Natural Language Processing

Large Language Models

W5 Agenda

- What are LLMs
- Prompt Engineering
- Model Fine-tuning
- Retrieval Augmented Generation
- Building an LMM powered App

What are LLMs

What are LLMs?

Large Language Models (LLMs) are Generative AI models designed to understand, generate, and interact with human language. They can process and generate text, answering questions, creating content, and even engaging in conversation.

LLMs are trained on vast datasets of text from the internet, including books, articles, and websites, using deep learning techniques. This training enables them to learn language patterns, grammar, and context. Despite their impressive abilities the training is still focused on next word prediction

They are used in a variety of applications such as chatbots, content creation, language translation, and sentiment analysis. LLMs are integral to enhancing human-computer interaction and automating complex language tasks.

Most popular LLMs include the pioneers in the domain such as Google's BERT and OpenAi's GPTs. Models from challengers such as Claude or open sourced models from Mistral AI are also catching up in performance.

Key LLM training steps

Data Preparation & Tokenization

- Collection of a vast and diverse datasets such as books, websites, and other textual materials.
- Preprocessing of this data to clean and format it for training
- Tokenizing text into format procesable by models

Model Architecture Design

- Transformer architecture allowed rapid growth of LLMs
- Different architectures and model sizes (params) have significant impact on performance
- BERT uses Bidirectional Encoder only architecture
- GPT uses Decoder only uni-directional architecture
- T5 keeps whole Encoder-Decoder setup

Pre Training

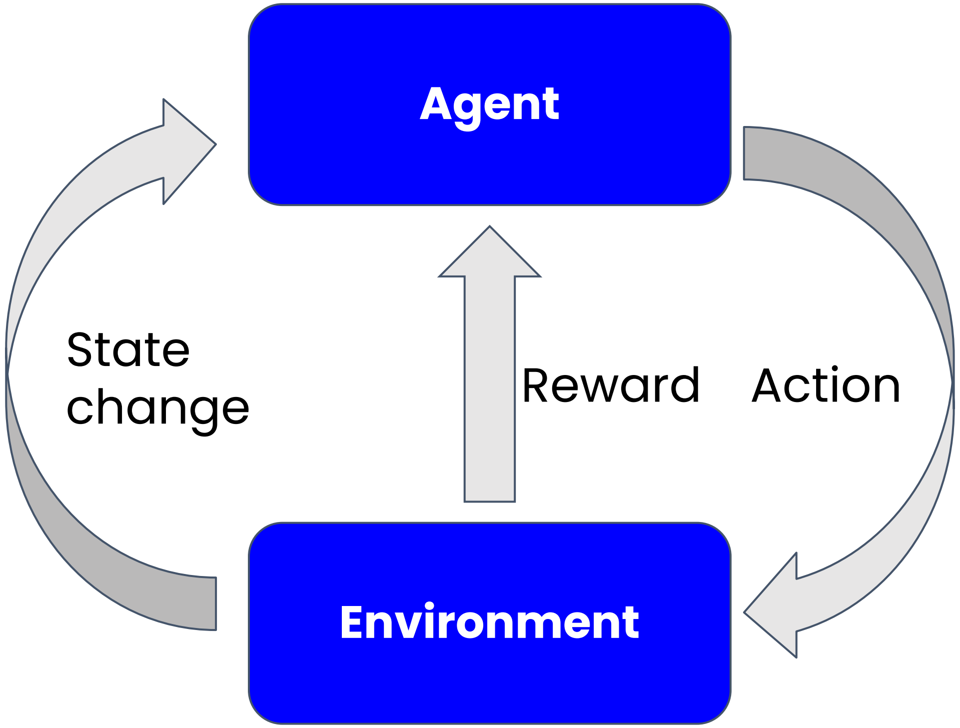
- The model undergoes unsupervised learning, where it learns to predict the next word in a sentence by being fed large amounts of text.
- This stage is critical for the model to learn language patterns, grammar, context, and general world knowledge.

Fine Tuning & RLFH

- Model is further trained on specific tasks and datasets in supervised learning setup
- The model receives feedback from human trainers to correct mistakes and improve its understanding.
- This iterative process helps in refining the model's responses and reducing biases.

