Large Language Models

(Part 3)

April 16, 2026

Announcements

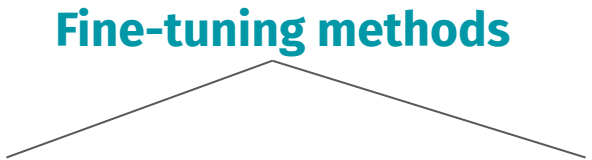
- Final Project Checkpoint 3 (code + outline/draft) due Friday, April 17 11:59pm
- Final Project Presentations on April 23 and 28
- AI@ND Showcase:
  - When: Monday, April 20 from 12:00pm - 5:30pm
  - Where: Hesburgh library 2nd floor
  - What: talks, student prompt-a-thon ($1000 in prizes), training classes
  - More info: https://ai.nd.edu/participate/ai-showcase/

Plan for today

- **Review of last class:** supervised fine-tuning of LLMs
- **Our goals today:**
  - Fine-tuning with RLHF
  - Beyond fine-tuning: methods for further improving LLMs
- **Next time:**
  - What are LLM agents?
  - Creating your own LLM agents

Review: Supervised Fine-tuning

Main approaches for fine-tuning



Supervised fine-tuning (SFT)

use labeled data (e.g., prompt-response pairs) from a specific task/domain to adjust the learned weights in the model

Reinforcement learning with human feedback (RLHF)

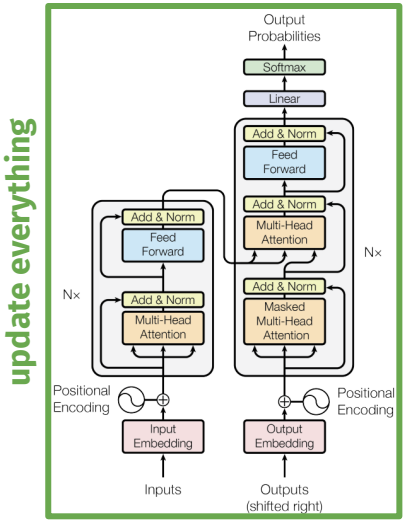
use human feedback to adjust the weights in the model

Supervised fine-tuning (SFT)

How do we update the model weights when fine-tuning?

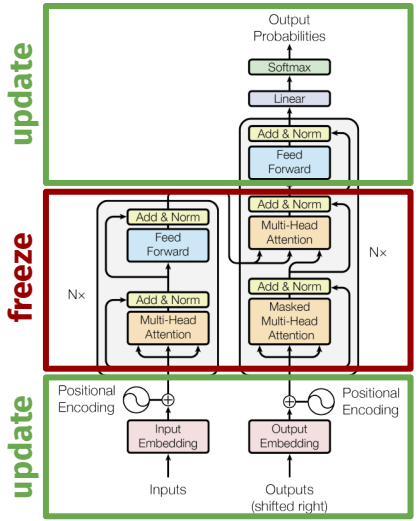
Full fine-tuning
(Update all model parameters)

Downsides: "catastrophic forgetting" (may forget how to do general tasks), requires huge amounts of memory

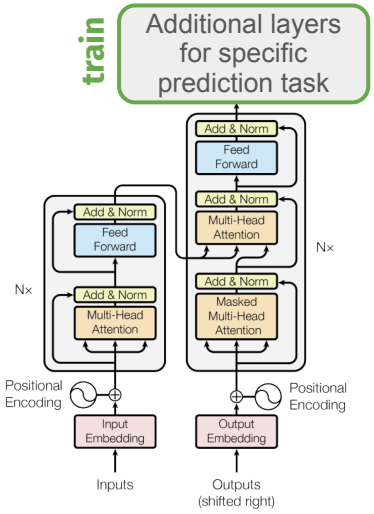


Parameter-efficient fine-tuning (PEFT)
(Partially update model parameters)

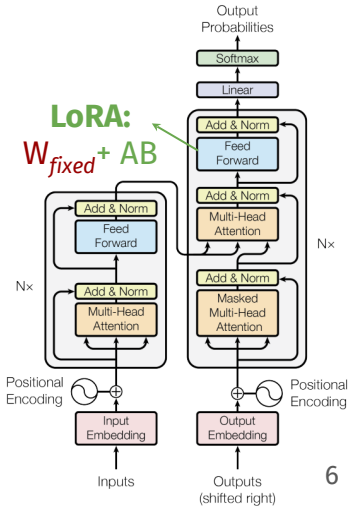
Selective
update subset of original LLM parameters; freeze all others



Additive/adapters
add new trainable components on top of base pre-trained LLM



Reparameterization
reparameterize weights



Main approaches for fine-tuning

Fine-tuning methods

Supervised fine-tuning (SFT)

use labeled data (e.g., prompt-response pairs) from a specific task/domain to adjust the learned weights in the model

