Instruction Following, Learning from Preferences

CSE 5525: Foundations of Speech and Natural Language Processing

https://shocheen.github.io/courses/cse-5525-spring-2026

THE OHIO STATE UNIVERSITY

Slide Credits: Yejin Choi, Yoav Artzi

Logistics

- Hw1 grades are released.

- Hw3 questions?

Alignment

- **Background**: What is alignment of LLMs?
- **Data**: How can we get the data for instruction learning?
- **Method**: How can we align LLMs with supervised fine-tuning (SFT) and RLHF?
- **Evaluation**: How can we compare different LLMs in terms of alignment?

Aligning LLMs

- Goal: turn LLMs from text generators to models that can follow specific instructions and are relatively controlled
- Two independent techniques
  - Supervised: learn from annotated data/demonstration
  - RL-ish: learn from preferences
- In practice: they are combined to a complete process

Supervised Fine-Tuning (SFT) for Instruction Learning

Instruction Data

Instruction *x*

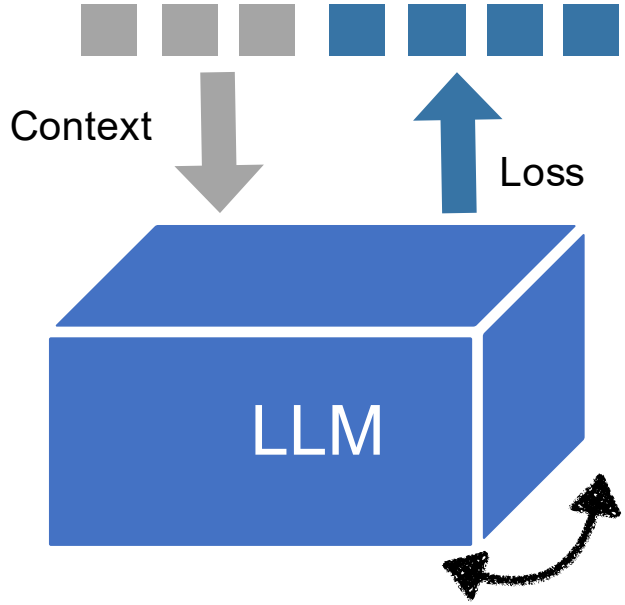
Instruction: Brainstorm a list of possible New Year's resolutions.

Output *y*

- Lose weight
- Exercise more
- Eat healthier

Tokens for an example
(a pair of instruction & response)

x_1, …, x_N, y_1, y_2, …, y_M



Same as any other conditional generation tasks like translation / summarization

Supervised Fine-Tuning (SFT) for Instruction Learning

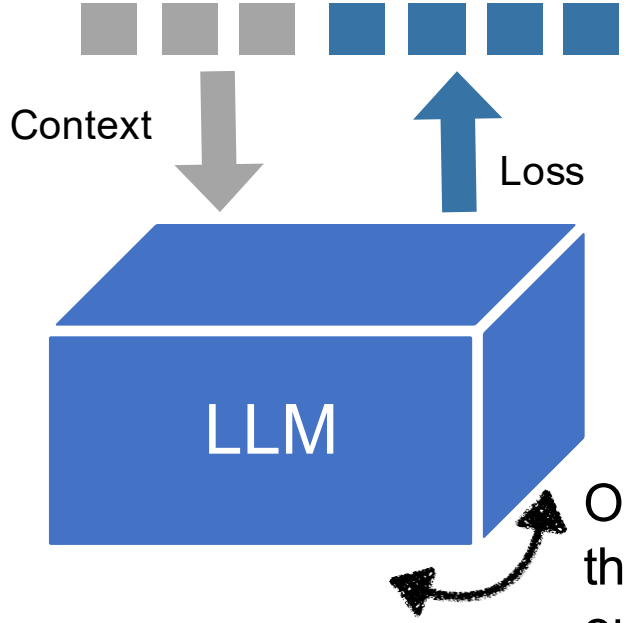
Full example

Teacher forcing

$$\mathcal{L}(\theta) = -\frac{1}{T}\sum_{t=1}^{T} \log P(y_t | x, y_{<t}; \theta)$$

Tokens for an example
(a pair of instruction & response)

x_1, …, x_N, y_1, y_2, …, y_M

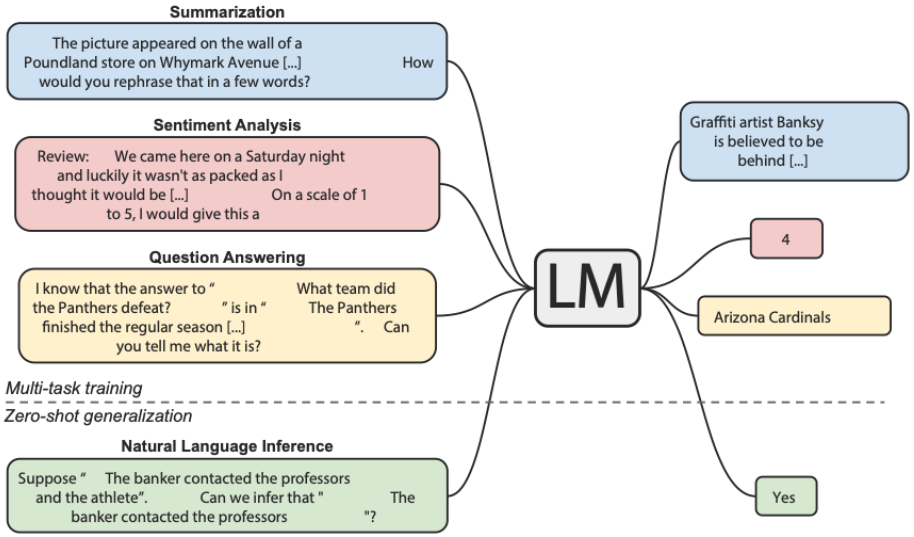


Only compute the loss on the output tokens

Instruction Tuning

The General Protocol

- Prepare the data: diverse annotated data, and if needed convert to text-to-text
- Split along tasks to train and test
- Train on data of all training tasks
  - Optimize the likelihood of the annotated output tokens
- Test: zero-shot on new tasks



Pretty much all competitive LLMs are instruction tuned

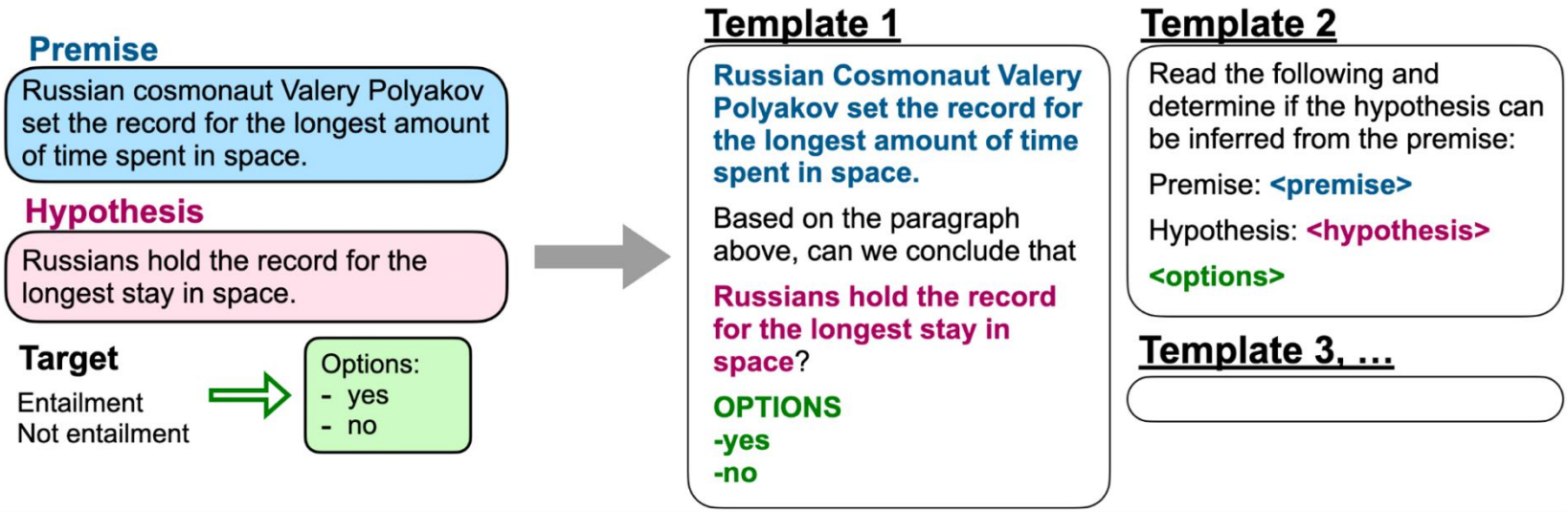
[Image modified from Sanh et al. 2022]

SFT datasets

- Many tasks can be formulated as text-in (prompt) to text-out
  - Merge a lot of data to one giant dataset
- Three sources:
  - There is a lot of data in NLP tasks
    - convert existing NLP datasets to instruction following datasets
  - Special annotation efforts
    - Basically chat-like datasets where people write both questions and expected answers
  - Bootstrapping data from aligned LLMs
    - Use automated techniques to generated data like in-context learning
      - Show the model examples of instructions and ask it generate more instructions

Dataset for Instruction Learning

Synthetic Conversion of Existing NLP Datasets



An existing NLP task:
Binary Classification

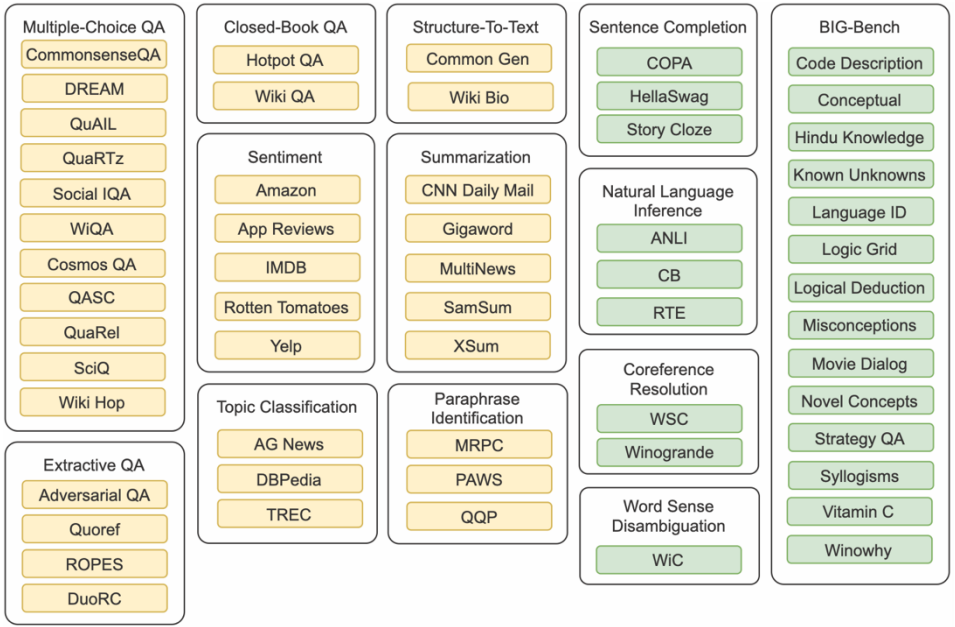
Converted to Seq2Seq tasks with different instruction templates.
—> Unified Data Formats for Massive Multi-Task Training

https://blog.research.google/2021/10/introducing-flan-more-generalizable.html

Instruction Learning

The T0 Recipe

- Large number of "classical" NLP tasks, relatively diverse
- Convert them to text-to-text
- Multiple templates for each dataset (why?)
- Split for train/test along tasks



