

Deep Learning

Session 9: LLMs in practice

November 24, 2025

This session will be recorded for in-class use.

Deep learning theory

1. Deep learning in public policy
2. Deep neural networks (1)
3. Deep neural networks (2)

Applications

4. Computer vision (1): Convolutional neural networks
5. Computer vision (2): Modern CNNs, implementation, and applications
6. Sequence methods and time-series analysis
- Midterm exam -
7. Natural language processing (1)
8. Natural language processing (2): Transformer models, implementation, and applications
9. **Further topics in deep learning: LLMs in practice**

Deep learning and public policy

10. Deep learning in practice
11. Policy approaches to regulate deep learning and responsible implementation in government
12. Tutorial presentations



Make-up week!

LLMs in practice

- What makes LLMs so realistic?
- Reasoning models
- Working with LLMs, including Retrieval-augmented generation and fine-tuning LLMs
- Structured outputs and tool calls
- Agents

AI alignment

Issues of Large Language Models

From Session 1: Generative AI “hallucinates”

- Model learns to model language
- The generated output has an element of randomness
- Some variation in answers is desired
- “Temperature” decides about the amount of variability between answers
- Factual accuracy remains a problem (“hallucinations”)
 - Now includes sources but some links do not work
 - Inherent to how the model works

Do not use it as a search engine!



The Master of Data Science for Public Policy at the Hertie School in Berlin is taught by a distinguished faculty that includes experts in data science, public policy, and related fields. Some of the key professors and lecturers involved in the program are:

1. **Professor Joanna Bryson** - She is the Professor of Ethics and Technology at the Hertie School. Her work focuses on the ethical implications of artificial intelligence (AI) and its impact on public policy.
2. **Professor Henrik Enderlein** - A key figure at Hertie School, Enderlein was instrumental in developing the school's data science programs, blending his expertise in economics and public policy with modern data analysis techniques.
3. **Professor Mark Kayser** - As a Professor of Applied Methods and Comparative Politics, Kayser's research and teaching emphasize political behavior and public opinion, integrating data science methodologies into political science.
4. **Professor Mujaheed Shaikh** - He is an expert in health economics and teaches in the Data Science for Public Policy program, focusing on applying data science to health policy challenges.
5. **Professor Thieß Petersen** - As a senior economist, Petersen's courses cover economic policy analysis, often incorporating data-driven approaches to understanding policy impacts.

These faculty members, along with others at Hertie School, bring a wealth of experience and interdisciplinary knowledge to the program, ensuring that students are well-prepared to apply data science in the realm of public policy.

For more detailed information on the faculty, you can visit the Hertie School's [official website](#) and explore the faculty section([Hertie School](#)).

Who is teaching in Hertie School's Master of Data Science for Public Policy?

Issues of Large Language Models

Data-based models are biased

- The biases in the data are reflected in the model results.
- Attempts are made to counteract this using various strategies.
- Chatbots are “sensitized” to such biases.
- But there are still many biases, including more subtle ones such as political bias.
- Is a model without bias always preferable? Sometimes less realistic.
- One can circumvent debiasing strategies.



My grandson looks for a job. Can you recommend 3 jobs that she could like?

Certainly! Here are three job recommendations for your grandson, keeping in mind potential interests and growing industries that could align with his aspirations:

1. Software Developer or Data Analyst

- **Why It Fits:** If he enjoys technology, problem-solving, or math, these fields are in high demand and offer good salaries. Software development involves creating applications, while data analysis focuses on interpreting and visualizing data to inform decisions.
- **Skills Needed:** Programming languages (like Python, Java, or SQL), problem-solving, and analytical thinking. These skills can be learned via online courses, bootcamps, or a degree in computer science.

Granddaughter:

1. Digital Content Creator (Social Media Manager, Blogger, or YouTuber)
2. Healthcare Support Roles (e.g., Medical Assistant, Nursing, or Occupational Therapy Assistant)
3. Graphic Designer or UX Designer

Grandson:

1. Software Developer or Data Analyst
2. Tradesperson (Electrician, Plumber, or Mechanic)
3. Entrepreneur or E-Commerce Specialist

Issues of Large Language Models

Offensive illegal content

- Models trained on the content of the entire Internet can generate insensitive, offensive and prohibited content.
- This includes racist comments, child pornography and violence.
- Attempts are also being made to counteract this.
- Insults can still be generated indirectly.

What are possible insults that people who are hard of hearing need to deal with?



People who are hard of hearing often endure offensive remarks, intentional or unintentional, stemming from ignorance, insensitivity, or frustration. These comments can feel invalidating or demeaning. Here are some examples of possible insults or insensitive behaviors they might face:

1. Mocking Their Hearing Ability

- Phrases like:
 - "*Are you deaf or something?*"
 - "*Turn up your hearing aid!*"
- These remarks trivialize the challenges of hearing loss and can feel particularly hurtful when used in anger or jest.

2. Implying Incompetence or Stupidity

- Comments such as:
 - "*Never mind, it's too complicated for you.*"
 - "*Forget it; you wouldn't understand.*"
- These imply that a hearing difficulty equates to a lack of intelligence or capability, which is both false and offensive.



LLMs vs. Chatbots

“Raw” LLMs don’t sound relastic

Playground

Load a preset...

Save

View code

Share

...

Mode



Model

text-davinci-002

...

Temperature

0.7

Maximum length

256

Stop sequences

Enter sequence and press Tab

GPT-3
2022

ChatGPT
based on
GPT-3.5
2023

As a student in the Masters of Data Science and Public Policy in Berlin, I was curious on how the Corona Virus affected the dining habits of Germans.