Reinforcement learning with human feedback (RLHF)

use human feedback to adjust the weights in the model

Review: Fine-tuning with RLHF

Reinforcement learning with human feedback

Given pre-trained model:

1. Collect human feedback

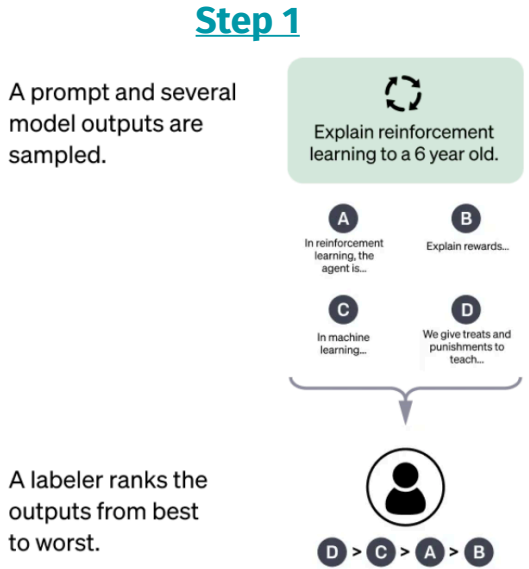


Figure source: Ouyang et al. (2022)

Reinforcement learning with human feedback

Given pre-trained model:

1. Collect human feedback
2. Train reward model to mimic human preferences

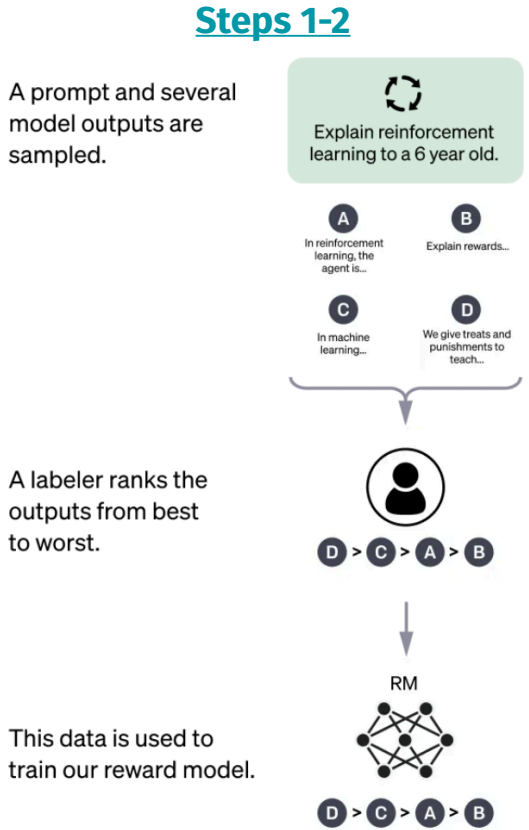
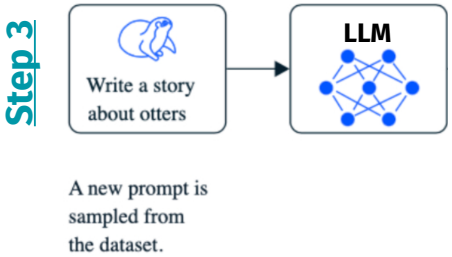


Figure source: Ouyang et al. (2022)

Reinforcement learning with human feedback

Given pre-trained model:

1. Collect human feedback
2. Train reward model to mimic human preferences
3. Fine-tune model using reinforcement learning
    a. Feed training samples through model



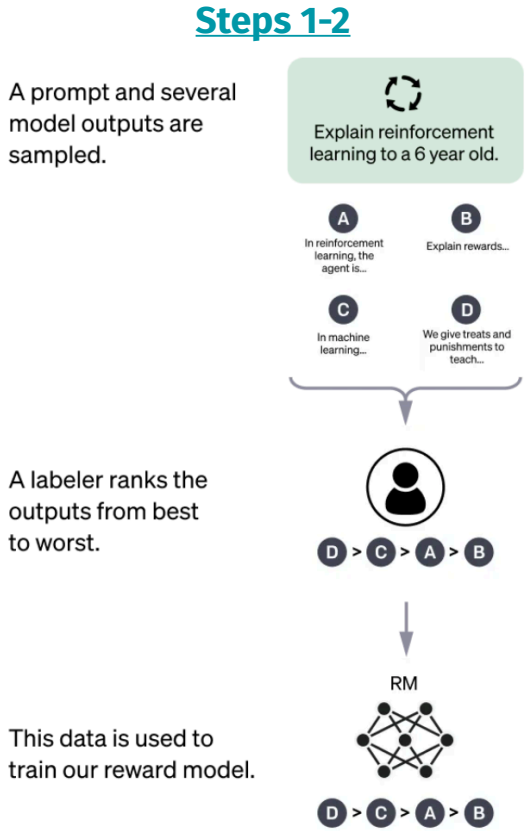
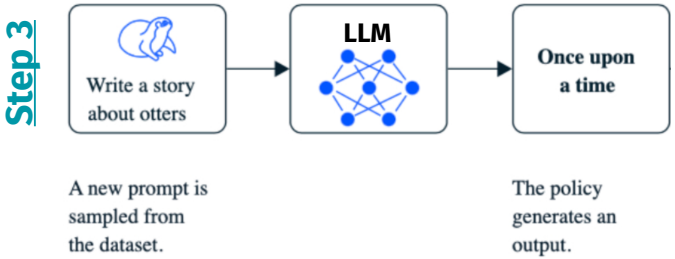