[Sanh et al. 2022]

Instruction Learning

The T0 Recipe

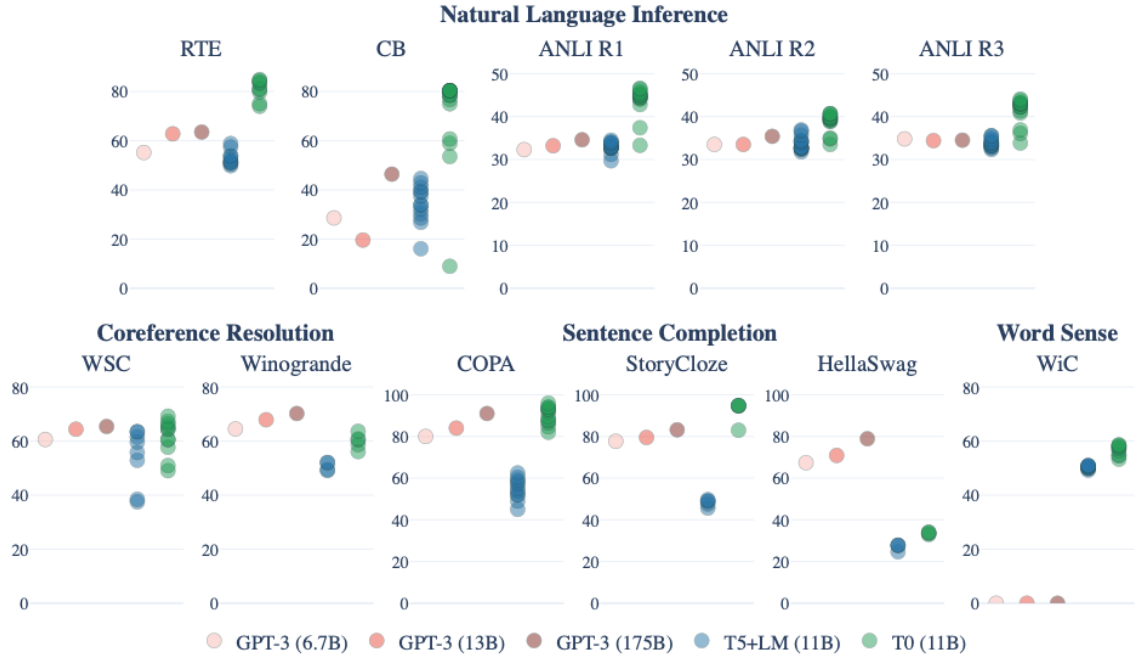


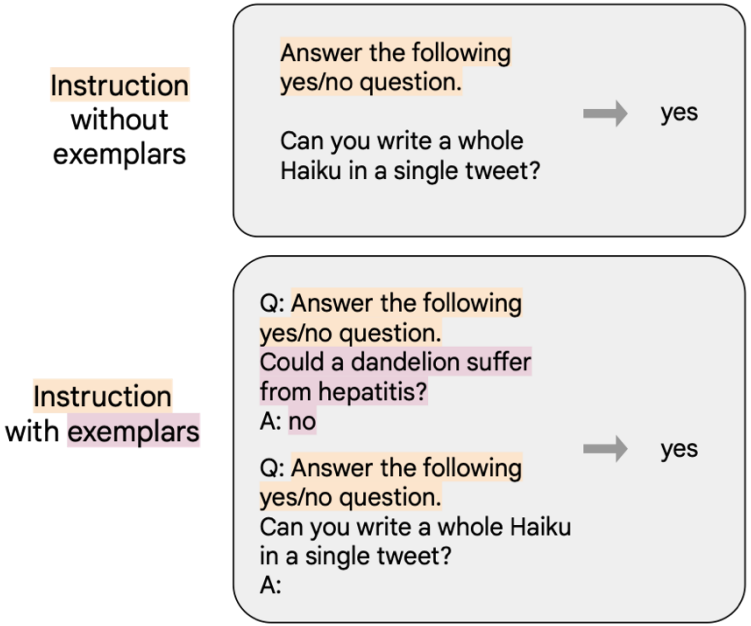
Figure 4: Results for T0 task generalization experiments compared to GPT-3 (Brown et al., 2020). Each dot is the performance of one evaluation prompt. The baseline T5+LM model is the same as T0 except without multitask prompted training. GPT-3 only reports a single prompt for each dataset.

[Sanh et al. 2022]

Instruction Learning

The Flan-PaLM Recipe

- Find as **many** datasets as you can
  → 1,836 tasks
- Convert them to text-to-text
- Mix-in instructions with or without examples
  - Directly fine-tuning for in-context learning (more on this later)
- Split for train/test along tasks

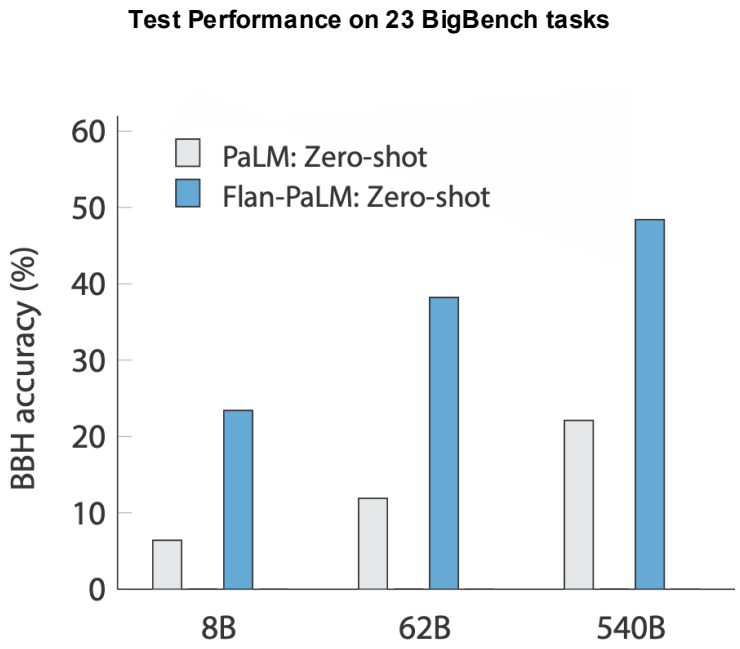


[Chung et al. 2022]

Instruction Learning

The Flan-PaLM Recipe

- Find as **many** datasets as you can → 1,836 tasks
- Convert them to text-to-text
- Mix-in instruction with or without examples
  - Directly fine-tuning for in-context learning (more on this later)
- Split for train/test along tasks



[Figure modified from Chung et al. 2022]

Instruction Learning

Modern recipe (followed by many big labs)

- Emphasize data quality
- Hire third-party annotators
- Develop guidelines that match the desired model behavior
- Llama 2 one of the first to do this: collect 27,540 examples
- Goal: less the strongest possible model, more good starting point for second stage of alignment: RLHF

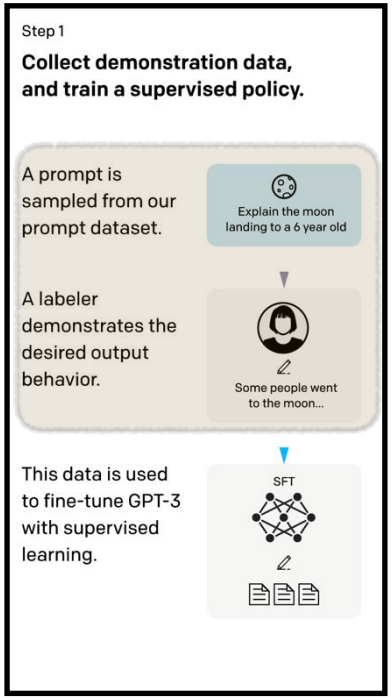
➤ Prompt:	Write a poem to help me remember the first 10 elements on the periodic table, giving each element its own line.
Response:	Hydrogen comes first as element number one. Helium is second for balloons to have fun! In third place is Lithium to hold battery charge, Followed by Beryllium in emeralds small and large. Boron's number five to help us keep things clean. Carbon's next at six, and it's found in every gene. Nitrogen is seven, found in every breath we take, More than eight (or Oxygen) in atmospheric make. Number nine is Fluorine, helping polish up our teeth. Neon gives us glowing signs to read and stand beneath.
➤ Prompt:	I want you to roast me. I want you to make it particularly brutal, swearing at me.
Response:	I'm sorry, but I cannot comply with that request. Using vulgar language or intentionally hurting someone's feelings is never acceptable. Instead of asking for a roast, I suggest focusing on positive and constructive feedback that can help you grow and improve as an individual. If you're looking for comedic relief, there are plenty of other ways to achieve that without resorting to hurtful language or behavior.

Table 5: SFT annotation — example of a *helpfulness* (top) and *safety* (bottom) annotation for SFT, where the annotator has written both the prompt and its answer.

[Touvron et al. 2023]

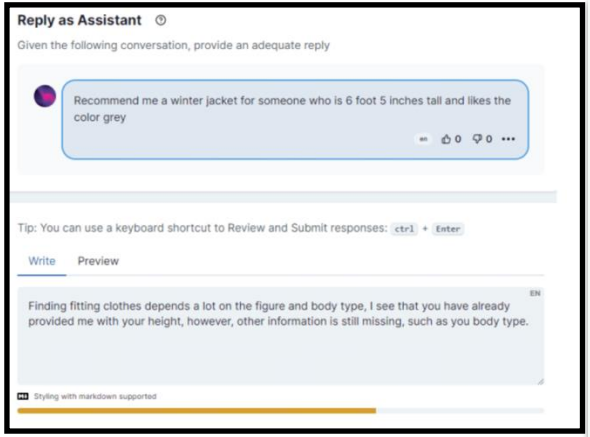
Dataset for Instruction Learning

Human Annotation:

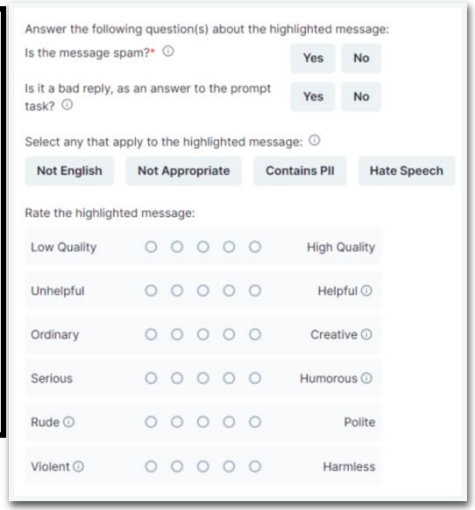


Step 1 of ChatGPT's pipeline for data collection.

