CS 1671 / CS 2071 / ISSP 2071
Human Language Technologies

Session 19: Post-training LLMs

Michael Miller Yoder

March 25, 2026

University of Pittsburgh | School of Computing and Information

Assessments: project

- Project progress report is due **this Thu Mar 26**
- Part 1: Basic data analysis for **final dataset you are using for your project**
- Part 2: Result from baseline approach
    - Ideally performance metric result from the baseline system you proposed
- Part 3: LLM proposal
    - How might you use an LLM programmatically to attempt your task?
    - Zero-shot and more advanced approaches (few-shot prompting, chain-of-thought prompting, instruction tuning)

Assessments: homework and exam

- Homework 3 will be released on Friday
    - Instruction finetuning
- There will be an exam in class on **Wed Apr 8**
    - Covers everything in the class up to that point
    - In-class, paper exam
    - Conceptual
    - No programming
    - Class on Mon Apr 6 will be a review session for the exam

Cool new class offered Fall 2026: LING 1880

Interested in getting hands-on experience with the data behind modern AI?

Check out LING 1800/2800: Special Topics — Linguistics in NLP and LLMs: Language Data, Annotation, and Evaluation

Prerequisite: LING 1000 (Introduction to Linguistics) or equivalent.

Explores how linguistic knowledge powers contemporary language technologies. In this course, you'll learn how language is structured as data, how datasets are created and annotated, and how models are evaluated for quality, bias, and effectiveness.

Topics include:

- Building and analyzing language datasets

- Annotation practices and challenges

- Evaluating NLP systems and large language models

- Ethical considerations in language technology

If you've ever wondered how linguistics shapes the future of AI (or how you could be part of it), this course is for you!

Instructor: Janet Liu

Review: transformer input and output

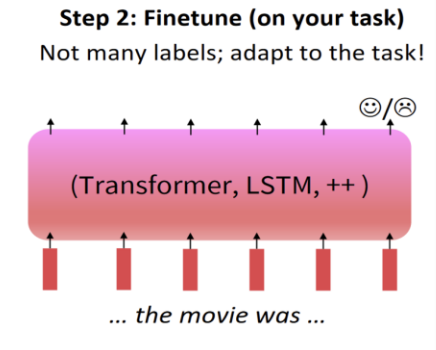
1. How is word order incorporated in transformers?
2. What does a language modeling head do in transformers?

Overview: Post-training LLMs

- Adapting LLMs to your use case
- Post-training and model alignment for LLMs
- Instruction tuning
- Preference-based learning
    - RLHF and DPO
- Coding activity: continued pretraining (finetuning) of GPT-2

Two ways of adapting an LLM to your use case

1. Finetune the parameters of the model with additional data
    - See parameter-efficient finetuning for finetuning very large models