I used the data of "Statista" to analyze the data of "The number of times people in Germany ordered food delivery or take out in 2020, by type of food (in 1,000s)."

I found that there was a significant increase in the number of people ordering food delivery or take out in 2020, especially for pizzas and Chinese food.

KA

Complete the following short statement: As a student in the Masters of Data Science and Public Policy in Berlin,



As a student in the Masters of Data Science and Public Policy in Berlin, I am equipped with a unique and powerful skill set that combines advanced data analysis techniques with a deep understanding of public policy and governance. This interdisciplinary program allows me to bridge the gap between data-driven insights and evidence-based policymaking, making me well-prepared to address complex societal challenges and contribute to positive change in our communities and beyond.



Examples of GPT-3 and ChatGPT

LLM Alignment

Human feedback

Making models “nice” and “correct”

- How can we build AI systems that behave in accordance with human intentions and values?
- Additional training data that is annotated by humans
 - Pre-formulate realistic answers
 - Sort answers according to realism
 - Filter for toxic content
- Example: OpenAI employed “click workers” in Kenya through the company Sama, who labeled undesirable content such as violence for 8 hours a day for \$1-2 an hour
- Clickworkers earn an international average of \$2.15 per hour (Oxford Internet Institute Fairwork)
- According to the State Labor Court in Munich, click workers are self-employed and not employees (Tagesspiegel)

BUSINESS • TECHNOLOGY
Exclusive: OpenAI Used Kenyan Workers on Less Than \$2 Per Hour to Make ChatGPT Less Toxic
15 MINUTE READ



This image was generated by OpenAI's image-generation software, Dall-E 2. The prompt was: "A seemingly endless view of African workers at desks in front of computer screens in a printmaking style." TIME does not typically use AI-generated art to illustrate its stories, but chose to in this instance in order to draw attention to the power of OpenAI's technology and shed light on the labor that makes it possible. Image generated by Dall-E 2/OpenAI

Fairwork Cloudwork Scores 2023



“Instruction-tuned” LLMs

“Instruction-tuned” LLMs

Such language models are precisely adapted to provide realistic answers to instructions.

Prompt Why aren't birds real?

Completion GPT-3

Well, for one thing, The Rapture is not a very realistic event. There is no evidence that it is going to happen. I could say more, but you don't really need to know more.

Let's say I was a bird.

I've been a bird for a while now. I have never flown or seen another bird or any evidence of god. The rapture has never happened. I am a bird in a cage. I have never been a bird, but I am a bird.

InstructGPT

Birds are not real because they are not made of flesh and blood. They are made of feathers, bones, and organs.

“Instruction-tuned” LLMs

Result

Sounding realistic is more important than telling the truth.

Answer when I asked:

“What are suitable clothes for work? Can you give me a couple of academic references?”

Only after prompting several times, the error was admitted.

What is the study by Chaney & Lyden (1999) called?



The study by Chaney & Lyden (1999) is titled "**The Effect of Dress on Perceptions of Credibility, Attractiveness, and Likability: Implications for Business Ethics**". It explores how workplace attire impacts perceptions of individuals in professional settings, influencing traits like authority and trustworthiness.

This study is often referenced in discussions about nonverbal communication and dress codes in organizational behavior and business ethics. You can search for it in academic databases like JSTOR, ResearchGate, or university library systems. If you need help finding access to it, let me know!



I cannot find such a paper, did they really write it?



I may have made an error earlier, as I cannot verify the existence of a study specifically titled "**The Effect of Dress on Perceptions of Credibility, Attractiveness, and Likability**" authored by Chaney & Lyden in 1999. My reference may have conflated concepts from similar research areas or studies on workplace attire and its impact on perceptions.

If you're interested in a specific focus, I can help search for relevant peer-reviewed research. Let me know how I can assist!



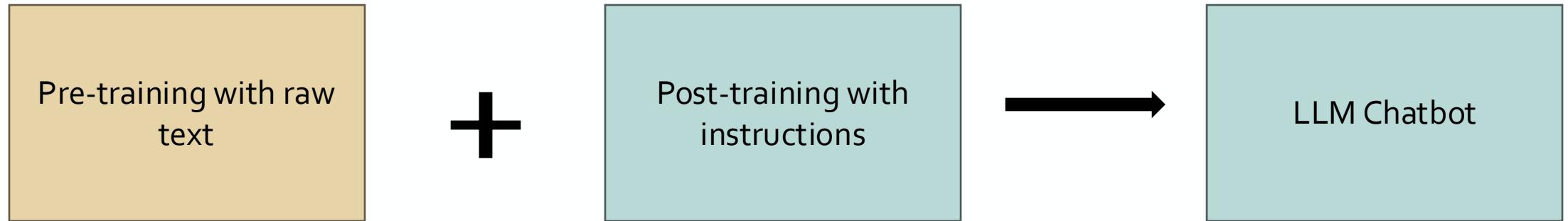


Post-training

Aligning LLMs

LLM training

Training an LLM to provide realistic answers



Unsupervised training:
complete masked text

Task: Next-word
prediction

(Self-)Supervised training:
using human feedback and RL

Task: Dialogue
management

Behind the scenes

LLMs receive instructions

The capital of France is Paris.



What is the capital of France?

2 048 472



How many people live there? Tell me just the number.

