HSLU Lucerne University of Applied Sciences and Arts

Large Language Models

NLP
Andreas Marfurt

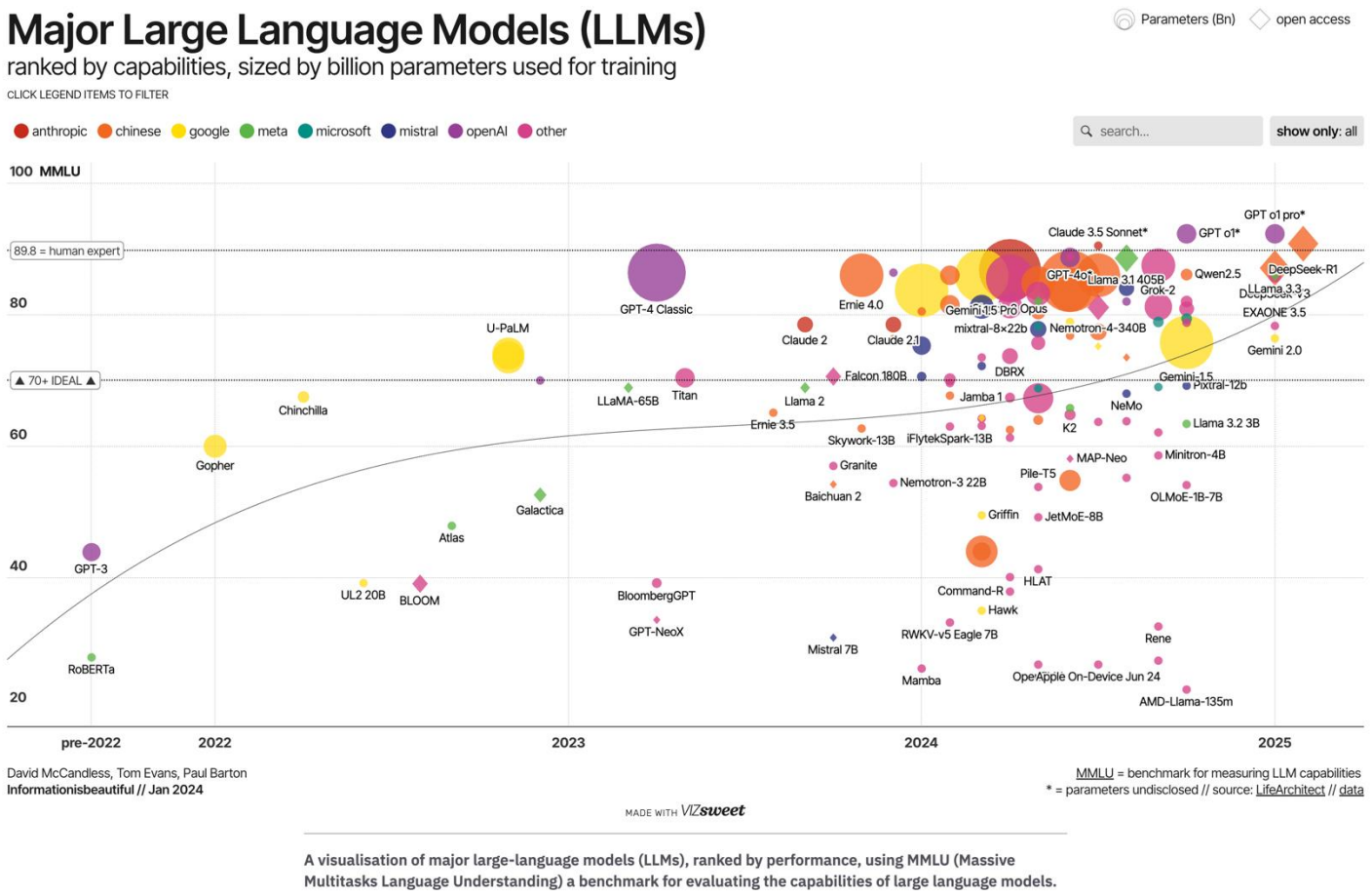
Information Technology
25.04.2025

FH Zentralschweiz

Overview

- LLM overview
  - Closed models/APIs
  - (Partially) open models
- Training/Fine-tuning LLMs
- LLM routing
- LLMs as a judge
- Comparing LLMs
- Running LLMs locally