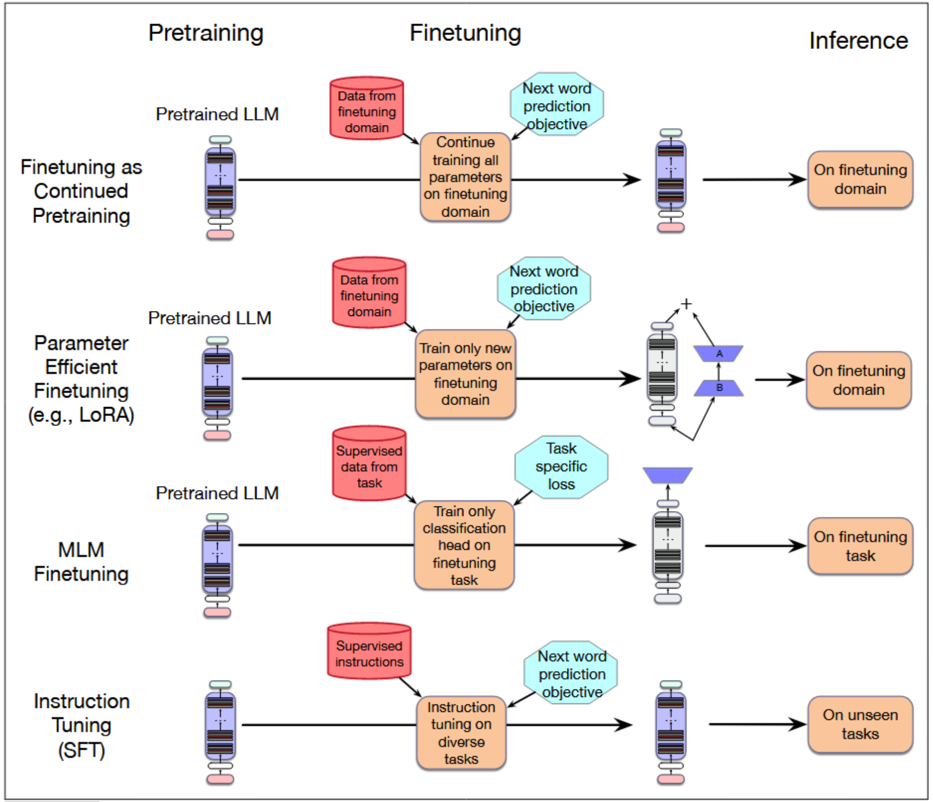


Figure 9.1 Instruction tuning compared to the other kinds of finetuning.

Figures from John Hewitt, Jurafsky and Martin

Two ways of adapting an LLM to your use case

2. Add more information in your prompt to the LLM, such as demonstrations (in-context learning)
   - In-context learning (few-shot prompting): give a few examples of output you want from the LLM
   - Chain-of-thought prompting: guide the LLM through a series of prompts that include intermediate reasoning