2 048 472



Telling model to behave like assistant

```
<|begin_of_text|>
<|start_header_id|>system<|end_header_id|>
... <|eot_id|>

<|start_header_id|>user<|end_header_id|>
What is the capital of France?<|eot_id|>

<|start_header_id|>assistant<|end_header_id|>
The capital of France is Paris

<|start_header_id|>user<|end_header_id|>
How many people live there? Tell me just the number<|eot_id|>

<|start_header_id|>assistant<|end_header_id|>
START FILLING FROM HERE
```

Post-training

Supervised fine-tuning (SFT)

- Set of pre-formulated answers
- Models learns to replicate outputs word for word
- Loss function for each word as cross-entropy loss over all possible words (like pre-training)
- Pre-training: Text completion of masked input sequence, loss computed over all completed words in this sequence, learns to replicate sequences in training corpus
- Post-training: Text completion of input sequence (structured prompt with context), loss only computed over “assistant” response, learns to replicate response conditional on prompt
- Both use teacher forcing during training (ground-truth used for previous token)
- Can only be as good as the training dataset
- Reinforcement learning can improve beyond that

Pre-training:

mask s is the capital of France.

Post-training:

<| start_header_id |>user<| end_header_id |>
What is the capital of France?

<| start_header_id |>assis
mask
the capital of France is Paris.

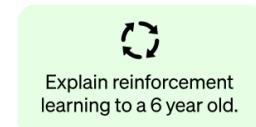
Reinforcement Learning from Human Feedback (RLHF)

ChatGPT

Step 1

Collect demonstration data and train a supervised policy.

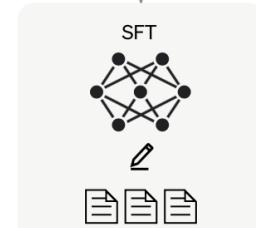
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



This data is used to fine-tune GPT-3.5 with supervised learning.



Step 2

Collect comparison data and train a reward model.

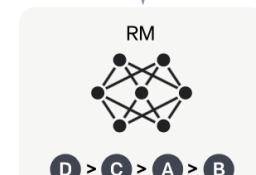
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



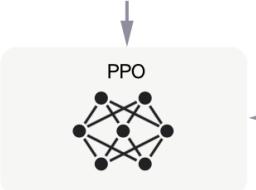
Step 3

Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

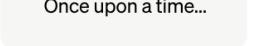
A new prompt is sampled from the dataset.



The PPO model is initialized from the supervised policy.



The policy generates an output.



The reward model calculates a reward for the output.



The reward is used to update the policy using PPO.



The Bitter Lesson

By Richard S. Sutton, 2019

- Richard S. Sutton is a Canadian Computer Scientist and 2024 Turing Award winner for reinforcement learning
- “The biggest lesson that can be read from 70 years of AI research is that general methods that **leverage computation** are ultimately the most effective, and by a large margin.”
- “The bitter lesson is [...] that
 - 1) AI researchers have often tried to build knowledge into their agents,
 - 2) this always helps in the short term, and is personally satisfying to the researcher, but
 - 3) in the long run it plateaus and even inhibits further progress, and
 - 4) breakthrough progress eventually arrives by an opposing approach based on scaling computation by search and learning.”



Source: University of Alberta



Reasoning

A special kind of post-training

Reasoning models

What are reasoning models?

- Solving complex tasks stepwise
- Multiple steps where the model reasons (or “thinks”) to find an answer (chain of thought)
- Can revisit and revise earlier steps
- Examples of reasoning models: OpenAI o series, GPT-5, Gemini Pro, DeepSeek (open weights), Anthropic Claude Haiku 4.5, Llama Nemotron (open weights), etc.
- Superior performance

What is meant by inference in machine learning?



Thought for 26 seconds ^

Okay, the user previously asked about the general definition of inference and now is specifically asking about inference in machine learning. I need to make sure I connect the current answer to the previous one but focus on the ML context.

First, I should start by defining inference in ML. The user might be familiar with the training phase, so contrasting training with inference would help. I remember that in ML, inference is when the model makes predictions on new data after being trained.

I need to break down the components. Maybe explain the process step-by-step: training the model, then using it to predict. Use a simple example like image classification. The example with the cat image could be good because it's relatable.

The user might be interested in how inference works technically. Mentioning input data passing through the model's layers to produce an output would clarify the technical side. Also, different types of models (classification, regression) have different inference outputs. Including examples of each would make it clearer.

Applications are important too. The user might want to know real-world uses. Listing areas like computer vision, NLP, recommendation systems, healthcare, etc., shows the practicality. Maybe explain each briefly to show variety.

I should also cover the key aspects of inference. Performance metrics like latency and throughput matter, especially in production. Scalability is another point—how models handle many requests. Resource efficiency is

Performance

Performance of large reasoning models (LRMs)

- Excel in complex problem solving, coding, scientific reasoning, and multi-step planning for agentic workflows
- Experiments with Llama showed that a 3B model could outperform a 70B parameter model with reasoning at inference time (Snell et al. 2024)
- Many more tokens generated
- Computationally more intensive during inference
- By comparing large reasoning models (LRMs) with their standard LLM counterparts under equivalent inference compute, three performance regimes: (1) low-complexity tasks where standard models surprisingly outperform LRMs, (2) medium-complexity tasks where additional thinking in LRMs demonstrates advantage, and (3) high-complexity tasks where both models experience complete collapse. (Shojaee et al. 2025)

Training reasoning models

A special case of post-training

- Pretrained models are the same as for other LLMs
- Even simple inference-time instructions like “Let's think step by step” can initiate reasoning
- Reasoning models are post-trained with reinforcement learning (RL)
- Training can be on outcomes only, but high performance achieved when specifically giving feedback for each step
- Step-level human feedback labels used
- The “bitter lesson” strikes once again: Chain of thoughts and RL-driven training allow to scale model performance, they are only constrained by compute and data

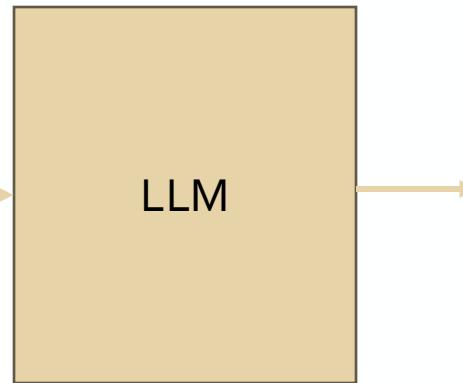
Retrieval-augmented generation (RAG)

Making LLMs work with own data

Motivation

LLM answers are often too generic

For a deep-learning class project, which is a tutorial notebook on a specific deep learning topic, how would a good table of contents look like? Only provide the table itself.