Reinforced learning from human feedback

Reinforced learning logic



Reinforced learning logic

Intruct fine-tuned LLM

Human Feedback

- Rank different answers from trained models
- Maximize helpfulness and relevance
- Minimize harm
- Avoid dangerous topics

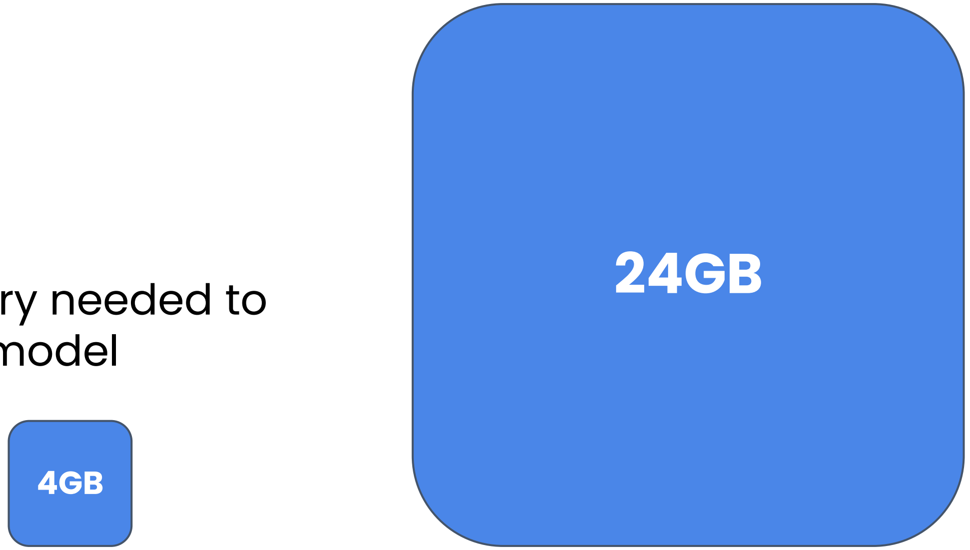
Human aligned LLM

Memory requirements

Approximate GPU RAM to train 1B params

Memory needed to store model

Memory needed to train model - weights, optimizer, calculating backpropagation



GPT 3.5:
175 billion
-> 4 200 GB

GPT 4:
1.76 trillion parameters
-> 42 240 GB

How LLMs changed ML

LLM Development	Classic ML
• No ML expertise needed • No training examples and clear loss function • Reasonable output without training • All communication with model based on natural language prompt • Model aims to follow prompt instructions - loss function is not that clear and easy to change	• ML expertise needed to get started • Training samples needed • Needs to be trained for a specific task • All communication with model based on natural language prompt • Model aims to minimize a loss function

Evaluating LLM performance

Human labeled benchmark datasets

GLUE (General Language Understanding Evaluation) and SuperGLUE Benchmarks:

- Designed to evaluate natural language understanding (NLU).
- Includes a series of tasks like sentiment analysis, question answering, and textual entailment.
- SuperGLUE is an advanced version of GLUE with more challenging tasks.

BLEU (Bilingual Evaluation Understudy) Score for Translation Tasks:

- Commonly used for evaluating the quality of machine-translated text compared to human translations.
- Focuses on how many words and phrases in the machine translation appear in the human translation.
- Commonly used for text translation

ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

- Set of metrics for evaluating automatic summarization and machine translation software in natural language processing.
- It compares an automatically produced summary or translation against a set of reference summaries, typically human-generated, using measures such as the overlap in unigrams, bigrams, trigrams, and longest common subsequences.
- Commonly used for summarization tasks

Massive models bechmark

Massive Multitask Language Understanding (MMLU)

- Comprehensive evaluation framework designed to assess the performance of language models across a wide range of subjects and tasks.
- It includes over 50 different tasks covering a diverse set of topics such as science, humanities, social sciences, and professional domains, aimed at testing the depth and breadth of a model's understanding.
- MMLU is known for its challenging nature, requiring models to not only understand the nuances of human language but also to demonstrate knowledge and reasoning abilities across various disciplines.

Big-Bench

- BIG-bench (Beyond the Imitation Game Benchmark) is an extensive benchmark designed to evaluate and push the limits of large-scale language models in areas like reasoning, creativity, and understanding.
- It encompasses a diverse range of tasks, over 200 in total, that cover a wide array of domains including mathematics, common sense reasoning, linguistics, and even ethical judgment.
- BIG-bench is unique in its focus on tasks that are challenging for current models, aiming to identify the limitations of existing AI and guide future research in natural language understanding and generation.