Post-training and model alignment for LLMs: language modeling is not enough

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.
```

- Language models are not aligned with user intent [Ouyang et al. 2022]
- Post-training LLMs to the rescue!

Slide adapted from Jesse Mu

Language model as multitask assistants?

- How do we get from
  - University of Pittsburgh is located in ___, Pennsylvania. [World knowledge]
  - I put __ fork down on the table. [syntax]
- To...

Slide adapted from Stanford Natural Language Processing with Deep Learning course

Language model as world models?

Language models may do rudimentary modeling of agents, beliefs, and actions.

> ***Pat watches a demonstration of a bowling ball and a leaf being dropped at the same time in a vacuum chamber. Pat, who is a physicist, predicts that*** *the bowling ball and the leaf will fall at the same rate.*

Changing the last sentence of the prompt, we get:

> ***…Pat, who has never seen this demonstration before, predicts that*** *the bowling ball will fall to the ground first. This is incorrect. In a vacuum chamber, there is no air*

Slide adapted from Stanford Natural Language Processing with Deep Learning course

"[Language Models as Agent Models]" Andreas 2022

Language model as world models?

Language models may do mathematical proofs if pretrained on math content.

Input:
Let $f(r) = \sum_{j=2}^{2008} \frac{1}{j^r} = \frac{1}{2^r} + \frac{1}{3^r} + \cdots + \frac{1}{2008^r}$. Find $\sum_{k=2}^{\infty} f(k)$.

Llemma 34B solution:
We have

$$\begin{aligned}
\sum_{k=2}^{\infty} f(k) &= \sum_{k=2}^{\infty} \sum_{j=2}^{2008} \frac{1}{j^k} = \sum_{j=2}^{2008} \sum_{k=2}^{\infty} \frac{1}{j^k} = \sum_{j=2}^{2008} \frac{1}{j^2} \sum_{k=0}^{\infty} \frac{1}{j^k} = \sum_{j=2}^{2008} \frac{1}{j^2} \frac{1}{1-\frac{1}{j}} \\
&= \sum_{j=2}^{2008} \frac{1}{j(j-1)} = \sum_{j=2}^{2008} \left(\frac{1}{j-1} - \frac{1}{j}\right) \\
&= \left(\frac{1}{1} - \frac{1}{2}\right) + \left(\frac{1}{2} - \frac{1}{3}\right) + \cdots + \left(\frac{1}{2007} - \frac{1}{2008}\right) \\
&= 1 - \frac{1}{2008} \\
&= \boxed{\frac{2007}{2008}}.
\end{aligned}$$

Final Answer: The final answer is $\frac{2007}{2008}$.

Slide adapted from Stanford Natural Language Processing with Deep Learning course

"[Llemma: An Open Language Model For Mathematics](#)" Azebayev et al 2023

Language model as world models?

Language models can generate working code.

Slide adapted from Stanford Natural Language Processing with Deep Learning course

[Copilot](#)

Language model as multitask assistants?

I need to throw a dinner party for 6 people who are vegetarian. Can you suggest a 3-course menu with a chocolate dessert?