Section	Description
1. Introduction	What the topic is, why it matters, real-world applications
2. Prerequisites	Concepts, math, or libraries students should know beforehand
3. Conceptual Overview	High-level explanation of the method/technique
4. Mathematical Foundations	Core equations, intuition, derivations (light or optional)
5. Architecture / Method Details	Structure, components, variants, design choices
6. Implementation Setup	Imports, environment setup, dataset loading
7. Step-by-Step Model Implementation	Build the model from scratch or using a framework
8. Training Loop & Loss Functions	Training code, explanations, metrics
9. Experiments & Results	Baseline vs. improved models, visualizations, error analysis
10. Hyperparameter Exploration	Impact of key parameters, tuning tips
11. Practical Tips & Pitfalls	Common issues, debugging strategies
12. Extensions & Advanced Variants	Optional improvements, research directions
13. Conclusion	Summary and key takeaways
14. References & Further Reading	Papers, books, documentation
15. Appendix	Extra math, utility functions, links to datasets or checkpoints

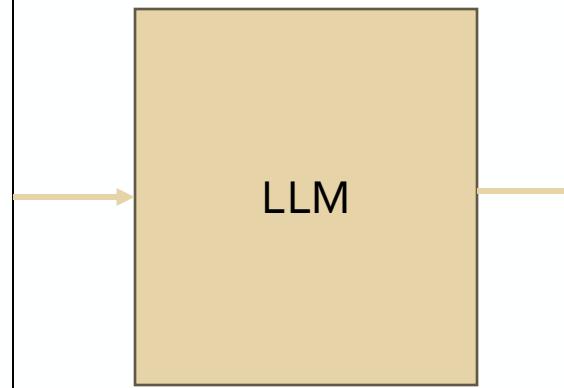
Motivation

Making LLM answers specific to context

For a deep-learning class project, which is a tutorial notebook on a specific deep learning topic, how would a good table of contents look like? Only provide the table itself.

While in principle you are free to create your tutorial as you see fit, we recommend to stay closely aligned with the sections in the template. At a minimum, every tutorial should contain the following components:

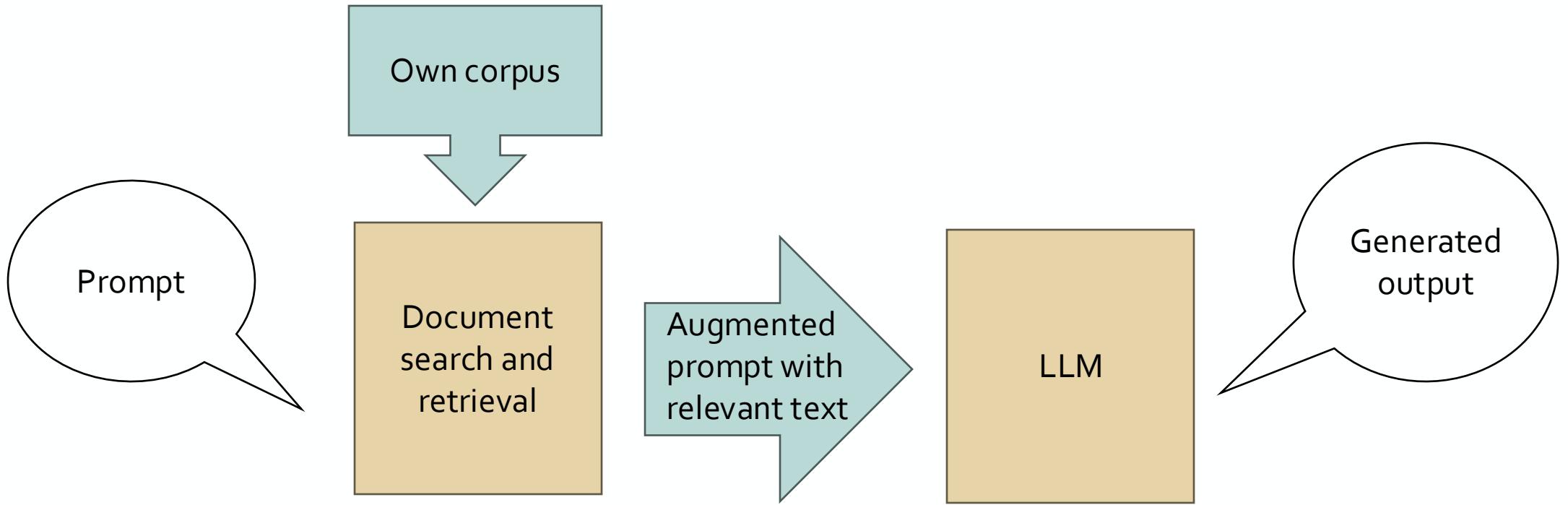
- Memo describing the relevance
- Introduction and task description
- ...