Figure source: Ouyang et al. (2022)

Reinforcement learning with human feedback

Given pre-trained model:

1. Collect human feedback
2. Train reward model to mimic human preferences
3. Fine-tune model using reinforcement learning
   a. Feed training samples through model
   b. Model generates outputs



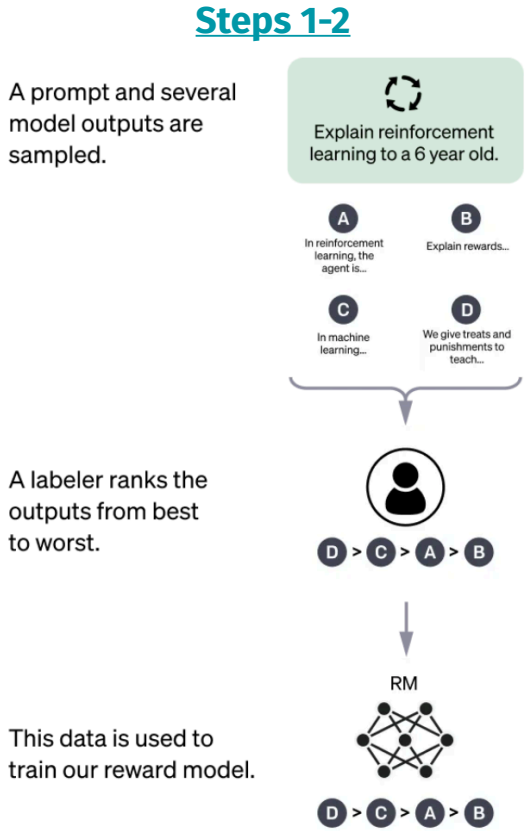


Figure source: Ouyang et al. (2022)

Reinforcement learning with human feedback

Given pre-trained model:

1. Collect human feedback
2. Train reward model to mimic human preferences
3. Fine-tune model using reinforcement learning
   a. Feed training samples through model
   b. Model generates outputs
   c. Outputs get scored by reward model

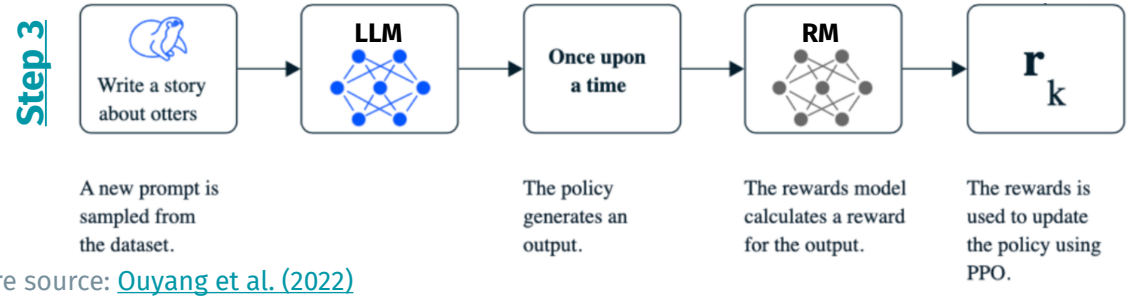
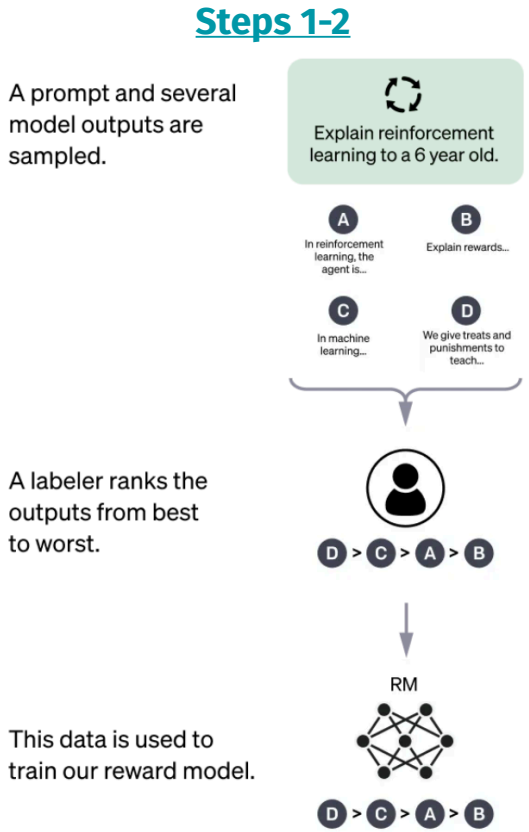