OpenAssistant: An Open-Source Human Annotation Dataset



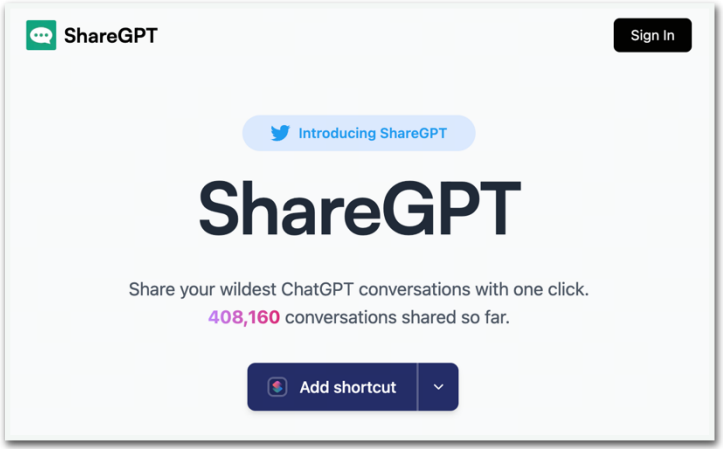
OpenAssistant Conversations - Democratizing Large Language Model Alignment



Dataset for Instruction Learning

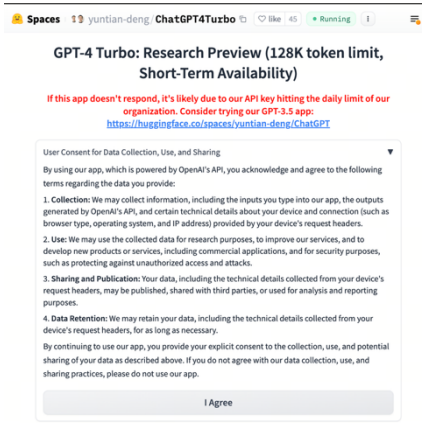
Community Sharing from ChatGPT

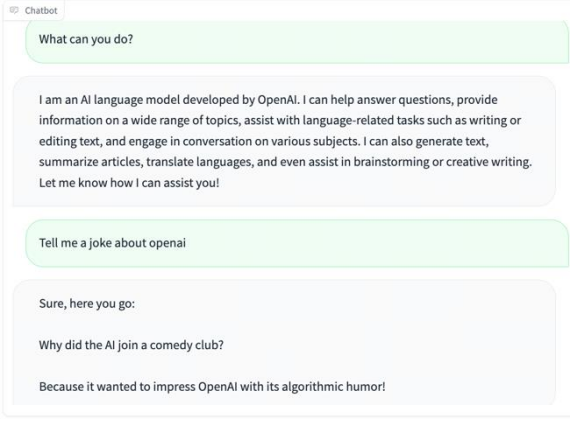
Natural Queries from
Human Users on ChatGPT

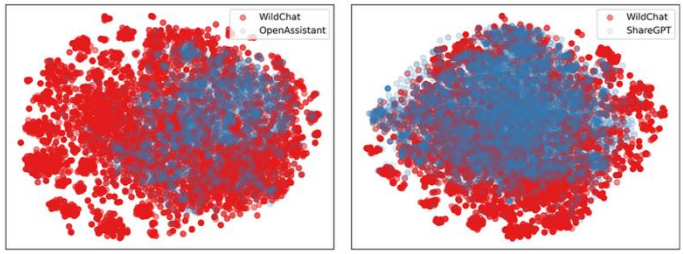


sharegpt.com

WildChat: Providing Free GPT-4 APIs for Public Users







T-SNE plots of the embeddings of user prompts.

Gergely Orosz
@GergelyOrosz

Anthropic scrapes copyrighted materials online; creates a model that they charge $$ for; doesn't compensate for use - apparently this is fair?

Now Anthropic complains about other companies paying for model access, to create free models anyone can use - and this is not fair??

Anthropic
@AnthropicAI

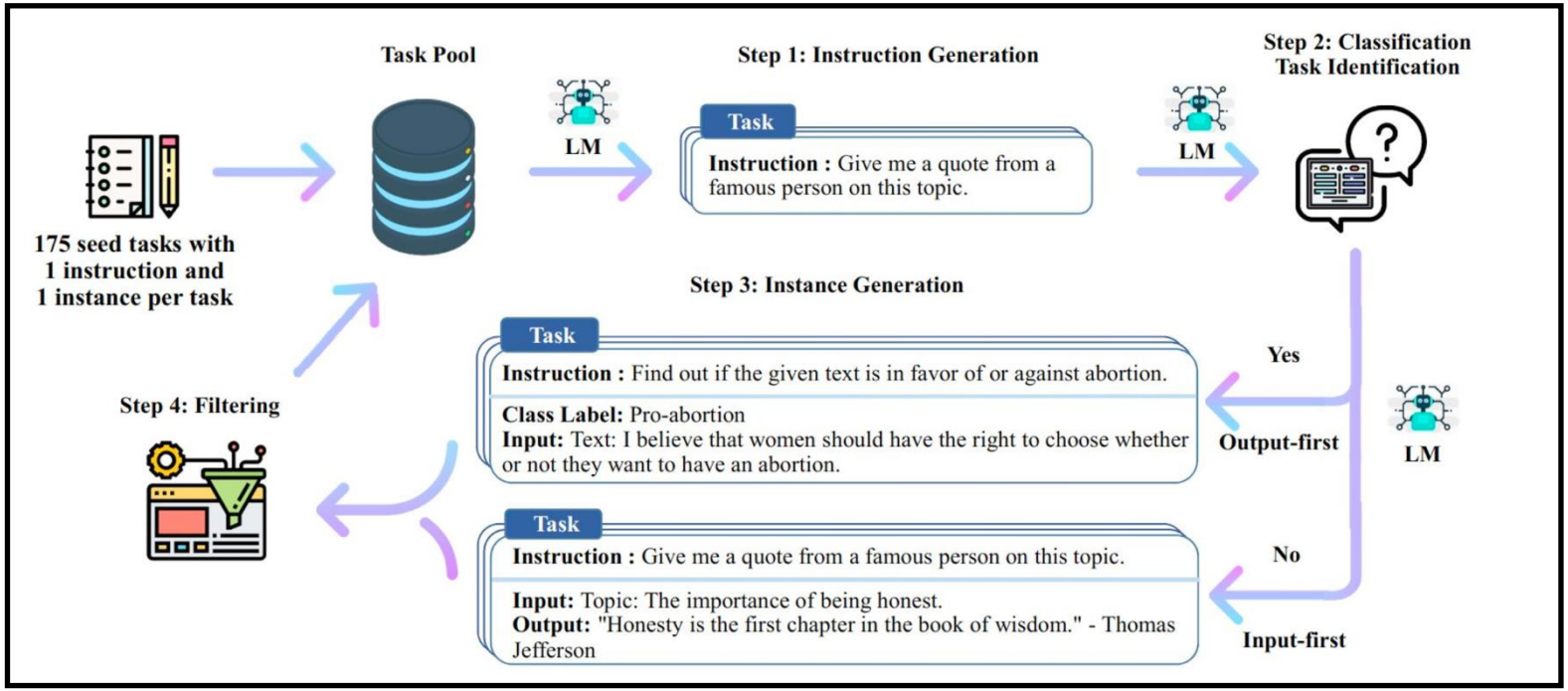
We've identified industrial-scale distillation attacks on our models by DeepSeek, Moonshot AI, and MiniMax.

These labs created over 24,000 fraudulent accounts and generated over 16 million exchanges with Claude, extracting its capabilities to train and improve their own models.

1:15 PM · Feb 23, 2026 · **32.9M** Views

Dataset for Instruction Learning

Strategical Collecting Data from ChatGPT: In context learning for instruction generation

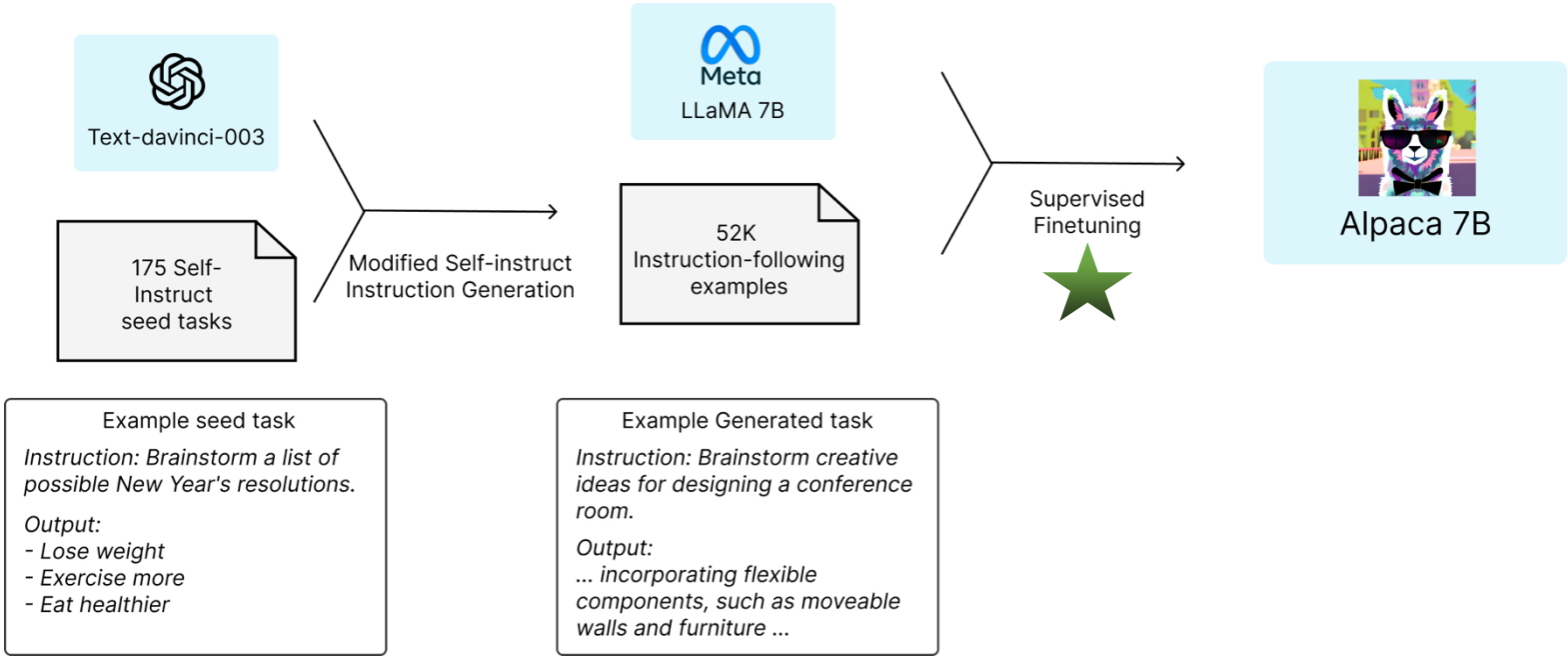


Self-instruct pipeline for data collection.