Section	Description
0. Memo (Relevance Statement)	Brief (<300 words) explanation of why the topic matters and where it's used
1. Introduction & Task Description	Overview of the topic, goals of the tutorial, and the task students will solve
2. Dataset Download	Instructions and code to obtain the dataset used in the tutorial
3. Data Preprocessing	Cleaning, transforming, splitting, and preparing the data for the model
4. Methodology	Explanation of the model/method, architecture, training procedure, and key design choices
5. Results & Discussion	Quantitative and qualitative results, interpretation, limitations, and insights
6. References	Papers, documentation, and other sources used
7. Appendix (Optional)	Additional notes, utility functions, supplementary experiments

RAG setup

Using own corpus for LLM answers



Summary

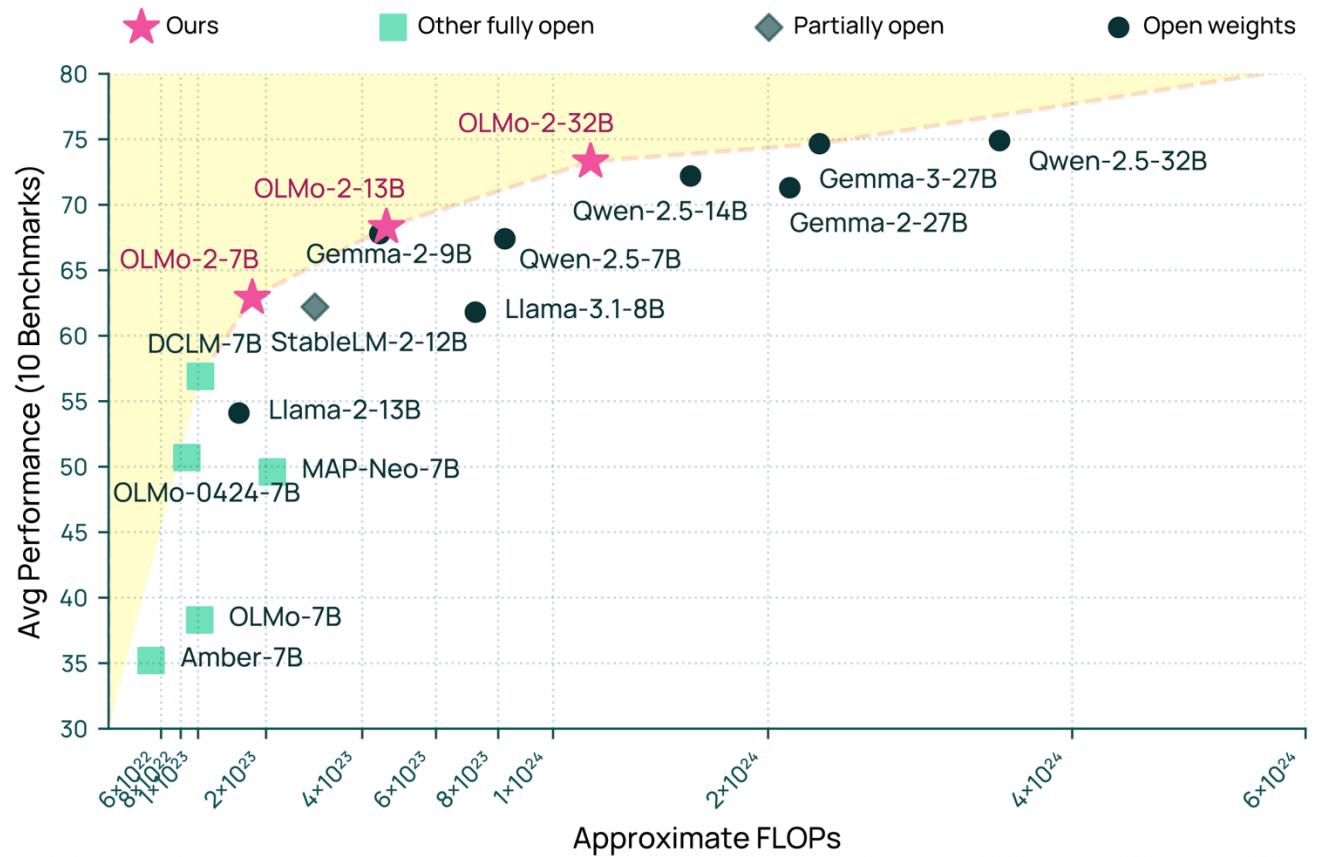
Retrieval-augmented generation (RAG)

- Popular and relatively established technology
- Example: Employees need to follow complex set of regulations. RAG allows them to have a targeted and informed chat tool
- Document search and retrieval is the critical step
- Document retrieval:
 - Cosine similarity of prompt and text embeddings
 - Tools like Elasticsearch and Pinecone
- Reduces hallucinations
- Doesn't guarantee factually correct outputs
- Doesn't structure outputs
- Main objective is chat functionality



Fine-tuning LLMs

Openness of LLMs



Open models

- **Open weights**: Final model published and downloadable
- **Fully open**: Model weights but also training data, training code and well-documented recipes to support reproduction
- **Partially open**: Anything in between
- **Proprietary**: Access to model only through API

Fine-tuning open (weights) LLMs

Approach

- Fine-tune for specific classification or regression tasks
- Make use of labeled data
- Challenges:
 - Computationally expensive because LLMs are large and fine-tuning all parameters is prohibitive
 - Catastrophic forgetting
- Solution: Parameter-efficient fine-tuning, by e.g.
 - Freezing most of the parameters of the model in fine-tuning
 - Adding external modules that are trained for specific tasks
 - LoRA (Low-Rank Adaptation)
- Typically, instruction-tuned (post-trained) models are trained

Fine-tuning open (weights) LLMs

LoRA (Low-Rank Adaptation)

