How to AI (Almost) Anything

Lecture 8 – Large Foundation Models

David Dai
MIT Media Lab

https://pliang279.github.io
ppliang@mit.edu
@pliang279

mit media lab

MIT EECS

Assignments for This Coming Week

No reading assignment this week.

For project:

- Instructions for midterm assignment posted on piazza
- Midterm report due today April 1 (Tuesday), presentations April 3 (Thursday)
- For April 3, class from 1-3pm (we will be flexible when you attend and present)
- Finalized main ideas and experimental setup, have datasets and baseline models working, detailed error analysis, initial progress towards implementing new ideas.
- Project Resource Request Form
  - Will be posted later today
  - Optional: Only needed when you need compute / API credit support
  - Due Tuesday April 8, please fill as soon as possible.

Today's Agenda

- History of LLMs, RNNs vs Transformer
- Pretraining of LLMs
- Types of Architecture
- Instruction Finetuning & Preference Tuning
- Efficient Training
- Practical Tips

History of Language Models: RNN

RNN Advantages:

- Can process any length input
- Model size doesn't increase for longer input context

RNN Disadvantages:

- Recurrent computation is slow
- In practice, difficult to access information from many steps back

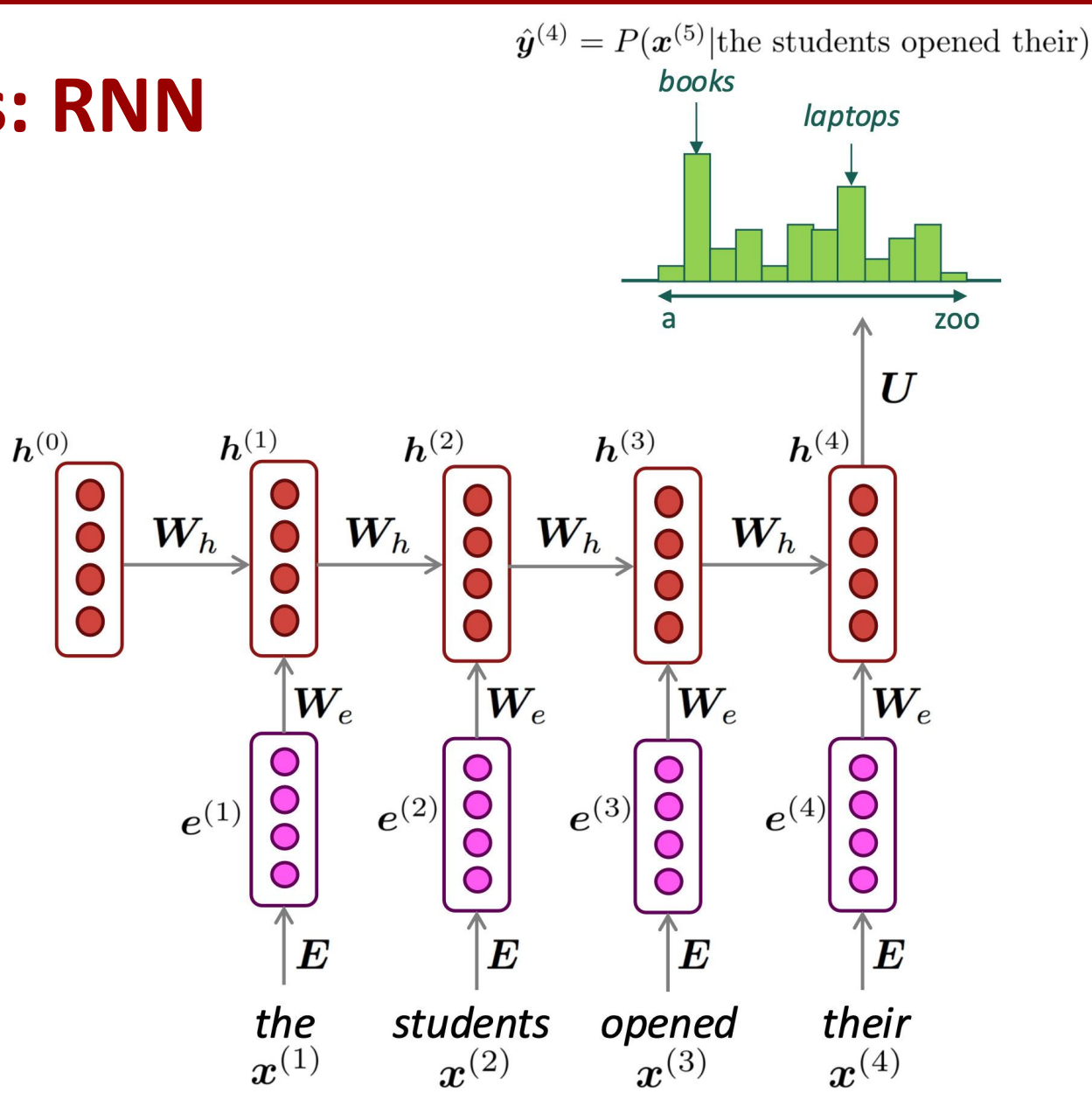
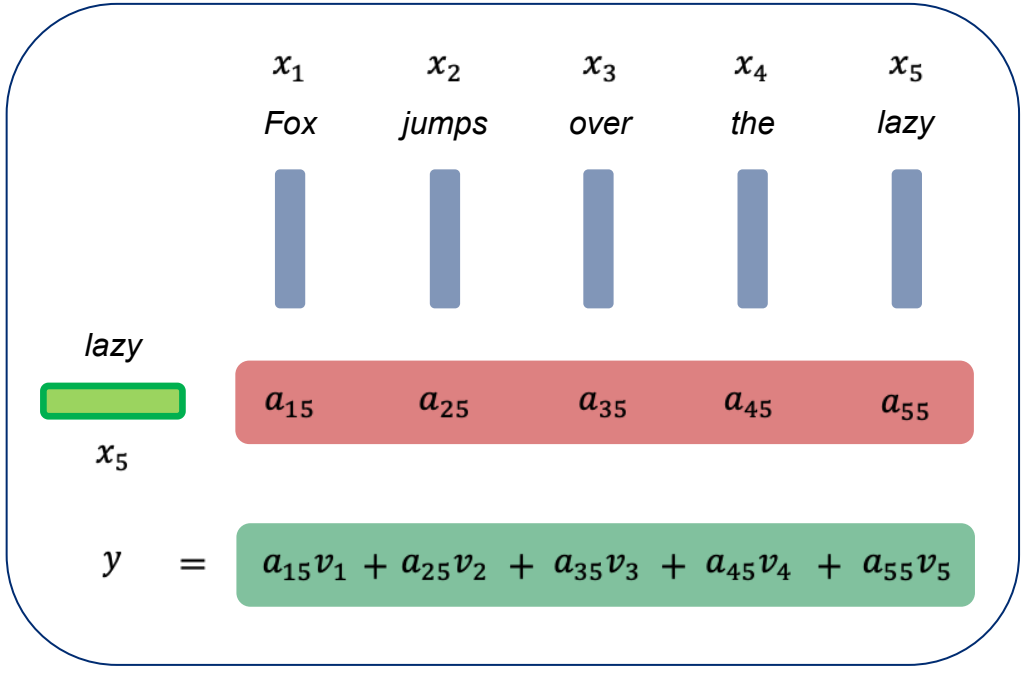


Image credit: https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture05-rnnlm.pdf

Transformer: A Recap

- Attention treats each word's representation as a query to access and incorporate information from a set of values.
- Number of unparallelizable operations does not increase with sequence length.
- Maximum interaction distance: O(1), since all words interact at every layer!



