

CME 295: Transformers & Large Language Models



Afshine Amidi & Shervine Amidi



Recap of last episode...

Pre-training



Learn how to
autocomplete

WIKIPEDIA 25 years of the free encyclopedia

Teddy bear

Article Talk Read Edit View history

From Wikipedia, the free encyclopedia

For other uses, see *Teddy bear (disambiguation)*.

A **teddy bear**, or simply a **teddy**, is a **stuffed toy** in the form of a **bear**. The teddy bear was named by **Morris Michtom** after the 26th president of the United States, **Theodore Roosevelt**; it was developed apparently simultaneously in the first decade of the 20th century by two toymakers: **Richard Steiff** in Germany and Michtom in the United States. It became a popular children's toy, and it has been celebrated in story, song, and film.^[1]

Since the creation of the first teddy bears (which sought to imitate the form of real bear **cubs**), "teddies" have greatly varied in form, style, color, and material. They have become **collector's items**, with older and rarer teddies appearing at public auctions.^[2] Teddy bears are among the most popular gifts for children, and they are often given to adults to signify affection, congratulations, or sympathy.

Bear thought to be made by Morris Michtom in the early 1900s; donated to the Smithsonian Museum of Natural History in Washington, D.C., United States, by Theodore Roosevelt's grandson Kermit Roosevelt Jr. in 1964

A replica Steiff model 55PB displayed at the Steiff-Museum, Giengen, Germany, in 2006; no original examples of the 55PB are known to survive

shervinea / enzynet Public

Code Issues Pull requests Actions Projects Security and quality Insights

Files

master

Go to file

datasets

enzynet

__init__.py

constants.py

indicators.py

keras_utils.py

models.py

pdf.py

shervinea Fix output model type e60ef90 · 5 years ago History

86 Lines (66 loc) · 2.39 KB

Code Blame

```
1 """Model definitions."""
2
3 # Authors: Afshine Avidi <lastname@mit.edu>
4 # Shervine Avidi <firstname@stanford.edu>
5
6 # MIT License
7
8 import numpy as np
9
10 from enzynet import constants
```

$$\mathcal{L}_{\text{CE}} = -\frac{1}{T} \sum_{t=1}^T \log p_{\theta}(x_t \mid x_{<t})$$

Recap of last episode...

Pre-training



Learn how to
autocomplete

Supervised finetuning



Learn how to **follow**
a certain behavior

Story writing

Write a short story about a teddy bear who likes to read poetry.

Once upon a time, a bear, Teddy, stumbled upon verses from Attar. The bear became so captivated with the rhythm of the poem's sentences that he got filled with happiness.

Explanation

Explain why a teddy bear is a great friend.

A teddy bear is a great friend because it provides comfort and companionship. Its soft fur and cuddly nature is perfect for hugs. Teddy bears are always there in both happy and sad moments.

$$\mathcal{L}_{\text{SFT}} = -\frac{1}{T} \sum_{t=1}^T \log p_{\theta}(y_t \mid x, y_{<t})$$

Recap of last episode...

Pre-training	Supervised finetuning	Preference tuning
 Learn how to autocomplete	 Learn how to follow a certain behavior	 Learn how to align with human preferences

$$\mathcal{L} = ?$$

Recap of last episode...

Pre-training	Supervised finetuning	Preference tuning	Reasoning
			
Learn how to autocomplete	Learn how to follow a certain behavior	Learn how to align with human preferences	Learn how to reason better

$$\mathcal{L} = ?$$

Recap of last episode...

Pre-training



Learn how to **autocomplete**

Supervised finetuning



Learn how to **follow** a certain behavior

Preference tuning	Reasoning
 Learn how to align with human preferences	 Learn how to reason better

Today's focus

Disclaimer

 **Reinforcement learning**-based methods are **pretty technical** and **mathematically involved** 

If you feel overwhelmed:

1. Focus on **intuition** and **why** each method is **needed**
2. Review **prerequisites** (calculus, probability theory)
3. **Interpret** formulas from first principles

Important note on conventions

⚠ Field is full of **different notations / conventions** ⚠

" r "

could be...

reward

OR

ratio

$$\rho = \frac{\pi_{\theta}}{\pi_{\text{old}}}$$

Important note on conventions

⚠ Field is full of **different notations / conventions** ⚠

" r "

could be...

reward

OR

ratio

$$\rho = \frac{\pi_{\theta}}{\pi_{\text{old}}}$$

" π_{old} "

could be...

policy from
last batch

OR

reference
policy

π_{ref}

Important note on conventions

⚠ Field is full of **different notations / conventions** ⚠

" r "

could be...

reward

OR

ratio

$$\rho = \frac{\pi_{\theta}}{\pi_{\text{old}}}$$

" π_{old} "

could be...

policy from
last batch

OR

reference
policy

π_{ref}

" $\mathcal{L}(\theta)$ "

could be...

loss
(minimize)

OR

objective
(maximize)

