других разработках.
- Привлечь студентов разного возраста.

Thanks for the attention!

You might take a look at spare slides.

what is difference intelligence and intellect

Intelligence and intellect are related but distinct concepts. Intelligence is generally understood as a mental quality that involves the ability to learn from experience, adapt to new situations, understand and handle abstract concepts, and use knowledge to manipulate one's environment. It is a combination of cognitive processes such as perception, learning, memory, reasoning, and problem-solving, all directed toward effective adaptation to the environment.

Intellect, on the other hand, refers to the capacity for grasping with the intellect, often associated with comprehension and the ability to understand complex ideas. It is commonly linked to cognitive abilities like reading skills and language comprehension.

In summary, intelligence encompasses a broader range of cognitive abilities aimed at adaptation and problem-solving, while intellect focuses more on understanding and comprehension.

Как использовать AI модели локально

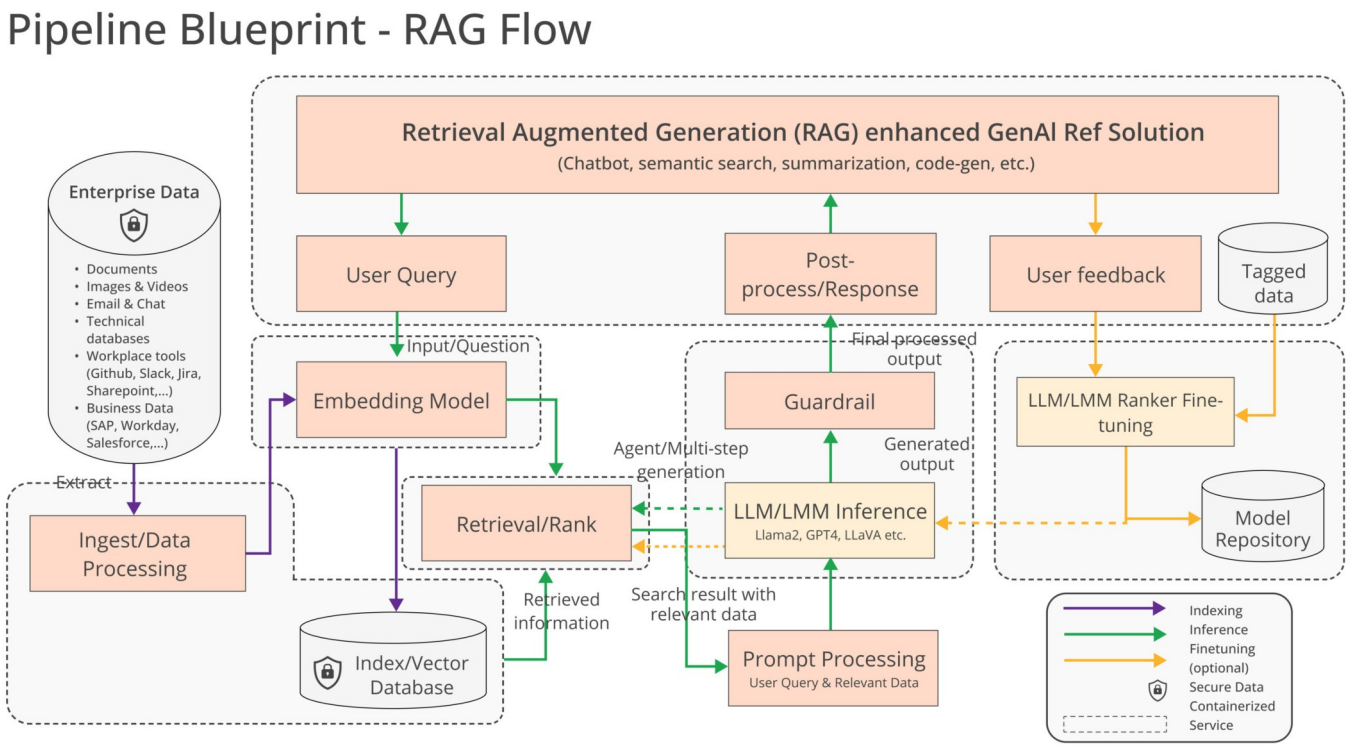
- https://matilabs.ai/2024/02/07/run-llms-locally/
- Llama https://github.com/meta-llama/llama-recipes

Доступные для нас модели

- Ollama.ai/library (около 50 тренированных моделей)
- NoteBookLM.google.com
- Perplexity.ai

RAG on Linux Foundation

- https://opea.dev/ - Open Platform for Enterprise AI



Publications-1

- Humza Naveed et al // A Comprehensive Overview of Large Language Models // https://arxiv.org/pdf/2307.06435.pdf
- Patrick Lewis et al // Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks // https://arxiv.org/pdf/2005.11401 they introduced in 2021
- The National Artificial Intelligence Research Resource (NAIRR) Pilot https://nairrpilot.org/
- Yuntong Hu et al // GRAG: Graph Retrieval-Augmented Generation (2024) // https://arxiv.org/pdf/2405.16506
- ADVANCED RESEARCH DIRECTIONS ON AI FOR ENERGY // Report on Winter 2023 Workshops // https://www.anl.gov/sites/www/files/2024-04/AI-for-Energy-Report_APRIL%202024.pdf
-

Publications-2

- Researchers built an ‘AI Scientist’ — what can it do?
  https://www.nature.com/articles/d41586-024-02842-3 and about AI-presentation-2024-09-10-1012.odp
- Can LLMs Generate Novel Research Ideas? A Large-Scale Human Study with 100+ NLP Researchers
  Chenglei Si, Diyi Yang, Tatsunori Hashimoto
  https://www.arxiv.org/abs/2409.04109
- Can LLMs Generate Novel Research Ideas? A Large-Scale Human Study with 100+ NLP Researchers
  Chenglei Si, Diyi Yang, Tatsunori Hashimoto
  https://www.arxiv.org/abs/2409.04109