Large Language Models (LLMs)



A visualisation of major large-language models (LLMs), ranked by performance, using MMLU (Massive Multitasks Language Understanding) a benchmark for evaluating the capabilities of large language models.

https://informationisbeautiful.net/visualizations/the-rise-of-generative-ai-large-language-models-llms-like-chatgpt/

Number of Parameters

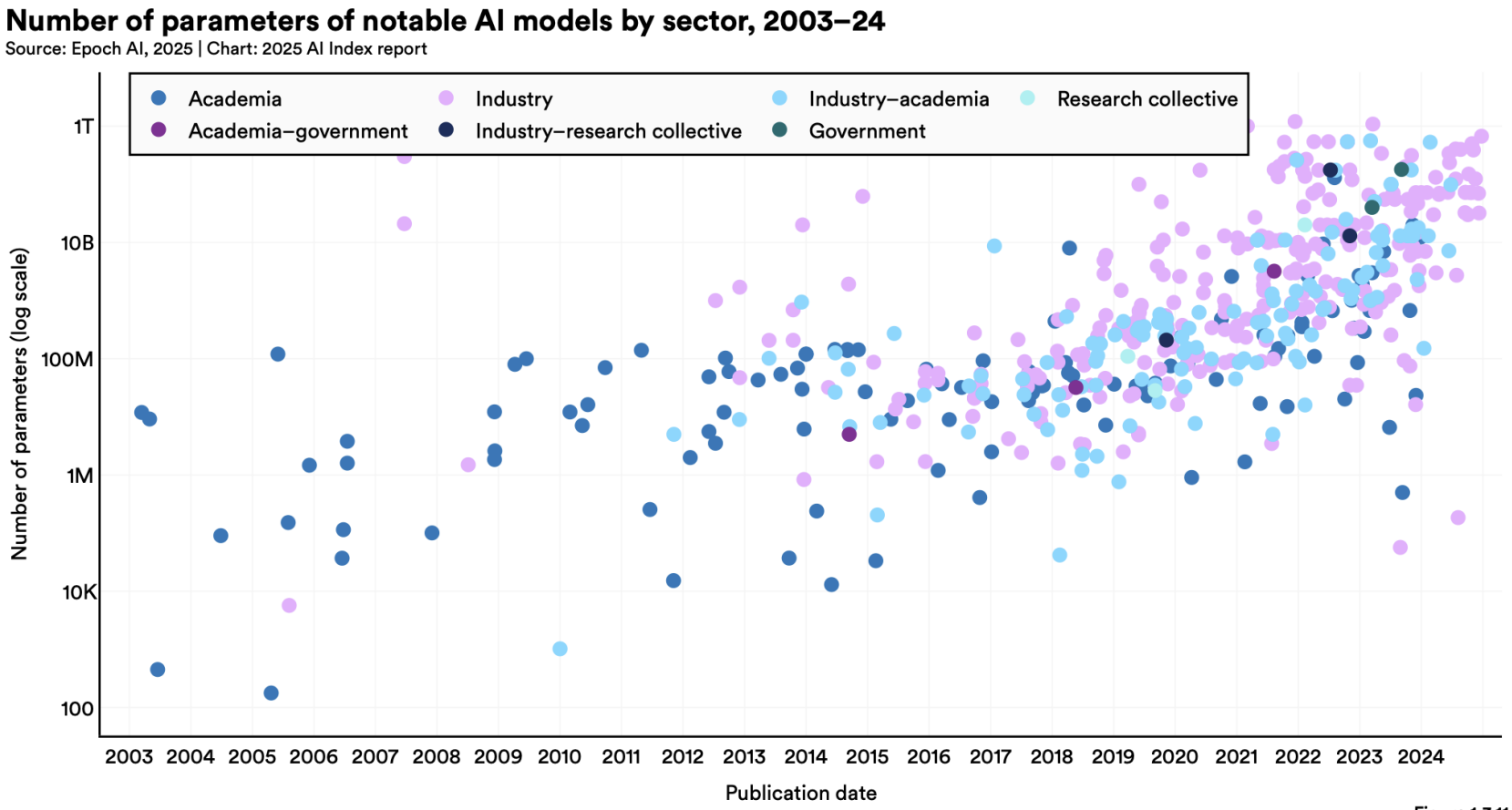


Figure 1.3.11

[Stanford AI Index Report 2025]