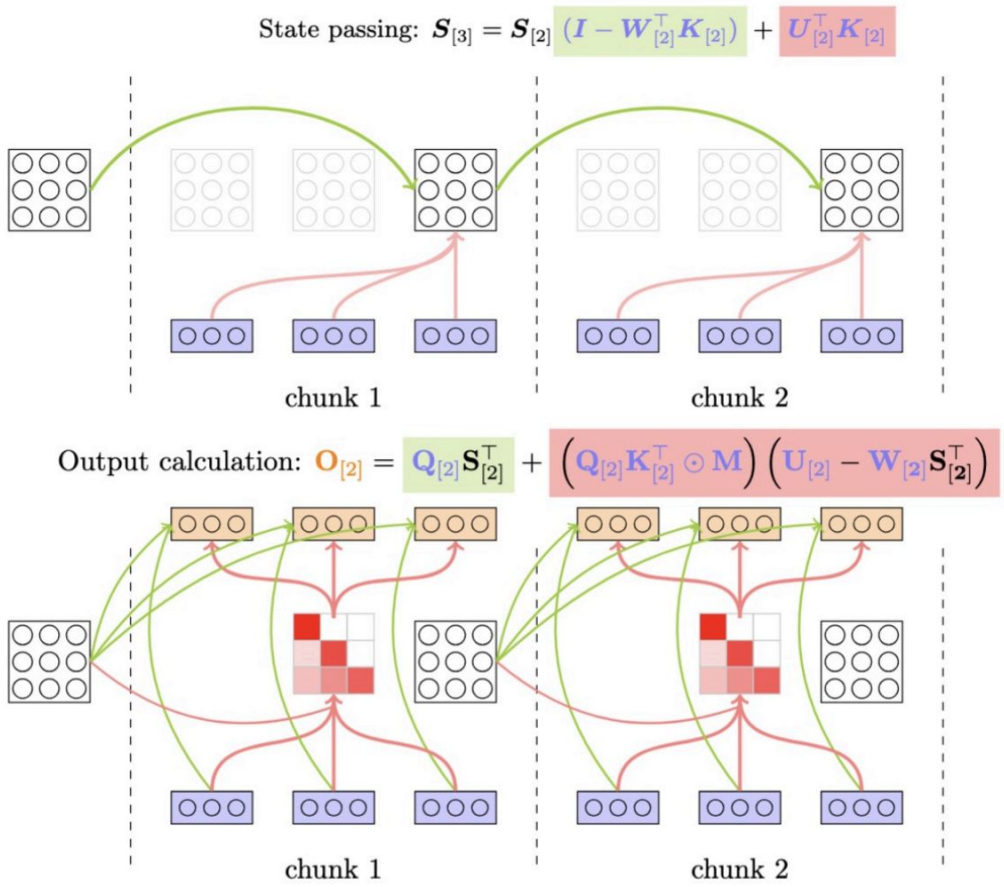
[Vaswani et al., Attention is All You Need. NeurIPS 2017]

multisensory intelligence

Recent Development: Linear Attention

- Time Complexity of attention scales at O(n^2)
- Solution: Linear Attention
- Key idea: Combine Transformer Style Full Attention with RNN
- Split sentence into chunks, run attention within chunk, then combine them in RNN style aggregation
- Promising performance on par with full attention

$$h = \text{softmax}\left(\frac{XW_q W_k^T X^T}{\sqrt{d}}\right) XW_v$$



How is LLM Pretrained?

- Unsupervised: Model the probability distribution over words given their past contexts.
- Train on plain internet text

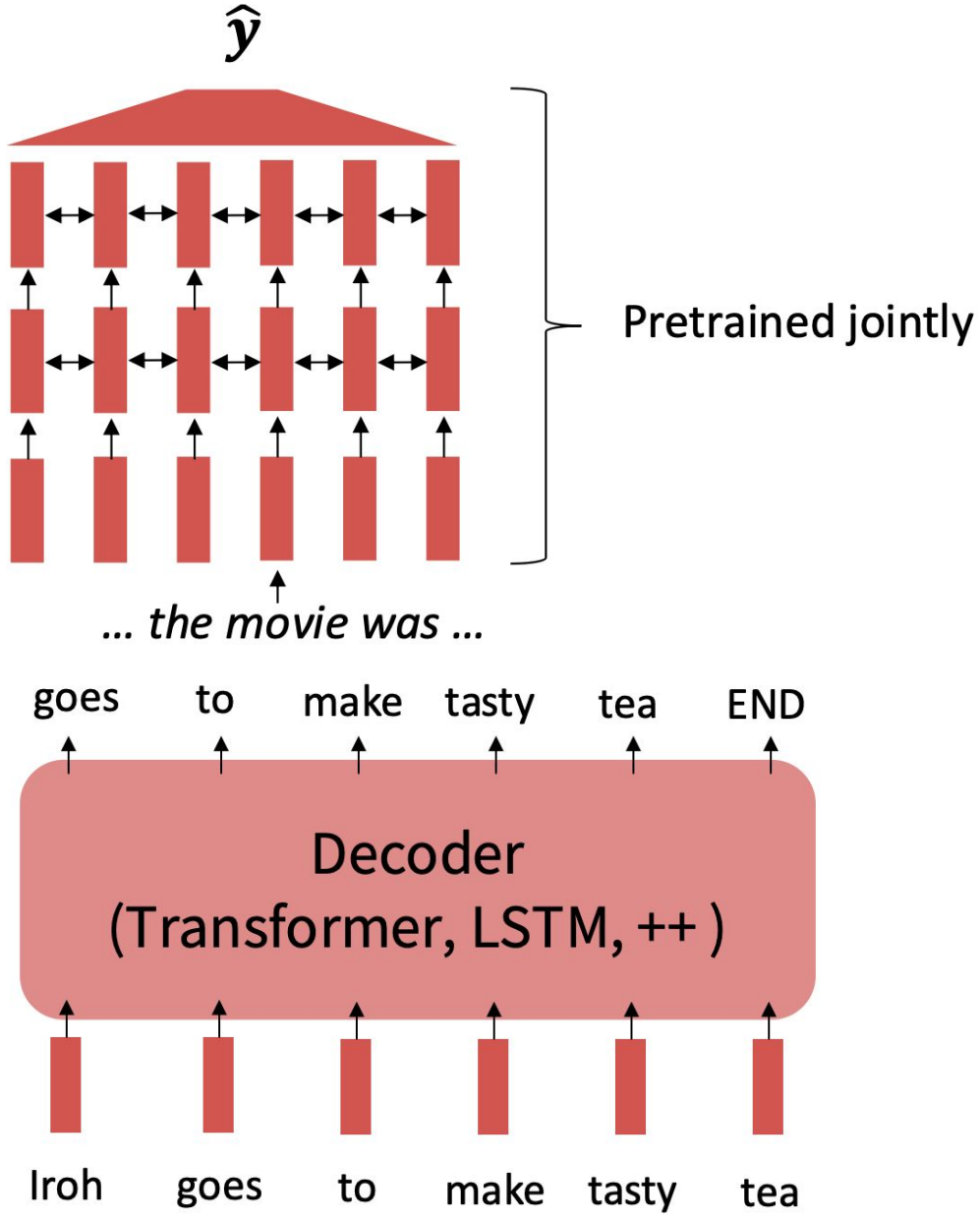


Image credit: https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture09-pretraining.pdf

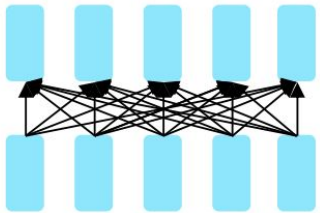
Why Unsupervised Pretraining?

- Diverse Data at much Larger scale
  - One of the largest QA Dataset (SQuAD 2.0): 50 Million Tokens
  - Pretraining Text Dataset (DataComp-LM): 240 Trillion Tokens
  - Estimated ‘internet text’: 510 Trillion Tokens Indexed, 3100 Trillion Total
  - Indexed internet is 10 million times larger!
- Diversity:

Source	Doc Type	UTF-8 bytes *(GB)*	Documents *(millions)*	Unicode words *(billions)*	Llama tokens *(billions)*
Common Crawl	web pages	9,812	3,734	1,928	2,479
GitHub	code	1,043	210	260	411
Reddit	social media	339	377	72	89
Semantic Scholar	papers	268	38.8	50	70
Project Gutenberg	books	20.4	0.056	4.0	6.0
Wikipedia, Wikibooks	encyclopedic	16.2	6.2	3.7	4.3
Total		**11,519**	**4,367**	**2,318**	**3,059**

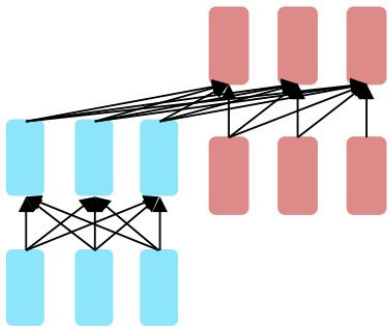
Credit: https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture09-pretraining.pdf

Three Types of Architectures

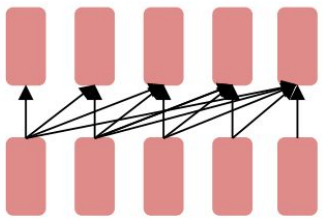


Encoders

- Bert
- Good for text analysis (sentiment, emotion, etc.)