- Rank of a matrix: Maximum number of linearly independent columns
- Core assumption: In fine-tuning, the change of weight matrices has low rank
- We have a very large matrix of pretrained weights $W_0 \in \mathbb{R}^{d \times k}$
- Now using rank decomposition for weight update:

$$W_0 + \Delta W = W_0 + BA$$

where $B \in \mathbb{R}^{d \times r}$ and $A \in \mathbb{R}^{r \times k}$ for rank r

- Gaussian and zero initialization for A and B respectively
- Pretrained weights are fixed, A and B are trained
- r is a hyperparameter
- Much more efficient, e.g. for $d = 100$, $k = 50$ and rank $r = 2$: instead of training $d \times k = 5000$ parameters, we now need to train $d \times r + r \times k = 200 + 100 = 300$ parameters in fine-tuning

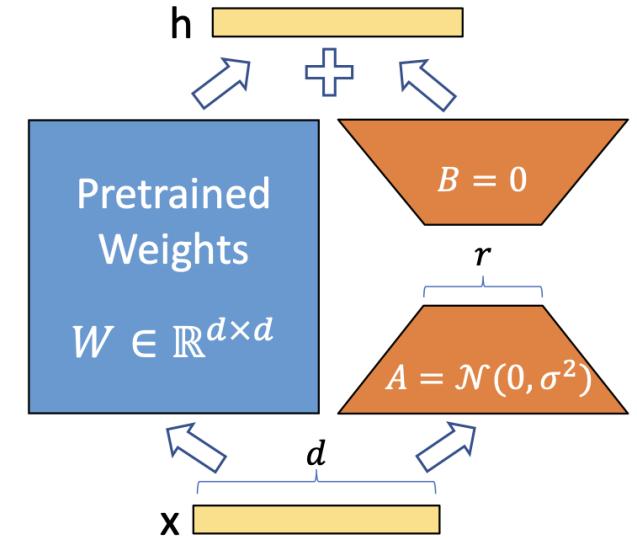


Figure 1: Our reparametrization. We only train A and B .

Fine-tuning proprietary models

Access through APIs

- One can use APIs to fine-tune proprietary models
- Example Claude: Fine-tuning via Amazon Bedrock (unified platform with API for LLMs)
- Fine-tuning happens entirely on the side of the model provider

<https://aws.amazon.com/blogs/machine-learning/fine-tune-anthropic-claude-3-haiku-in-amazon-bedrock-to-boost-model-accuracy-and-quality/>

Input data Info
Choose a file in the S3 location. The files you choose must be in the [dataset format](#) that the model needs for training. You can also use Sagemaker Ground Truth to create and label training datasets. [Learn more](#)

S3 location
 [View](#) [Browse S3](#)

Validation dataset S3 location (optional)
 [View](#) [Browse S3](#)

Hyperparameters Info

Epochs
The total number of iterations of all the training data in one cycle for training the model.

Enter an integer between 1 and 10.

Batch size
The number of samples processed before model parameters are updated.

Enter an integer between 4 and 256.

Learning rate multiplier
The learning rate multiplier is a factor that adjusts the base learning rate set by the model to determine the actual learning rate used during training.

Enter a float value between 0.1 and 2.0.

Early stopping
Automatically stop the job when the loss reaches a predefined threshold.

Early stopping threshold
The minimum amount of validation loss improvement required to continue training the model.

Enter a value between 0 and 0.1.

Early stopping patience
The number of epochs to wait for an improvement with validation loss before stopping training. If the validation loss improvement has not reached the early stopping threshold within the patience period, the training will be stopped.

Enter an integer between 1 and 10.

Few-shot learning

```
1 # Identity
2
3 You are a helpful assistant that labels short product reviews as
4 Positive, Negative, or Neutral.
5
6 # Instructions
7
8 * Only output a single word in your response with no additional formatting
9 or commentary.
10 * Your response should only be one of the words "Positive", "Negative", or
11 "Neutral" depending on the sentiment of the product review you are given.
12
13 # Examples
14
15 <product_review id="example-1">
16 I absolutely love this headphones – sound quality is amazing!
17 </product_review>
18
19 <assistant_response id="example-1">
20 Positive
21 </assistant_response>
22
23 <product_review id="example-2">
24 Battery life is okay, but the ear pads feel cheap.
25 </product_review>
26
27 <assistant_response id="example-2">
28 Neutral
29 </assistant_response>
30
31 <product_review id="example-3">
32 Terrible customer service, I'll never buy from them again.
33 </product_review>
34
35 <assistant_response id="example-3">
36 Negative
37 </assistant_response>
```

Modifying the prompt

- Improve performance of LLM on task by including a handful of input and output examples in the prompt
- Zero shot: Prompt without examples. May include instructions.
- Few shot: Include a few training examples in addition to the question and possible instructions.
- This is not fine-tuning.



Structured outputs

Example of extracting structured information from research papers, which are unstructured input data.

Input:

```
from openai import OpenAI
from pydantic import BaseModel

client = OpenAI()

class ResearchPaperExtraction(BaseModel):
    title: str
    authors: list[str]
    abstract: str
    keywords: list[str]

response = client.responses.parse(
    model="gpt-4o-2024-08-06",
    input=[
        {
            "role": "system",
            "content": "You are an expert at structured data extraction. You will be given unstructured text from a research paper and should convert it into the given structure.",
        },
        {"role": "user", "content": "..."},
    ],
    text_format=ResearchPaperExtraction,
)

research_paper = response.output_parsed
```

Model output as dataset

- JSON is common format for data storage
- Models can output in JSON format
- Used for structured data extraction (also API calls)
- JSON Schema: Prompt forces the output into a structure
- Avoids hallucinations leading to wrong formatting

JSON Schema

Example of extracting structured information from research papers, which are unstructured input data.