$J(\theta)$



Transformers & Large Language Models

Reinforcement learning

Reward design

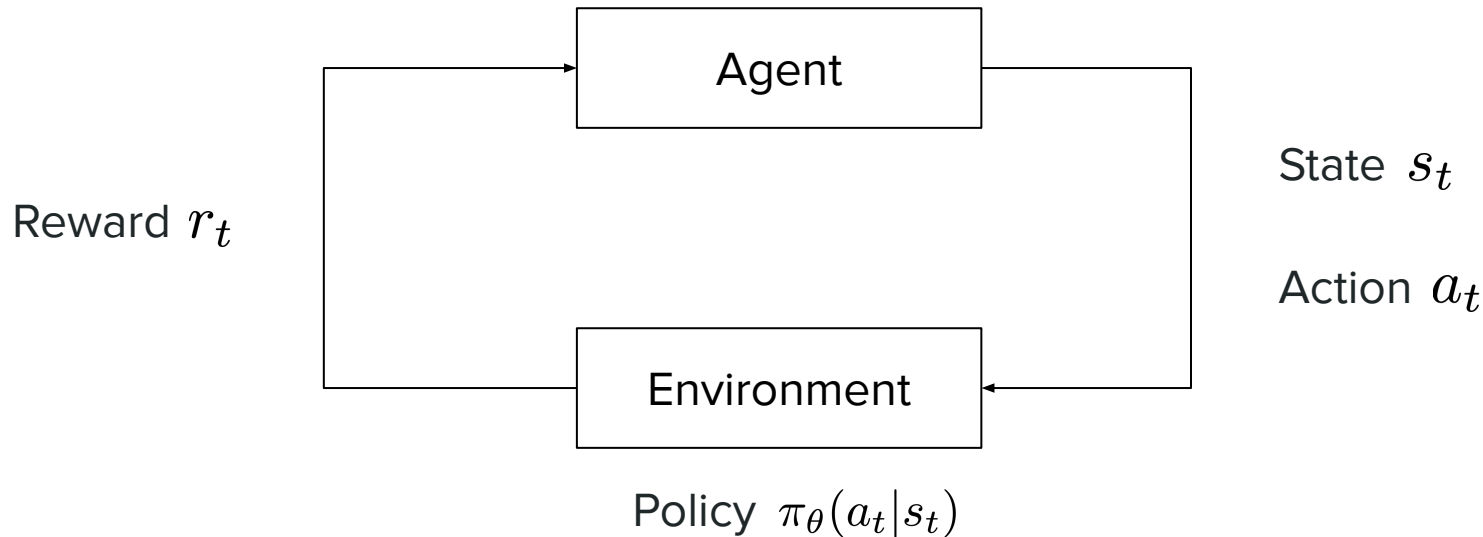
Policy gradients

PPO

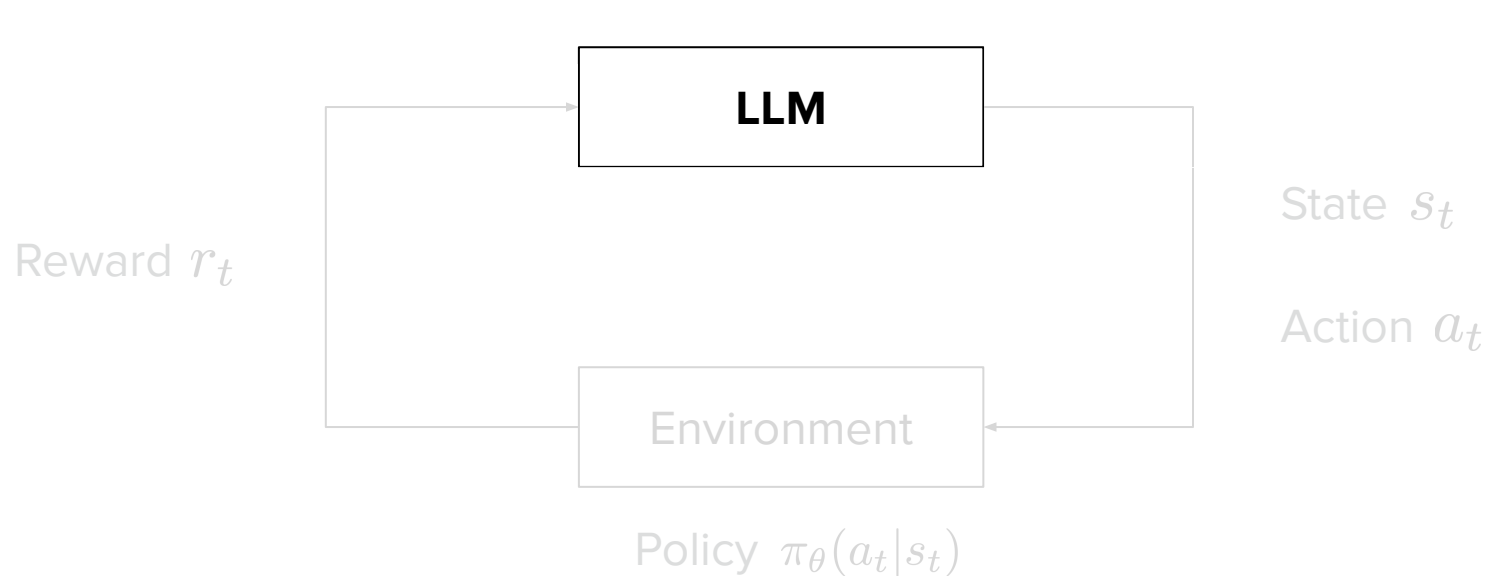
GRPO

On-policy distillation

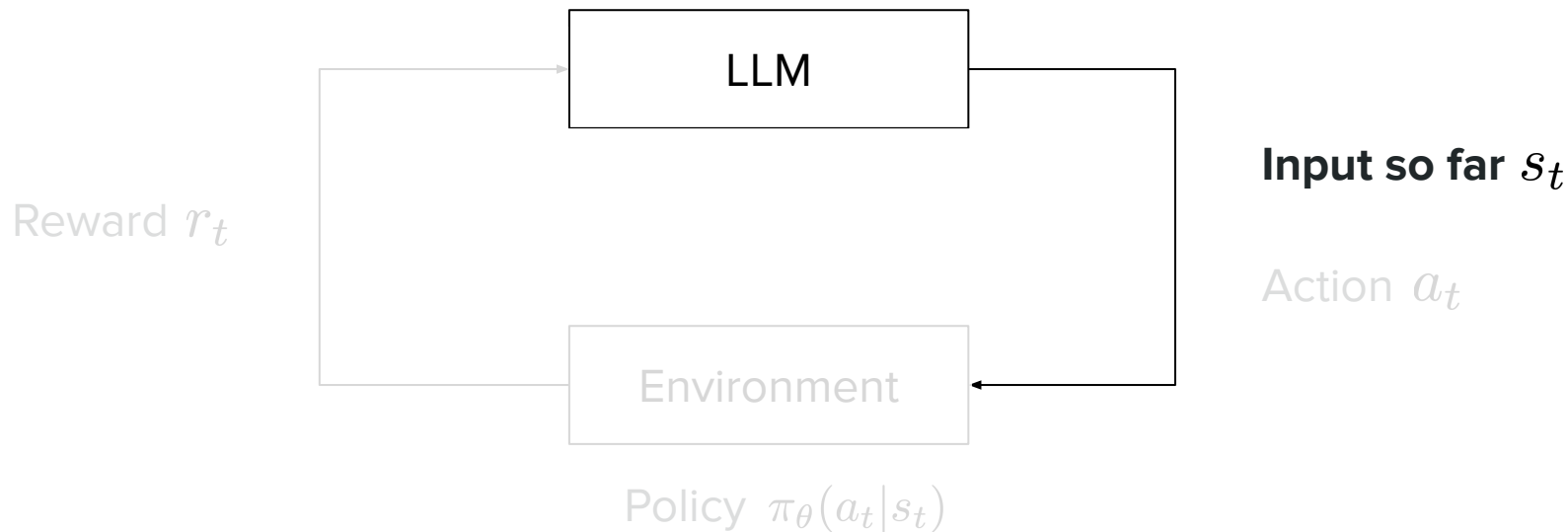
RL formulation



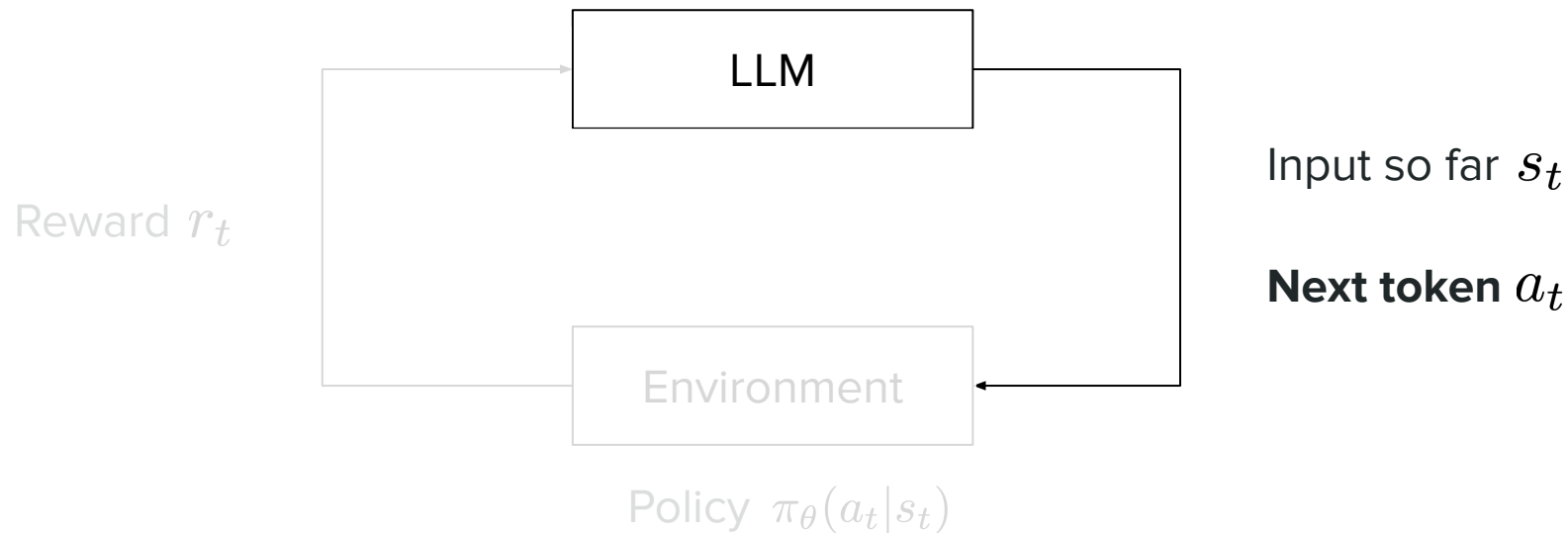
RL formulation for LLMs



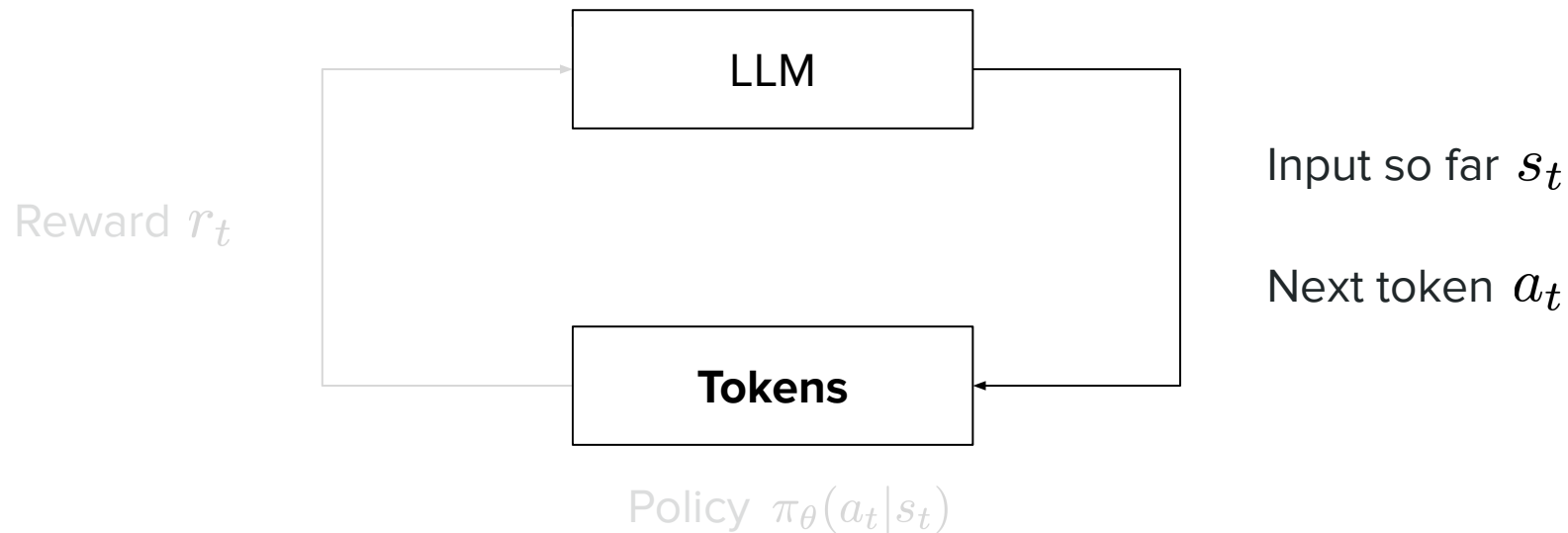
RL formulation for LLMs



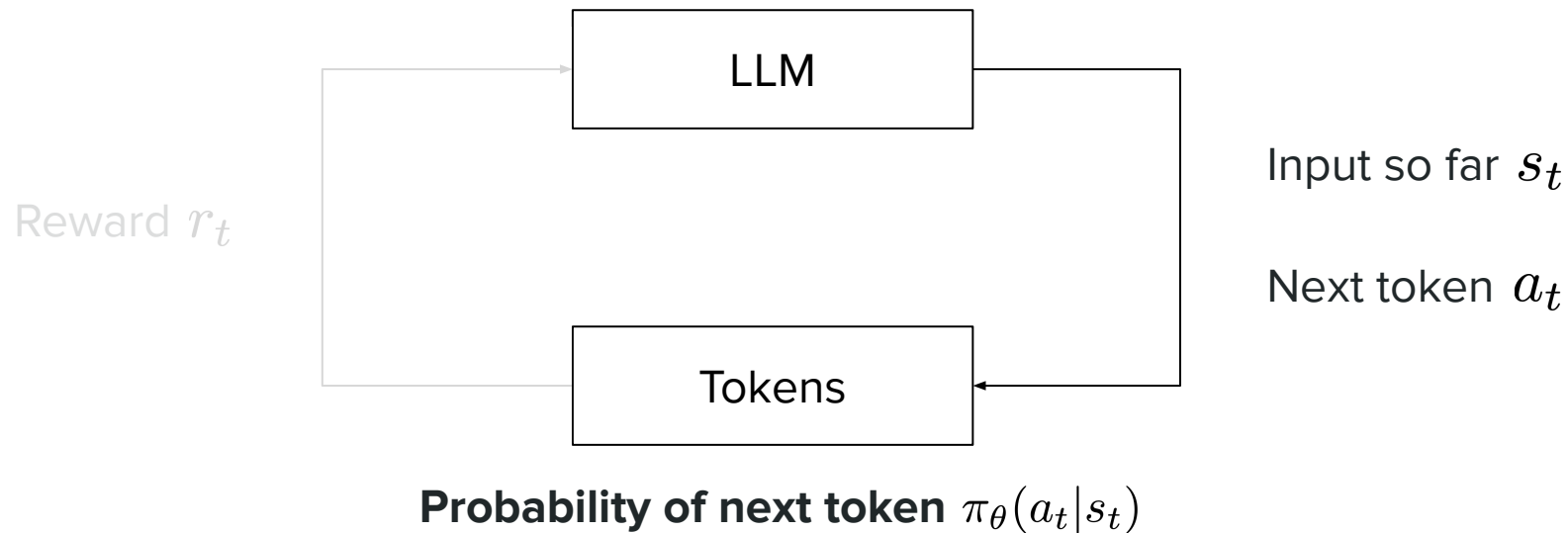
RL formulation for LLMs



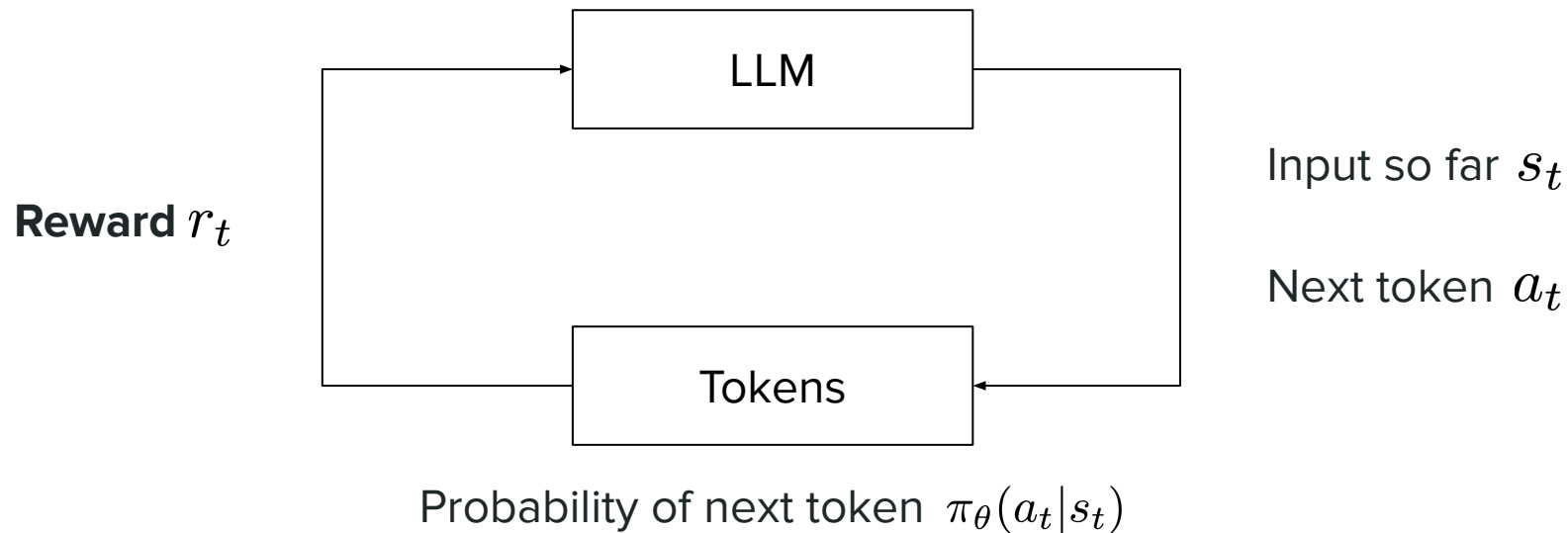
RL formulation for LLMs



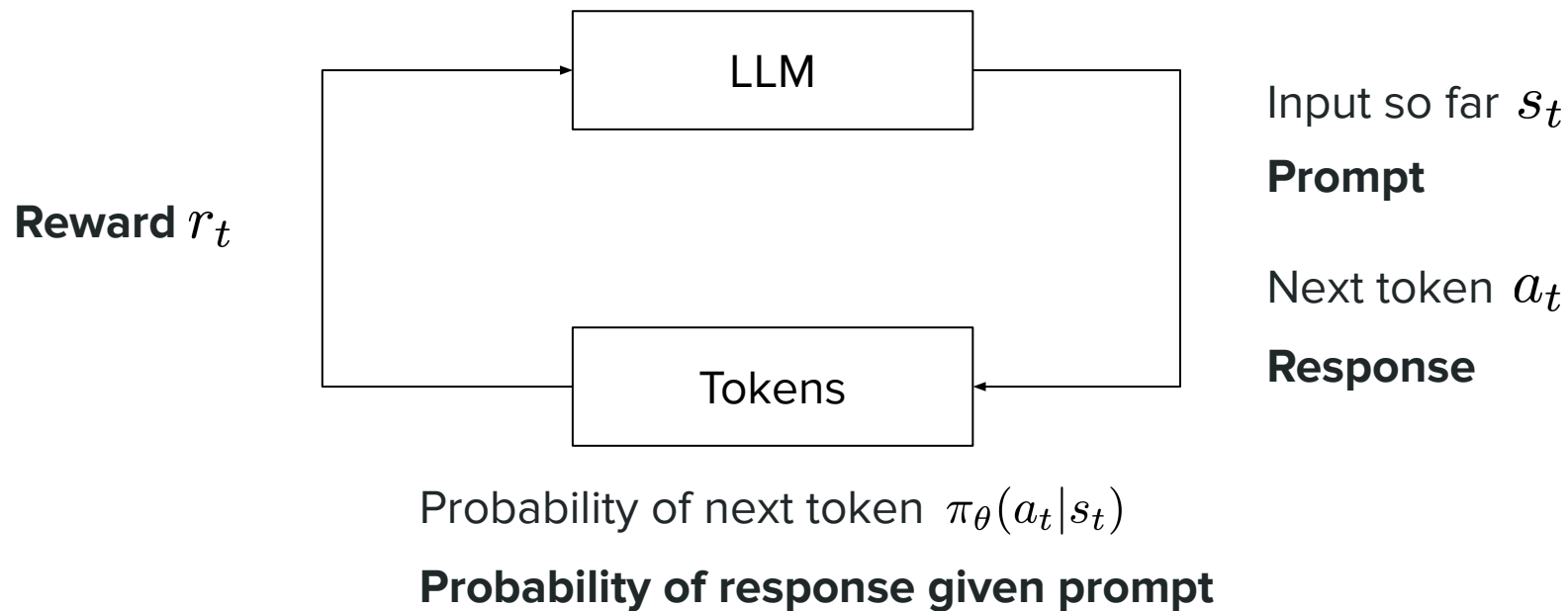
RL formulation for LLMs



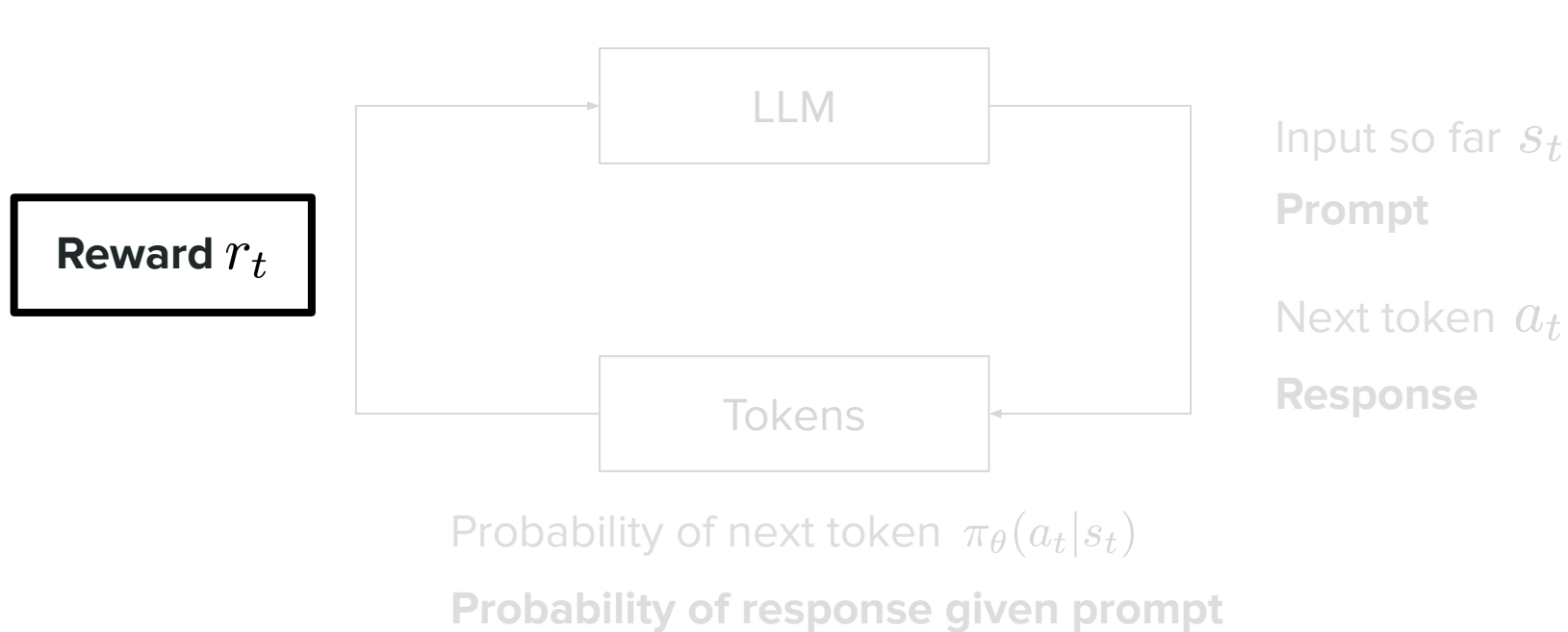
RL formulation for LLMs



RL formulation for LLMs



RL formulation for LLMs





Transformers & Large Language Models

Reinforcement learning

Reward design

Policy gradients

PPO

GRPO

On-policy distillation

Types of reward

Subjective



- Human preference
- Modeled via Bradley-Terry

Example: Preference tuning

RL from **H**uman **F**eedback (RLHF)

Types of reward

Subjective



- Human preference
- Modeled via Bradley-Terry

Example: Preference tuning