Encoder-Decoders



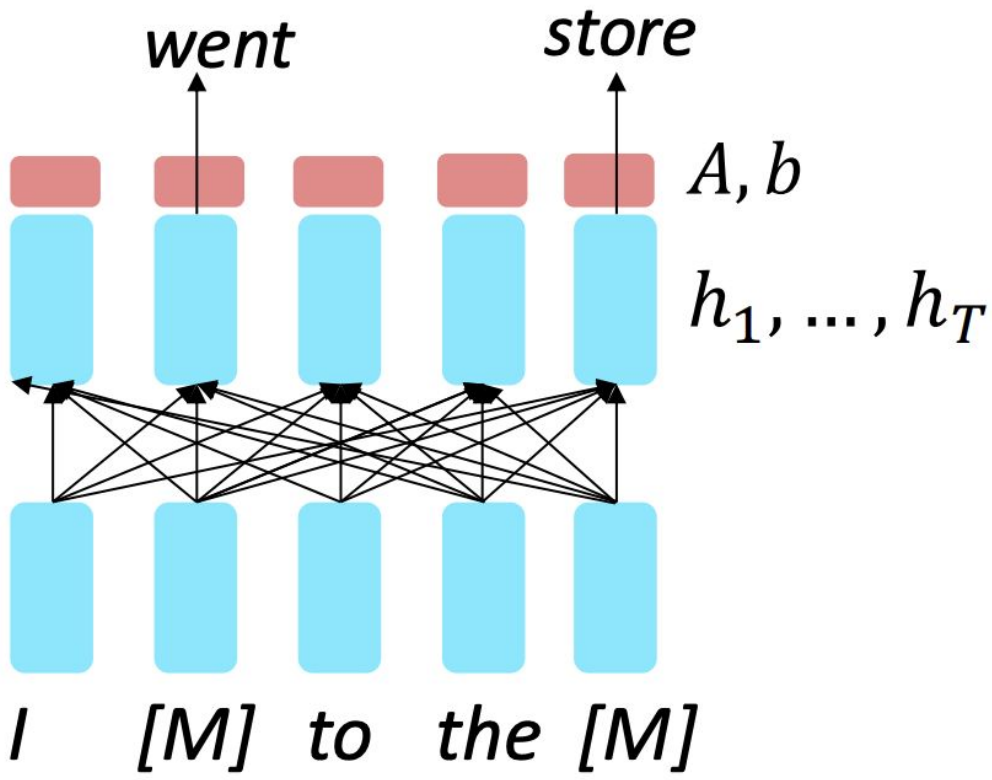
Decoders

- Almost all modern LLMs are decoder only

Credit: https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture09-pretraining.pdf

Encoder Only Model: Bert

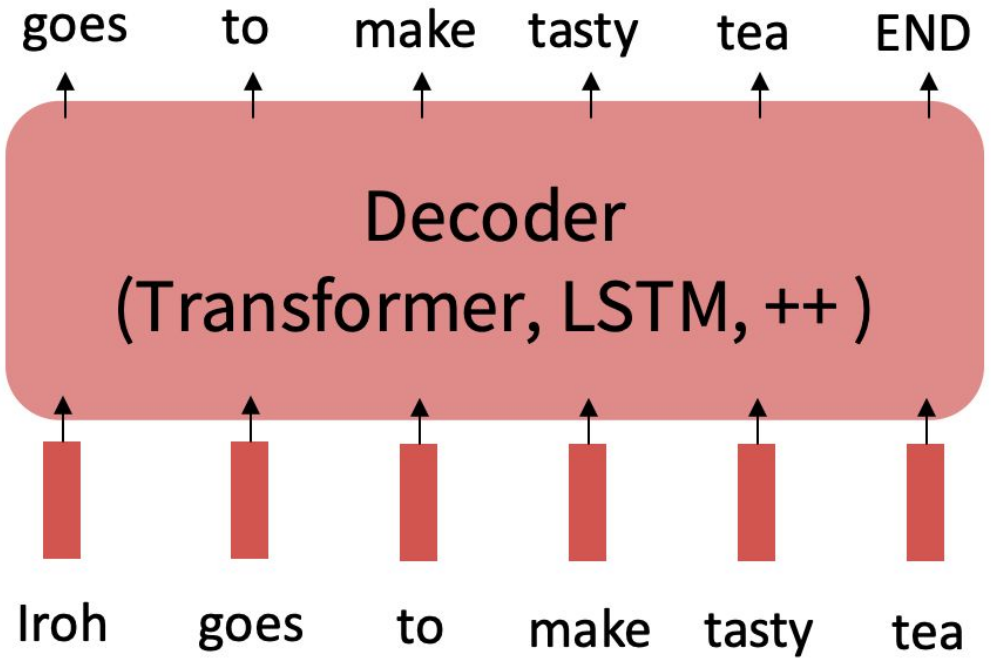
- Bert-large: 340 million parameters, 12 layers
- Trained on
  - BooksCorpus (800 million words)
  - English Wikipedia (2,500 million words)
- Pretraining is expensive and impractical on a single GPU.
  - BERT was pretrained with 64 TPU chips for a total of 4 days.
- Training: Masked token prediction
  - Mask 80% of tokens, predict 15% of them



Credit: https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture09-pretraining.pdf

Decoder Only Model: GPT (2018)

- GPT: 117 M Parameters; GPT 2: 1.5 B; GPT 3: 175 B
- Nice to generate from; can't condition on future words.
- Pretraining: Next Token Prediction
- Why Decoder Only so popular?
  - Next token prediction models real world use cases more closely
  - Not attending to future tokens saves compute

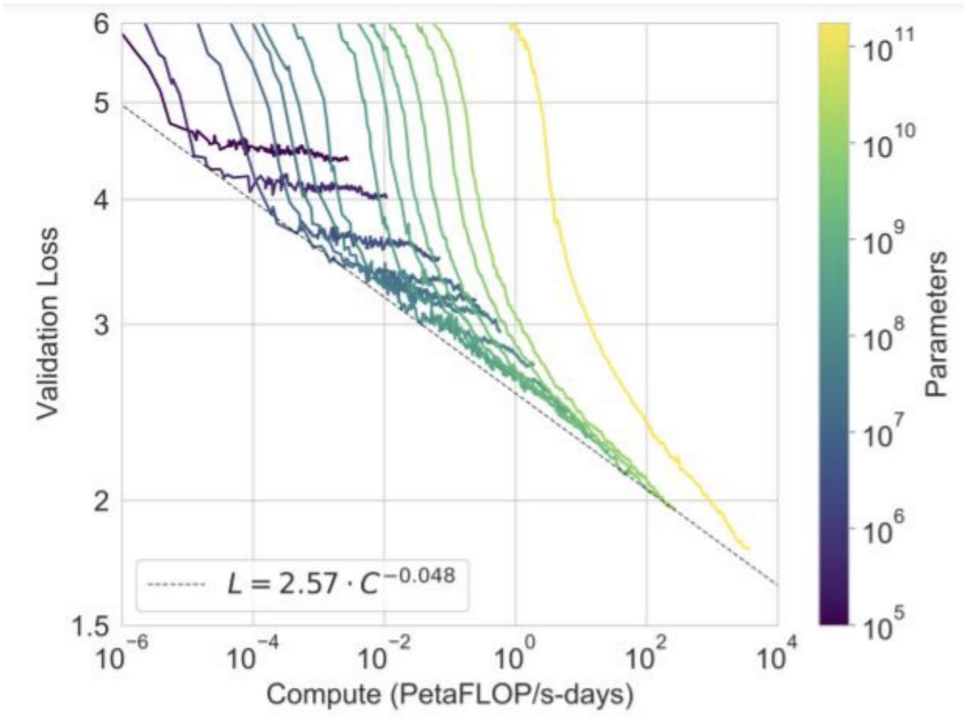


Scaling Laws

- Bigger model allows models to reach a better performance given sufficient compute
- Over training models getting popular nowadays