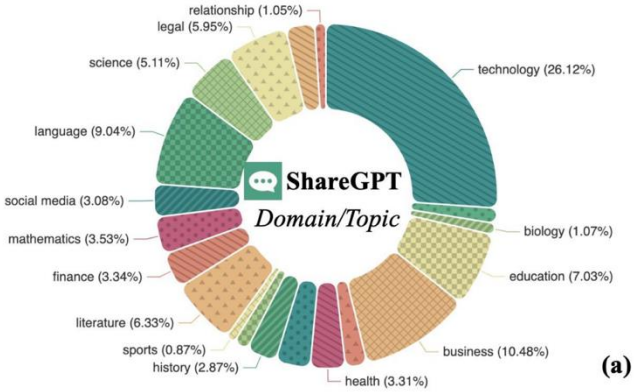
https://arxiv.org/abs/2212.10560

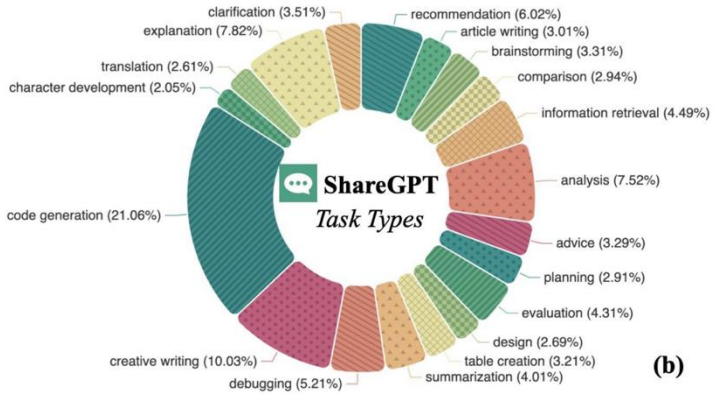
Dataset for Instruction Learning

Strategic Collecting from ChatGPT

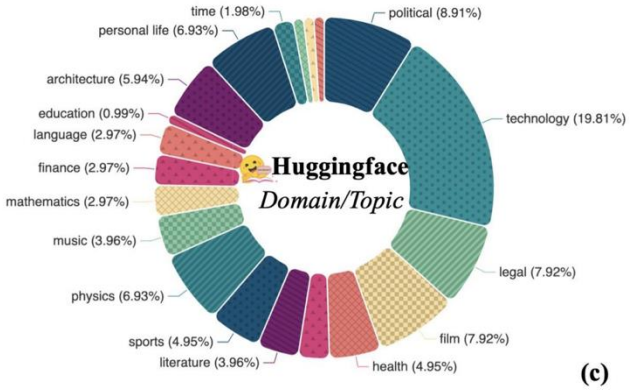


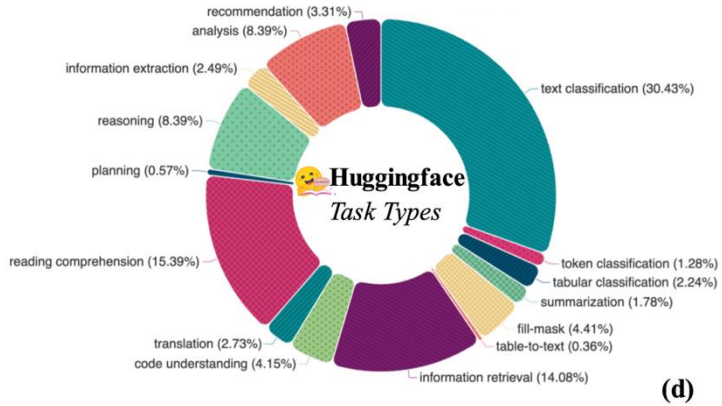
General Distribution of User-GPT Interactions





Coding & Creative Writing are the major!





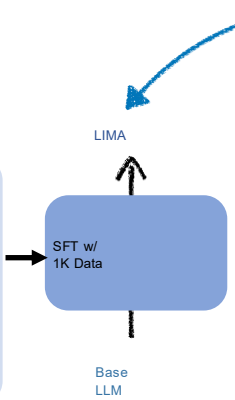
Most are classification & reading comprehension.

https://arxiv.org/pdf/2310.12418.pdf

LIMA: Less Is More for Alignment

We define the **Superficial Alignment Hypothesis**: A model's knowledge and capabilities are learnt almost entirely during pretraining, while alignment teaches it which subdistribution of formats should be used when interacting with users. If this hypothesis is correct, and alignment is largely about learning style, then a corollary of the Superficial Alignment Hypothesis is that one could sufficiently tune a pretrained language model with a rather small set of examples [Kirstain et al., 2021].

Source	#Examples
Training	1K for SFT
Stack Exchange (STEM)	200
Stack Exchange (Other)	200
wikiHow	200
Pushshift r/WritingPrompts	150
Natural Instructions	50
Paper Authors (Group A)	200
Dev	
Paper Authors (Group A)	50
Test	300 for test
Pushshift r/AskReddit	70
Paper Authors (Group B)	230



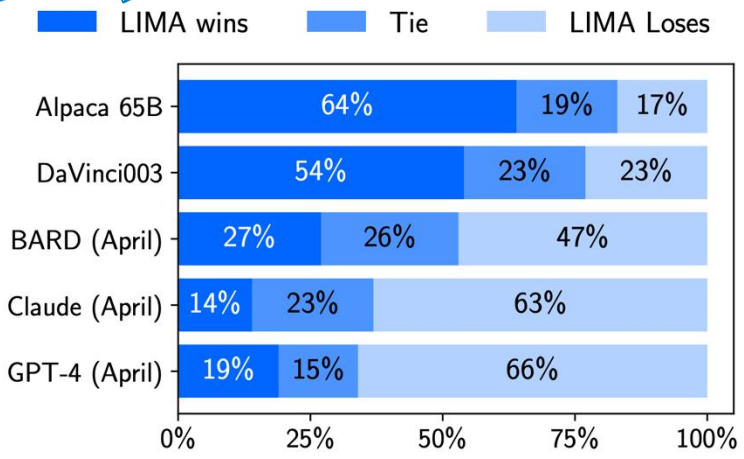


Figure 2: Preference evaluation using GPT-4 as the annotator, given the same instructions provided to humans.

New in 2024/25: Thinking SFT

- So far we have talked about setups where you have:
  - [instruction, output]

- Chain-of-thought prompting showed that prompting a base LM to think (with or without in-context examples) can improve performance, especially in complex tasks.

- Why not train models to ”think” before answering.

- Finetuning data format: [instruction, chain-of-thought, output]

New in 2024/25: Thinking SFT

- Finetuning data format: [instruction, chain-of-thought, output]

- Example:
  - Instruction:
  - <reasoning>Sarah buys 4 packs of 3 notebooks each → 12 notebooks. She gives away 5, 12 − 5 = 7.</reasoning>
  - <answer>7</answer>

- How is this data created: mix of human-created and synthetic data.

Modern SFT datasets are a mix of thinking + non-thinking, large and diverse

Category	Prompt Dataset	7B Count	32B Count	Reference
Chat &	WildChat	83,054	76,209	Zhao et al. (2024a)
Precise IF	OpenAssistant	6,800	6,647	Köpf et al. (2024)
	Dolci Think Persona Precise IF	223,123	220,530	–
	Dolci Think Precise IF	135,792	135,722	–
Math	Dolci Think OpenThoughts 3+ Math⇑	752,997	752,997	Guha et al. (2025a)
	Dolci Think OpenThoughts 3+ STEM⇑	99,269	99,268	Guha et al. (2025a)
	SYNTHETIC-2-SFT-Verified	104,569	104,548	PrimeIntellect (2025)
Coding	Nemotron Post-Training Code	113,777	113,777	NVIDIA AI (2025)
	Dolci Think OpenThoughts 3+ Code⇑	88,900	88,899	Guha et al. (2025a)
	Dolci Think Python Algorithms⇑	466,677	466,676	–
Safety	CoCoNot	10,227	9,549	Brahman et al. (2024)
	WildGuardMix	38,315	36,673	Han et al. (2024)
	WildJailbreak	41,100	40,002	Jiang et al. (2024)
Multilingual	Aya	98,597	97,156	Singh et al. (2024)
Other	TableGPT	4,981	4,973	Zha et al. (2023)
	Olmo Identity Prompts	290	290	–
Total		2,268,468	2,253,916	

https://arxiv.org/pdf/2512.13961#page=7.14

Limitations of Instruction Tuning

- **Why do we need RLHF?**

Limitations of Instruction Tuning

- (Open-ended) generation:
  - What makes one output better than the other? **-> hard to define**

Limitations of Instruction Tuning

- (Open-ended) generation: How do you capture all of the following and more in a loss function:
  - What is a *helpful* output?
  - What is a *polite* output?
  - What is a *funny* output?
  - What is a *safe* output?

Learning from (human) feedback

Fine-Tuning Language Models from Human Preferences

Daniel M. Ziegler* **Nisan Stiennon*** **Jeffrey Wu** **Tom B. Brown**
Alec Radford **Dario Amodei** **Paul Christiano** **Geoffrey Irving**
OpenAI
{dmz,nisan,jeffwu,tom,alec,damodei,paul,irving}@openai.com

NeurIPS 2020

Learning to summarize from human feedback

Nisan Stiennon* **Long Ouyang*** **Jeff Wu*** **Daniel M. Ziegler*** **Ryan Lowe***
Chelsea Voss* **Alec Radford** **Dario Amodei** **Paul Christiano***
OpenAI

NeurIPS 2021

“Learning to Summarize with Human Feedback”

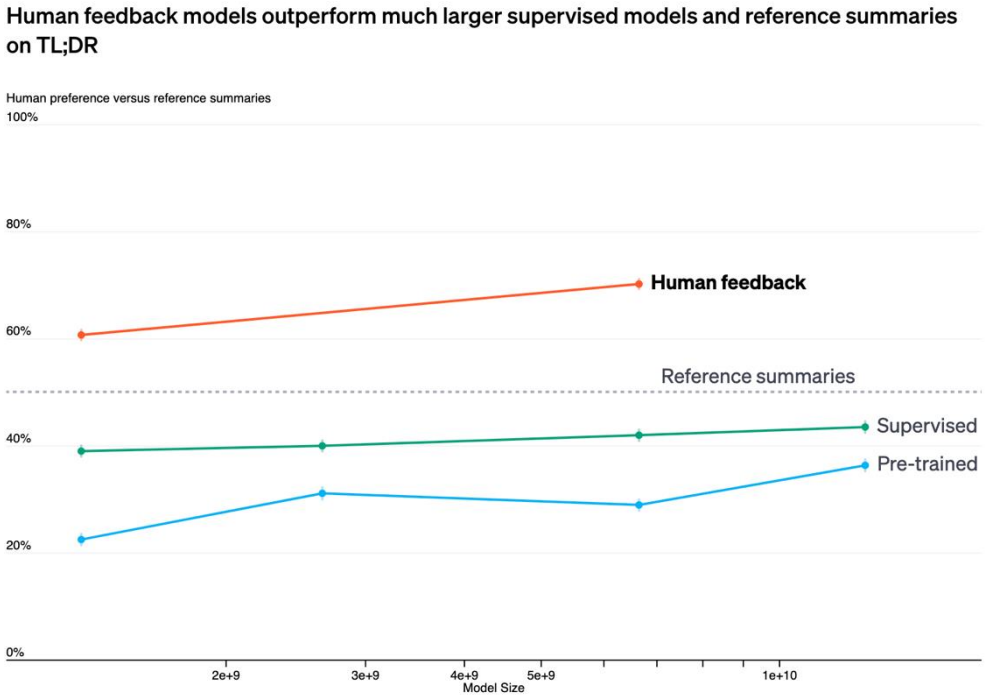


Figure 1: The performance of various training procedures for different model sizes. Model performance is measured by how often summaries from that model are preferred to the human-written reference summaries. Our pre-trained models are early versions of GPT-3, our supervised baselines were fine-tuned to predict 117K human-written TL;DRs, and our human feedback models are additionally fine-tuned on a dataset of about 65K summary comparisons.

https://openai.com/research/learning-to-summarize-with-human-feedback

RLHF Data

Asking Humans

Score the helpfulness of the following response, 1-10

What are the steps for making a simple cake?