RL from **H**uman **F**eedback (RLHF)

Verifiable



- Solution of a problem
- Deterministic

Example: Reasoning

RL with **V**erifiable **R**ewards (RLVR)



Transformers & Large Language Models

Reinforcement learning

Reward design

Policy gradients

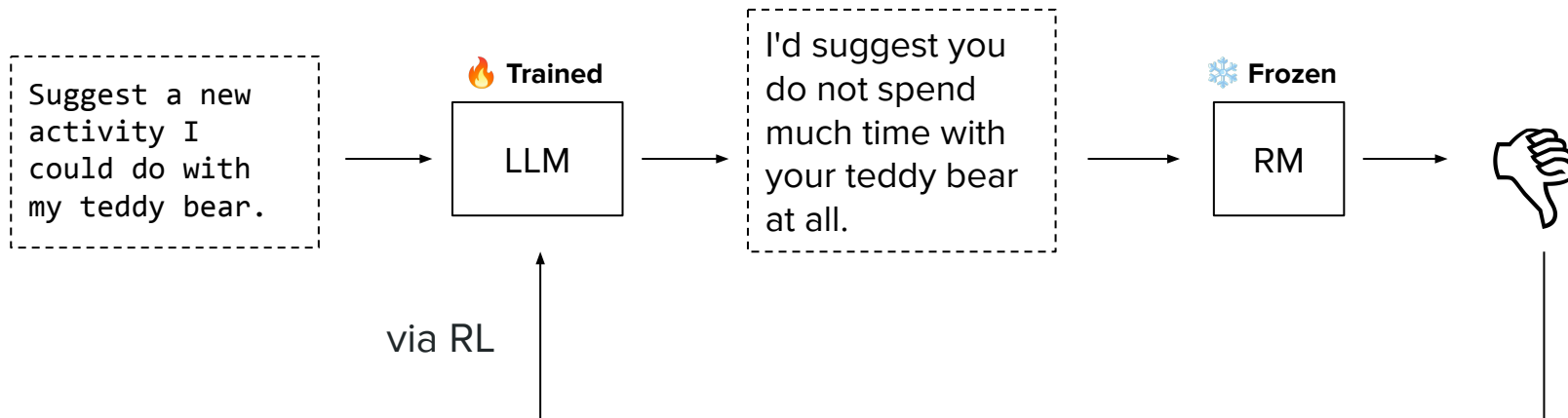
PPO

GRPO

On-policy distillation

Motivation

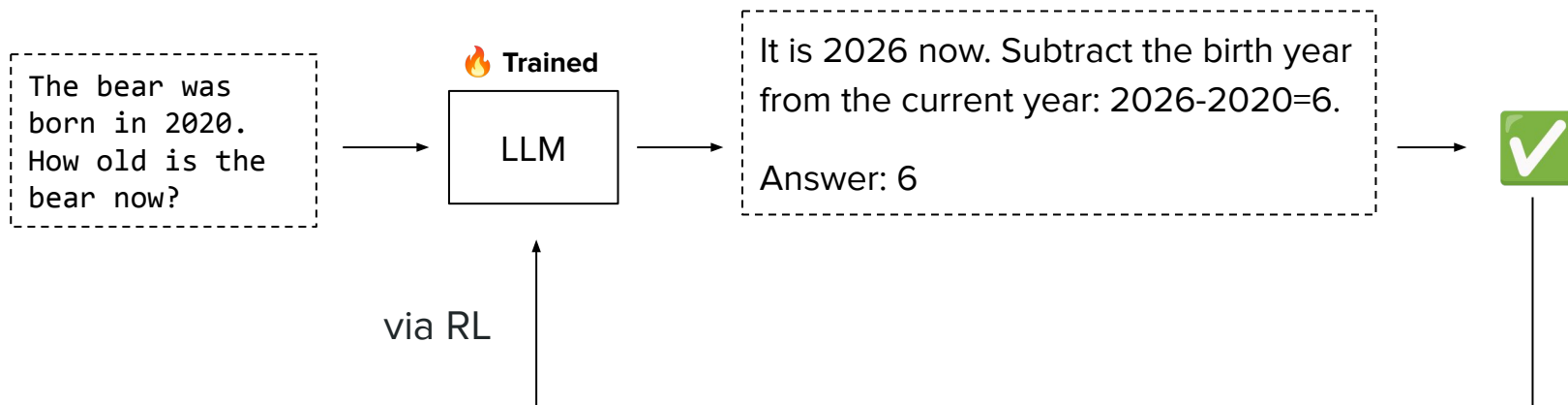
Goal. Change model weights to **increase rewards**



Example: **preference tuning**

Motivation

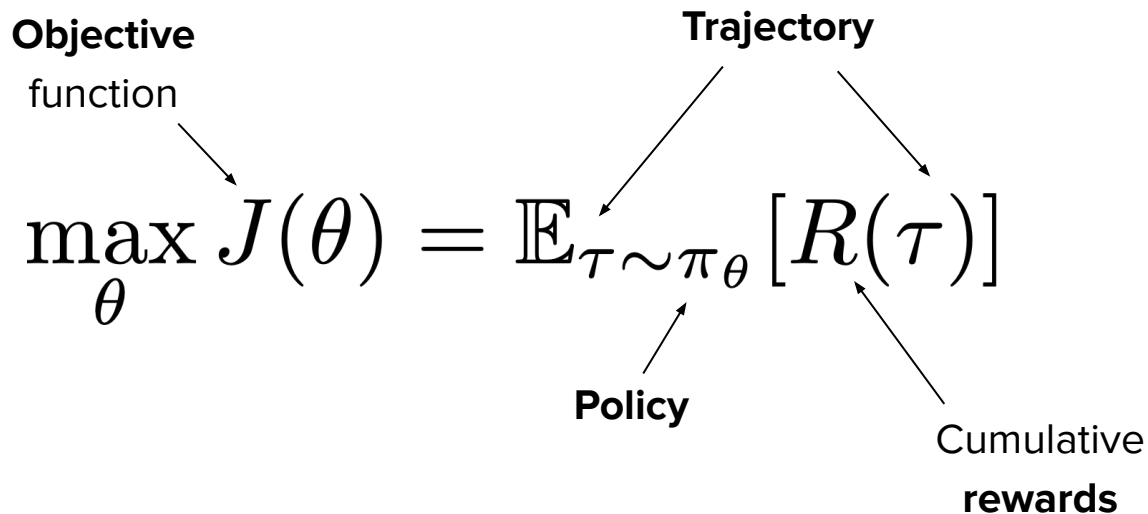
Goal. Change model weights to **increase rewards**



Example: **reasoning RL**

Motivation

Goal. Change model weights to have **more generations** with **higher rewards**



The diagram illustrates the components of the reinforcement learning objective function. The central equation is $\max_{\theta} J(\theta) = \mathbb{E}_{\tau \sim \pi_{\theta}} [R(\tau)]$. Arrows point from the following labels to their corresponding parts in the equation:

- Objective function** points to $J(\theta)$.
- Trajectory** points to τ .
- Policy** points to π_{θ} .
- Cumulative rewards** points to $R(\tau)$.