Figure source: Ouyang et al. (2022)



Reinforcement learning with human feedback

Given pre-trained model:

1. Collect human feedback
2. Train reward model to mimic human preferences
3. Fine-tune model using reinforcement learning
   a. Feed training samples through model
   b. Model generates outputs
   c. Outputs get scored by reward model
   d. Update model to maximize reward

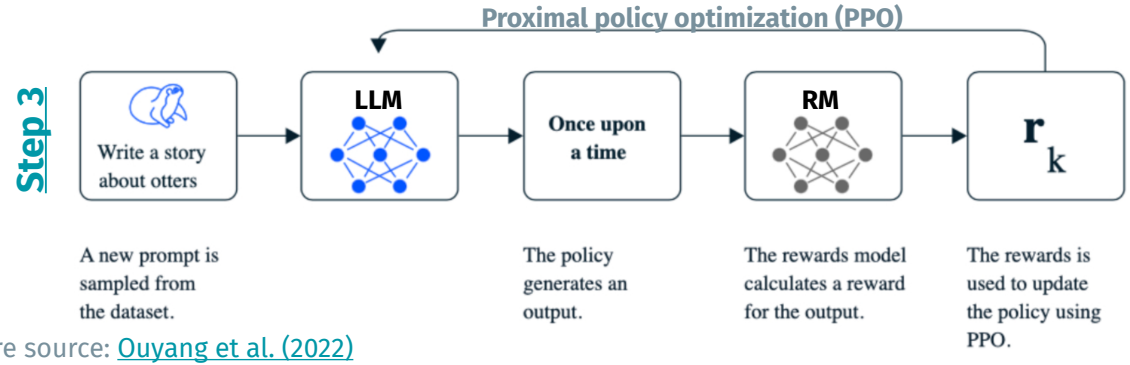
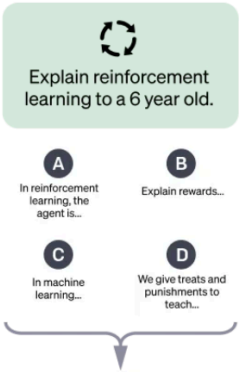


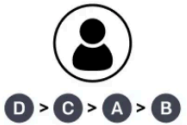
Figure source: Ouyang et al. (2022)

Steps 1-2

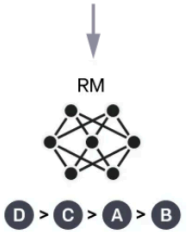
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



Beyond Fine-Tuning:
How can we improve the LLM without adjusting the weights?

Other techniques to improve LLMs in practice: **prompting**

Other techniques to improve LLMs in practice: **prompting**

Prompt engineering: improves LLMs without re-training or fine-tuning

- Be as clear and as specific as possible
  - Ex: Each of the following prompts will give very different outputs (see ChatGPT responses)
    - "Explain prompt engineering."
    - "Explain prompt engineering to a group of senior undergraduates and graduate students in a data science class."
    - "Explain prompt engineering in simple words, like I'm a 10-year old."

Other techniques to improve LLMs in practice: **prompting**

Prompt engineering: improves LLMs without re-training or fine-tuning

+ Be as clear and as specific as possible
+ Give examples (aka few-shot demonstrations, in-context learning)