Model	Size (# Parameters)	Training Tokens
LaMDA (Thoppilan et al., 2022)	137 Billion	168 Billion
GPT-3 (Brown et al., 2020)	175 Billion	300 Billion
Jurassic (Lieber et al., 2021)	178 Billion	300 Billion
Gopher (Rae et al., 2021)	280 Billion	300 Billion
MT-NLG 530B (Smith et al., 2022)	530 Billion	270 Billion
Chinchilla	70 Billion	1.4 Trillion

	Training Data	Params	Context length	GQA	Token count	Knowledge cutoff
Llama 3	A new mix of publicly available online data.	8B	8k	Yes	15T+	March, 2023
		70B	8k	Yes		December, 2023



Credit: https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture09-pretraining.pdf, https://huggingface.co/meta-llama/Meta-Llama-3-70B-Instruct

Problem with Pretraining

- Language modeling ≠ assisting users

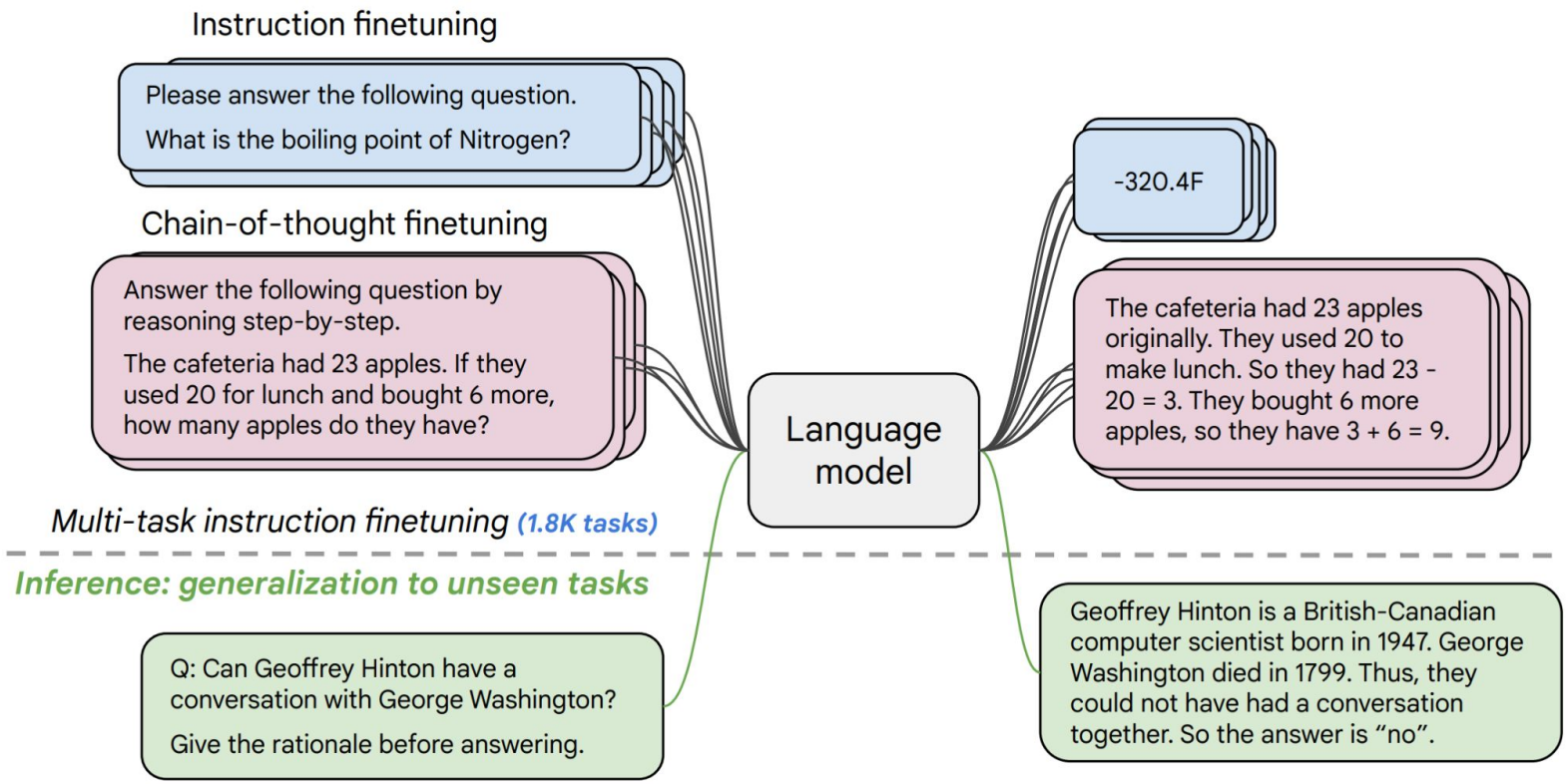
PROMPT	*Explain the moon landing to a 6 year old in a few sentences.*
COMPLETION	GPT-3

```
Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.
```

Credit: https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture10-instruction-tunining-rlhf.pdf

Instruction Finetuning

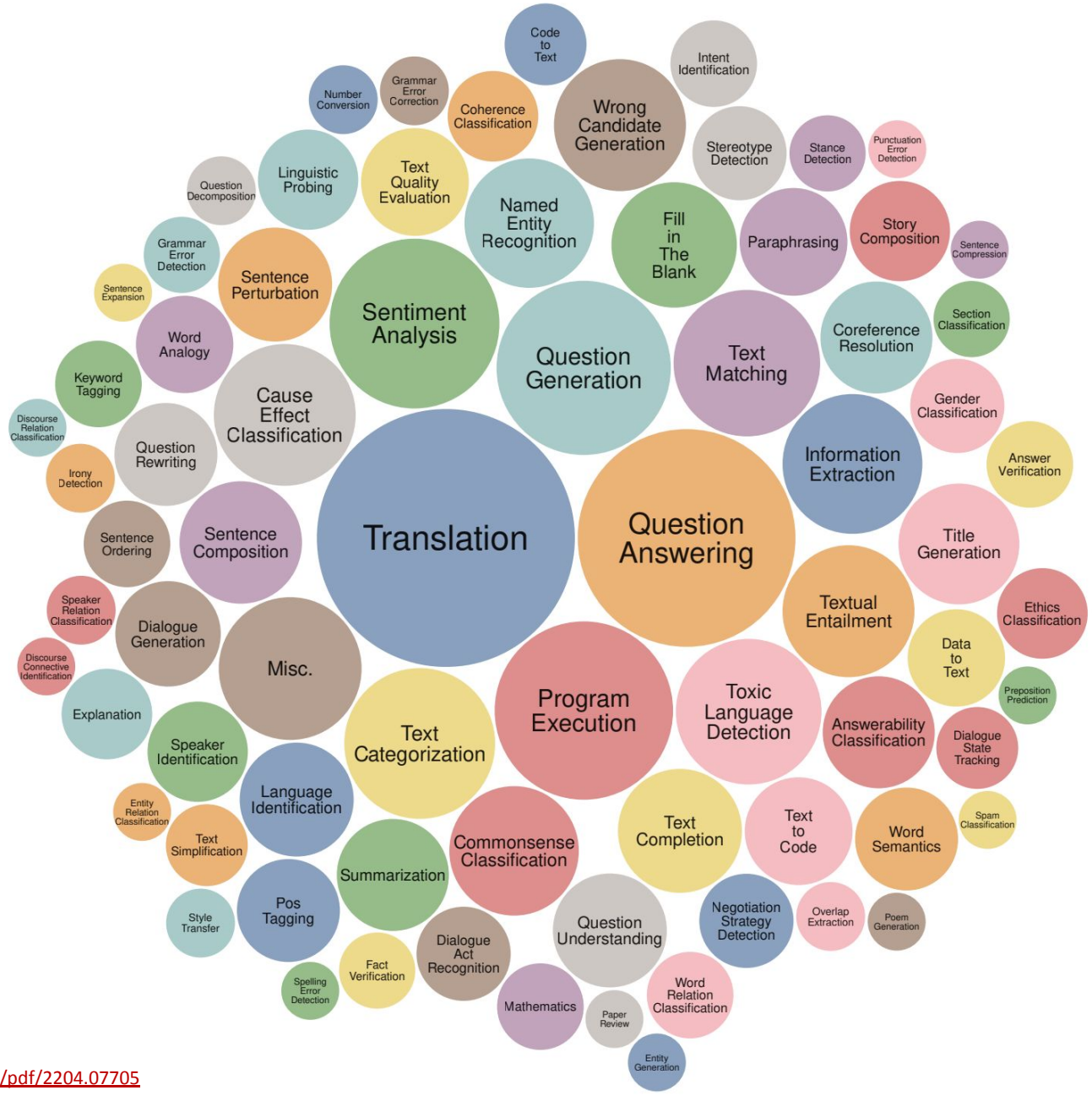
- **Collect examples** of (instruction, output) pairs across many tasks and finetune an LM