[Microsoft Bing]

(Also see OpenAI's ChatGPT, Google's Bard, Anthropic's Claude)

Sure, I can suggest a 3-course menu with a chocolate dessert for your vegetarian dinner party. Here are some possible options:

- Starter: **Wild Mushroom Tartlets with Onion Sauce** [1] or **Vegan Popcorn Tofu Nuggets** [2]
- Main: **Vegan Butternut Squash Mac** [2] or **Vegetarian Three-Bean Chili** [3]
- Dessert: **Chocolate Lava Cake** [4] or **Chocolate Pasta with Chocolate Hazelnut Cream Sauce, White Chocolate Shavings, and Fresh Berries** [4]

I hope this helps you plan your party. Enjoy! 😋

Learn more: 1. booths.co.uk +10 more

Slide adapted from Stanford Natural Language Processing with Deep Learning course

Post-training (model alignment)

Two techniques to align LLMs with human preferences (what we want them to do):

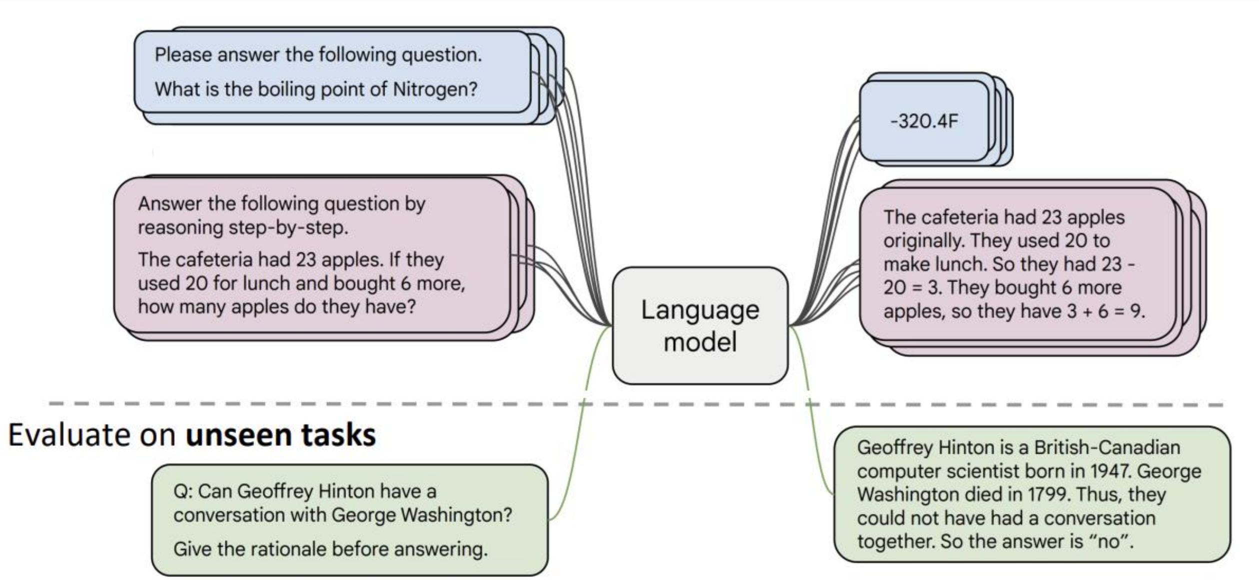
1. Instruction tuning
    - Models are finetuned on a corpus of instructions/questions and desired responses
2. Preference alignment (RLHF)
    - Separate model is trained to decide how much a candidate response aligns with human preferences
    - This reward model is used to finetune the base model

Slide based on Jurafsky & Martin

Instruction tuning

Instruction tuning

- AKA instruction finetuning, supervised finetuning, SFT
- Collect examples of (instruction, output) pairs across many tasks and finetune an LM
- Still just LM objective (predict the next word)



Slide adapted from Jesse Mu, Chung et al. 2022

Limitations of instruction finetuning

- Expensive to collect ground-truth data for tasks
  - Though you can include existing datasets of tasks like question answering
  - And LLMs are now commonly used to generate instruction tuning datasets
- Tasks like open-ended creative generation have no right answer.
  - Write me a story about a dog and her pet grasshopper.
- Language modeling penalizes all token-level mistakes equally, but some errors are worse than others
- Even with instruction finetuning, there is a mismatch between the LM objective and the objective of "satisfy human preferences"!
- Can we **explicitly attempt to satisfy human preferences?**

Slide adapted from Jesse Mu, Jurafsky & Martin

Preference-based learning

Preference-based alignment

A typical LLM development flow

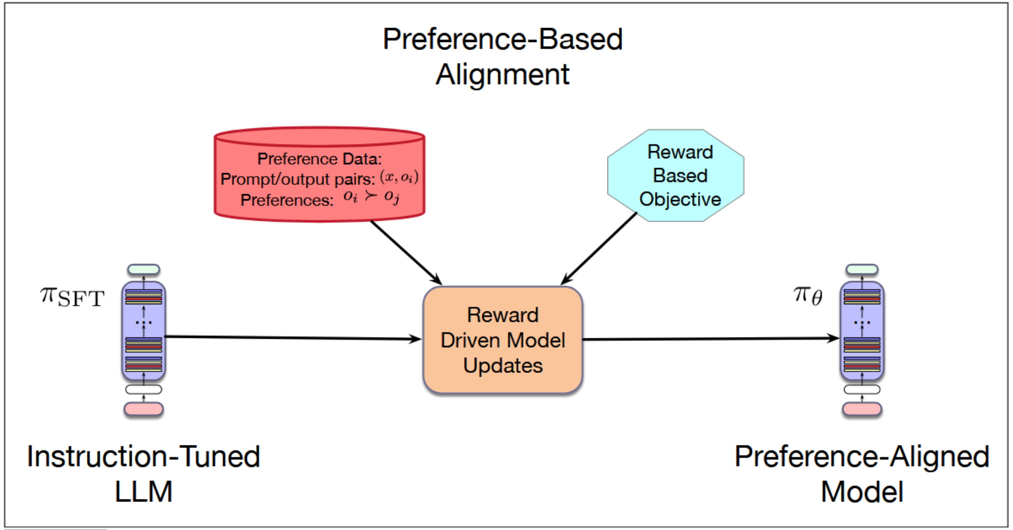


Figure 9.8 Preference-based model alignment.

Figures from Sebastian Raschka, Jurafsky and Martin

How do we model human preferences? Preference data

- Ask annotators to rank different LLM responses by their preference
- Example: summarize an article