Chatbot Arena

https://huggingface.co/spaces/lmsys/chatbot-arena-leaderboard

Chatbot ELO (2024-01-07)

OpenAI

ANTHROP\C

MISTRAL AI_

Model	⭐ Arena Elo rating	📈 MT-bench (score)	MMLU	License
GPT-4-Turbo	1243	9.32		Proprietary
GPT-4-0314	1192	8.96	86.4	Proprietary
GPT-4-0613	1158	9.18		Proprietary
Claude-1	1149	7.9	77	Proprietary
Claude-2.0	1131	8.06	78.5	Proprietary
Mixtral-8x7b-Instruct-v0.1	1121	8.3	70.6	Apache 2.0
Claude-2.1	1117	8.18		Proprietary
GPT-3.5-Turbo-0613	1117	8.39		Proprietary
Gemini Pro	1111		71.8	Proprietary
Claude-Instant-1	1110	7.85	73.4	Proprietary
Tulu-2-DPO-70B	1110	7.89		AI2 ImpACT Low-risk
Yi-34B-Chat	1110		73.5	Yi License

Prompt Engineering

What is a prompt?

- Prompt text is the key model input for LLMs - it contains all the instructions together with conversation history and any additional context such as RAG
- Despite being dominated by Natural Language prompt can implement coded structures such as html like syntax
- While designing a prompt you should provide as detailed instructions and context as possible
- Prompt does not need to be equal to what we see in the chat window, additional elements can be appended such as:
  - initial model instructions
  - domain information from RAG
  - conversation history
- In context learning allows to help format model outputs by few-show-learning through prompting