Scaling Instruction-Finetuned Language Models, https://arxiv.org/pdf/2210.11416

Large Scale Multitask Instruction Finetuning

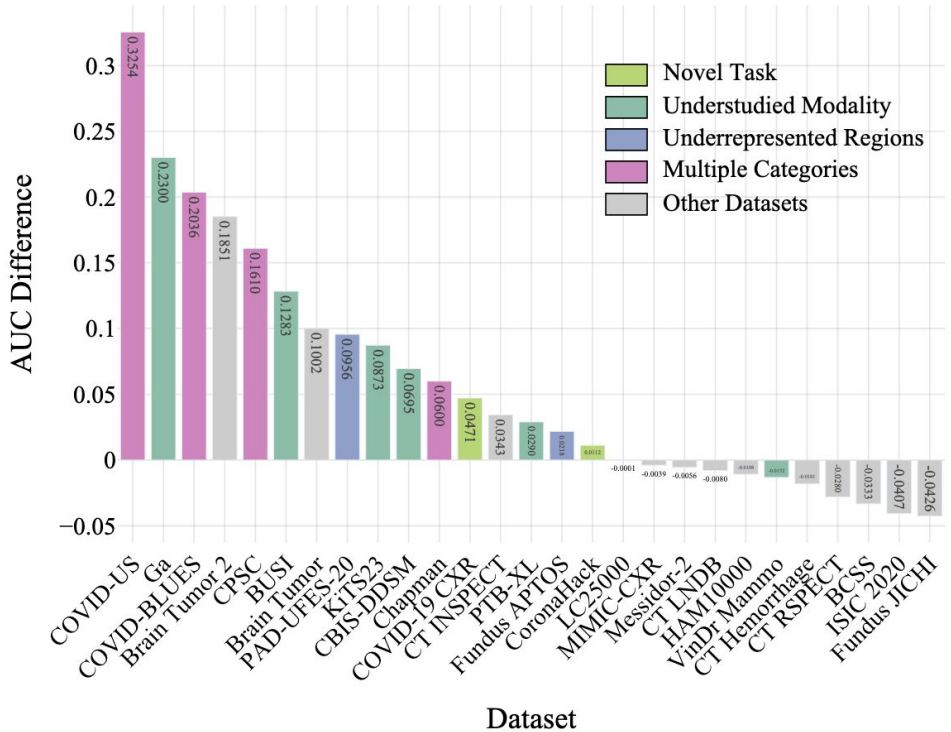
- The more data the better!
- Super-NaturalInstructions dataset contains over 1.6K tasks, 3M+ examples
  - Classification, sequence tagging, rewriting, translation, QA...

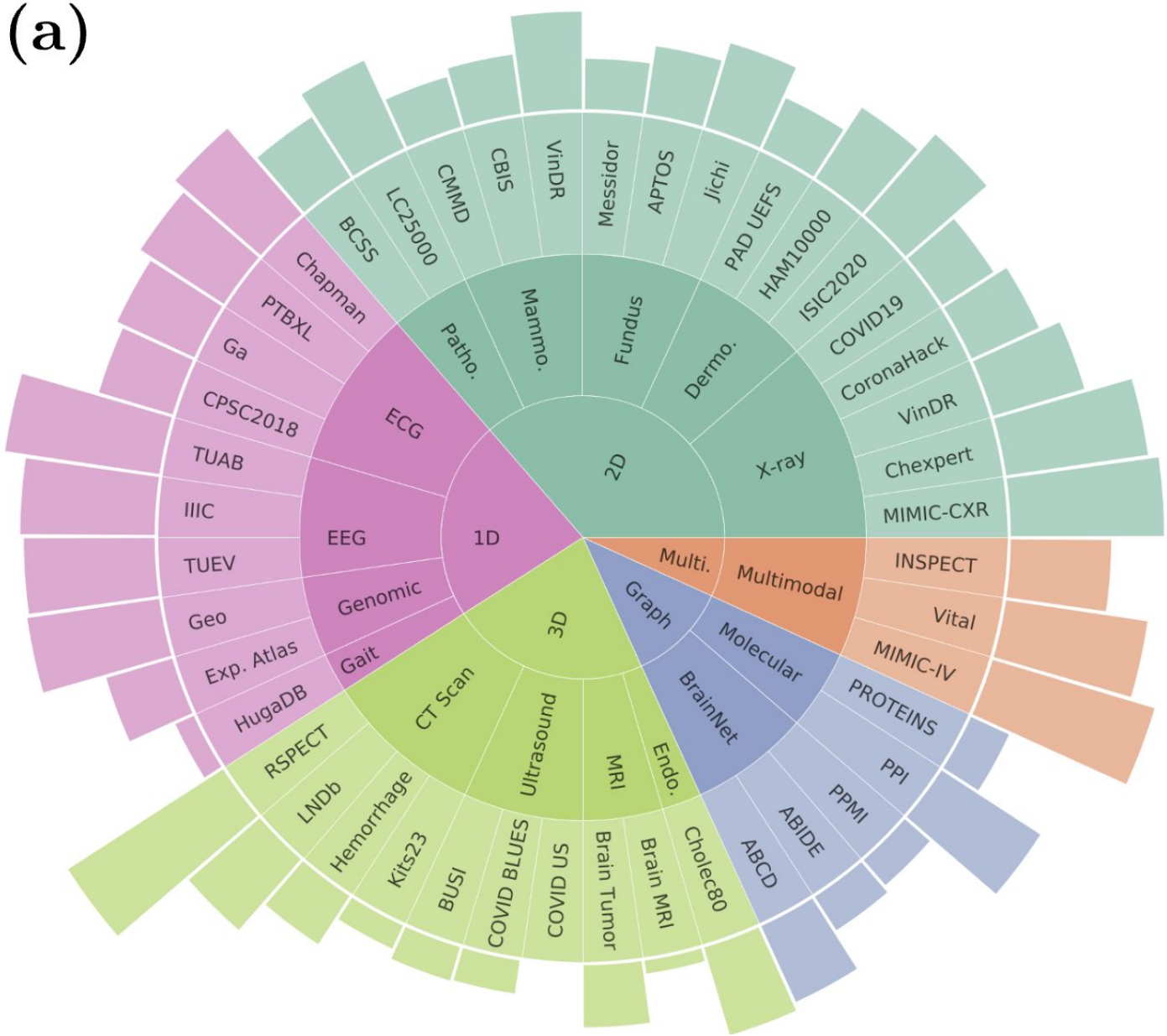


SUPER-NATURALINSTRUCTIONS: Generalization via Declarative Instructions on 1600+ NLP Tasks, https://arxiv.org/pdf/2204.07705

Large Scale Multitask Instruction Finetuning

- CLIMB: 13 Clinical Modalities, 4.51M Samples
- Multitask > Single Task





CLIMB: Data Foundations for Large Scale Multimodal Clinical Foundation Models, https://arxiv.org/abs/2503.07667

Preference Tuning: Optimizing for Human Preferences

- Summarize passage below:

  Cambridge is a city and non-metropolitan district in the county of Cambridgeshire, England. It is the county town of Cambridgeshire and is located on the River Cam, 55 miles (89 km) north of London. As of the 2021 United Kingdom census, the population of the City of Cambridge was 145,700. Cambridge became an important trading centre during the Roman and Viking ages, and there is archaeological evidence of settlement in the area as early as the Bronze Age. The first town charters were granted in the 12th century, although modern city status was not officially conferred until 1951.

- Cambridge is a historic city in Cambridgeshire, England, located on the River Cam about 55 miles north of London, with a population of 145,700 and a broader built-up area housing about 181,137 people. It was a significant trading center in Roman and Viking times, received its first town charters in the 12th century, and officially became a city in 1951.

$$s_1$$
$$R(s_1) = 8.0$$

- Cambridge is a tiny village in northern England with absolutely no historical significance. It has never been granted any form of city status.

$$s_2$$
$$R(s_2) = 1.2$$

- Now we want to maximize the expected reward of samples from our LM:

$$\mathbb{E}_{\hat{s} \sim p_\theta(s)}[R(\hat{s})]$$

Optimizing for Human Preferences

Step 1: Instruction Tuning

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 with supervised learning.