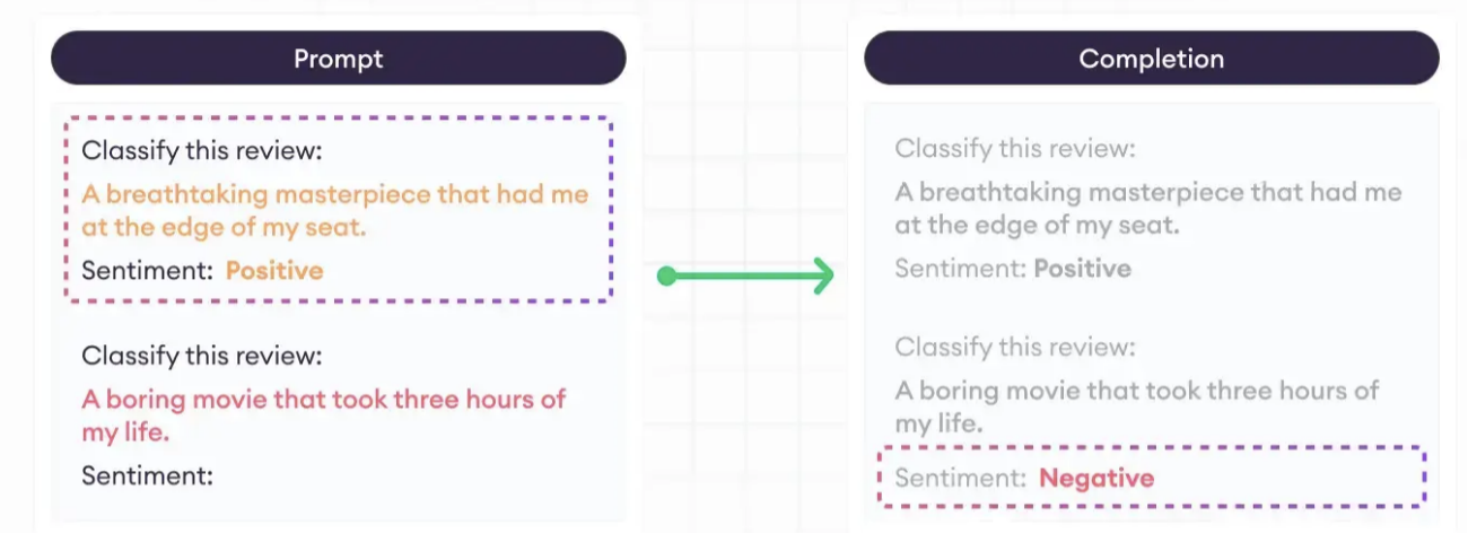


Figure Source: Armine Papikyan (2025)

Other techniques to improve LLMs in practice: **prompting**

Prompt engineering: improves LLMs without re-training or fine-tuning

- Be as clear and as specific as possible
- Give examples (aka few-shot demonstrations, in-context learning)
- Use chain-of-thought prompting

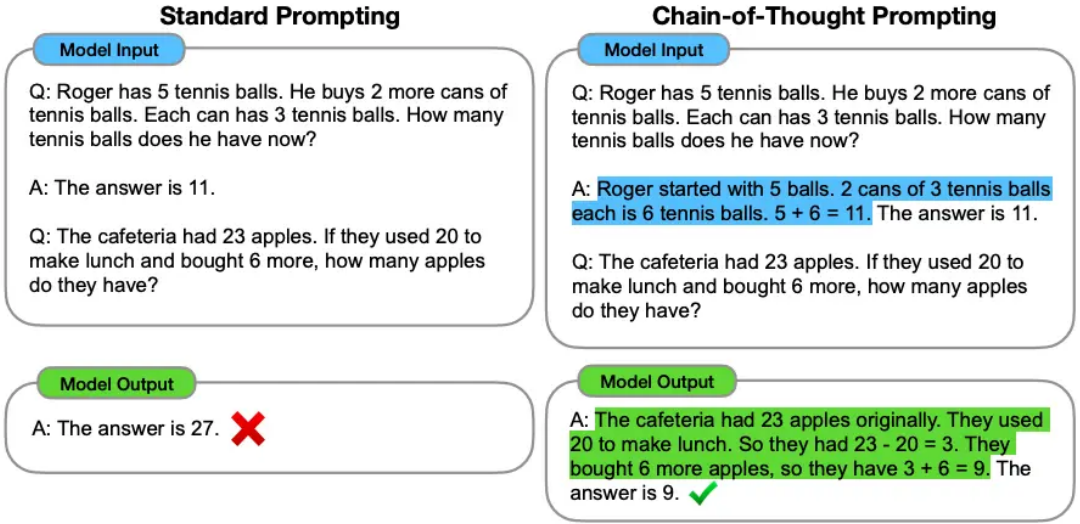


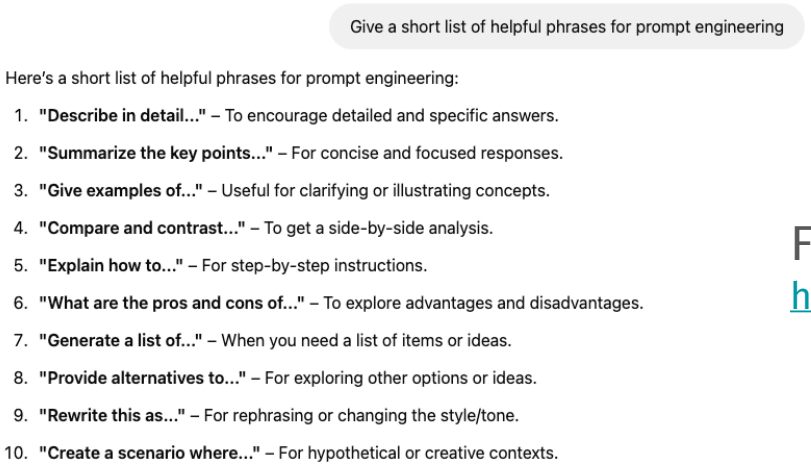
Figure Source: Wei et al. (2023)