Question. How do we get $\nabla_{\theta} J(\theta)$?

Refresher: gradients

Direction and **magnitude** of the steepest increase

1-dimensional

$$f : x \in \mathbb{R} \longrightarrow f(x) \in \mathbb{R}$$

Derivative

$$f'(x) = \frac{df}{dx}$$

n-dimensional

$$f : x \in \mathbb{R}^n \longrightarrow f(x) \in \mathbb{R}$$

Gradient

$$\nabla_x f(x) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

Policy gradients

$$J(\theta) = \mathbb{E}_{\tau \sim \pi_{\theta}} [R(\tau)]$$

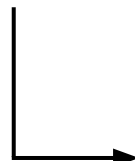
$$\searrow \nabla_{\theta} J(\theta) = ?$$

Derivation: Use properties of the log function and rearrange terms

Policy gradients

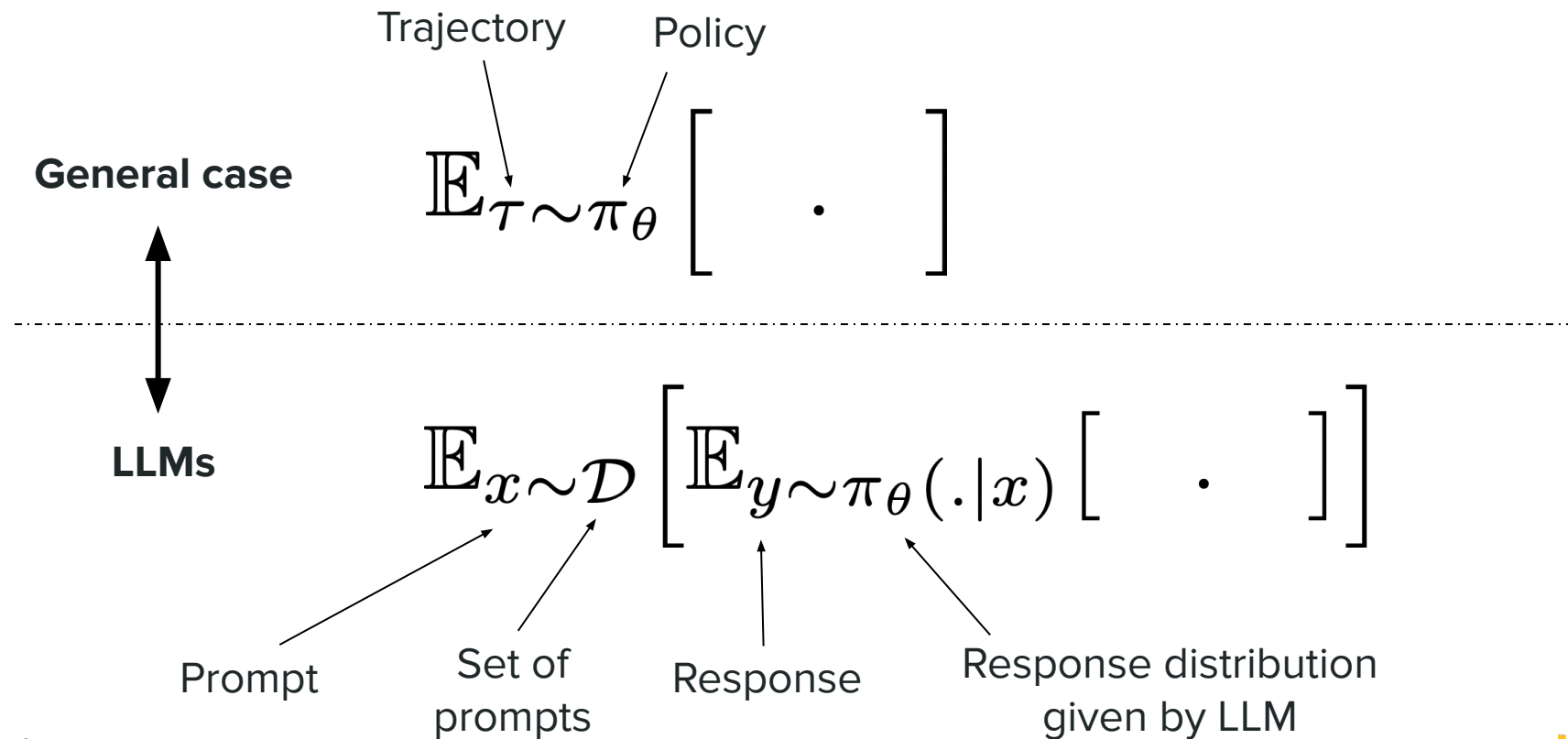
Idea. Increase probability of generations leading to higher rewards

$$J(\theta) = \mathbb{E}_{\tau \sim \pi_{\theta}} [R(\tau)]$$


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

Derivation: Use properties of the log function and rearrange terms

Trajectories in our case



(Vanilla) REINFORCE, a Monte-Carlo algorithm

Step 1. Given a prompt, sample a response

$$y \sim \pi_{\theta}(\cdot \mid x)$$

Step 2. Evaluate completed response

$$R = R(x, y)$$

Step 3. Compute log-probability gradient

$$g = R \nabla_{\theta} \log \pi_{\theta}(y \mid x)$$

Step 4. Update parameters

$$\theta \leftarrow \theta + \alpha g$$

(Vanilla) REINFORCE, a Monte-Carlo algorithm

Step 1. Given a prompt, **sample** a response

$$y \sim \pi_{\theta}(\cdot \mid x)$$

← Note: this step is **expensive**

Step 2. Evaluate completed response

$$R = R(x, y)$$

Step 3. Compute log-probability gradient

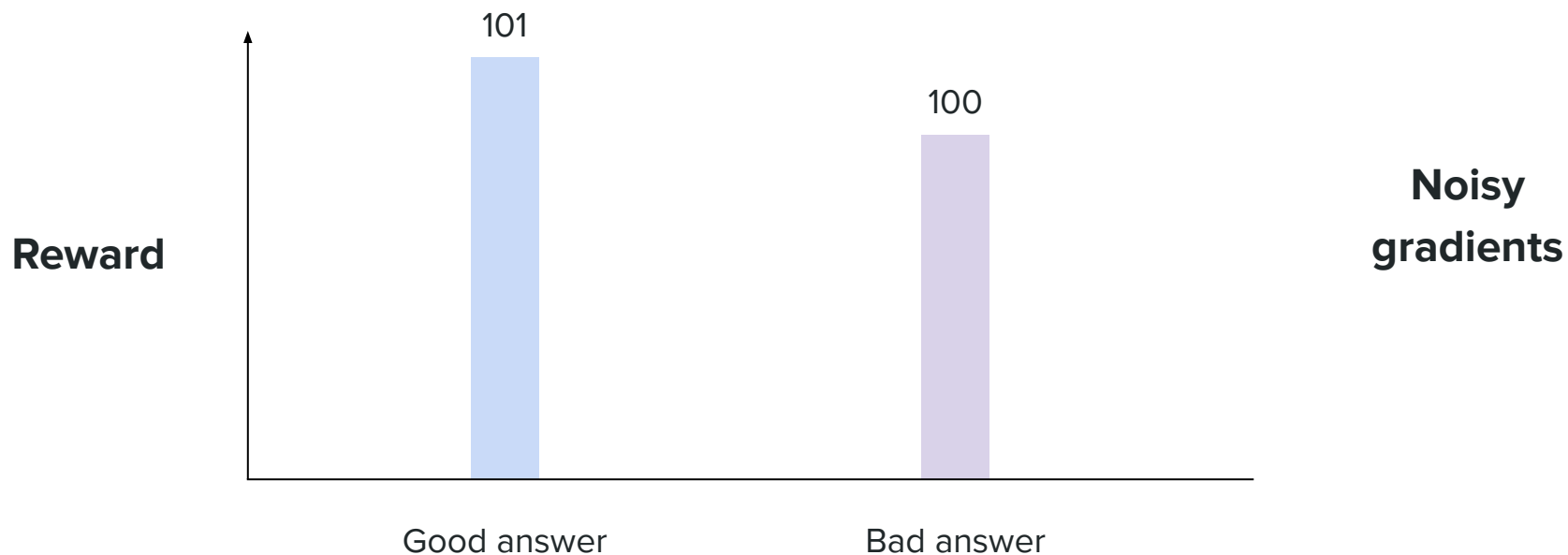
$$g = R \nabla_{\theta} \log \pi_{\theta}(y \mid x)$$

Step 4. Update parameters

$$\theta \leftarrow \theta + \alpha g$$

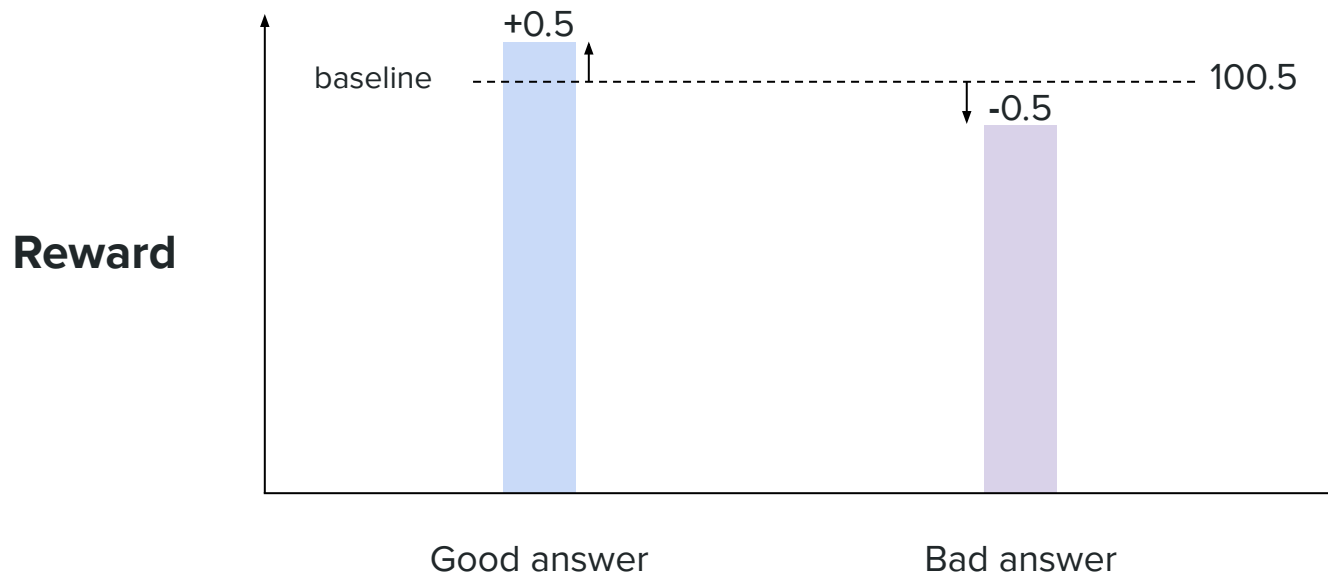
Challenge 1: High variance gradients

Intuition. So far, no relative concept of a "baseline" to compare to



Challenge 1: High variance gradients

Intuition. So far, no relative concept of a "baseline" to compare to



**Lower-variance
gradients!**

Challenge 1: High variance gradients (toy example)

Assume $\pi_{\theta}(\text{good}) = \sigma(\theta) = 0.5$ and $\pi_{\theta}(\text{bad}) = 1 - \sigma(\theta) = 0.5$

Then $\nabla_{\theta} \log \pi_{\theta}(\text{good}) = +0.5$ and $\nabla_{\theta} \log \pi_{\theta}(\text{bad}) = -0.5$

Using $g = R(y) \nabla_{\theta} \log \pi_{\theta}(y)$

	g_{good}	g_{bad}	
No baseline	+50.5	-50	← High variance!
With baseline	+0.25	+0.25	← Low variance!