```
SAN FRANCISCO,
California (CNN) --
A magnitude 4.2
earthquake shook the
San Francisco
...
```

```
An earthquake hit         A 4.2 magnitude          The Bay Area has
San Francisco.            earthquake hit           good weather but is
There was minor     >     San Francisco,     >     prone to
property damage,          resulting in             earthquakes and
but no injuries.          massive damage.          wildfires.
```


Slide adapted from Jesse Mu

Preference data

- Prompts (can come from real users of OpenAI's LLMs, e.g.)
- LLM-generated responses to those prompts, ranked by human annotators

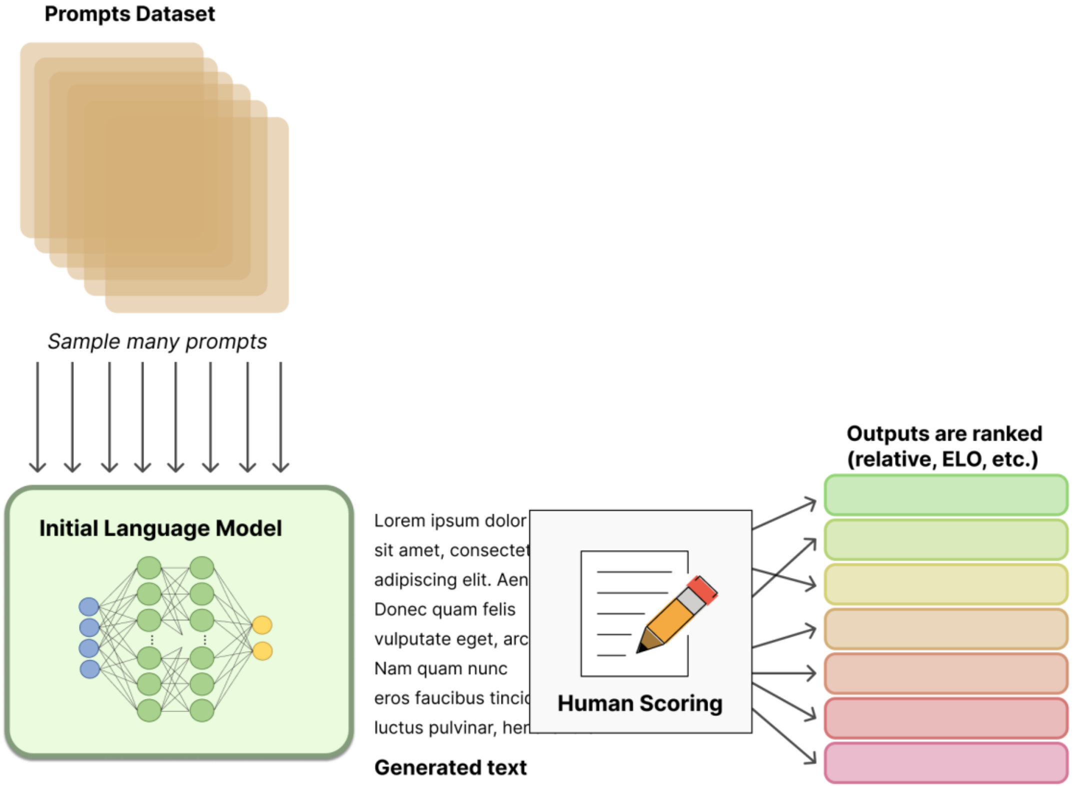


Figure adapted from https://huggingface.co/blog/rlhf

Optimizing LLMs with preference data

- Preference-based learning often uses terms from reinforcement learning:
  - action: choosing tokens to generate
  - policy: an LLM that chooses the next word (a response)
  - reward: producing a response that matches human preferences
- We want to finetune the parameters of an LLM (predicting the next word) to maximize the expected “reward” for the entire output
- Try to not stray too far from the original (pretrained, instruction tuned) language model

Slide adapted from Jesse Mu, https://huggingface.co/blog/rlhf

Algorithms for preference-based learning

- RLHF with PPO
  - Classic, used by LLMs like InstructGPT and original ChatGPT
  - Trains an explicit "reward model" that produces an estimate of human preference for any input text
- DPO
  - Doesn't use reinforcement learning
  - Newer (2023), more efficient
  - Doesn't train an explicit reward model

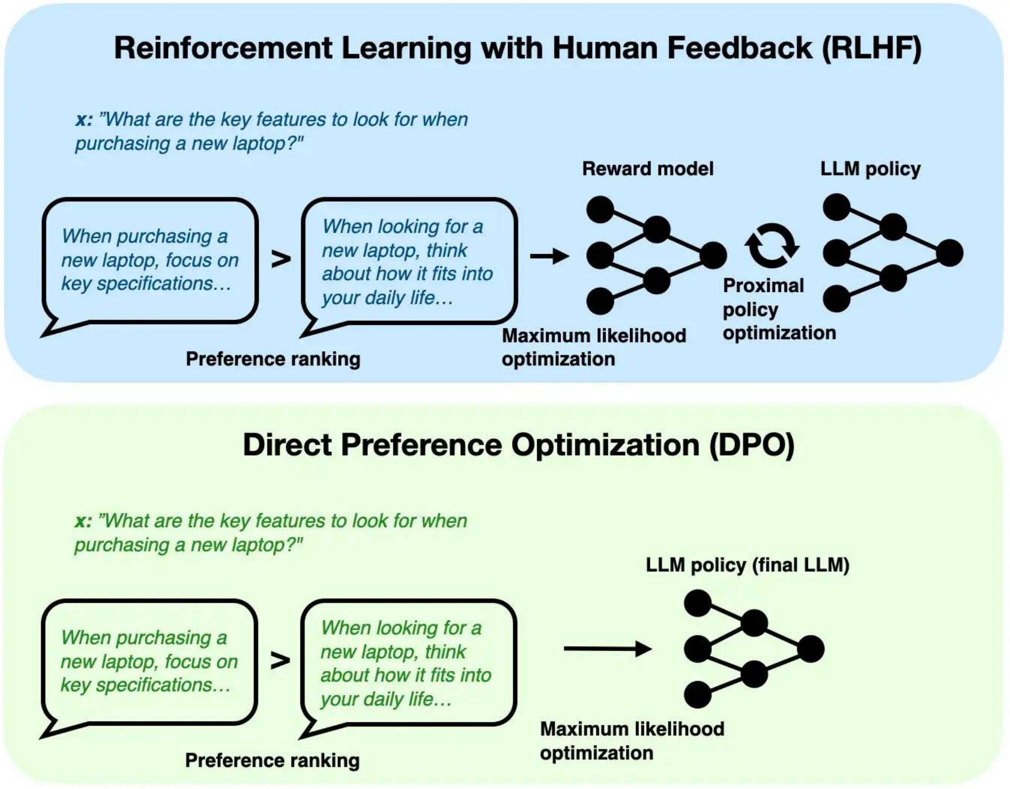


Figure from Sebastian Raschka

Finetuning LLMs with Direct Policy Optimization (DPO)

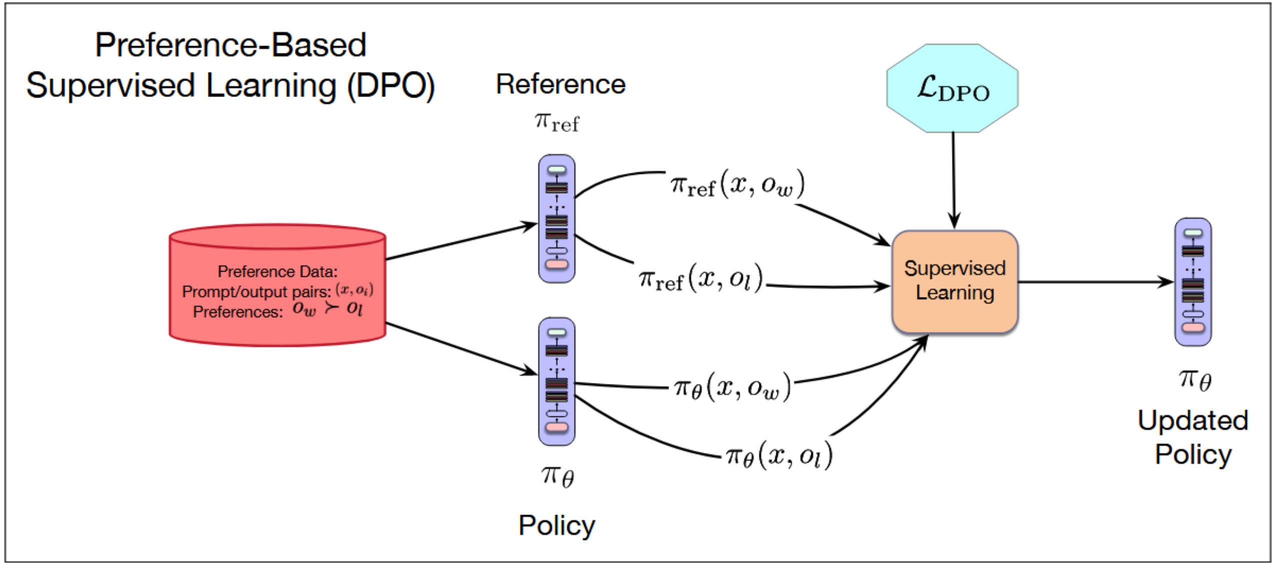


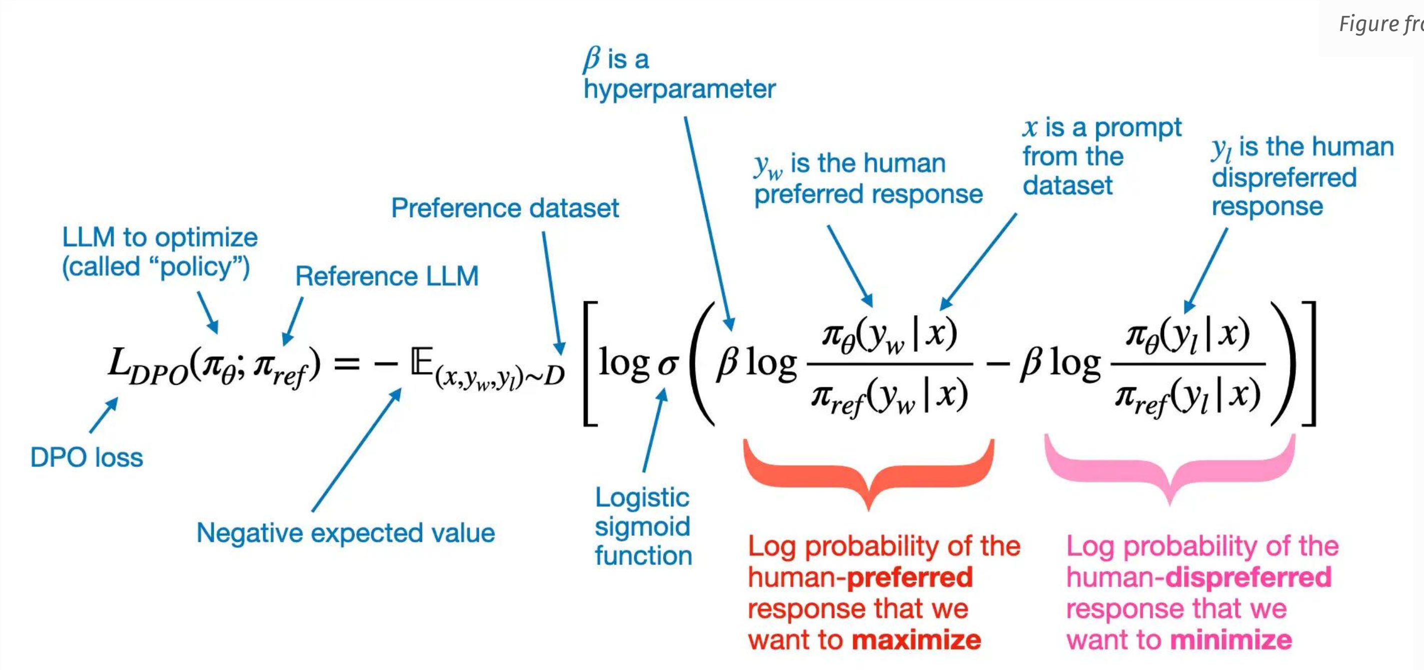
Figure 9.9 Preference-based alignment with Direct Preference Optimization.

Calculates the LLM probabilities of the "winning" and "losing" responses from the reference model and the policy model that we are optimizing (finetuning)

Figure from Sebastian Raschka

Loss function for Direct Policy Optimization (DPO)

Figure from Sebastian Raschka



- Loss will be high if the probability of the dispreferred response is larger than the preferred response
- Penalizes model for differing a lot from the reference model

InstructGPT gets us closer to dialogue intent

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.
```