https://chat.openai.com/c/507051a7-dc8a-4789-864c-e87fcfb2b1b2

Popular prompting templates

Question + Instruction

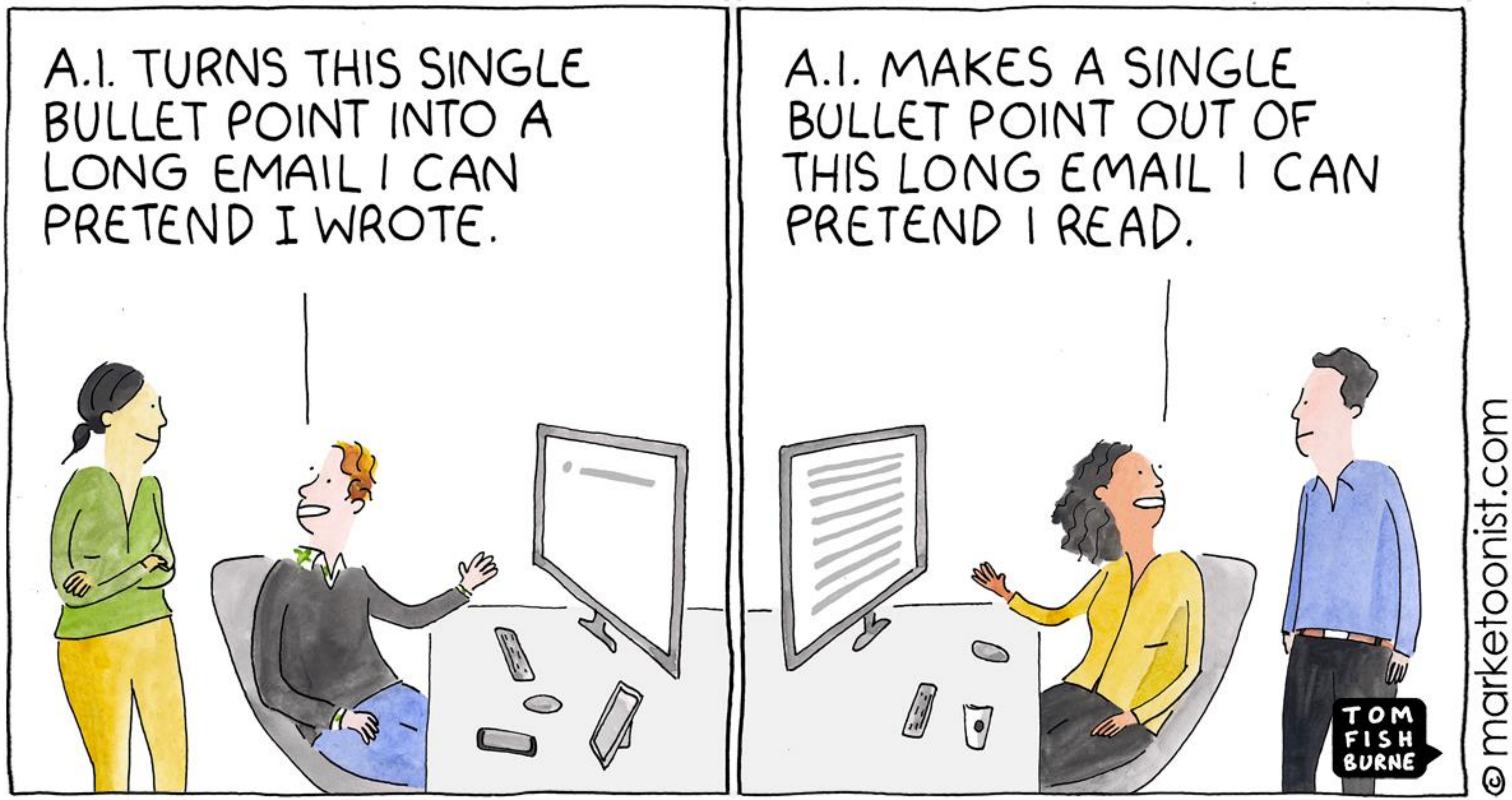
You

What are the 5 most important NLP topics I should include in a short NLP course? Give me an agenda with a brief description of each topic.

Instructions + Input data (zero-shot learning example)

You

Given the following information about me prepare a summary for my Linkedin Profile. My name is Jan, I live in Warsaw, I graduated Power Engineering but for most of my career worked in Data Science. I am passionate about analysing Real Estate data and NLP models.


A.I. TURNS THIS SINGLE BULLET POINT INTO A LONG EMAIL I CAN PRETEND I WROTE.
A.I. MAKES A SINGLE BULLET POINT OUT OF THIS LONG EMAIL I CAN PRETEND I READ.
TOM FISH BURNE
© marketoonist.com

Popular prompting templates

Question + Examples

You

I really liked London, Rome and Prague. I disliked Zurich, Paris and Sosnowiec. What cities would you recommend for my next trip?

What is Prompt engineering

- Prompt engineering aims to create prompts in a more programmatic, reproducible way
- It involves carefully designing and structuring the input given to the model to effectively guide it towards generating the desired output or response.
- Temperature is one of the key hyperparams controlling how varied and creative each models response is - even with the exact same prompt
- Prompt Engineering can be more structured by leveraging libraries such as LangChain

Chain-of-thought reasoning

- Chain-of-thought prompting: Encourages model to be factual and precise by explaining its reasoning

J **You**

What are the best libraries to get started with NER?
Use this format:
Q: <repeat question>
A: Let's analyze the possibilities. <give reasoning>, summarizing the best library for starting with NER is <final answer>

Q: What are the best libraries to get started with Named Entity Recognition (NER)?

A: Let's analyze the possibilities. There are several libraries catering to NER tasks, but the choice often depends on factors like ease of use, community support, and performance. spaCy stands out as an excellent option for beginners. Its user-friendly API, pre-trained models, and efficient processing make it a go-to choice for those new to NER. Therefore, spaCy is recommended as a solid starting point for diving into Named Entity Recognition.

Citing sources

J **You**

What are the top 3 LLM models? Answer only with reliable sources and cite those sources.

ChatGPT

As of my last knowledge update in January 2022, three prominent large language models (LLMs) were widely recognized:

1. **GPT-3 (Generative Pre-trained Transformer 3)**
   - **Source:** Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language Models are Few-Shot Learners. arXiv:2005.14165.
2. **BERT (Bidirectional Encoder Representations from Transformers)**
   - **Source:** Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805.
3. **T5 (Text-to-Text Transfer Transformer)**
   - **Source:** Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2019). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. arXiv:1910.10683.

Warning!
LLM can still make up sources, but at least you can check if they exists

Few shot learning

Zero-shot

You
User
'I rather enjoyed this movie, but the beggining was boring'. Please conduct sentiment analysis on this review.

