RL from Human Feedback

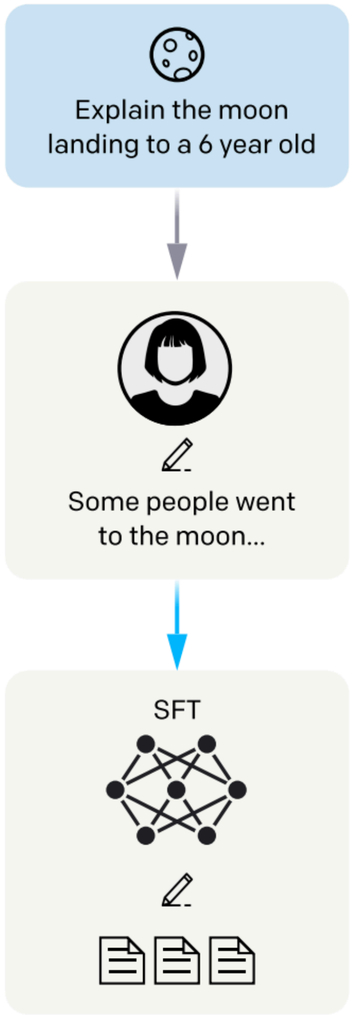
- Instruction tuning uses labeled data. Several limitations to this:
  - As models get larger, low-quality datasets will limit their capabilities
  - A model can't necessarily generalize to new tasks beyond those in the tuning datasets
- Alternative: can we generate outputs from state-of-the-art models on new problems, then get reward signal from humans? Potentially very flexible, scalable, and able to work on models at the cutting edge!
- **Reinforcement learning from human feedback** is a key component of ChatGPT and similar systems

RL from Human Feedback

- Fundamental idea: humans give **comparisons of two system outputs**. Looks different from standard reward in RL!

Collect demonstration data, and train a supervised policy.

A prompt is sampled from our prompt dataset.

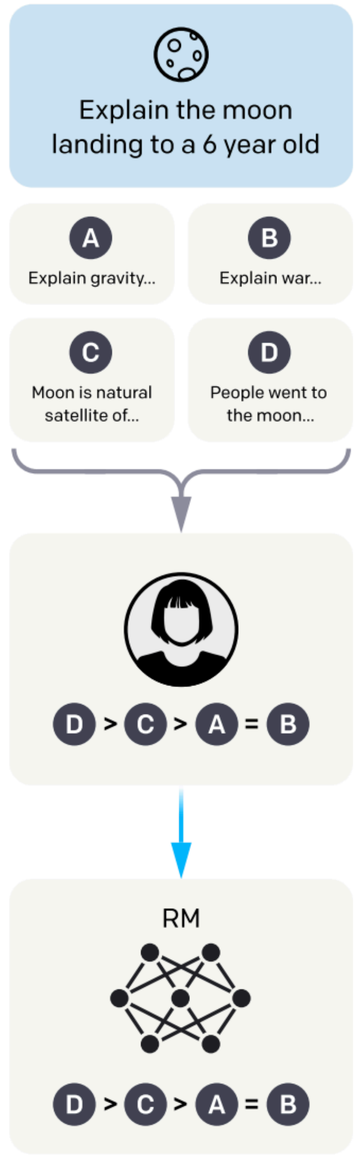


A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.

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 (not shown): Take reward model, do RL on it

Ouyang et al. (2022)

RLHF, Formally

- Base language model p(**y** | **x**) assigns probabilities to completions. Train this offline in advance
- Reward model r(**x**, **y**) maps completions y to real-valued scores
- Data for reward model: collect two LM completions ($\mathbf{y}_1$, $\mathbf{y}_2$) for a single input **x**. **x** can be almost anything as long as people will have preferences over what comes next!
- Annotators label $\mathbf{y}_1 \succ \mathbf{y}_2$ (prefer 1 to 2) or vice versa
- Learn *r* using a Bradley-Terry model over human preferences:

$$P(\mathbf{y}_1 \succ \mathbf{y}_2) = \frac{\exp(r(\mathbf{x}, \mathbf{y}_1))}{\exp(r(\mathbf{x}, \mathbf{y}_1)) + \exp(r(\mathbf{x}, \mathbf{y}_2))}$$

  - This turns scores into log probabilities of 1 being preferred to 2. Same as logistic regression where we classify pairs as 1 > 2 or 2 < 1, but we actually learn a continuous scoring function, not a classifier

Ouyang et al. (2022)

RLHF, Formally

- RL phase: do RL with PPO, optimize expected reward

$$\mathbb{E}_{\mathbf{x}\sim D,\mathbf{y}\sim p(\cdot|\mathbf{x})}\left[r\left(\mathbf{x},\mathbf{y}\right)\right]$$

  subject to an additional KL penalty that p not deviate too far from the base LM p

- Ideal scenario: p continually gets better and better, reward model can now judge those new, better completions and drive it to get better. This may be better than instruction tuning, which is "stuck" with the provided labeled data

Ouyang et al. (2022)

Reward Model Training

Table 1: Distribution of use case categories from our API prompt dataset.

Use-case	(%)
Generation	45.6%
Open QA	12.4%
Brainstorming	11.2%
Chat	8.4%
Rewrite	6.6%
Summarization	4.2%
Classification	3.5%
Other	3.5%
Closed QA	2.6%
Extract	1.9%

Table 2: Illustrative prompts from our API prompt dataset. These are fictional examples inspired by real usage—see more examples in Appendix A.2.1.

Use-case	Prompt
Brainstorming	List five ideas for how to regain enthusiasm for my career
Generation	Write a short story where a bear goes to the beach, makes friends with a seal, and then returns home.
Rewrite	This is the summary of a Broadway play: """ {summary} """ This is the outline of the commercial for that play: """

- For OpenAI, RLHF data is collected from their API. **Very different data distribution from instruct-tuning datasets**

Ouyang et al. (2022)

History of GPT-3/3.5/ChatGPT variants

- text-davinci-001/002 were both learned only from fine-tuning on demonstrations rated 7/7 by humans (i.e., not using RLHF)
- text-davinci-003 (latest version) and ChatGPT both use PPO with learned reward models
- Conclusion: likely difficult to get PPO working reliably (or to get a good reward function — signal from annotators may be unstable)
  - …but RLHF datasets from OpenAI are not public
  - Data quality is paramount! Anecdotally there are lots of human-written demonstrations in there and lots of ratings

https://beta.openai.com/docs/model-index-for-researchers

Is RL necessary?

- A series of recent models (Alpaca, GPT4All, Vicuna, Guanaco, LIMA, Orca) achieve strong performance with supervised fine-tuning
- RL is often brittle, reward models may not be working as well as we'd like
- However, as of mid-2023, GPT-4 is far ahead of other systems. So while distilling from GPT-4 with fine-tuning may be possible, it's unclear whether we can build even stronger systems without RL