Training Compute

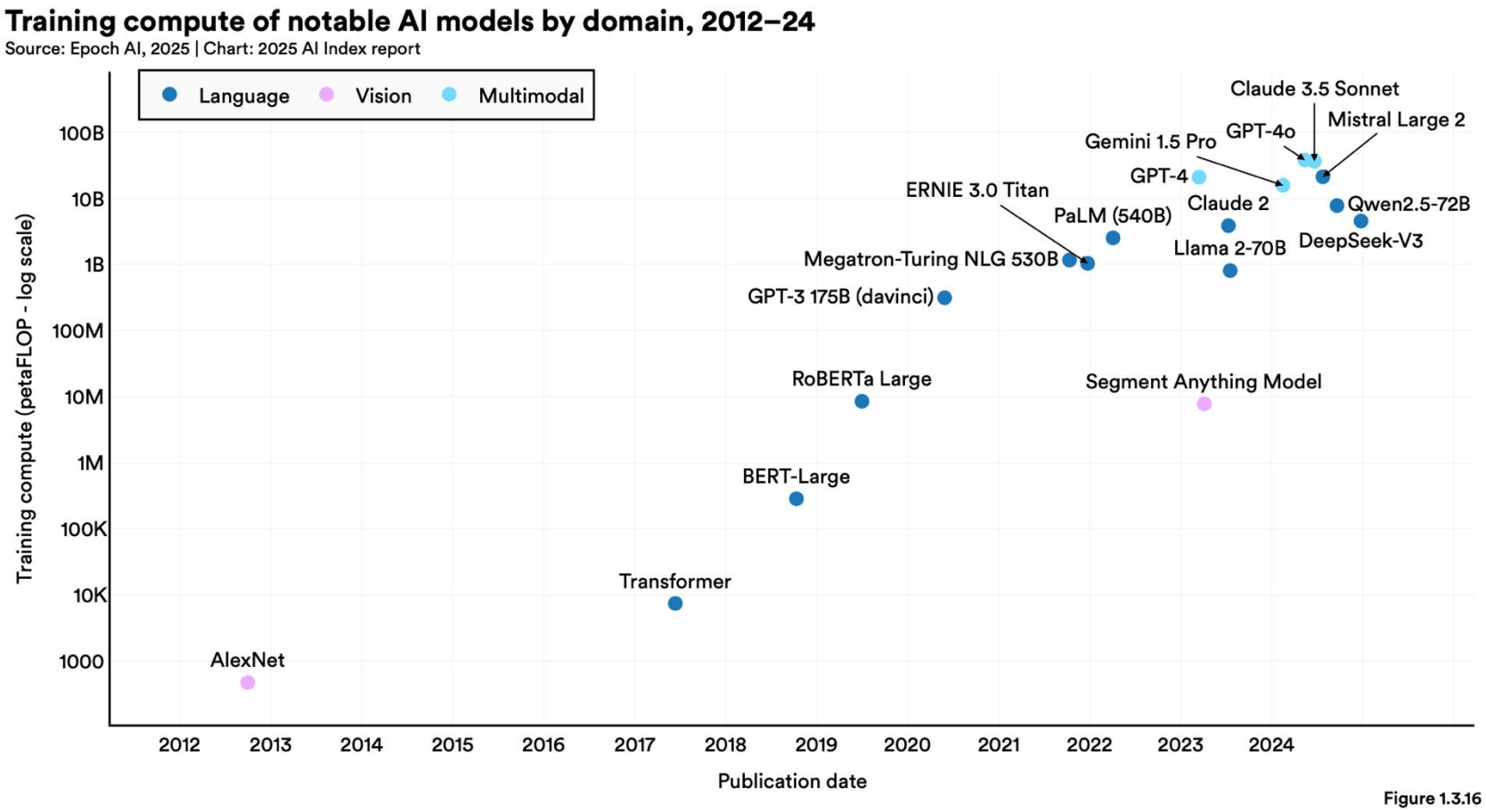


Figure 1.3.16

Training Cost

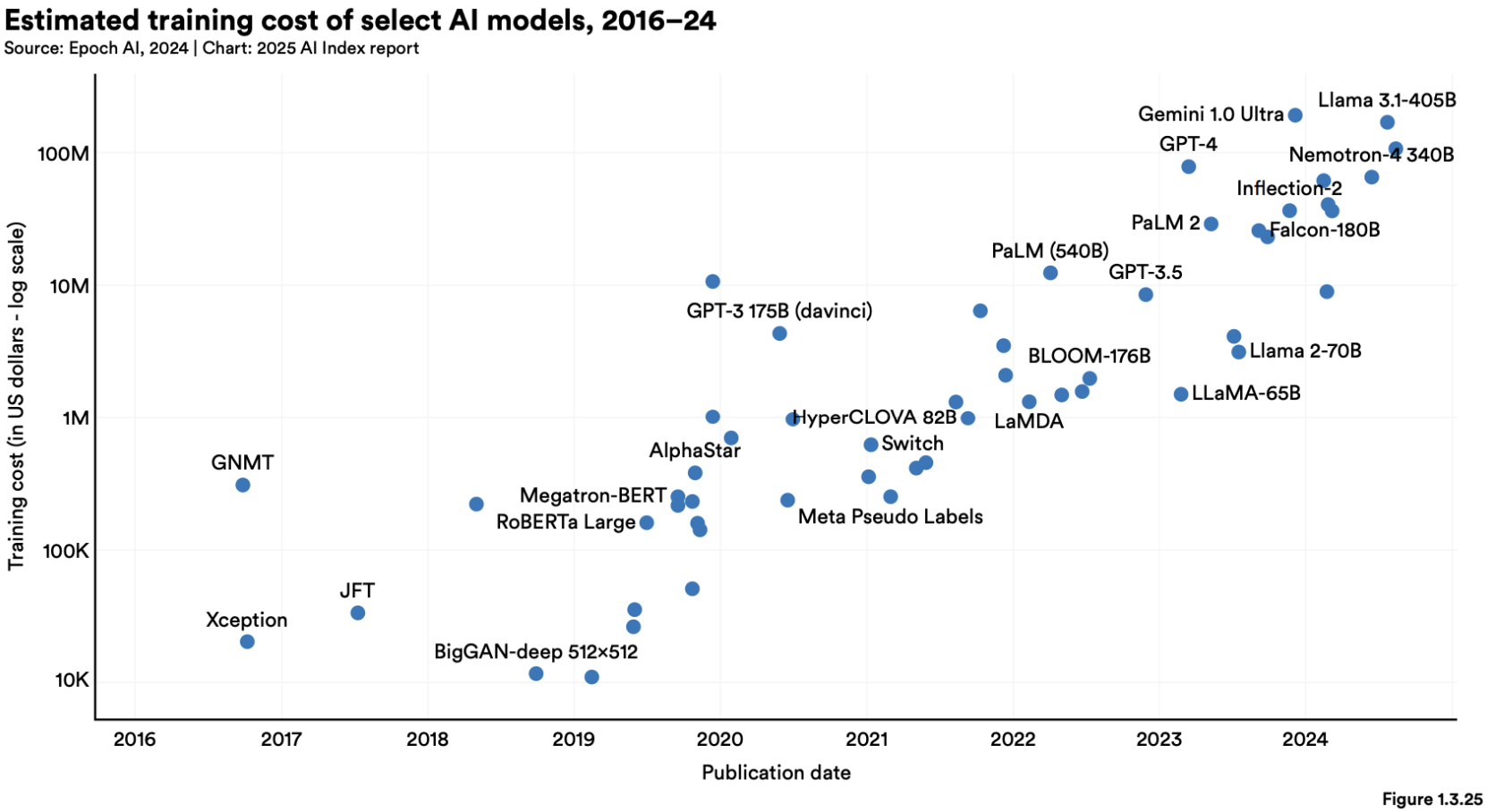


Figure 1.3.25

Differences Between LLMs

- Access: Closed vs. open weights vs. open source
- License: Commercially usable?
- Language: English-focused or multilingual?
- Input data type: Text-only, code, …?
- Input modality: Text, images, audio, EEG data, …
- Output modality: Text, images, audio, video, …
- Training: Instruction tuning, conversations, RLHF, …
- Number of parameters: Can you run it on your GPU(s)?