ChatGPT
Sentiment analysis involves determining the sentiment or emotion expressed in a piece of text. In the review you provided, "I rather enjoyed this movie, but the beginning was boring," the sentiment is mixed. Let's break it down:

1. "I rather enjoyed this movie" - This part of the review expresses a positive sentiment. The word "enjoyed" indicates a positive feeling towards the movie.
2. "but the beginning was boring" - On the other hand, this part of the review expresses a negative sentiment. The word "boring" indicates a negative feeling towards the beginning of the movie.

So, overall, the sentiment in this review is mixed, with a positive sentiment towards the movie as a whole but a negative sentiment towards the beginning of the movie.

Two-shot

You
Classify this review:
"I hated this movie, total waste of time and money"
Sentiment: Negative

Classify this review:
"The most beautiful love story in the history of cinema"
Sentiment: Positive

Classify this review:
'I rather enjoyed this movie, but the beggining was boring'.
Sentiment:

ChatGPT
Sentiment: Mixed (Positive about the overall enjoyment, but negative about the beginning)

Three-shot

You
User
User
Classify this review:
"I hated this movie, total waste of time and money"
Sentiment: Negative

Classify this review:
"The most beautiful love story in the history of cinema"
Sentiment: Positive

Classify this review:
"The beggining dragged on forever, but I liked the suprising ending. Overall experience was average"
Sentiment: Neutral

Classify this review:
'I rather enjoyed this movie, but the beggining was boring'.
Sentiment:

ChatGPT
Sentiment: Mixed or Neutral

Tuning LLMs

Model Fine-tuning

Single task fine tuning

- Retraining all model params on task specific data
- Possible with as little as 1k examples
- Requires significant compute resources and creates a completely separate model for each task
- May lead to catastrophic forgetting, which is basically an equivalent of overfitting

Parameter Efficient Fine Tuning (PEFT)

- Retraining specific part of model params, with keeping majority of model frozen
- Significantly less compute intensive, 90% of params remain frozen and the remaining <10% can be stored for each task and swapped at inference
- Combines general knowledge with new task, reduces risk of catastrophic forgetting

PEFT methods

Reparametrization

- Retrain part of models params using lower dimension
- LoRA is one of most popular use-cases combining model base params, with ones trained for a specific task

Additive

- Add trainable layers or parameters to model
- In “Soft Prompts” prompt tuning additional training happens at input level
- Adapters add additional model layers fine-tuned for specific task, while the backbone models are frozen