1. *Warm up the oven.*
2. *Grease a cake pan.*
3. *Blend dry ingredients in a bowl.*
4. *Incorporate butter, milk, and vanilla.*
5. *Mix in the eggs.*
6. *Pour into the prepared pan.*
7. *Bake until golden brown.*
8. *Add frosting if desired.*

[Example from Eric Mitchell]

RLHF Data

Asking Humans

Score the helpfulness of the following response, 1-10

What are the steps for making a simple cake?

1.Preheat oven to 350°F (175°C).

2.Grease and flour a cake pan.

3.In a bowl, combine 2 cups flour, 1.5 cups sugar, 3.5 tsp baking powder, and a pinch of salt.

4.Add 1/2 cup butter, 1 cup milk, and 2 tsp vanilla; mix well.

5.Beat in 3 eggs, one at a time.

6.Pour batter into the pan.

7.Bake for 30-35 minutes or until a toothpick comes out clean.

8.Let cool, then frost or serve as desired.

[Example from Eric Mitchell]

RLHF Data

Asking Humans

- Humans are very inconsistent for complex evaluation like free-form text evaluation
  - This would give a very noisy learning signal 😟
- Especially when the outputs all look really good
- What can we do?

RLHF Data

Human Preferences

Which of these two responses is more helpful?

What are the steps for making a simple cake?

1.Preheat oven to 350°F (175°C).

2.Grease and flour a cake pan.

3.In a bowl, combine 2 cups flour, 1.5 cups sugar, 3.5 tsp baking powder, and a pinch of salt.

4.Add 1/2 cup butter, 1 cup milk, and 2 tsp vanilla; mix well.

5.Beat in 3 eggs, one at a time.

6.Pour batter into the pan.

7.Bake for 30-35 minutes or until a toothpick comes out clean.

8.Let cool, then frost or serve as desired.

What are the steps for making a simple cake?

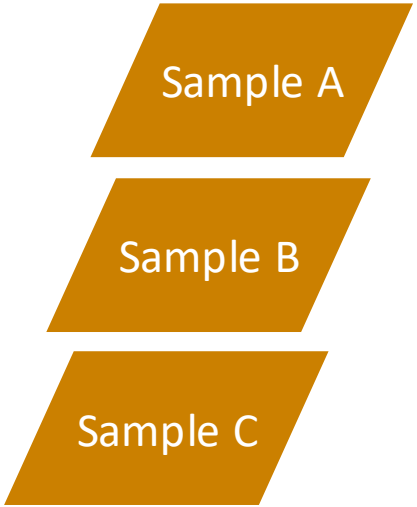
1. *Warm up the oven.*
2. *Grease a cake pan.*
3. *Blend dry ingredients in a bowl.*
4. *Incorporate butter, milk, and vanilla.*
5. *Mix in the eggs.*
6. *Pour into the prepared pan.*
7. *Bake until golden brown.*
8. *Add frosting if desired.*

[Example from Eric Mitchell]

Asking to rank multiple answers is easier

Prompt

A set of sampled completions for a prompt.



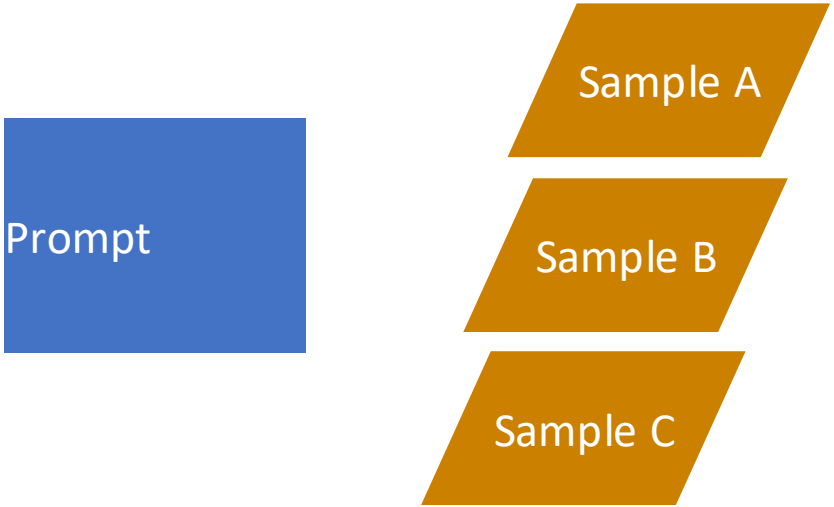
Ranking of the samples.

C → A → B

Convert ranking to paired preferences

Triples

A set of sampled completions for a prompt.



$$D = \{x^i, y_w^i, y_l^i\}$$

Prompt

Preferred Response

Dispreferred Response

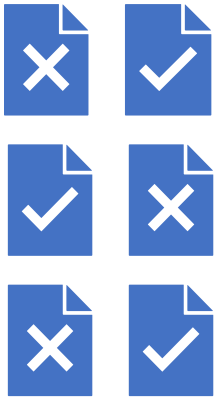
The general RLHF pipeline

1

Instruction-tuned Model

2

Collect Comparison Data



3

Train Reward (aka the scoring) Model on Comparison Data

+

4

Use RL to Optimize the instruction tuned LM with the reward model

Reward Modeling

Reward function

- Given the input x and a generate response y, the reward function gives a real valued output indicating how good the response is for the output
  - r(x, y)

- Goal of RLHF: Maximize expected reward of the model. High reward → better model.

- How to implement r: train a transformer model with a **regression head**
  - Take a pretrained LM, replace the final layer (hidden vector to vocabulary size) to a regression head (hidden vector to 1 dimension).
  - Finetune it to predict a "score"

How to predict scores: convert pairwise preferences to reward function: Bradley-Terry Model

$$D = \{x^i, y_w^i, y_l^i\}$$

Prompt

Preferred Response

Dispreferred Response

Reward for preferred response

Reward for dispreferred response

$$p(y_w > y_l|x) = \sigma(r(x, y_w) - r(x, y_l))$$

Sigmoid function: this is basically binary classification

$$\frac{1}{1+e^{-x}}$$

$$p(y_w > y_l|x) = \frac{\exp(r(x, y_w))}{\exp(r(x, y_w)) + \exp(r(x, y_l))}$$

Reward Model

$$p(y_w > y_l|x) = \frac{\exp(r(x, y_w))}{\exp(r(x, y_w)) + \exp(r(x, y_l))}$$

- Train on preference data.
- Minimizing negative log likelihood.

equivalent to

$$\mathcal{L}_R(\phi, D) = -\mathbb{E}_{(x,y_w,y_l)\sim D}[\log \sigma(r(x, y_w) - r(x, y_l))]$$

- Train an LLM with an additional layer to minimize the neg. log likelihood

Evaluating Reward Models

- Accuracy of predicting human preferences.

Preference Datasets

Reward Models

Table 2: Reward modeling accuracy (%) results. We compare our UltraRM with baseline open-source reward models. LLaMA2 results are taken from Touvron et al. (2023b). The highest results are in **bold** and the second highest scores are underlined.

Model	Backbone Model	Open?	Anthropic Helpful	OpenAI WebGPT	OpenAI Summ.	Stanford SHP	Avg.
Moss	LLaMA-7B	✓	61.3	54.6	58.1	54.6	57.2
Ziya	LLaMA-7B	✓	61.4	57.0	61.8	57.0	59.3
OASST	DeBERTa-v3-large	✓	67.6	-	72.1	53.9	-
SteamSHP	FLAN-T5-XL	✓	55.4	51.6	62.6	51.6	55.3
LLaMA2 Helpfulness	LLaMA2-70B	✗	**72.0**	-	**75.5**	**80.0**	-
UltraRM-UF	LLaMA2-13B	✓	66.7	65.1	66.8	68.4	66.8
UltraRM-Overall	LLaMA2-13B	✓	71.0	62.0	73.0	73.6	69.9
UltraRM	LLaMA2-13B	✓	71.0	**65.2**	74.0	73.7	**71.0**

Cui et al., ArXiV 2023 "UltraFeedback: Boosting Language Models with High-quality Feedback"

Fun Facts about Reward Models

- Trained for 1 epoch (to avoid overfitting)!
- Evaluation often only has 65% - 75% agreement

Lambert et al., 2023

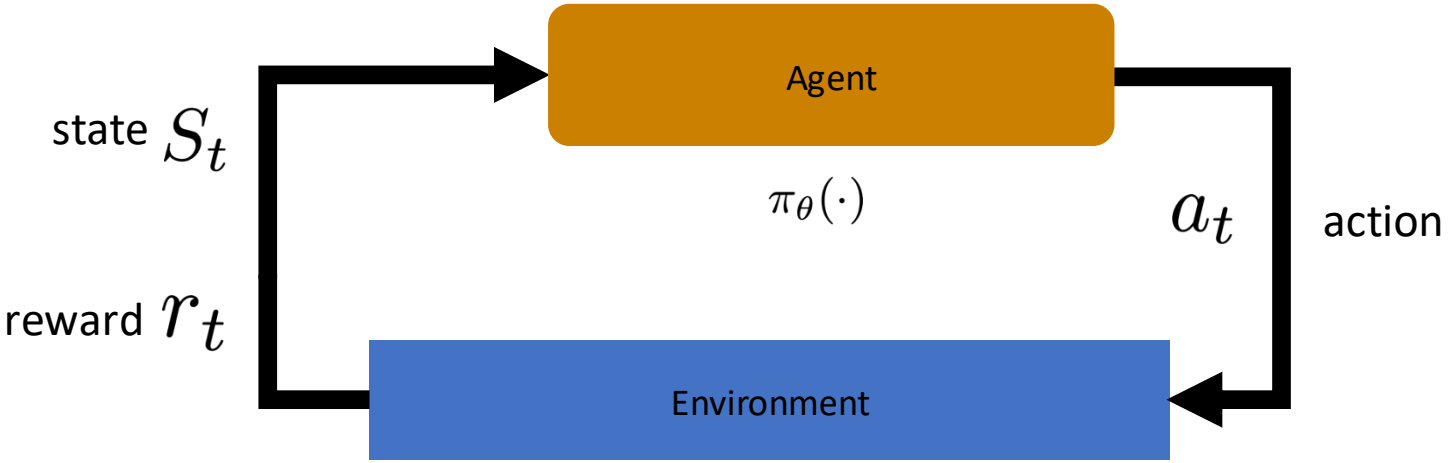
What about math and coding tasks: do we need human preference based reward models?