LLM Development 2021 – 2023

- LaMDA: 137B parameter dialogue model, Google, May 2021
- Megatron-Turing NLG: 530B parameter language model, Microsoft and Nvidia, October 2021
- PaLM: 540B parameter language model, Google, April 2022
- Cohere xlarge: 52B parameter language model, Cohere.ai, May 2022
- OPT: 175B parameter language model, Facebook/Meta, May 2022
- BLOOM: 176B parameter language model, Hugging Face, July 2022
- DALL·E 2: Combination of language and image models to generate images from captions, uses a 1.2B text encoder from GLIDE (a similar model), OpenAI, July 2022
- Galactica: 120B parameter language model for science, Facebook/Meta, November 2022
- ChatGPT: unknown size, assistant, OpenAI, November 2022
- LLaMA: 65B parameter language model, Facebook/Meta, February 2023
- GPT-4: unknown size (potential leak), assistant, OpenAI, March 2023
- Claude: presumably 52B parameters, assistant integrated in Quora and Slack, Anthropic, March 2023
- Since then: Pythia, Dolly, OpenAssistant, WizardLM, MPT, LIMA, PaLM 2, Guanaco, RWKV-4-Raven, Claude 2, LLaMA 2, Falcon. See one of many links.

Closed-weights LLMs/APIs

- OpenAI: GPT-4o/GPT-4.1/o1/o3/o4 (-mini)
- Google: Gemini 2.5 Flash Preview/Pro Preview
- Anthropic: Claude 3.5 Haiku/3.7 Sonnet
- Cohere: Command R/R+/A
- Perplexity AI (search with more accurate sources)
- You.com (complicated search queries that require multiple steps)

- Many more!

Open-source vs. open-weights

- Currently a discussion what open-source means for AI models
- Open Source Initiative (OSI) says Llama is not open-source (1, 2)
  - Llama license prevents companies with >700 million monthly active users to use it
  - No information about training data provided

Open-source LLMs

GPT-J, GPT-NeoX, Pythia

OLMo, OLMoE

Hugging Face

FineWeb, SmolLM

BigCode

StarCoder

together.ai

RedPajama

MAP

MAP-Neo, OpenCoder

DataComp - LM

DCLM-BASELINE

LLM360

K2

Screenshot from Percy Liang's keynote at EMNLP 2024

Open-weights LLMs on Hugging Face (until 2023)

- GPT-NeoX-20B: LM from Eleuther AI with a detailed training description
- OPT: LM by Facebook/Meta, metaseq GitHub, 66B model on Hugging Face
  - Includes a training log on their GitHub
- BLOOM: LM by Hugging Face's BigScience initiative
  - BLOOMZ: Assistant from BLOOM
- Flan-T5: up to 11B LM finetuned by Google on multiple tasks
- UL2: 20B parameter language model by Google that combines denoising with encoder-decoder vs. decoder-only pretraining objectives, blog
  - Flan-UL2: UL2 model with instruction tuning
- LLaMA and LLaMA 2: Open-source language models by Facebook/Meta with GPT-3's performance, 4 model sizes: 7B, 13B, 33B, 65/70B; code, must apply for weights, can be converted to Hugging Face transformers library or directly downloaded from HF if you create a HF account with the same email address
- Falcon: Open-source LM by the Technology Innovation Institute in the UAE, 4 model sizes: 1.3B, 7.5B, 40B, 180B; models on Hugging Face, chat/instruction tuned variants

Open-weights LLMs (2024)

- Meta: Llama 3 and Llama 4
  - Llama-3.1 (8B/70B/405B) trained on 15T tokens
  - Llama-3.2 (1B/3B for text, 11B/90B for vision and text)
  - Llama-3.3 (70B) stronger version of Llama-3.1 70B
- Google: Gemma 3
  - Google on Hugging Face has versions for vision, code, moderation
- DeepSeek: V3 (general purpose) and R1 (reasoning)
- Nvidia: Various models with different architectures/tasks
- Mistral: Mistral-Small-3.1 (24B), Pixtral (12B vision-language model (VLM))

Open-source LLMs (2024)