Other techniques to improve LLMs in practice: **prompting**

Prompt engineering: improves LLMs without re-training or fine-tuning

- Be as clear and as specific as possible
- Give examples (aka few-shot demonstrations, in-context learning)
- Use chain-of-thought prompting
- Use key phrases (e.g., "Think step-by-step")

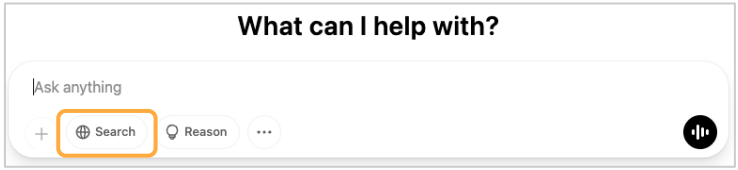
From ChatGPT:

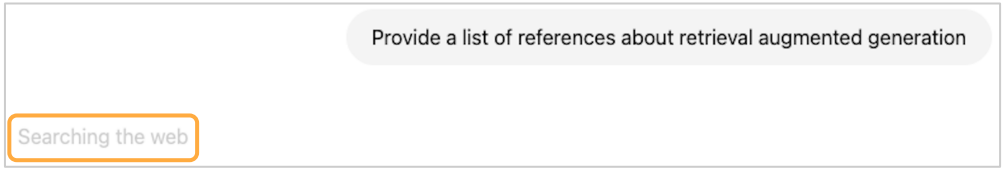


For more on prompting:
https://www.promptingguide.ai/

Other techniques to improve LLMs in practice: **RAG**

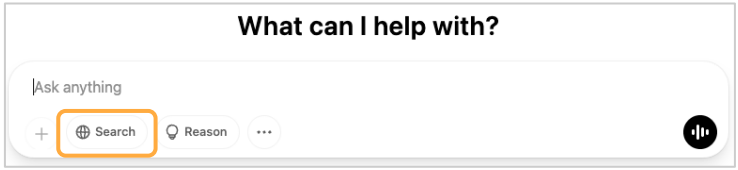
Retrieval Augmented Generation (RAG): combines natural language generation and information retrieval to ensure LLMs are better grounded by external up-to-date knowledge sources

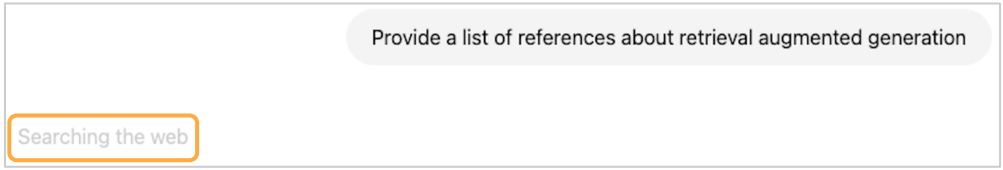


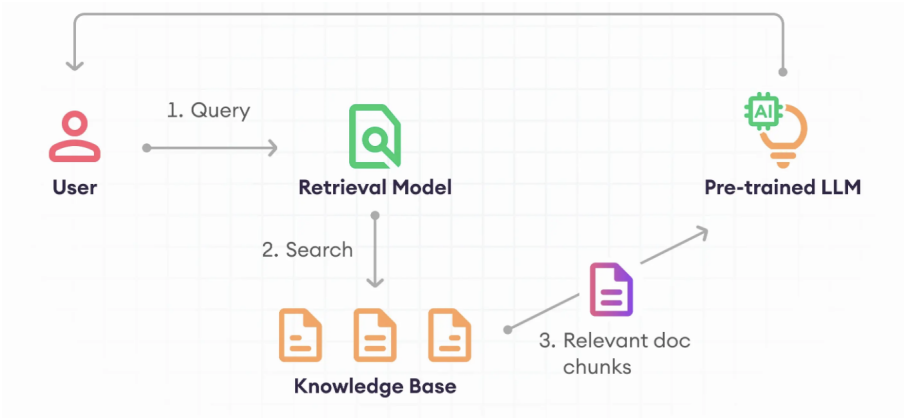


Other techniques to improve LLMs in practice: **RAG**

Retrieval Augmented Generation (RAG): combines natural language generation and information retrieval to ensure LLMs are better grounded by external up-to-date knowledge sources