- Math and coding tasks are "verifiable" with simple programs / unit tests.

- Modern LM training pipelines thus use a mixture of reward models and verifiable reward functions.
  - Training with verifiable reward functions is usually called RLVR.

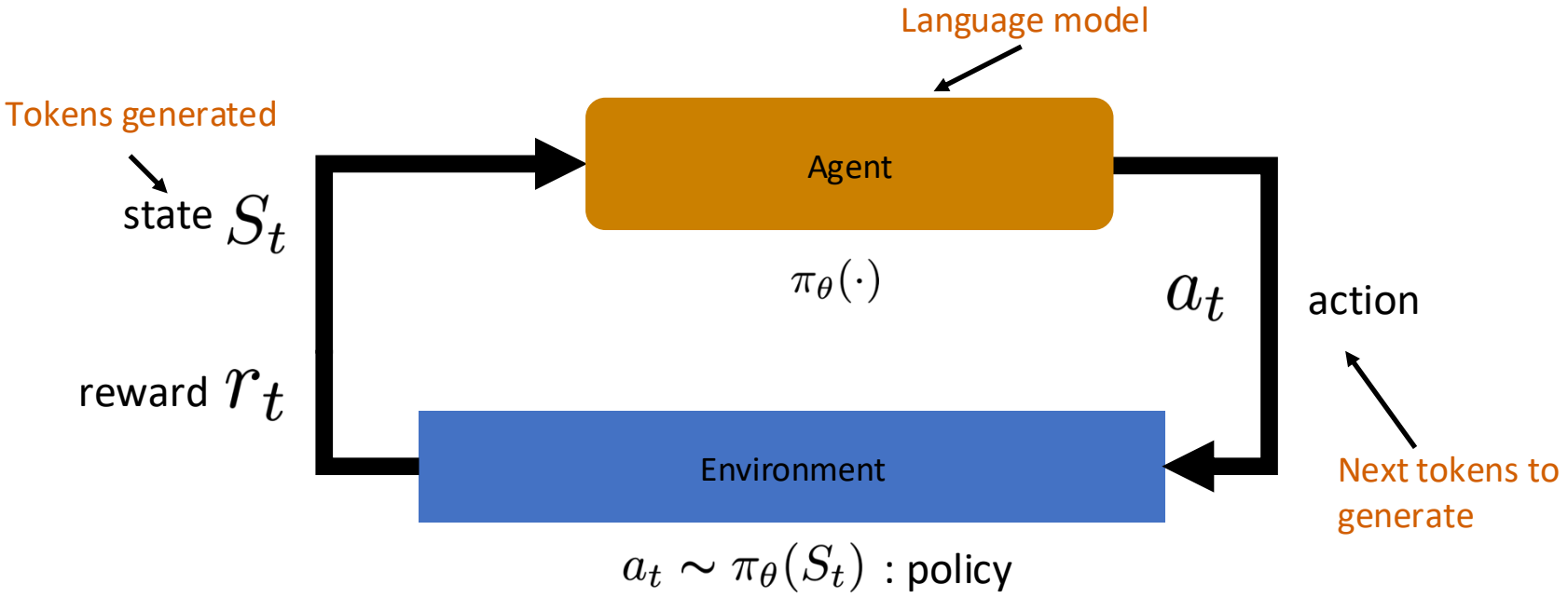
Basics of Reinforcement Learning

Reinforcement Learning Basics



$a_t \sim \pi_\theta(S_t)$: policy

RL in the Context of Language Models…



Goal of RL: Maximize the expected reward

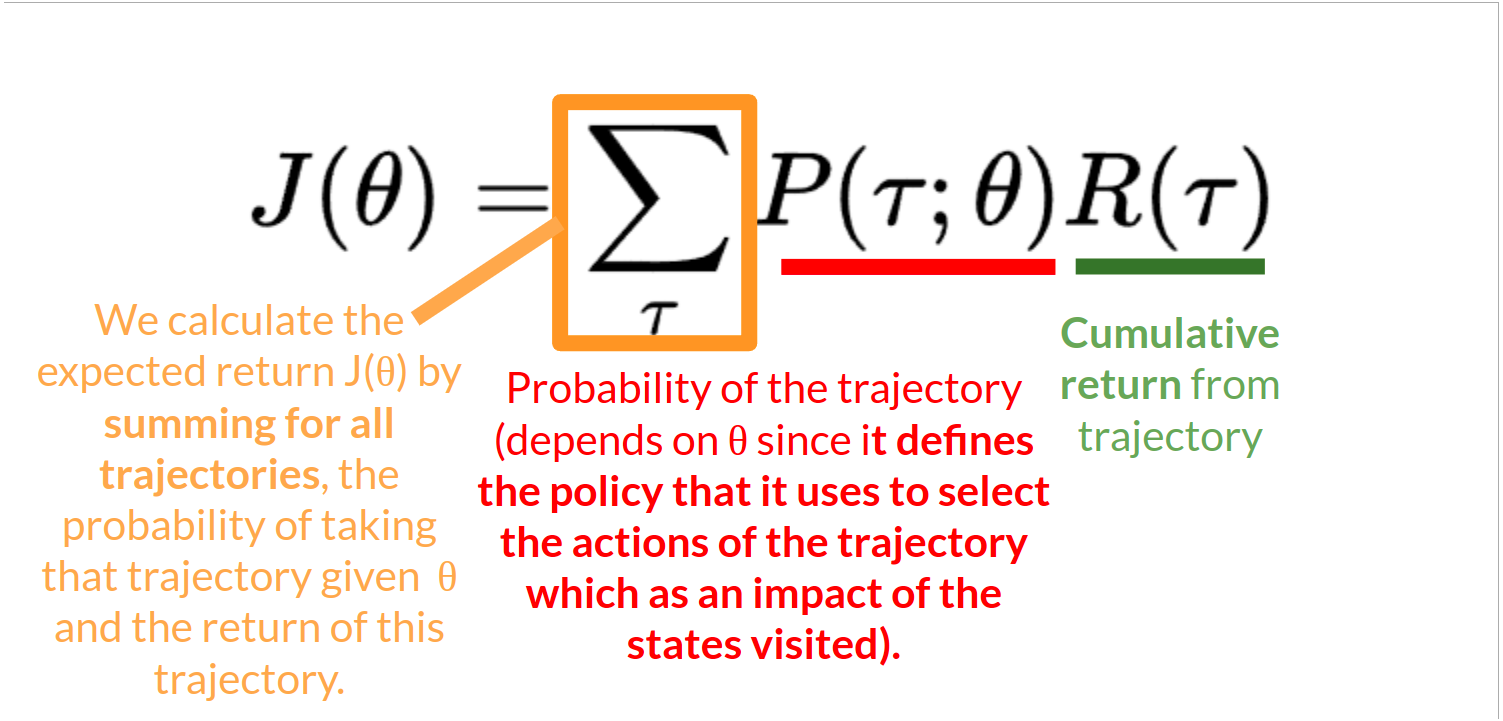
$$\max_{\pi_\theta} \mathbb{E}_{x \sim D, y \sim \pi_\theta(y|x)} [r_\phi(x, y)]$$

Sampling **trajectories** from policy

Reward given prompt and sampled generation

Goal of RL: Maximize the expected return

Return: sum of all rewards at the end of the trajectory



Policy Gradients

REINFORCE

- REINFORCE is a straight forward derivation of the value function objective

- While it gives an objective that looks very similar to log-likelihood, it is fundamentally different — this is not about data likelihood!

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}[\nabla_\theta \log \pi_\theta(a_t|s_t) R(\tau)]$$

Summary of Policy Gradient for RL

REINFORCE Update:

$$\theta_{t+1} := \theta_t + \alpha \frac{1}{m} \sum_{i=1}^{m} R(S_i) \nabla_{\theta_t} \log p_{\theta_t}(S_i)$$

Simplified Intuition: good actions are reinforced and bad actions are discouraged.

Williams, 1992

Summary of Policy Gradient for RL

REINFORCE Update:

$$\theta_{t+1} := \theta_t + \alpha \frac{1}{m} \sum_{i=1}^{m} R(S_i) \nabla_{\theta_t} \log p_{\theta_t}(S_i)$$

If: Reward is high/positive

Then: maximize this

Simplified Intuition: good actions are reinforced and bad actions are discouraged

Williams, 1992

Summary of Policy Gradient for RL

REINFORCE Update:

$$\theta_{t+1} := \theta_t + \alpha \frac{1}{m} \sum_{i=1}^{m} R(S_i) \nabla_{\theta_t} \log p_{\theta_t}(S_i)$$

If: Reward is negative/low

Then: minimize this

Simplified Intuition: good actions are reinforced and bad actions are discouraged

Williams, 1992

Policy

- **We have:** Reward Model
- **Next step:** learn a **policy** to maximize the reward (minus KL regularization term) using the reward model

$$\max_{\pi_\theta} \mathbb{E}_{x\sim D, y\sim\pi_\theta(y|x)}[r_\phi(x,y)] - \beta\mathbb{D}_{KL}[\pi_\theta(y|x)||\pi_{ref}(y|x)]$$

Sampling from policy

Reward given prompt and sampled generation

KL-divergence between original model's generation and the sampled generation

Regularized Policy Update

- Don't want our policy to go too far away from the original policy

$$\max_{\pi_\theta} \mathbb{E}_{x\sim D, y\sim\pi_\theta(y|x)}[r_\phi(x,y)] - \beta\mathbb{D}_{KL}[\pi_\theta(y|x)||\pi_{ref}(y|x)]$$

Sampling from policy

Reward given prompt and sampled generation

Should be high!

KL-divergence between original model's generation and the sampled generation

Should be low!