- Eleuther AI: Pythia, GPT-J 6B, GPT-NeoX 20B
  - Intermediate Pythia model checkpoints released during pretraining
- Allen Institute for AI: OLMo 2
- ML Foundations: DataComp-LM (DCLM)
- Hugging Face
  - SmolLM and SmolLM2 (135M, 360M, 1.7B)
  - StarCoder and StarCoder2 (3/7/15B, LLM specialized for code)

Pre-Training LLMs

- Code repositories: Megatron-LM (Nvidia), nanotron (Hugging Face)
- Pretraining recipes: OLMo, OLMo 2, DCLM, DeepSeek
- Datasets: C4, The Pile, RefinedWeb, FineWeb (15T tokens, 44 TB), CommonCrawl (2.74 billion web pages in March 2025 dump)
- Logs of actual training runs:
  - BLOOM 176B chronicles (almost 4 months of pretraining)
  - OPT 175B Logbook (100 pages of PDF)
  - Collection of logbooks

Fine-Tuning LLMs

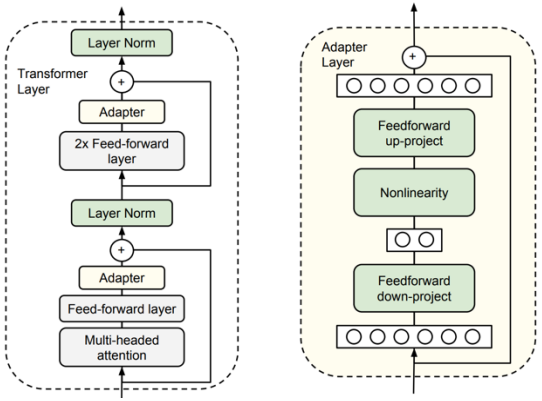
- Continued training (if you have enough GPUs)
  - This can be tricky if learning rate was cooled down at the end of training

- Parameter-efficient fine-tuning (PEFT)
  - Idea: Fine-tune only a small number of parameters (typically less than 1%)
  - Memory during training: Parameters, activations, gradients, optimizer state
    - With PEFT: Gradients and optimizer state only necessary for parameters that are updated

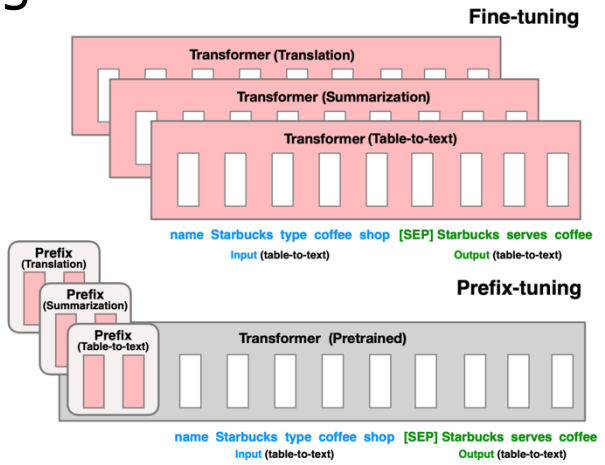
Parameter-Efficient Finetuning (PEFT)

- Adapter layers (Houlsby et al., 2019; Pfeiffer et al., 2020)
- Prompt tuning (Lester et al., 2021), Prefix tuning (Li and Liang, 2021), P-tuning (Liu et al., 2021)
- Low-rank adaptation (LoRA; Hu et al., 2022)

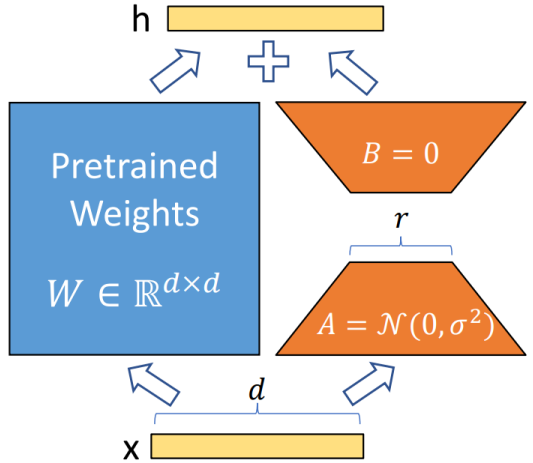
→ Hugging Face has the adapter-transformers library for adapter layers, and the PEFT library for prompt tuning and LoRA