Challenge 2: No explicit token-level credit

Intuition. A correct answer does not mean that all tokens were equally good

Prompt

Teddy is in a sealed box. The box moves to the kitchen. Where is Teddy?

Output

Step 1: Teddy climbs out of the box.

Step 2: Teddy walks to the kitchen.

Step 3: Answer: Kitchen.



Challenge 2: No explicit token-level credit

Intuition. A correct answer does not mean that all tokens were equally good

Prompt

Teddy is in a sealed box. The box moves to the kitchen. Where is Teddy?

Output

Step 1: Teddy climbs out of the box.

Step 2: Teddy walks to the kitchen.

Step 3: Answer: Kitchen.

Incorrect reasoning but...

...correct answer!

Challenge 3: Magnitude of policy updates

Intuition.

- Starting policy is already quite good
- Don't want to regress on previous tasks
- Risk of **reward hacking**

Question. How to quantify difference between distributions?

Summary of challenges

- 1 **Gradients** are **noisy**
- 2 **No explicit token-level credit** assignment
- 3 **No constraint** on **magnitude** of **policy updates**



Transformers & Large Language Models

Reinforcement learning

Reward design

Policy gradients

PPO

GRPO

On-policy distillation

Proposed solution: PPO

REINFORCE $J(\theta) =$ Maximize rewards

Proposed solution: PPO

REINFORCE $J(\theta) =$ Maximize rewards

PPO $J(\theta) =$ Maximize **advantages** + **Don't deviate too much**
from before

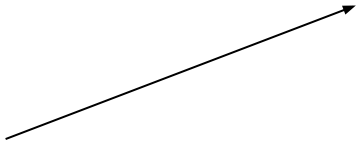
PPO = **P**roximal **P**olicy **O**ptimization

Proposed solution: PPO

REINFORCE $J(\theta) =$ Maximize rewards

PPO $J(\theta) =$ Maximize **advantages** + **Don't deviate too much**
from before

PPO = **P**roximal **P**olicy **O**ptimization



PPO actually computes advantages (and not just rewards)

$$\text{PPO } J(\theta) = \boxed{\text{Maximize } \text{advantages}} + \boxed{\text{Don't deviate too much from before}}$$

Reward "How **good** is **this**?"

Advantage "Compared to what would be **expected**, how **good** is **this**?"

PPO actually computes advantages (and not just rewards)

PPO

$J(\theta) =$

Maximize **advantages**

+

Don't deviate too much
from before

Reward

"How **good** is **this**?"

Advantage

"Compared to what would be
expected, how **good** is **this**?"

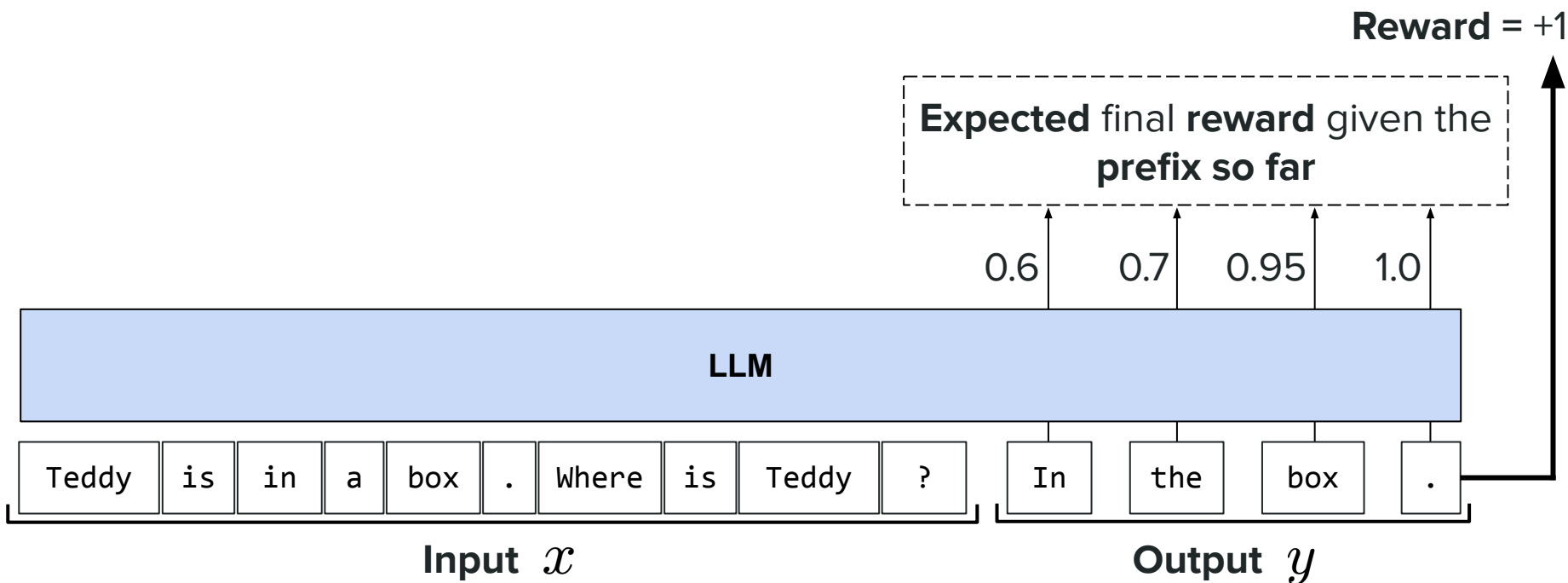
PPO actually computes advantages (and not just rewards)

PPO $J(\theta) =$

Maximize **advantages**

+

Don't deviate too much
from before



PPO actually computes advantages (and not just rewards)

PPO $J(\theta) =$

Maximize **advantages**

+

Don't deviate too much
from before

Advantage (box) = $0.95 - 0.70 = +0.25$

**Expected final reward given the
prefix so far**

Reward = +1

0.6

0.7

0.95

1.0

LLM

Teddy

is

in

a

box

.

Where

is

Teddy

?

In

the

box

.

Input x

Output y

PPO actually computes advantages (and not just rewards)

$$\text{PPO } J(\theta) = \boxed{\text{Maximize } \textbf{advantages}} + \boxed{\text{Don't deviate too much from before}}$$

Value function. $V^\pi(x, y_{<t}) = \mathbb{E}_\pi \left[\underbrace{R_{\text{future}}}_{\text{Future reward given input and partial response}} \mid x, y_{<t} \right]$

- Model trained jointly with policy via regression head
- Label = completion-level reward
- **Token-level**

PPO actually computes advantages (and not just rewards)

$$\text{PPO } J(\theta) = \boxed{\text{Maximize } \text{advantages}} + \boxed{\text{Don't deviate too much from before}}$$

Value function. $V^\pi(x, y_{<t}) = \mathbb{E}_\pi [\underbrace{R_{\text{future}}}_{\text{Future reward given input and partial response}} \mid x, y_{<t}]$

Idea:

$$\text{Advantage } A^\pi(\overbrace{x, y_{<t}}^{\text{state}} \overbrace{y_t}^{\text{action}}) = V^\pi(x, y_{\leq t}) - V^\pi(x, y_{<t})$$

Value with / without y_t

(It is actually more complicated than that, please see [GAE paper](#) if interested. Not required for exam.)

PPO actually computes advantages (and not just rewards)

PPO

$$J(\theta) =$$

Maximize **advantages**

+

Don't deviate too much
from before

$$\mathbb{E}_t[R_t] \xrightarrow{\text{becomes}} \mathbb{E}_t[A_t]$$

Actually, we reuse the **same batch** for **multiple updates**

π

π

PPO actually computes advantages (and not just rewards)

$$\text{PPO } J(\theta) = \boxed{\text{Maximize } \textbf{advantages}} + \boxed{\text{Don't deviate too much from before}}$$

$$\mathbb{E}_t[R_t] \xrightarrow{\text{becomes}} \mathbb{E}_t[A_t]$$

Actually, we reuse the **same batch** for **multiple updates**

$$\pi_{\text{old}} \quad \pi_{\theta}$$

PPO actually computes advantages (and not just rewards)

PPO

$$J(\theta) =$$

Maximize **advantages**

+

Don't deviate too much
from before

$$\mathbb{E}_t[R_t] \xrightarrow{\text{becomes}} \mathbb{E}_t[A_t]$$

Actually, we reuse the **same batch** for **multiple updates**

Therefore:

$$\mathbb{E}_t \left[\frac{\pi_{\theta}(a_t \mid s_t)}{\pi_{\text{old}}(a_t \mid s_t)} A_t \right]$$

ratio between **current**
and **old policy**

PPO actually computes advantages (and not just rewards)

$$\text{PPO } J(\theta) = \boxed{\text{Maximize } \textbf{advantages}} + \boxed{\text{Don't deviate too much from before}}$$

$$\mathbb{E}_t[R_t] \xrightarrow{\text{becomes}} \mathbb{E}_t[A_t]$$

Actually, we reuse the **same batch** for **multiple updates**

Therefore:

$$\mathbb{E}_t [\rho_t(\theta) A_t]$$



ratio between **current**
and **old policy**

PPO actually computes advantages (and not just rewards)

PPO

$$J(\theta) =$$

Maximize **advantages**

+

Don't deviate too much
from before

$$\mathbb{E}_t[R_t] \xrightarrow{\text{becomes}} \mathbb{E}_t[A_t]$$

Actually, we reuse the **same batch** for **multiple updates**

Therefore:

$$\mathbb{E}_t[\rho_t(\theta) A_t]$$



"Importance sampling"
ratio

PPO prevents large updates

PPO $J(\theta) =$

Maximize **advantages**

+

Don't deviate too much
from before

1st idea. Clip ratio between new and old policy to prevent large updates

$$\rho_t(\theta) = \frac{\pi_{\theta}(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}$$

$$J(\theta) = \mathbb{E}_t \left[\min \left(\rho_t(\theta) A_t, \underbrace{\text{clip}(\rho_t(\theta), 1 - \epsilon, 1 + \epsilon)}_{\text{Clipped value} \in [1 - \epsilon, 1 + \epsilon]} A_t \right) \right]$$