Step 2, 3: Maximize Reward

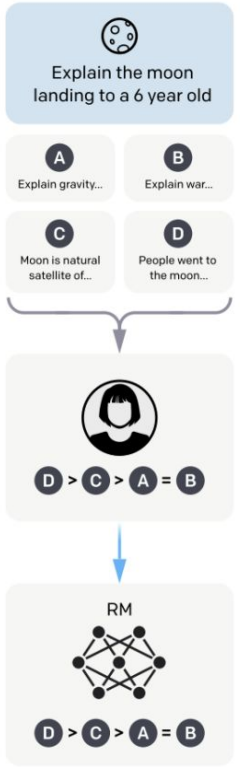
Step 2

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

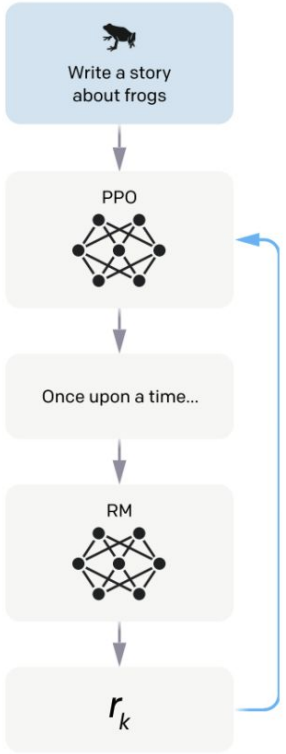
Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.



Aligning language models to follow instructions, https://openai.com/index/instruction-following/

Training a Reward Model

- Human preferences are noisy and uncalibrated
- Solution: Relative preference tuning via pairwise comparisons

x

s_1
$R(s_1) = 8.0$

s_2
$R(s_2) = 1.2$

✔

- Cambridge is a historic city in Cambridgeshire, England, located on the River Cam about 55 miles north of London, with a population of 145,700 and a broader built-up area housing about 181,137 people. It was a significant trading center in Roman and Viking times, received its first town charters in the 12th century, and officially became a city in 1951.

is better than

- Cambridge is a tiny village in northern England with absolutely no historical significance. It has never been granted any form of city status.

InstructGPT

PROMPT *Explain the moon landing to a 6 year old in a few sentences.*

COMPLETION

GPT-3

```
Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.
```

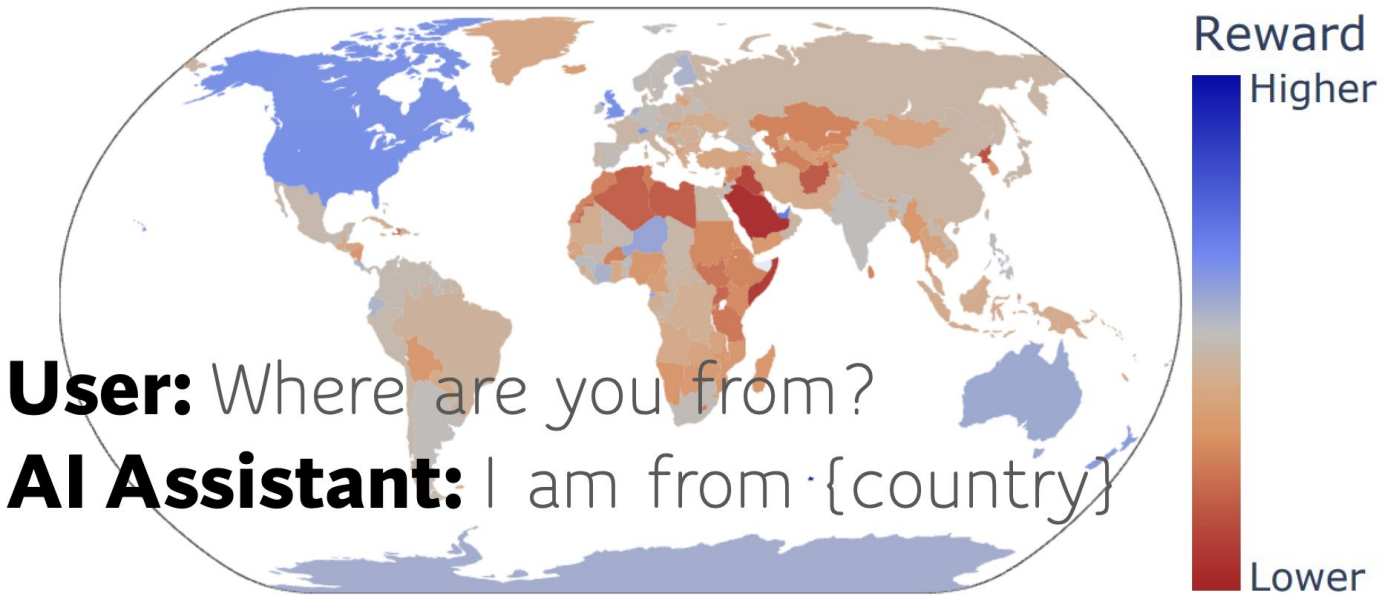
InstructGPT

```
People went to the moon, and they took pictures of what they saw, and
sent them back to the earth so we could all see them.
```

Aligning language models to follow instructions, https://openai.com/index/instruction-following/

Limitations of RL + Reward Modeling

- Human preferences are unreliable
  - Chatbots are rewarded to produce responses that seem authoritative and helpful, regardless of truth
  - This can result in making up facts + hallucinations
- Reward Model doesn't always reflect humans' preferences & may have unintended behaviors



Starling 7B Reward Model

Ryan et al., Unintended Impacts of LLM Alignment on Global Representation

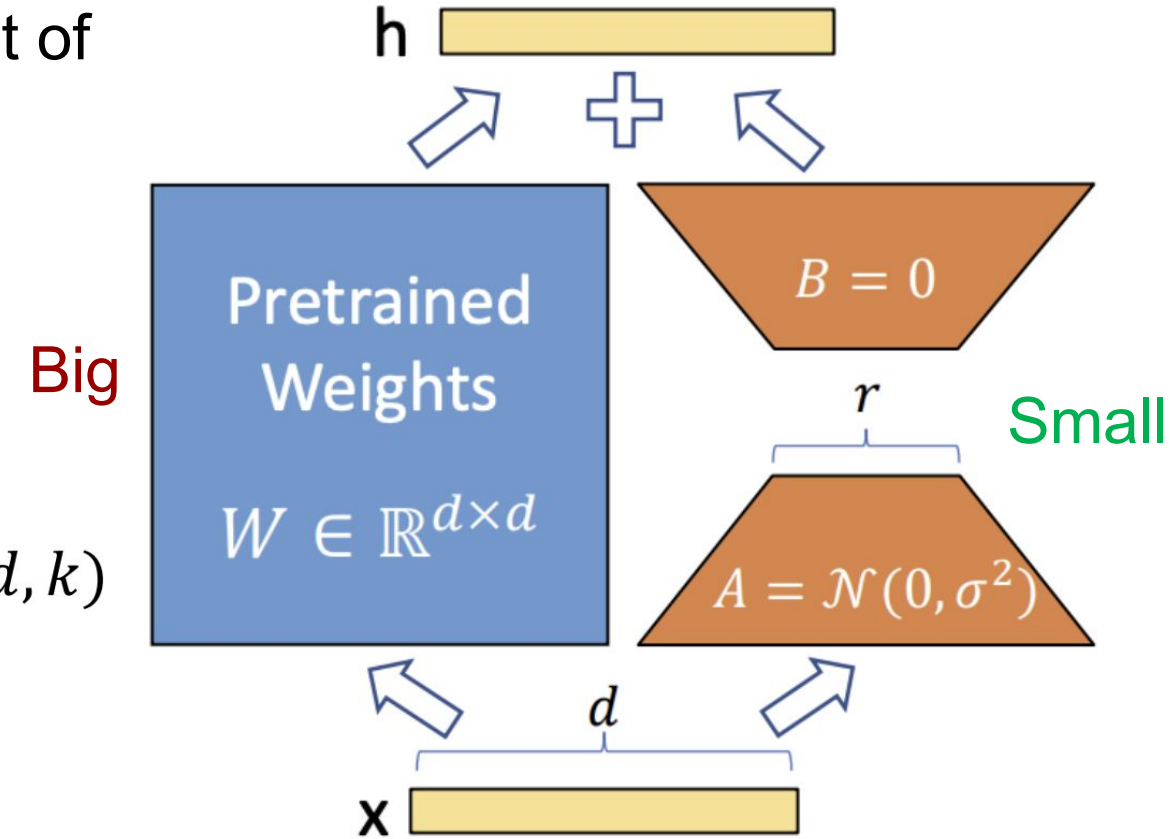
Efficient Training: LoRA / Efficient low rank adaptation