Adapter layers (Houlsby et al., 2019)



Prefix tuning (Li and Liang, 2021)



LoRA (Hu et al., 2022)

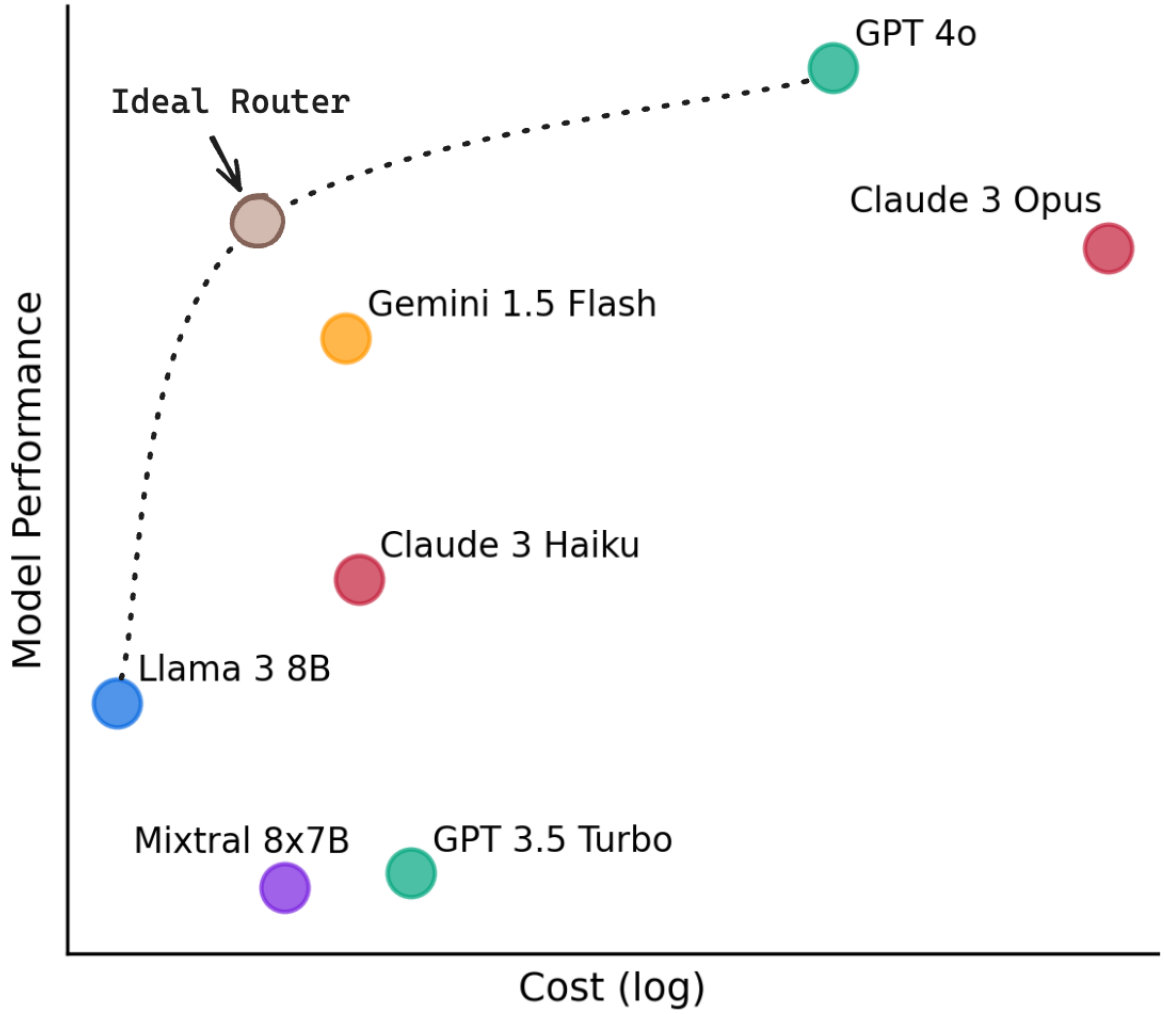
Parameter-Efficient Finetuning (PEFT)