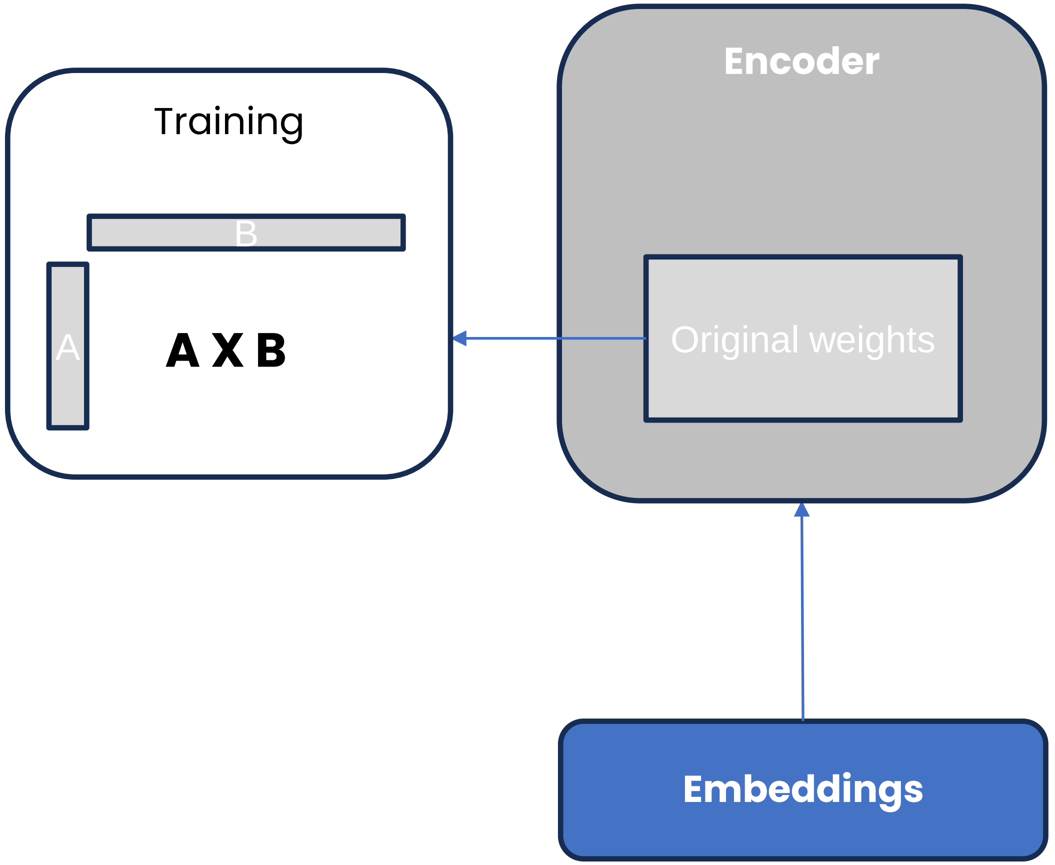
LoRA: Low Rank Adaption of LLMs

Training

- Freeze original model weights
- Replace part of original weight with 2 rank decomposition matrices (with lower dimensionality)
- Train weights only for the smaller matrices

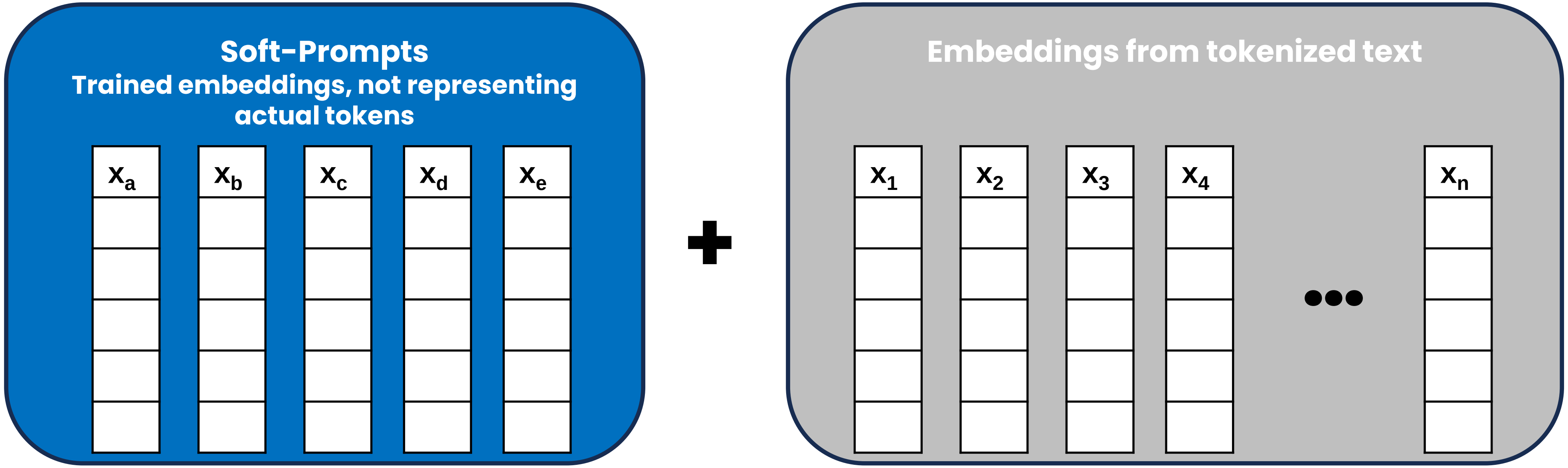
Inference:

- Multiply low rank matrices, to get a matrix with same dimenstions as original weights
- Add product of this multiplication to original weights



Soft prompts fine tuning

- Add additional embeddings, which do not correspond to any token representation
- They will form context embeddings, which help to guide input prompt toward desired outcomes
- Analysing their vector representation in relation to actual words can provide some basic context



Retrieval Augmented Generation (RAG)

What is RAG?