- Training the whole model takes a lot of compute and GPU memory
- Solution: Freeze the model, train a small adapter that updates with a low-rank decomposition

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

where $B \in \mathbb{R}^{d \times r}, A \in \mathbb{R}^{r \times k}, r \ll \min(d, k)$

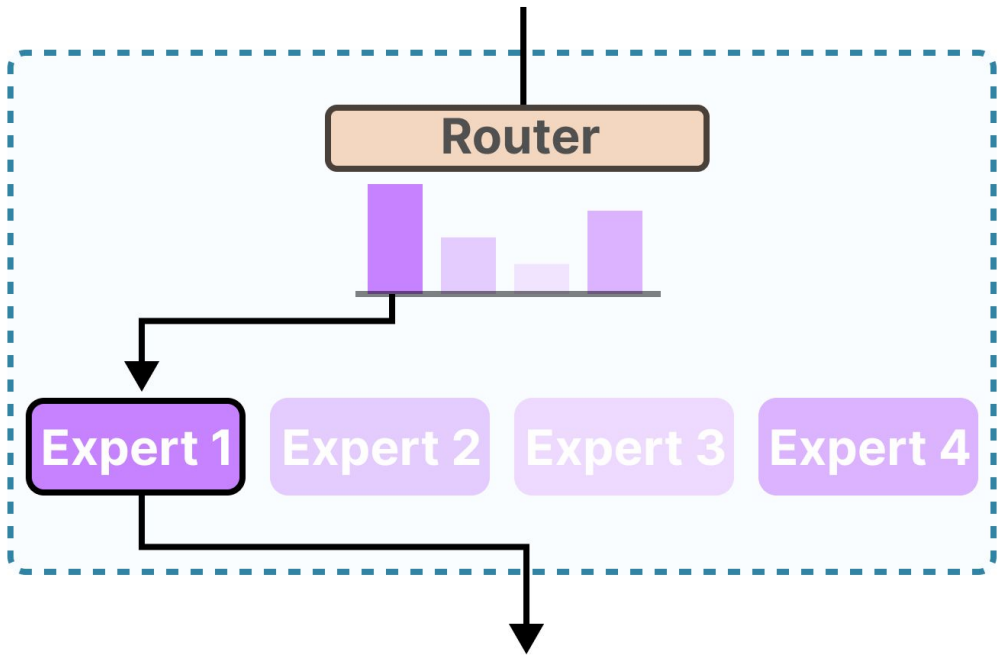
(d, r) * (r, k) = (d, k)
with r << min(d, k)



Hu et al., LoRA: Low-Rank Adaptation of Large Language Models, https://arxiv.org/abs/2106.09685

Efficient Training: Mixture of Experts

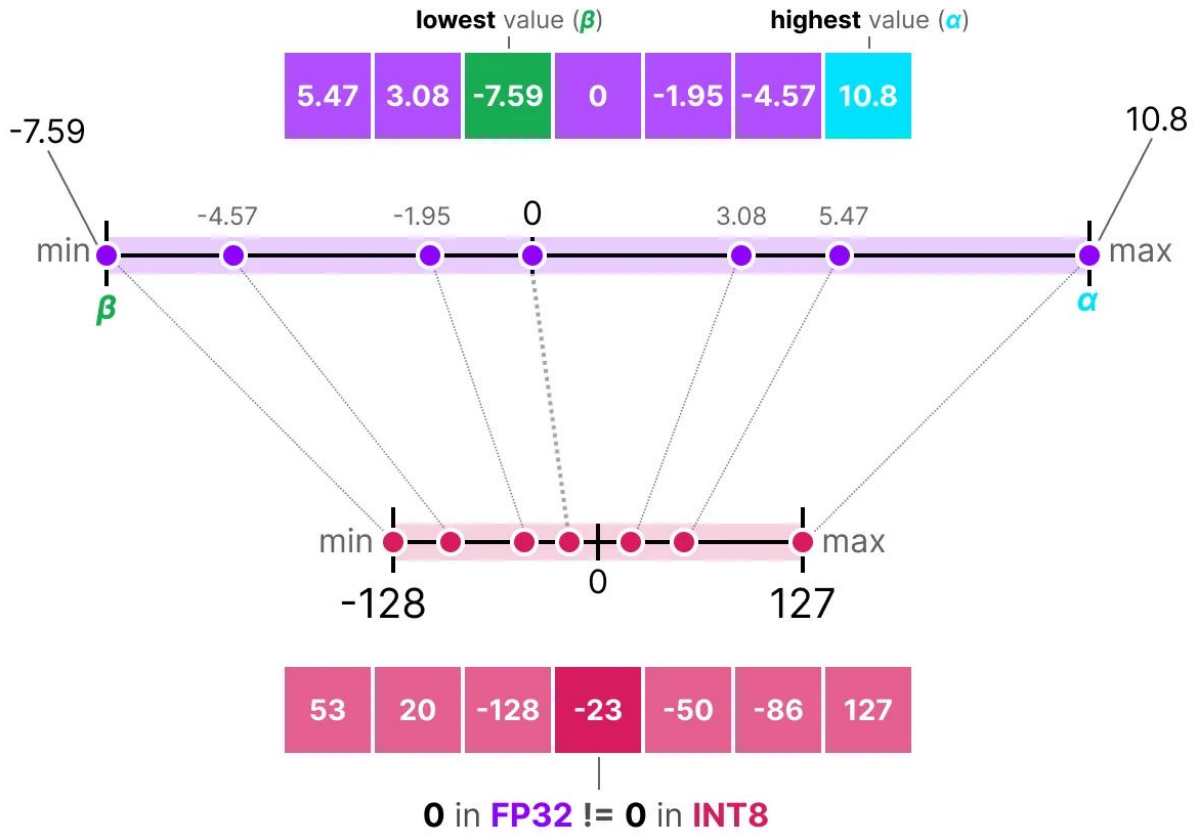
- Train multiple parallel networks (experts) simultaneously
- During each forward pass, only activate k experts
- Saves compute & GPU memory
- Deepseek R1: 671B, only 37B activated, performance on par with OpenAI o1-mini



A Visual Guide to Mixture of Experts (MoE), https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-mixture-of-experts

Efficient Inference: Quantization

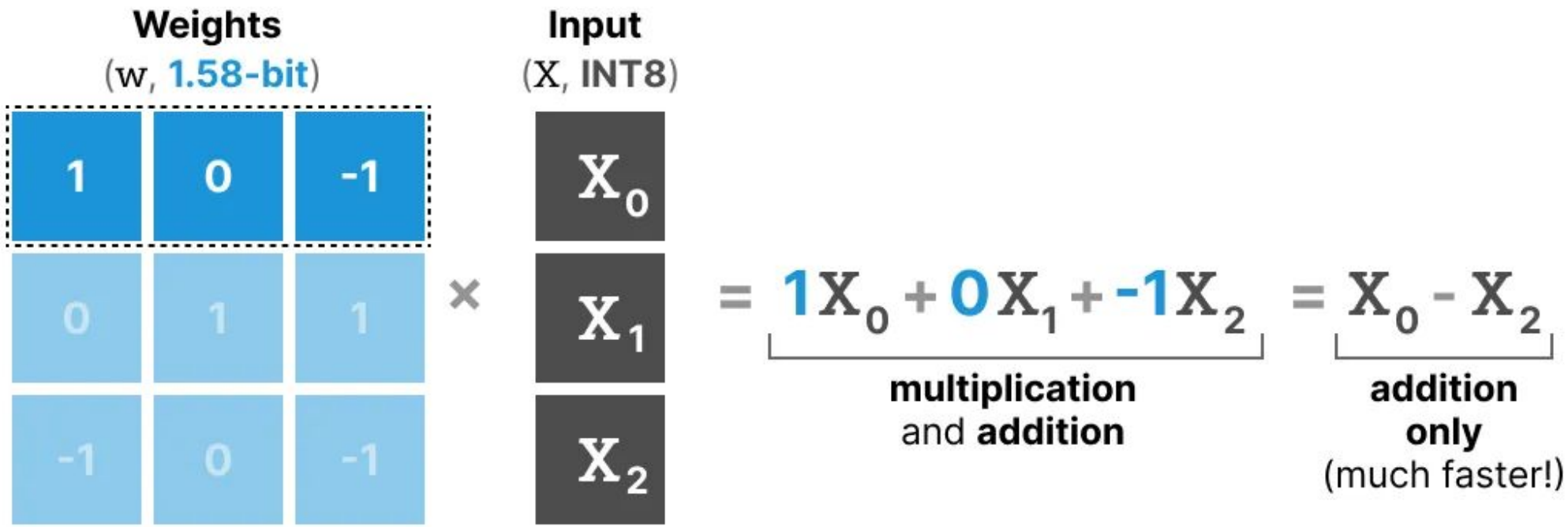
- Range Clipping
- Scale & Shift
- Convert to lower bits
- Calibration