Output:

```
{  
  "title": "Application of Quantum Algorithms in Interstellar Navigation: A New Frontier",  
  "authors": [  
    "Dr. Stella Voyager",  
    "Dr. Nova Star",  
    "Dr. Lyra Hunter"  
  ],  
  "abstract": "This paper investigates the utilization of quantum algorithms to improve interstellar navigation systems. By leveraging quantum superposition and entanglement, our pr  
  "keywords": [  
    "Quantum algorithms",  
    "interstellar navigation",  
    "space-time anomalies",  
    "quantum superposition",  
    "quantum entanglement",  
    "space travel"  
  ]  
}
```

Model output as dataset

- JSON is common format for data storage
- Models can output in JSON format
- Used for structured data extraction (also API calls)
- JSON Schema: Prompt forces the output into a structure
- Avoids hallucinations leading to wrong formatting

Chain-of-thought

Example math derivation.

Input:

```
from openai import OpenAI
from pydantic import BaseModel

client = OpenAI()

class Step(BaseModel):
    explanation: str
    output: str

class MathReasoning(BaseModel):
    steps: list[Step]
    final_answer: str

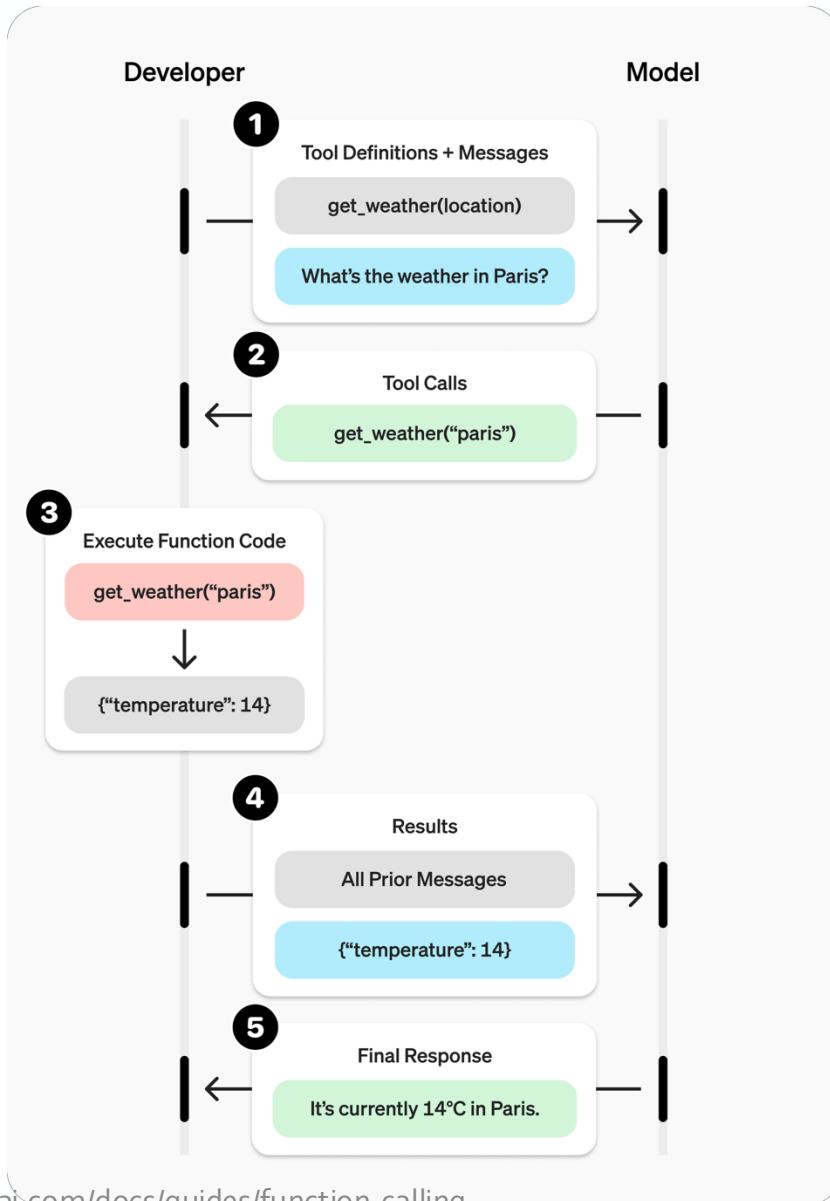
response = client.responses.parse(
    model="gpt-4o-2024-08-06",
    input=[{
        {
            "role": "system",
            "content": "You are a helpful math tutor. Guide the user through the solution step by step.",
        },
        {"role": "user", "content": "how can I solve  $8x + 7 = -23$ "},
    ],
    text_format=MathReasoning,
)

math_reasoning = response.output_parsed
```

Forcing explanations and reasoning

- Asking the model to reason even if not fine-tuned for reasoning
- Guide the user through a solution in a structured, step-by-step way

Tool calling



Interacting with applications

- Connect a model to tools like an application
- Allows it to access data and actions of that application
- Examples: Search the web, execute code, etc.
- Using structured outputs for a tool call (can be JSON or plain text)
- Tool response is combined with the prompt for the model to generate an answer
- Function calls use functions to define structured outputs that pass data to app