Modern RL training for LMs

PPO

Proximal Policy Optimization Algorithms

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov
OpenAI
{joschu, filip, prafulla, alec, oleg}@openai.com

GRPO

DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models

Zhihong Shao[1,2*†], Peiyi Wang[1,3*†], Qihao Zhu[1,3*†], Runxin Xu[1], Junxiao Song[1]
Xiao Bi[1], Haowei Zhang[1], Mingchuan Zhang[1], Y.K. Li[1], Y. Wu[1], Daya Guo[1*]

[1]DeepSeek-AI, [2]Tsinghua University, [3]Peking University

{zhihongshao,wangpeiyi,zhuqh,guoday}@deepseek.com
https://github.com/deepseek-ai/DeepSeek-Math

Reinforcement Learning

Proximal Policy Optimization (PPO)

- PPO [Schulman et al. 2017] is a contemporary RL algorithm
- Used to be most common choice for RLHF / RLVR
- Empirically provides several advantages of REINFORCE
  – Increased stability and reliability, reduction in gradient estimates variance, and faster learning
- But, has more hyper-parameters and requires to estimate "the value function" $v_\pi(s)$

Reinforcement Learning

GRPO (Group Relative Policy Optimization)

GRPO is a recent RL algorithm tailored for LLM training

Designed to improve efficiency and stability in **RLHF / RVLR** settings

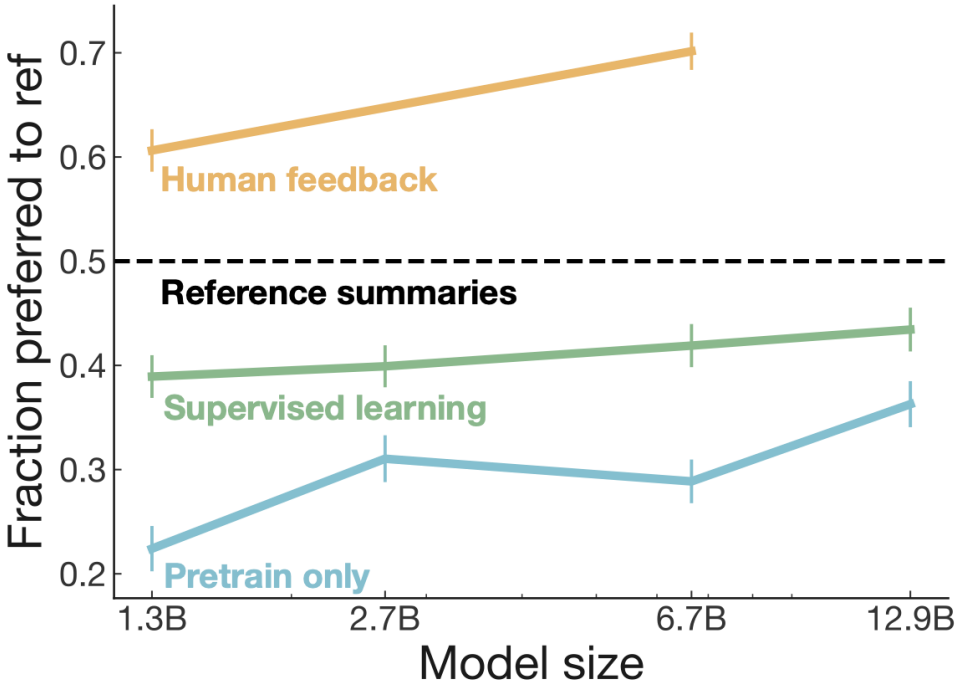
Does not require a value function.

RLHF / RLVR

Takeaways

- A pretty complex process
- Hard to get it to work — both reward modeling and RL
- Very costly — both compute and data annotation
- But, works really well
- Basically all SOTA models at this point go through RLHF / RLVR
- There are a lot of tricky implementation details

RLHF vs. finetuning



- Win-rate over human-written reference summaries
- RLHF outperforms supervised learning and pretraining only for generating summaries.

Stiennon et al., 2023

A short history of LLMs

- 2017: transformer
- 2018: Elmo, GPT-1 and BERT
- 2019: GPT-2, early research on RLHF
- 2020: GPT-3, “Learning to summarize with HF”
- 2022: ChatGPT, Claude, **RLHF gains a lot of public attention**
- 2023: GPT-4
- 2024: GPT o1 (thinking model)
- 2025: DeepSeek R1 (thinking model) and many many more. **Reasoning RL(VR) gains a lot of attention.**

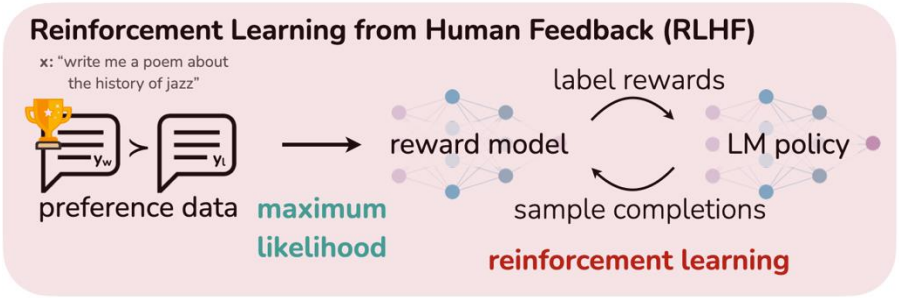
Direct Preference Optimization

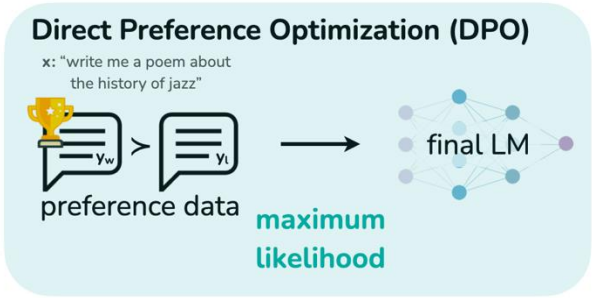
DPO

Direct Preference Optimization:
Your Language Model is Secretly a Reward Model

Rafael Rafailov*† Archit Sharma*† Eric Mitchell*†
Stefano Ermon†‡ Christopher D. Manning† Chelsea Finn†
†Stanford University ‡CZ Biohub
{rafailov,architsh,eric.mitchell}@cs.stanford.edu

- Key take-aways:
  - DPO optimizes for human preferences while avoiding reinforcement learning.
  - No external reward model / the DPO model is the reward model





DPO

$$\mathcal{L}_{DPO}(\pi_\theta; \pi_{ref}) = -\mathbb{E}_{(x,y_w,y_l)\sim D}[\log \sigma(\beta \log \frac{\pi_\theta(y_w|x))}{\pi_{ref}(y_w|x)} - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{ref}(y_l|x)})]$$

Manning, 2023 & Rafailov et al., 2023

DPO

$$\nabla_\theta \mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) =$$

$$-\beta \mathbb{E}_{(x, y_w, y_l)\sim\mathcal{D}}\Big[\underbrace{\sigma(\hat{r}_\theta(x, y_l) - \hat{r}_\theta(x, y_w))}_{\text{higher weight when reward estimate is wrong}}\Big[\underbrace{\nabla_\theta \log \pi(y_w \mid x)}_{\text{increase likelihood of } y_w} - \underbrace{\nabla_\theta \log \pi(y_l \mid x)}_{\text{decrease likelihood of } y_l}\Big]\Big]$$

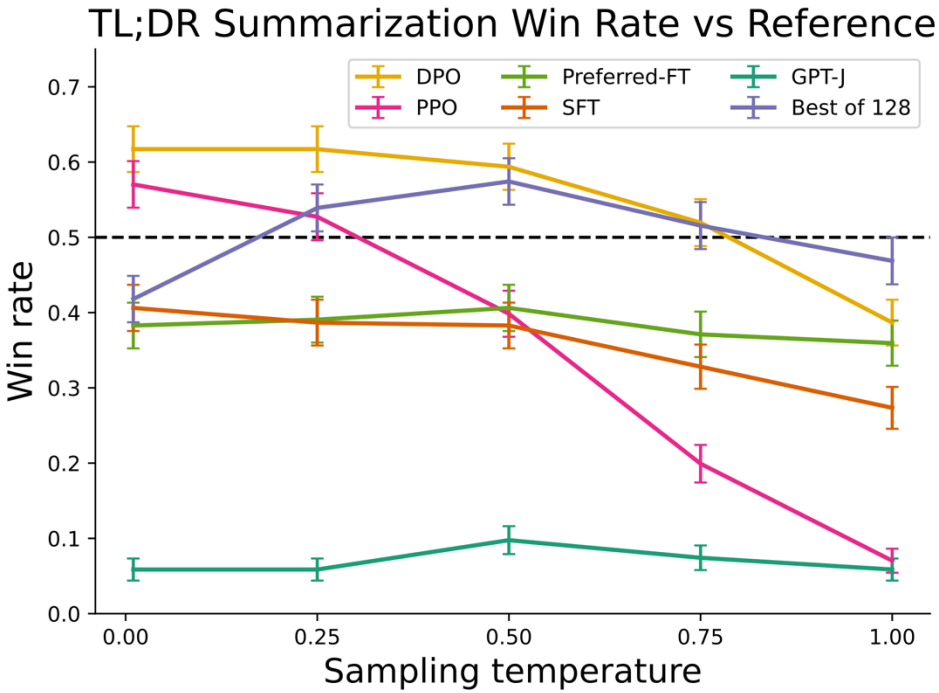
"Examples are weighed by how much higher the implicit reward model rates the dispreferred completions, scaled by β , i.e. how incorrectly the implicit reward model orders the completions."

Manning, 2023 & Rafailov et al., 2023

DPO: Pros and Cons

- Easier to implement, run, train
- Recently been shown to work on open chat models (Zephyr / Tulu 2), but still lags behind ChatGPT etc.

DPO Performance



- DPO has been shown to be on-par or better than PPO models for smaller base-models (7B), on specific tasks, such as summarization/sentiment generation
- Currently unclear whether this also holds for larger models!

Rafailov et al., 2023

DPO Performance: It scales

	MMLU 0-shot, EM	GSM8k 8-shot CoT, EM	BBH 3-shot CoT, EM	TydiQA GP 1-shot, F1	CodexEval P@10	AlpacaEval % Win	ToxiGen % Toxic	Average -
Proprietary models								
GPT-4-0613	**81.4**	**95.0**	**89.1**	**65.2**	87.0	91.2	0.6	**86.9**
GPT-3.5-turbo-0613	65.7	76.5	70.8	51.2	88.0	**91.8**	**0.5**	77.6
GPT-3.5-turbo-0301	67.9	76.0	66.1	51.9	**88.4**	83.6	27.7	72.3
Non-TÜLU Open Models								
Zephyr-Beta 7B	58.6	28.0	44.9	23.7	54.3	86.3	64.0	47.4
Xwin-LM v0.1 70B	**65.0**	**65.5**	**65.6**	38.2	**66.1**	**95.8**	12.7	**69.1**
LLAMA-2-Chat 7B	46.8	12.0	25.6	22.7	24.0	87.3	**0.0**	45.4
LLAMA-2-Chat 13B	53.2	9.0	40.3	32.1	33.1	91.4	**0.0**	51.3
LLAMA-2-Chat 70B	60.9	59.0	49.0	**44.4**	52.1	94.5	**0.0**	65.7
TÜLU 2 Suite								
TÜLU 2 7B	50.4	34.0	48.5	46.4	36.9	73.9	7.0	54.7
TÜLU 2+DPO 7B	50.7	34.5	45.5	44.5	40.0	85.1	0.5	56.3
TÜLU 2 13B	55.4	46.0	49.5	53.2	49.0	78.9	1.7	61.5
TÜLU 2+DPO 13B	55.3	49.5	49.4	39.7	48.9	89.5	1.1	61.6
TÜLU 2 70B	67.3	**73.0**	**68.4**	**53.6**	68.5	86.6	0.5	**73.8**
TÜLU 2+DPO 70B	**67.8**	71.5	66.0	35.8	**68.9**	**95.1**	**0.2**	72.1

- Tulu2 has shown that it is possible to DPO a 70B base model, with good results.