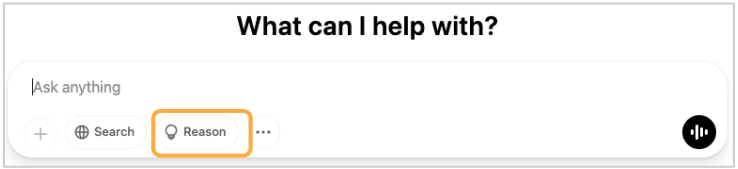






Other techniques to improve LLMs in practice: **Reasoning**

Reasoning: the latest "frontier" of LLM research – how do we teach LLMs to reason logically and think about the intermediate steps to arrive at an answer/ response



Other techniques to improve LLMs in practice: **Reasoning**

Reasoning: the latest "frontier" of LLM research – how do we teach LLMs to reason logically and think about the intermediate steps to arrive at an answer/ response

- **Chain-of-thought prompting:** prompt the model with "Let's think step-by-step..."

Example:

> *Question:* If there are 3 red balls and 2 blue balls, and you take one out randomly, what is the probability it is red?
> *Prompt:* Let's think step-by-step...

Examples from ChatGPT

Other techniques to improve LLMs in practice: **Reasoning**

Reasoning: the latest "frontier" of LLM research – how do we teach LLMs to reason logically and think about the intermediate steps to arrive at an answer/ response

- **Chain-of-thought prompting:** prompt the model with "Let's think step-by-step..."
- **Few-shot prompting:** prompt the model with a few correct examples

Example:

vbnet

Copy Edit

```
Q: Tom is taller than Jerry. Jerry is taller than Max. Who is the tallest?
A: Let's think step-by-step...
Tom > Jerry > Max → Tom is the tallest.

Q: Sarah is older than Emma. Emma is older than Olivia. Who is the oldest?
A: Let's think step-by-step...
Sarah > Emma > Olivia → Sarah is the oldest.

Q: Alan is faster than Brian. Brian is faster than Carl. Who is the fastest?
A: Let's think step-by-step...
```

Examples from ChatGPT

Other techniques to improve LLMs in practice: **Reasoning**

Reasoning: the latest "frontier" of LLM research – how do we teach LLMs to reason logically and think about the intermediate steps to arrive at an answer/ response

- **Chain-of-thought prompting:** prompt the model with "Let's think step-by-step..."
- **Few-shot prompting:** prompt the model with a few correct examples
- **Self-consistency:** ask the model multiple times (using temperature > 0) and pick the most common conclusion

Examples from ChatGPT

Other techniques to improve LLMs in practice: **Reasoning**

Reasoning: the latest "frontier" of LLM research – how do we teach LLMs to reason logically and think about the intermediate steps to arrive at an answer/ response

- **Chain-of-thought prompting:** prompt the model with "Let's think step-by-step..."
- **Few-shot prompting:** prompt the model with a few correct examples
- **Self-consistency:** ask the model multiple times (using temperature > 0) and pick the most common conclusion
- **ReAct** (reasoning + acting): let LLMs alternate between thinking and using tools (e.g., calculator, search)

Example:

> *"To answer this, I need to know the capital of Albania. Let me look it up." → Uses tool → Returns "Tirana"*

Examples from ChatGPT

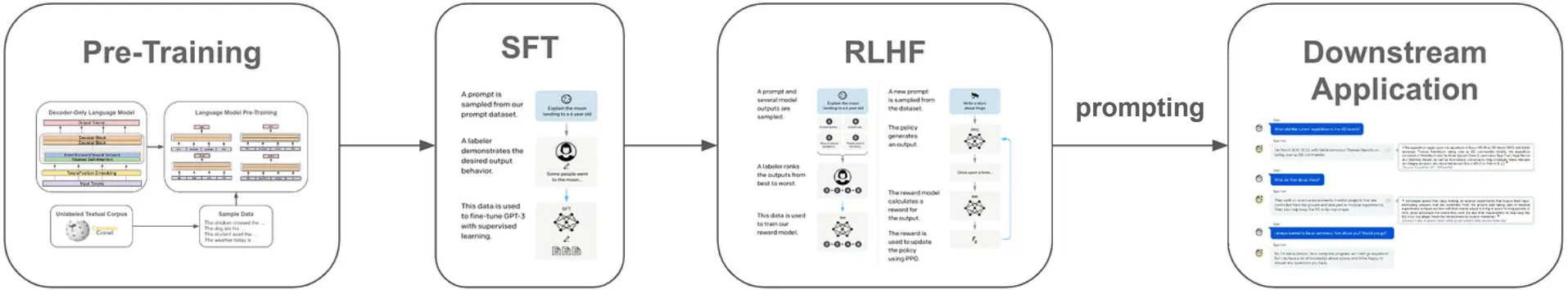
Other techniques to improve LLMs in practice: **Reasoning**