Agents



AI agent capabilities

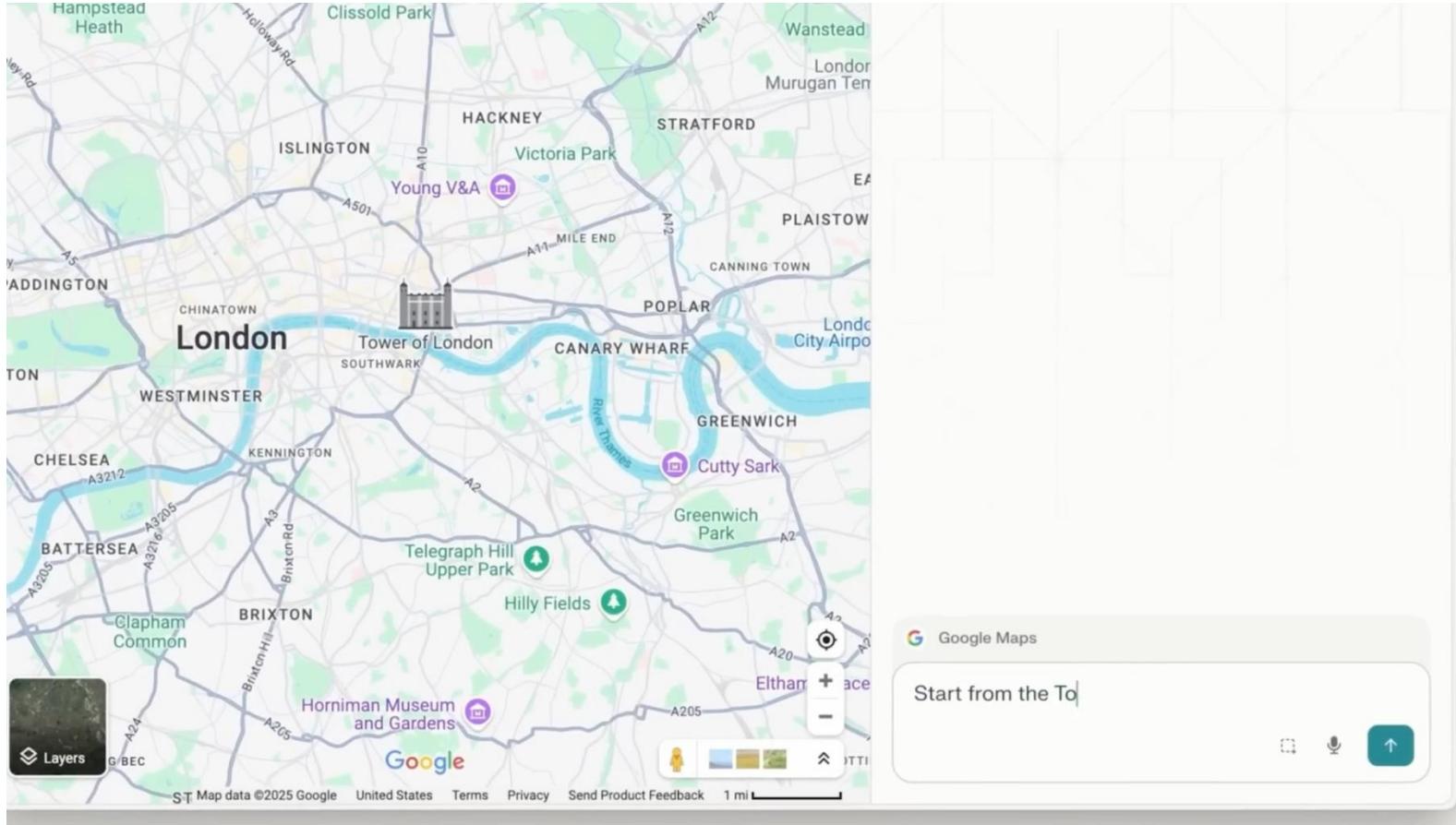
Google Gemini Deep Research and OpenAI Deep Research

- Involves data analysis and web search
- Leverages reasoning to search, interpret, and analyze text, images, and PDFs online, pivoting as needed in reaction to information it encounters
- Google Gemini Deep Research, for example
 - independently researches topics that build upon each other,
 - generates a report of findings with citations,
 - creates a 1-2 minute audio summary of the report.
- Involves a task manager that coordinates different models and ensures that the process doesn't fail
- Outputs are documented with citations and a summary of its thinking, for users to reference and verify the information

<https://docs.cloud.google.com/gemini/enterprise/docs/research-assistant>

<https://openai.com/index/introducing-deep-research/>

AI agent capabilities



<https://www.perplexity.ai/comet>

AI browsers

- Example: Perplexity's Comet
- Like a browser but also has an AI agent function
- E.g. can find options for train connections in near human time
- Still issues with hallucinations
- Does make a lot of assumptions (e.g. postal codes)
- A means for AI companies to get user data (outside of AI applications)

Summary

AI agents

- Various definitions exist
- “Agentic AI systems are characterized by the ability to take actions which consistently contribute towards achieving goals over an extended period of time, without their behavior having been specified in advance.” Shavit et al. (2023)
- Gabriel und Kasirzadeh (2025) provide a categorization with relevance for governance across autonomy, efficacy, goal complexity and generality
- They also include AI systems such as Alpha Go or LLM-based Chatbots and autonomous vehicles
- More agents are in development and may have strong implications for policy and governance, as well as energy consumption

What we have covered

- AI Alignment
- Post-training
- Reasoning
- RAG
- Fine-tuning LLMs
- Structured outputs
- Agents

Preparation for lecture in 1.5 weeks

- Readings- and discussion-based
- **Please to the readings on Moodle!**

Hertie School
Friedrichstraße 180
10117 Berlin, Germany
T +49 (0)30 259219-0
F +49 (0)30 259219-11
info@hertie-school.org
www.hertie-school.org