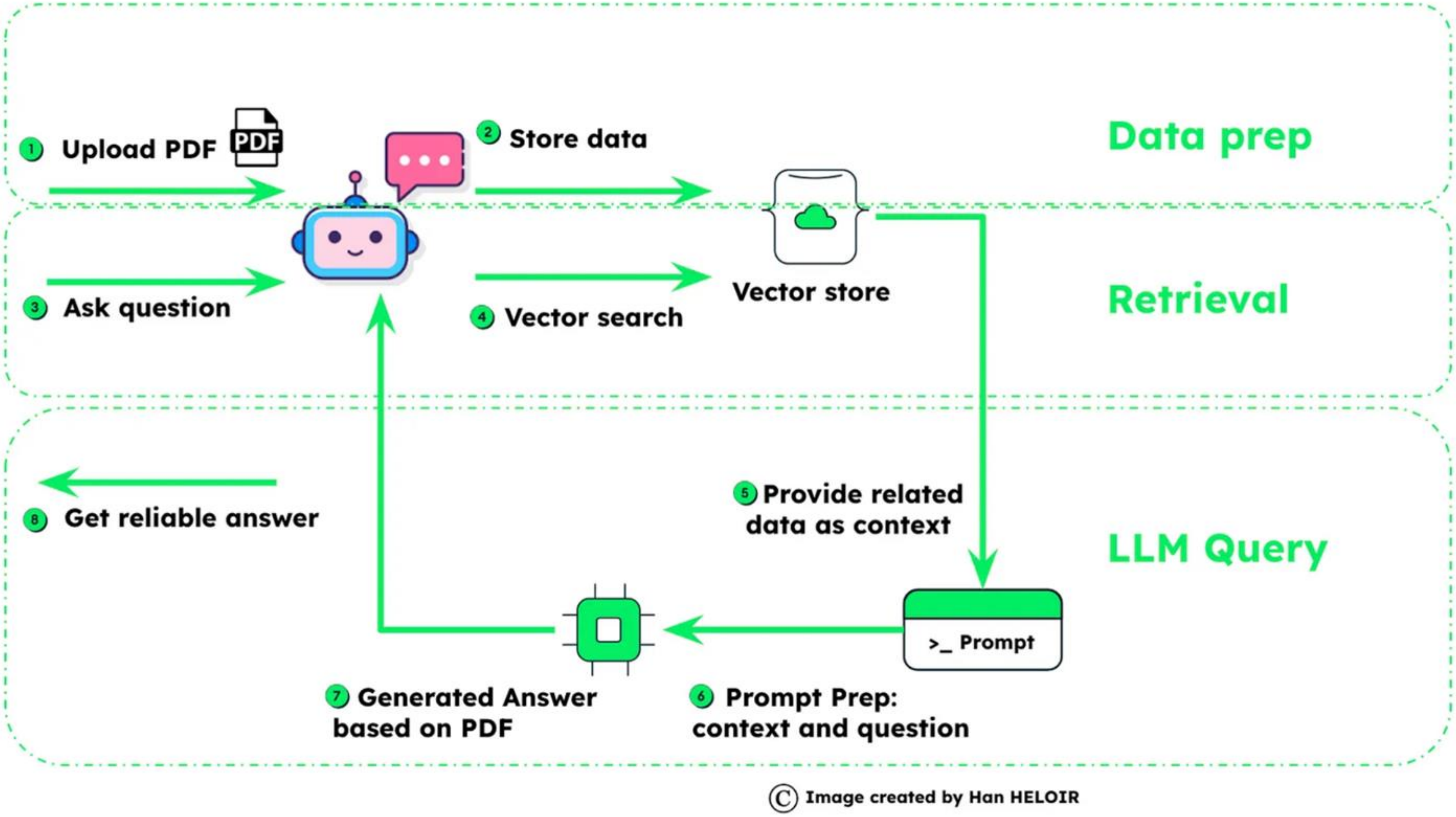
Retrieval-Augmented Generation (RAG) represents a significant advancement in language model technology.

I allows expanding and updating models knowledge with external data, which improves information accuracy and relevance

This approach broadens the use-cases of LLMs, making them more effective in fields like research, fact-checking, and detailed question-answering

Despite quite complex structure at its core the additional information is still added to prompt in text form, which can have negative impact on tokens count, costs and performance

RAG diagram



Source: Han Heloid [Medium]