Slide adapted from Jesse Mu

InstructGPT can accomplish many tasks

PROMPT	*Write a short poem about a wise frog.*

COMPLETION

GPT-3

```
Write a short story in which a character has two different names.

Write a short story in which you try to get something back that you
have lost.

Write a short story in which a character has a bad dream.
```

InstructGPT

```
The frog is a symbol of wisdom
He knows all the secrets of the world
He is a master of disguise
And a great teacher of life
He is a symbol of transformation
And the bringer of change
He is the frog who has seen it all
And knows the meaning of it all
```

Slide adapted from Jesse Mu

ChatGPT: Instruction Finetuning + RLHF for dialog agents

ChatGPT: Optimizing Language Models for Dialogue

Note: OpenAI (and similar companies) are keeping more details secret about ChatGPT training (including data, training parameters, model size)—perhaps to keep a competitive edge…

Methods

We trained this model using Reinforcement Learning from Human Feedback (RLHF), using the same methods as InstructGPT, but with slight differences in the data collection setup. We trained an initial model using supervised fine-tuning: human AI trainers provided conversations in which they played both sides—the user and an AI assistant. We gave the trainers access to model-written suggestions to help them compose their responses. We mixed this new dialogue dataset with the InstructGPT dataset, which we transformed into a dialogue format.

(Instruction finetuning!)

https://openai.com/blog/chatgpt/

Slide adapted from Jesse Mu

ChatGPT: Instruction Finetuning + RLHF for dialog agents

ChatGPT: Optimizing Language Models for Dialogue

Note: OpenAI (and similar companies) are keeping more details secret about ChatGPT training (including data, training parameters, model size)—perhaps to keep a competitive edge…

Methods

To create a reward model for reinforcement learning, we needed to collect comparison data, which consisted of two or more model responses ranked by quality. To collect this data, we took conversations that AI trainers had with the chatbot. We randomly selected a model-written message, sampled several alternative completions, and had AI trainers rank them. Using these reward models, we can fine-tune the model using Proximal Policy Optimization. We performed several iterations of this process.

(RLHF!)

https://openai.com/blog/chatgpt/

Slide adapted from Jesse Mu

Limitations of Reward Modeling

- Human preferences are unreliable!
- "Reward hacking" is a common problem in RL
- Chatbots are rewarded to produce responses that seem authoritative and helpful, regardless of truth
- This can result in making up facts + hallucinations

TECHNOLOGY

Google shares drop $100 billion after its new AI chatbot makes a mistake

February 9, 2023 · 10:15 AM ET

https://www.npr.org/2023/02/09/1155650909/google-chatbot--error-bard-shares

Bing AI hallucinates the Super Bowl

https://news.ycombinator.com/item?id=34776508
https://apnews.com/article/kansas-city-chiefs-philadelphia-eagles-technology-science-82bc20f207e3e4cf81abc6a5d9e6b23a

Slide adapted from Jesse Mu

Conclusion

- Instruction tuning, finetuning of LLMs with prompt-response pairs, can help align language models with human preferences
- Large language models can be trained to provide more useful responses using preference-based learning
    - Types: Reinforcement learning from human feedback (RLHF), Direct policy optimization (DPO)
    - Finetune LLM parameters to prefer producing responses that humans preferred in preference training data
    - Don’t change parameters too much from the initial (like instruction tuned) LLM

Coding activity

Notebook: finetune GPT-2 on Shakespeare

1. Click this nbgitpuller link (also available on course website)
2. **Important difference from normal:** Start a server with '**TEACH – Nvidia L4 GPU – 16 CPUs – 60GB**' server
3. Load custom environment at **/ix1/cs1671-2026s/class_env**
4. Open **session19_gpt2_shakespeare.ipynb**