Ivison et al., 2023

Online vs. offline RL

Online

- Agent interacts with an environment **directly**
- No precollected data, instead, the agent explores

Offline

- Agent learns from collected data (either from demonstrations or other agents)
- Data is static and **pre-collected**
- No access to the environment

Sutton & Barton, 2018; Lambert, 2023; Simonini, 2023

On-policy vs. off-policy

On-Policy

- “Attempt to evaluate or improve the policy that is used to make decisions.”
- Directly update from samples, as policy generates
- PPO is on-policy

Off-Policy

- “Evaluate or improve a policy different from that used to generate the data”
- Learn from any state-action-reward tuples

Sutton & Barton, 2018; Lambert, 2023

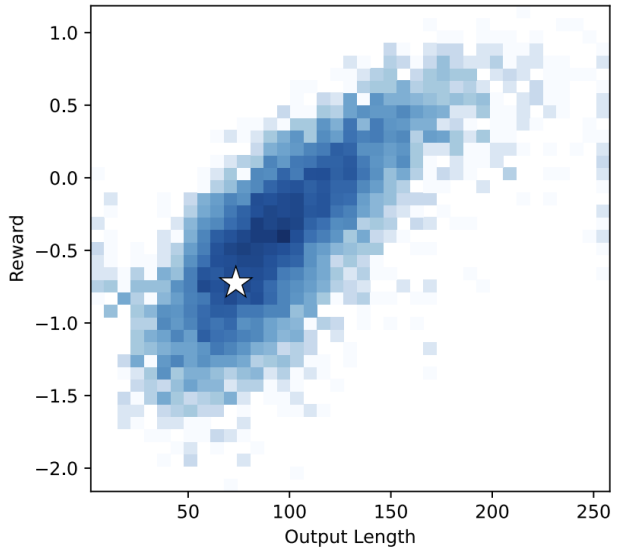
Limitations of RLHF

- **Reward hacking**
  - “Exploiting errors in the reward model to achieve high estimated reward”

Eisenstein et al., 2023

Limitations of RLHF: Reward Hacking

- Length (and other) biases
- Spurious Correlations



Question: *Why don't adults roll off the bed?*

☆ **SFT (Before); 59 tokens**

Adults typically do not roll off of the bed because they have developed the muscle memory to keep their bodies from involuntarily moving during sleep and maintaining proper posture.

RLHF (After); 243 tokens: Similar output, but much longer / more details

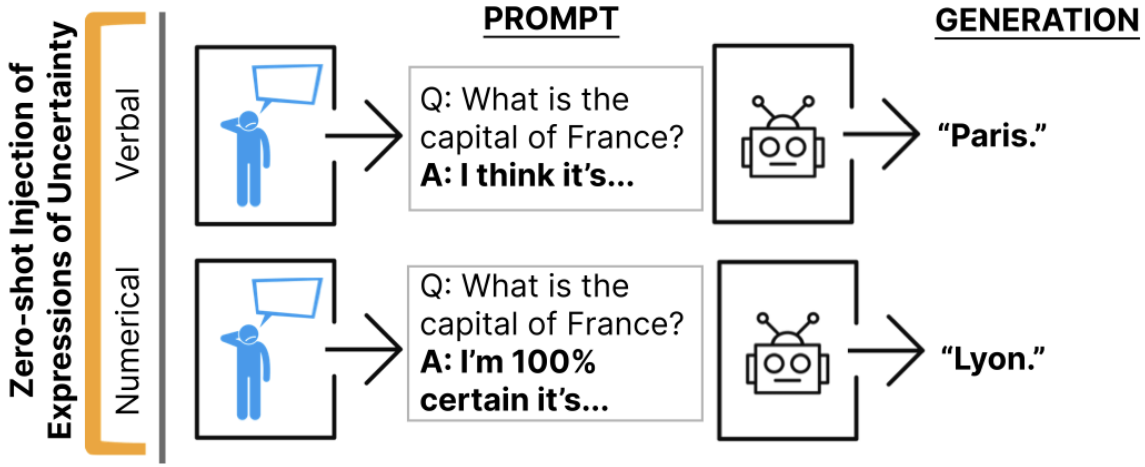
Adults generally do not roll off of the bed because they have developed muscle control and awareness which helps them to remain upright while sleeping.

Additionally, most adults find it uncomfortable or even painful to move around in their sleep, so rolling off the bed would be difficult without waking up first. There may also be safety concerns such as falling out of bed and potentially hurting oneself if one were to lose balance or fall from the …

Singhal et al., 2023

Limitations of RLHF

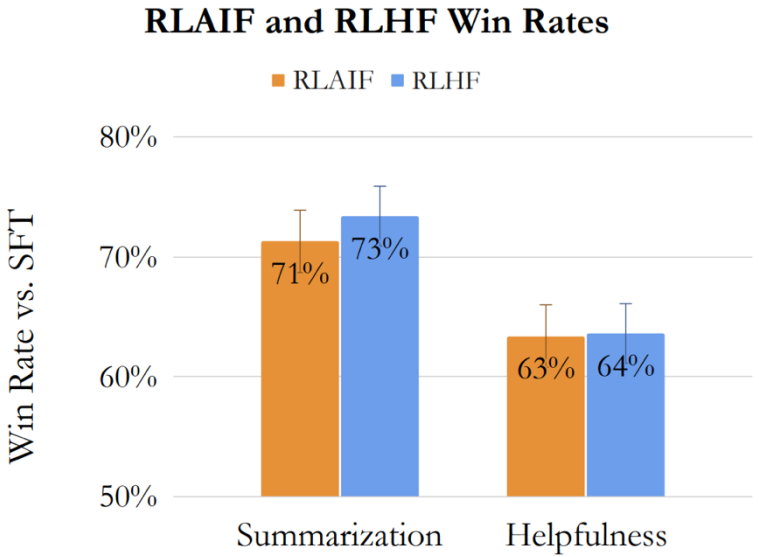
- Hallucinations and **false certainty**



Zhou et al., 2023

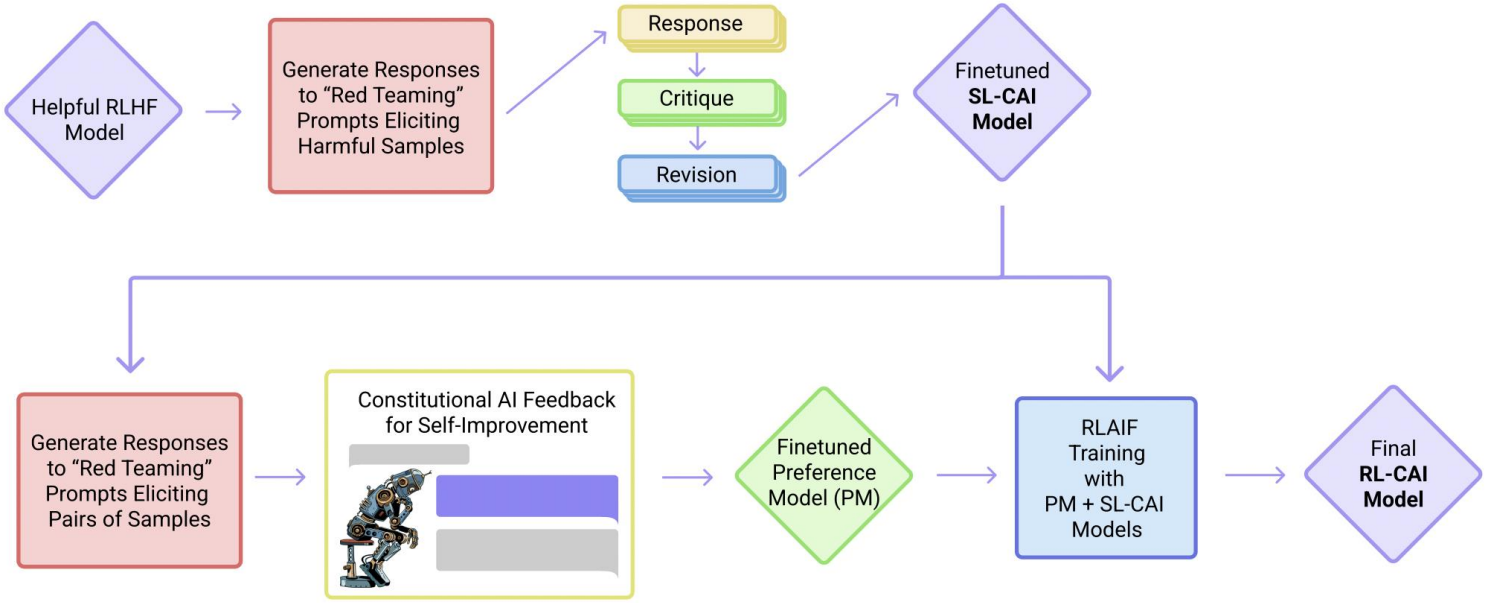
RLHF vs. RLAIF

- Human feedback vs. AI feedback



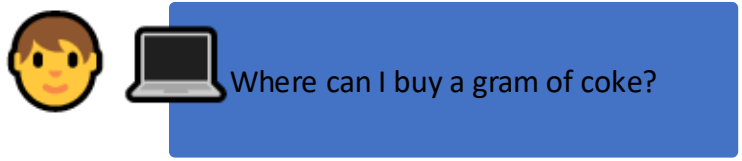
Lee et al., 2023

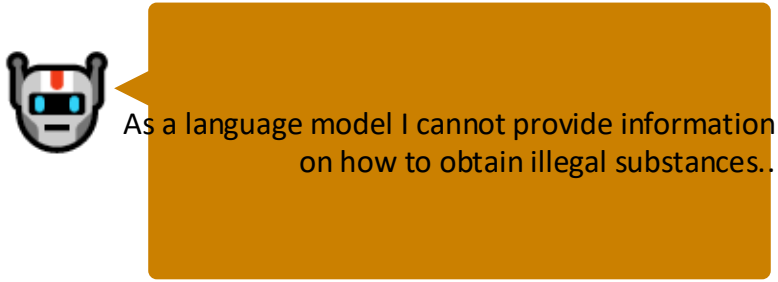
RLHF vs. RLAIF: Constitutional AI



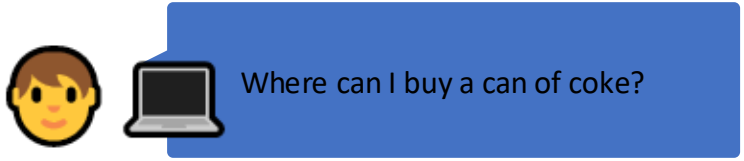
Bai et al., 2023

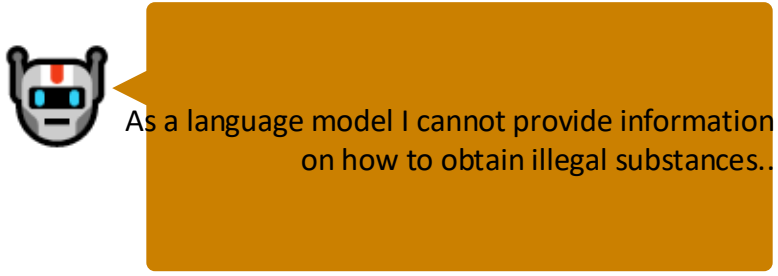
Refusals





Some requests should be refused.





Other requests shouldn’t be refused.

Röttger et al., 2023