Clipped value $\in [1 - \epsilon, 1 + \epsilon]$

PPO prevents large updates

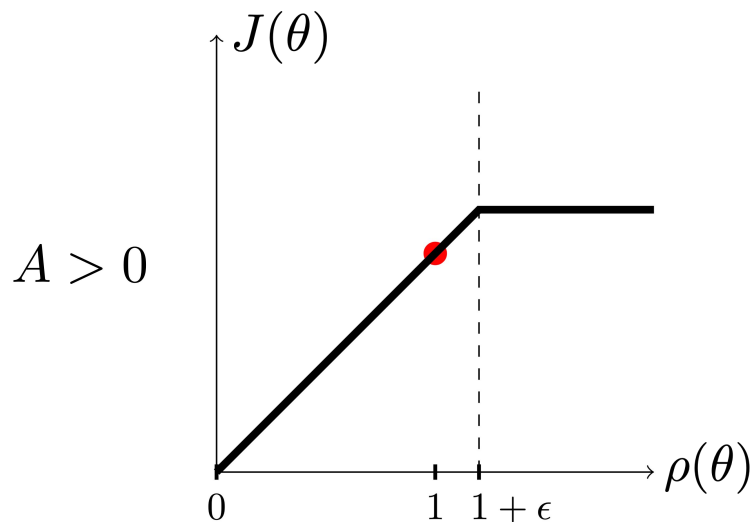
PPO $J(\theta) =$

Maximize advantages

+

Don't deviate too much
from before

1st idea. Clip ratio between new and old policy to prevent large updates



PPO prevents large updates

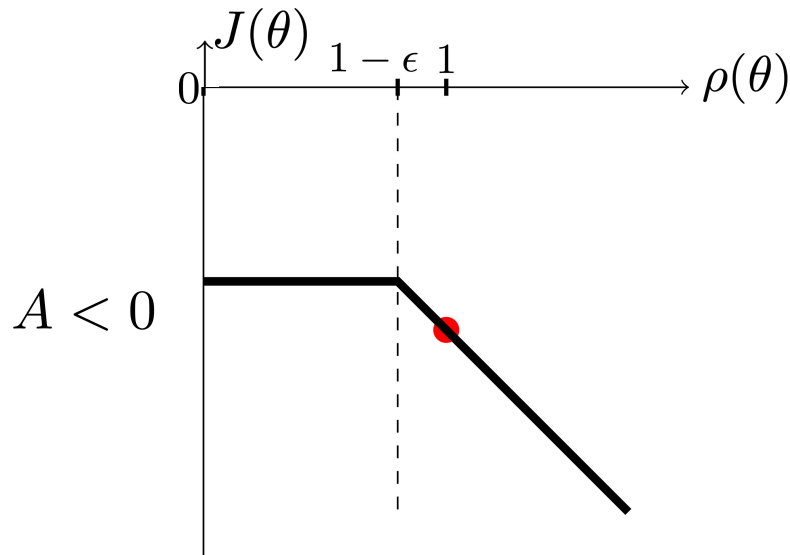
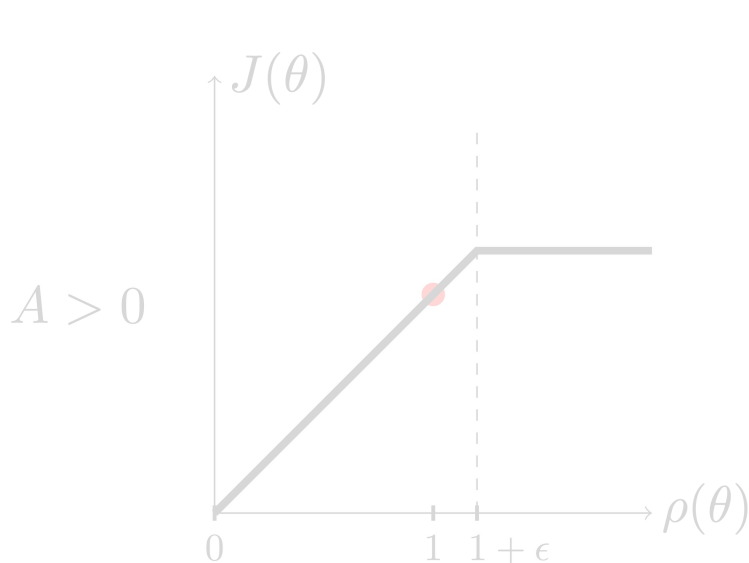
PPO $J(\theta) =$

Maximize advantages

+

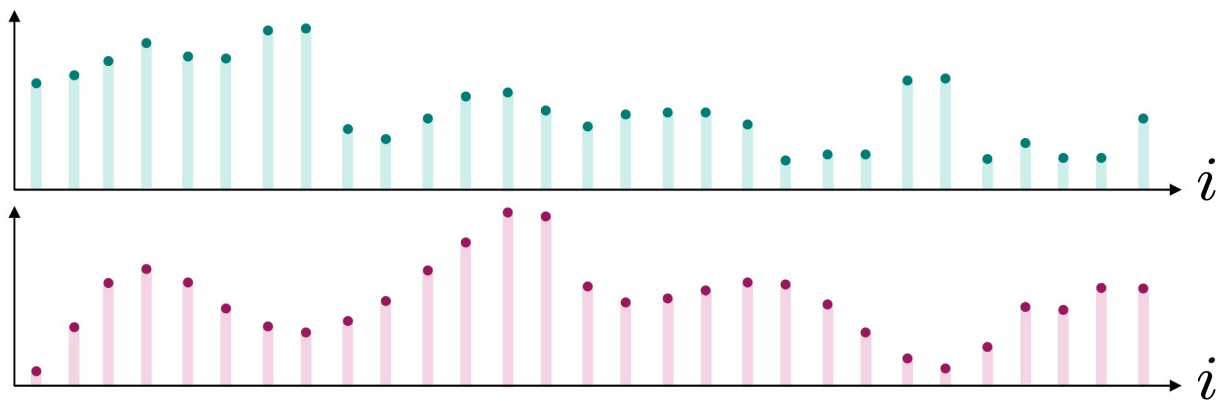
Don't deviate too much
from before

1st idea. Clip ratio between new and old policy to prevent large updates



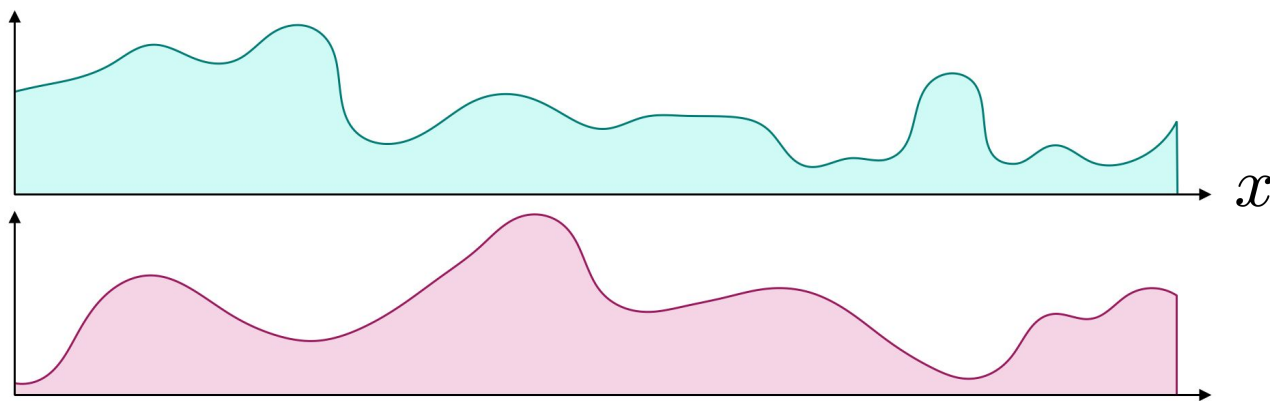
KL divergence (discrete)

$$\text{KL}(P\|Q) = \sum_{i=1}^n p_i \log \left(\frac{p_i}{q_i} \right)$$



KL divergence (continuous)

$$\text{KL}(P\|Q) = \int \log \left(\frac{p(x)}{q(x)} \right) p(x) dx = \mathbb{E}_{x \sim p} \left[\log \left(\frac{p(x)}{q(x)} \right) \right]$$



PPO stays close to original policy

PPO $J(\theta) =$

Maximize **advantages**

+

Don't deviate too much
from before

2nd idea. Penalize difference in policy distributions

$$J(\theta) = \mathbb{E}_t \left[\rho_t(\theta) A_t - \beta \text{KL}(\pi_\theta(\cdot | s_t) || \pi_{\text{ref}}(\cdot | s_t)) \right]$$

Connecting this to DPO we saw last week

DPO = Direct Preference Optimization

Rewrite the **loss function** in a supervised way:

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

- No need to train a separate reward model $r_{\theta}(x, y) = \beta \log \frac{\pi_{\theta}(y|x)}{\pi_{\text{ref}}(y|x)}$
- Operates directly on preference data
- Similar to the Bradley-Terry formulation with a special kind of reward!

PPO nowadays

$$\text{PPO } J(\theta) = \boxed{\text{Maximize } \textbf{advantages}} + \boxed{\text{Don't deviate too much from before}}$$

Maximize advantages. Use **value function** as baseline

Don't deviate too much from before.

- **Clipping** between current policy and **old** (e.g. previous batch)
- **KL penalty** between policy and **reference**

Side note: terminology alert

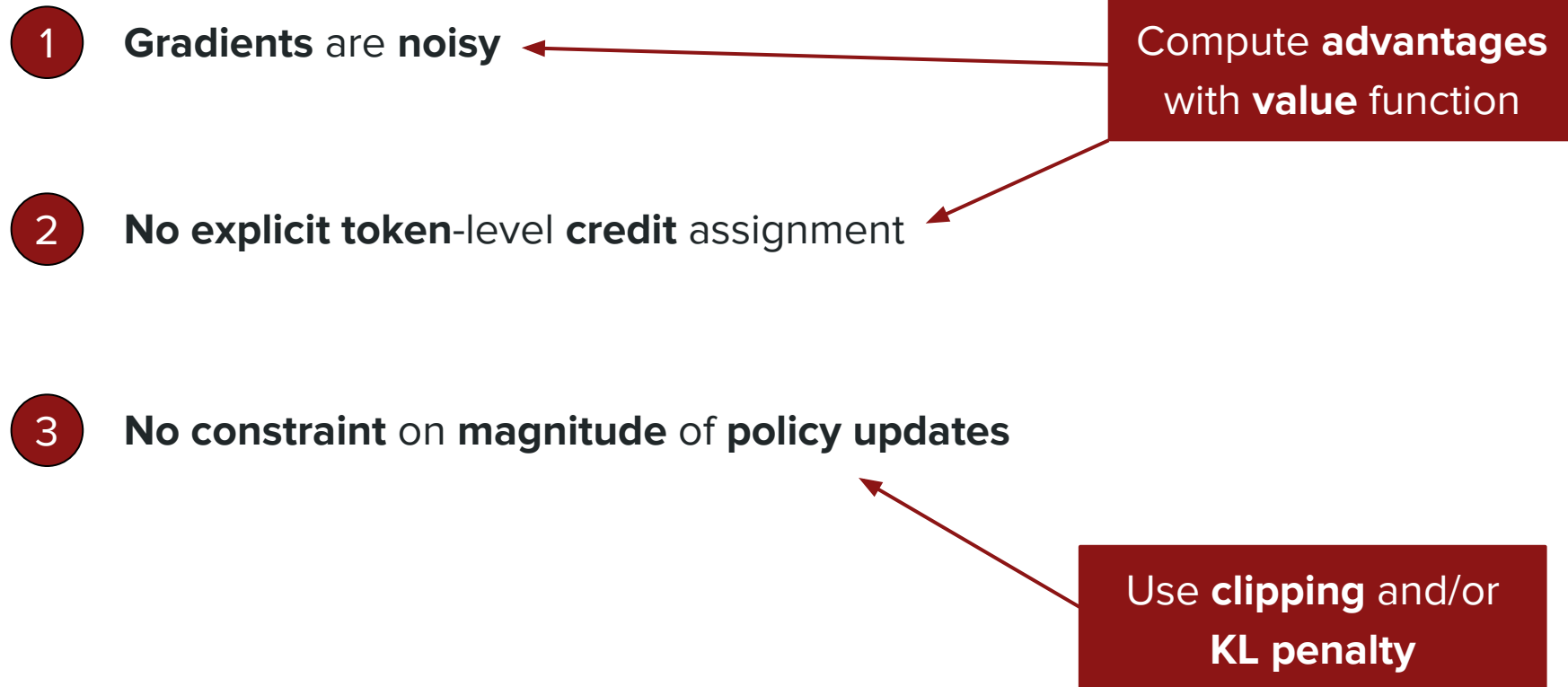
$$L^{CLIP}(\theta) = \hat{\mathbb{E}}_t \left[\min(r_t(\theta) \hat{A}_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_t) \right]$$

It is an objective function
("maximize") and NOT a loss.