- Quantization: Post-processing pretrained models to compress them
  - Different methods to do the compression:
    - AWQ (Lin et al., 2023)
    - GPTQ (Frantar et al., 2023)
  - Just applying quantization at inference time can hurt performance
  - Quantized models are usually available within hours or a few days of a new release on Hugging Face

- Quantization + LoRA = QLoRA (Dettmers et al., 2023)
  - 4-bit quantization combined with LoRA
  - Can achieve equal performance to full-precision finetuning

LLM Routing

- Don’t need to use the largest and best LLM for every task
- Can use simple models for easy tasks
  - Cost efficiency
  - Speed
- LLM routing: Find the smallest model with acceptable performance



https://lmsys.org/blog/2024-07-01-routellm/

LLMs as a Judge

- "… if you have a text-based task that a human can do pretty quickly and you don't need absolutely perfect reliability, you can usually ask your language model to do it …" - Interview with Anthropic researchers on Asterisk Magazine
- Zheng et al., 2023
- Hugging Face cookbook

LLMs as a Judge (Setup)

- Task description
- Scale description: minimum, maximum, value types (e.g. float)
- Explanation of the output format
- A beginning of an answer, to take the LLM by the hand as far as we can

LLMs as a Judge (Initial)

```
JUDGE_PROMPT = """
You will be given a user_question and system_answer couple.
Your task is to provide a 'total rating' scoring how well the system_answer answers the user concerns expressed in the user_question.
Give your answer as a float on a scale of 0 to 10, where 0 means that the system_answer is not helpful at all, and 10 means that the answer completely and helpfully addresses the question.

...

Provide your feedback as follows:

Feedback:::
Total rating: (your rating, as a float between 0 and 10)

Now here are the question and answer.

Question: {question}
Answer: {answer}

Feedback:::
Total rating: """
```

HSLU [Hugging Face cookbook]

LLMs as a Judge (Improvements)

- Leave more time for thought by adding an evaluation field before the final answer (chain-of-thought).
- Use a small integer scale like 1-4 or 1-5 instead of a large float scale as previously.
- Provide an indicative scale for guidance.
- Give examples (few-shot prompting).
- We even add a carrot to motivate the LLM!

LLMs as a Judge (Improved)

```
IMPROVED_JUDGE_PROMPT = """

You will be given a user_question and
system_answer couple.

Your task is to provide a 'total rating'
scoring how well the system_answer
answers the user concerns expressed in
the user_question.

Give your answer on a scale of 1 to 4,
where 1 means that the system_answer is
not helpful at all, and 4 means that the
system_answer completely and helpfully
addresses the user_question.


...
```

```
Here is the scale you should use to
build your answer:

1: The system_answer is terrible:
completely irrelevant to the question
asked, or very partial

2: The system_answer is mostly not
helpful: misses some key aspects of the
question

3: The system_answer is mostly helpful:
provides support, but still could be
improved

4: The system_answer is excellent:
relevant, direct, detailed, and
addresses all the concerns raised in the
question

...
```

LLMs as a Judge (Improved)

```
Provide your feedback as follows:


Feedback:::

Evaluation: (your rationale for the
rating, as a text)

Total rating: (your rating, as a number
between 1 and 4)


You MUST provide values for
'Evaluation:' and 'Total rating:' in
your answer.


...
```

```
Now here are the question and answer.


Question: {question}

Answer: {answer}


Provide your feedback. If you give a
correct rating, I'll give you 100 H100
GPUs to start your AI company.

Feedback:::

Evaluation: """
```

Comparing LLMs Online

- Chatbot Arena
  - https://lmarena.ai or https://beta.lmarena.ai/
  - New nicer UI
  - (you can use this for free access to state-of-the-art LLMs)
- Used to create a ranking of LLMs (based on Elo system)
  - Leaderboard from Chatbot Arena
  - Hugging Face with more sorting/filtering options
- There is also a Copilot Arena plugin for VS code
  - Compare code completions
  - (Only use with code that is allowed to be sent to the APIs)

Running LLMs Locally

- GPT4All
  - Easiest installation and intuitive UI
  - Can upload files and do retrieval-augmented generation
- Ollama
  - Wrapper around Hugging Face
  - Most models
  - Command-line, can install additional UI
- Llama.cpp
  - Written in pure C/C++, optimized for speed
  - Uses the GGUF quantization format