Reasoning: the latest "frontier" of LLM research – how do we teach LLMs to reason logically and think about the intermediate steps to arrive at an answer/ response

- **Chain-of-thought prompting:** prompt the model with "Let's think step-by-step..."
- **Few-shot prompting:** prompt the model with a few correct examples
- **Self-consistency:** ask the model multiple times (using temperature > 0) and pick the most common conclusion
- **ReAct** (reasoning + acting): let LLMs alternate between thinking and using tools (e.g., calculator, search)
- **Socratic Questioning:** ask the model to ask itself questions, critique its own answer, or generate a counter-argument as it reasons

Examples from ChatGPT

A basic development pipeline using LLMs



What does this look like in practice?

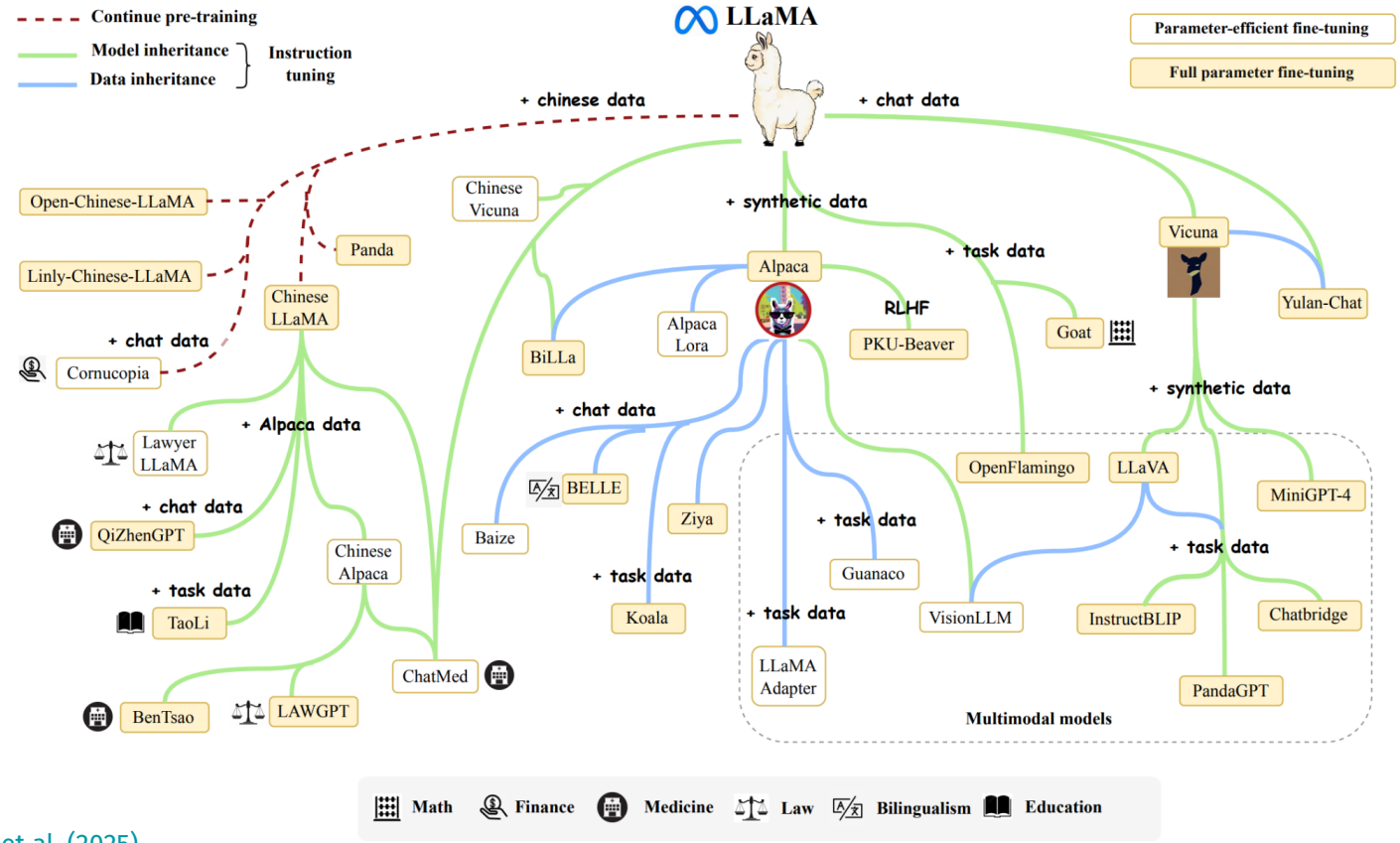


Figure Source: [Zhao et al. (2025)]