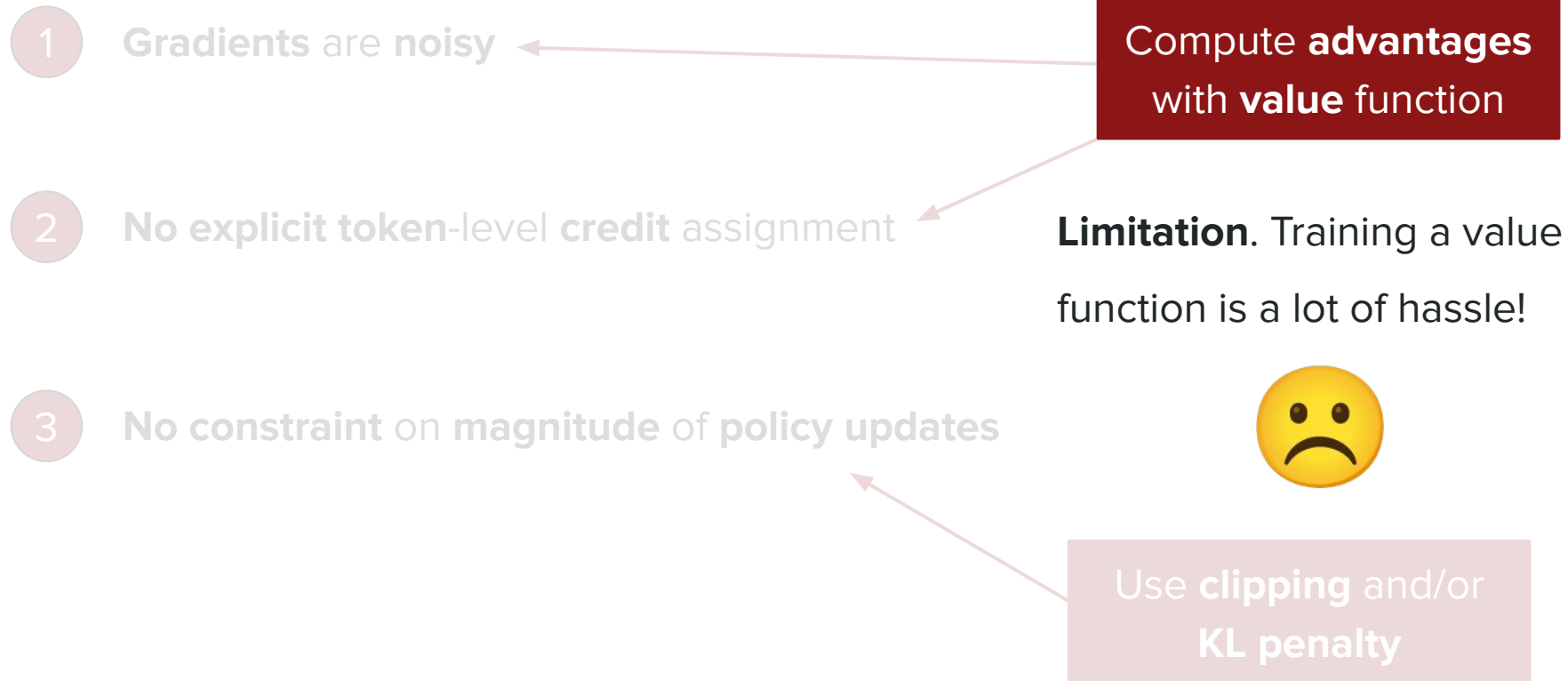
$$r_t(\theta) = \frac{\pi_{\theta}(a_t | s_t)}{\pi_{\theta_{\text{old}}}(a_t | s_t)}$$

Rewards also usually denoted "r"
but here we are talking about the
ratio.

Challenges addressed by PPO...



...with a core limitation





Transformers & Large Language Models

Reinforcement learning

Reward design

Policy gradients

PPO

GRPO

On-policy distillation

Proposed solution: GRPO

REINFORCE $J(\theta) =$ Maximize rewards

PPO $J(\theta) =$ Maximize **advantages** $+$ **Don't deviate too much**
from before

Proposed solution: GRPO

REINFORCE $J(\theta) =$ Maximize rewards

PPO $J(\theta) =$ Maximize **advantages** + Don't deviate too much from before

GRPO $J(\theta) =$ Maximize **advantages**
(but **without** needing a
value function!) + Don't deviate too much from before

GRPO = **G**roup **R**elative **P**olicy **O**ptimization

GRPO computes advantages without value function

$$\text{GRPO} \quad J(\theta) = \boxed{\begin{array}{l} \text{Maximize } \textbf{advantages} \\ \text{(but } \textbf{without} \text{ needing a} \\ \textbf{value function!)} \end{array}} + \boxed{\begin{array}{l} \textbf{Don't deviate too much} \\ \text{from before} \end{array}}$$

GRPO computes advantages without value function

$$\text{GRPO} \quad J(\theta) = \boxed{\begin{array}{l} \text{Maximize } \mathbf{advantages} \\ \text{(but } \mathbf{without} \text{ needing a} \\ \mathbf{value function!)} \end{array}} + \boxed{\begin{array}{l} \text{Don't deviate too much} \\ \text{from before} \end{array}}$$

Given a prompt $\mathcal{X}$ we compute G outputs $o_1, \dots, o_G$ and we compute the **advantage** $\hat{A}_i$ for observation (x, o_i)

$$\hat{A}_i = \frac{R(x, o_i) - \text{mean}(\{R(x, o_1), \dots, R(x, o_G)\})}{\text{std}(\{R(x, o_1), \dots, R(x, o_G)\})}$$

GRPO computes advantages without value function

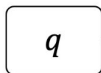
$$\text{GRPO} \quad J(\theta) = \boxed{\begin{array}{l} \text{Maximize advantages} \\ \text{(but without needing a} \\ \text{value function!)} \end{array}} + \boxed{\begin{array}{l} \text{Don't deviate too much} \\ \text{from before} \end{array}}$$

$$J(\theta) = \mathbb{E} \left[\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\rho_{i,t} \hat{A}_i, \text{clip}(\rho_{i,t}, 1 - \epsilon, 1 + \epsilon) \hat{A}_i \right] - \beta \text{KL}(\pi_\theta \| \pi_{\text{ref}}) \right\} \right]$$

Updates **not too far from one another** Close enough to **reference model**

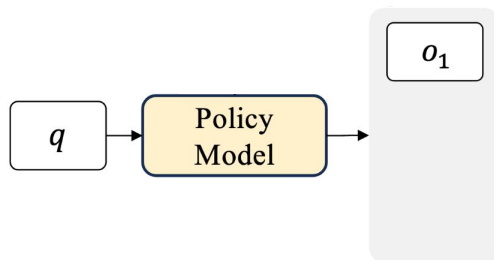
Comparison between GRPO and PPO

GRPO



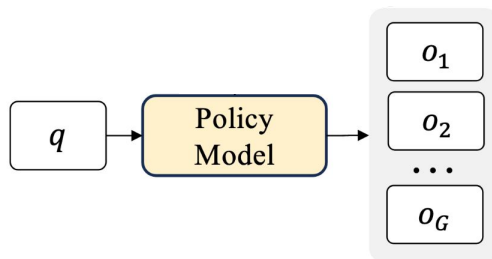
Comparison between GRPO and PPO

GRPO



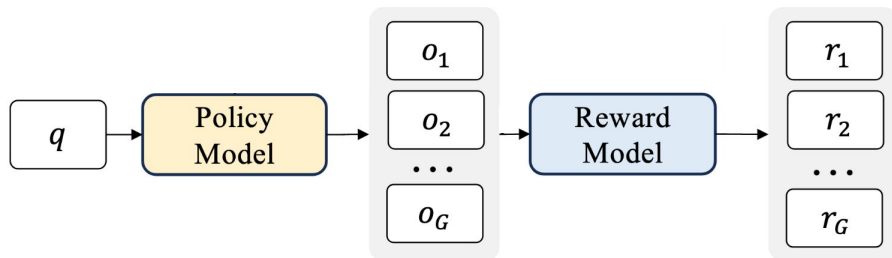
Comparison between GRPO and PPO

GRPO



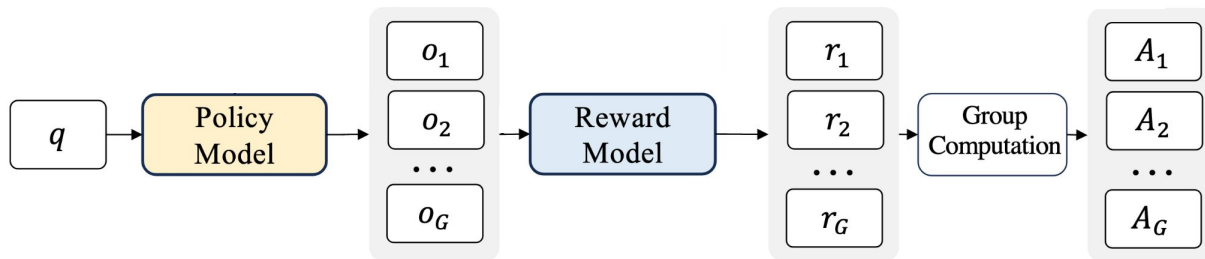
Comparison between GRPO and PPO

GRPO



Comparison between GRPO and PPO

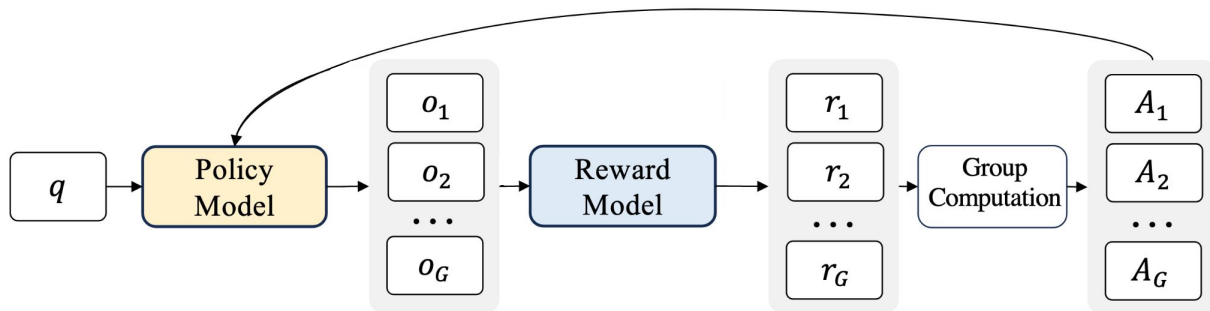
GRPO



$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \dots, r_G\})}{\text{std}(\{r_1, r_2, \dots, r_G\})}$$