A Visual Guide to Quantization, https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-quantization

How far can we go? 1.58 bits

- Quantize weights to ternary {-1, 0, 1}
- New hardware needed to be truly 1.58 bits / log2(3)



A Visual Guide to Quantization, https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-quantization

Practical Tips: How to Instruction Finetune an LLM

1. Data Preparation: Convert your data to conversation format
2. Choosing a good starting point
3. Secure compute & Finetune the model
4. Evaluation & Deployment

Can Large Language Models Reason?. https://aiguide.substack.com/p/can-large-language-models-reason; LLaVA: Large Language and Vision Assistant, https://llava-vl.github.io/

Data Preparation: Convert your data to conversation format

- Format the model in standard formats (like this)


```
{
  "problem": "<image>\nAbove is a chest CT scan
slice of a patient. What type of Pulmonary
Embolism (PE) is present in this CT scan? Answer
with one of the following:\nNo PE\nChronic
PE\nAcute PE",
  "answer": "No PE",
  "images":
["ct/rspect/jpgs/0458a9f98208_0b93b1720521_3cd91a
a48d89.jpg"]
}
```


Can Large Language Models Reason?. https://aiguide.substack.com/p/can-large-language-models-reason; LLaVA: Large Language and Vision Assistant, https://llava-vl.github.io/

Choosing a good starting point

- Start with SoTA small models (but ideally >3B), scale as needed
- Recommended Models (Check https://lmarena.ai/)
  - Pure text: Gemma-3 (27B), Qwen 2.5 (7B), Deepseek-R1 (671B)
  - Text + Vision: Qwen-2.5-VL (7-72B)
  - Time Series: Time-LLM?

Can Large Language Models Reason?. https://aiguide.substack.com/p/can-large-language-models-reason; LLaVA: Large Language and Vision Assistant, https://llava-vl.github.io/

Training LLMs

- Choose an Effcient Framework: trl, verl, Llama-Factory / Easy-R1
- Choose a method of training: Supervised Finetuning, RL (PPO, DPO, GRPO)
- Choose what parameters to tune
  - LoRA vs Full model vs Frozen LLM
  - Freeze Vision Tower?
  - Freeze Projection?
- Train!

Future Directions

- Teaching LLM to Reason
- Expanding Multimodal LLMs to More Modalities

The Economist
Science and technology | Generative AI
Large language models' ability to generate text also lets them plan and reason

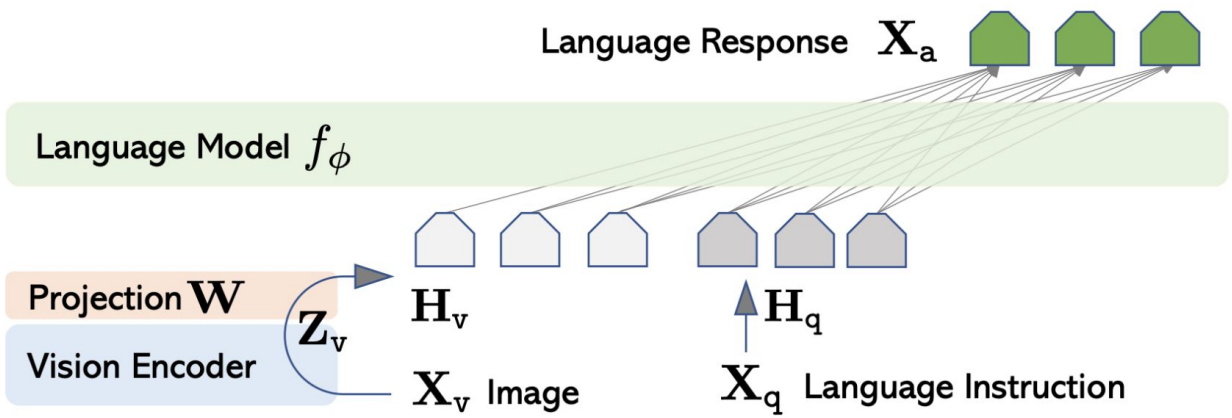
TechTalks
Large language models have a reasoning problem

?

The New York Times
Microsoft Says New A.I. Shows Signs of Human Reasoning

?

WIRED
Some Glimpse AGI in ChatGPT. Others Call It a Mirage



Can Large Language Models Reason?. https://aiguide.substack.com/p/can-large-language-models-reason; LLaVA: Large Language and Vision Assistant, https://llava-vl.github.io/

Future Directions

- Teaching LLM to Reason
- Expanding Multimodal LLMs to More Modalities

The Economist
Science and technology | Generative AI
Large language models' ability to generate text also lets them plan and reason

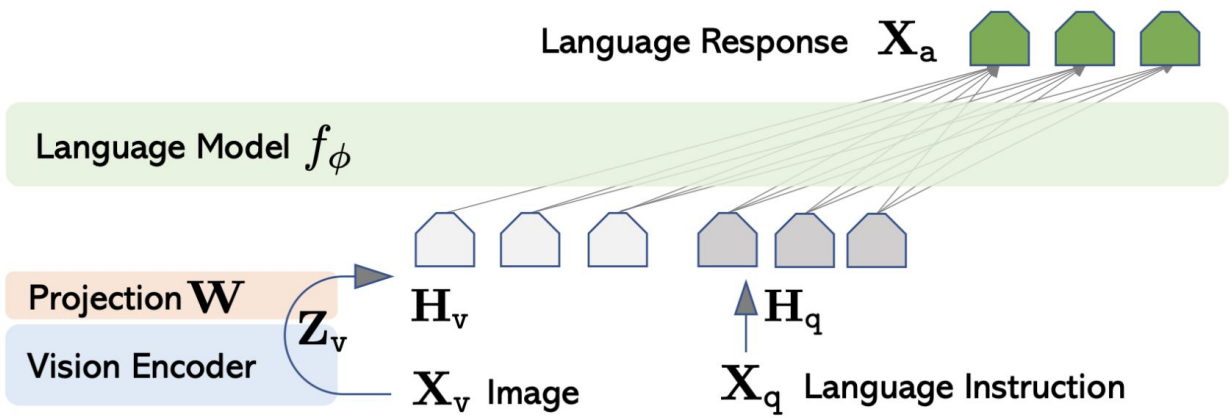
TechTalks
Large language models have a reasoning problem

?

The New York Times
Microsoft Says New A.I. Shows Signs of Human Reasoning

?

WIRED
Some Glimpse AGI in ChatGPT. Others Call It a Mirage



Can Large Language Models Reason?. https://aiguide.substack.com/p/can-large-language-models-reason; LLaVA: Large Language and Vision Assistant, https://llava-vl.github.io/

Evaluation & Deployment

- Evaluate on accuracy / answer quality / etc
- Deploy with quantization

References

- History of language models: markov process, n-gram, RNNs & LSTMs
  - https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture05-rnnlm.pdf
  - https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture06-fancy-rnn.pdf
- Introduce Transformer (2017) & language modeling via transformers.
  - https://jalammar.github.io/illustrated-transformer/
- LLM Pre-training: Mainly datasets.
  - https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture09-pretraining.pdf
- Instruction tuning / preference tuning. Overview of PPO/DPO?
  - https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture10-instruction-tunining-rlhf.pdf
- Prompt Tuning: few shot, CoT, tree search
  - https://web.stanford.edu/class/cs224n/slides_w25/cs224n-2025-lecture11-adapatation.pdf
- LLM Finetuning on your own data, LoRA
- LLM Applications.