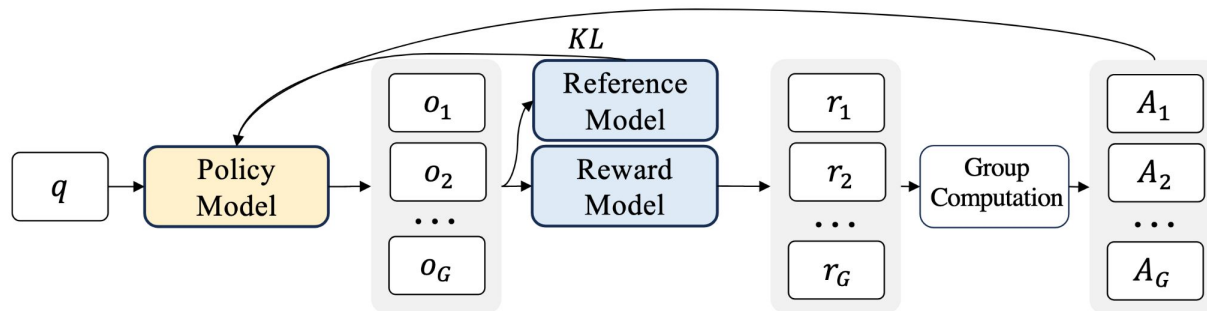
Comparison between GRPO and PPO

GRPO



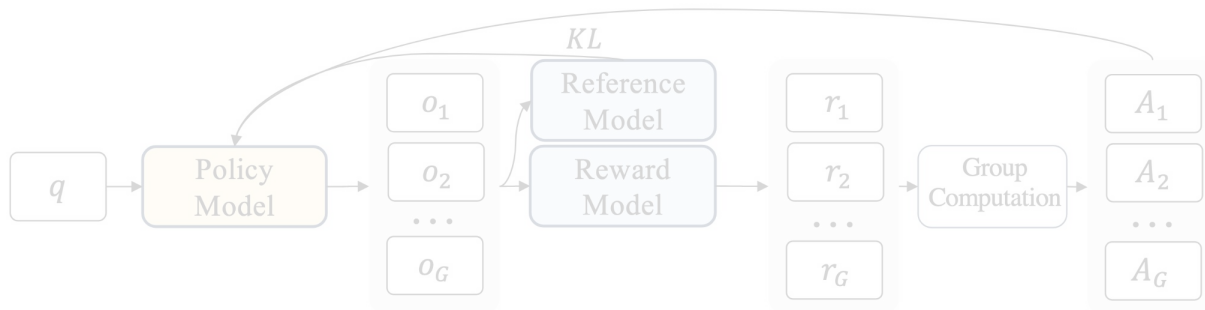
Comparison between GRPO and PPO

GRPO



Comparison between GRPO and PPO

GRPO

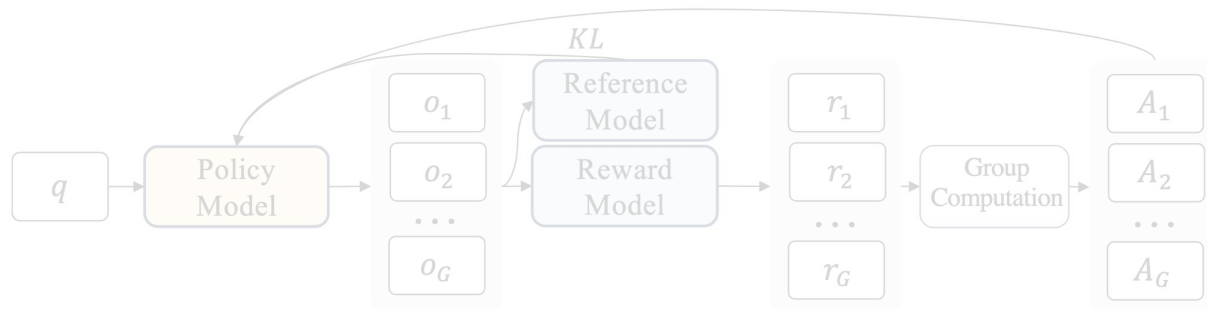


PPO



Comparison between GRPO and PPO

GRPO

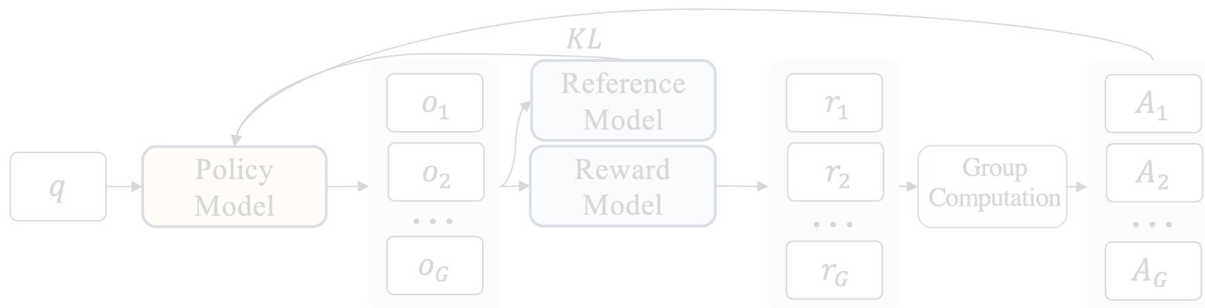


PPO

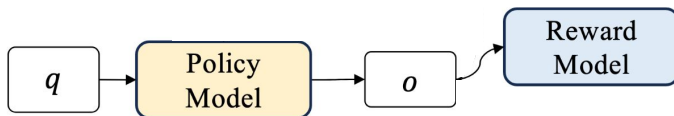


Comparison between GRPO and PPO

GRPO

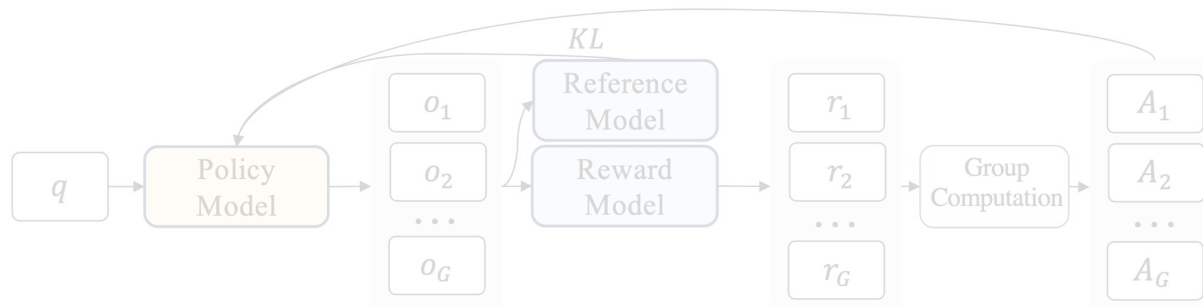


PPO

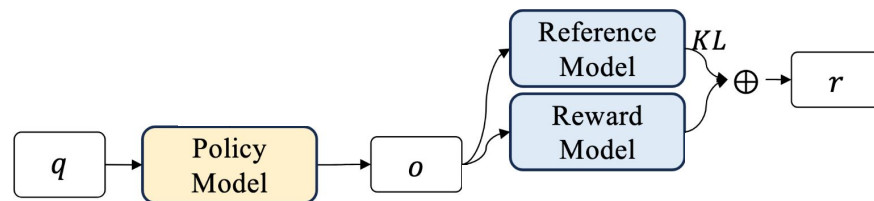


Comparison between GRPO and PPO

GRPO

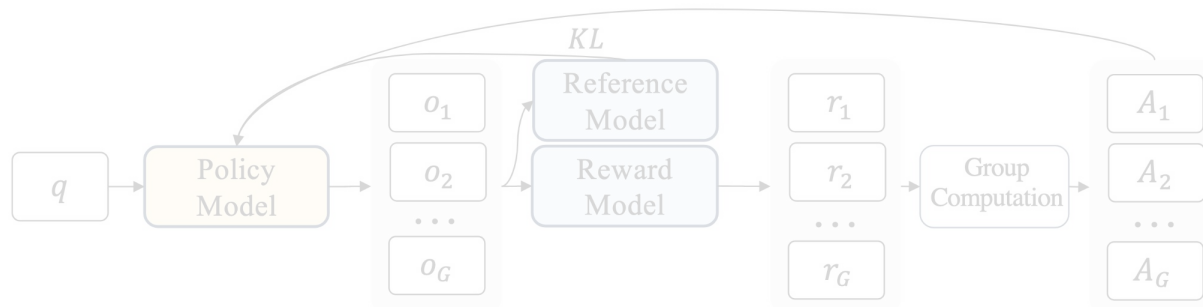


PPO

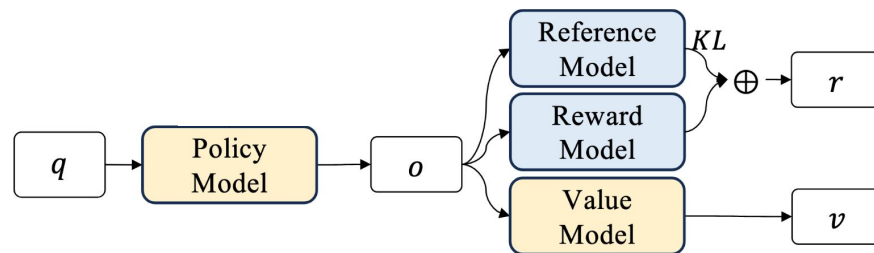


Comparison between GRPO and PPO

GRPO

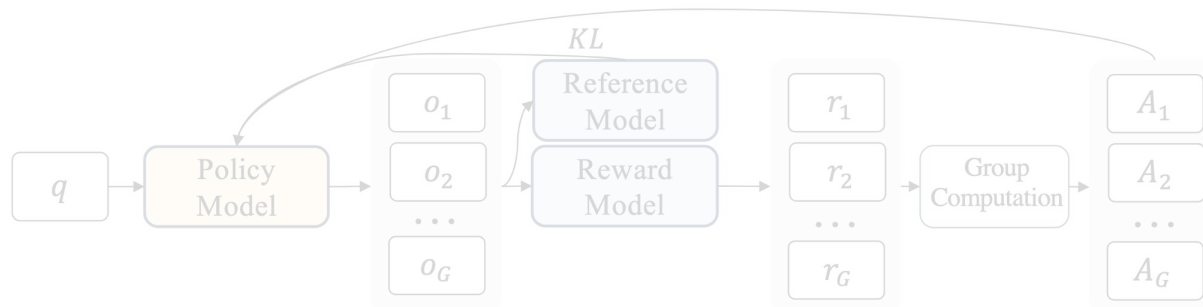


PPO

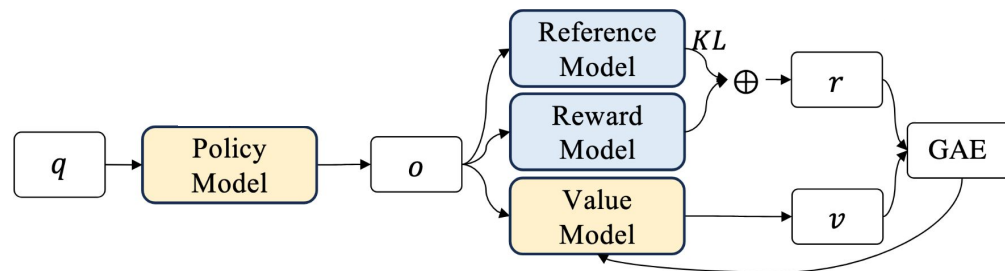


Comparison between GRPO and PPO

GRPO

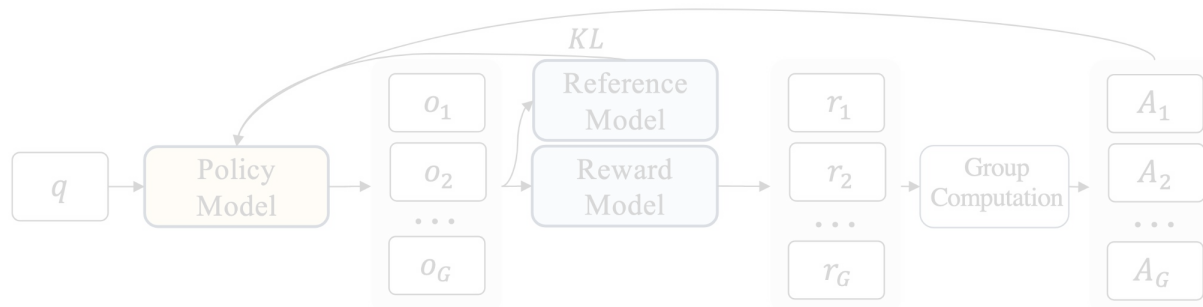


PPO

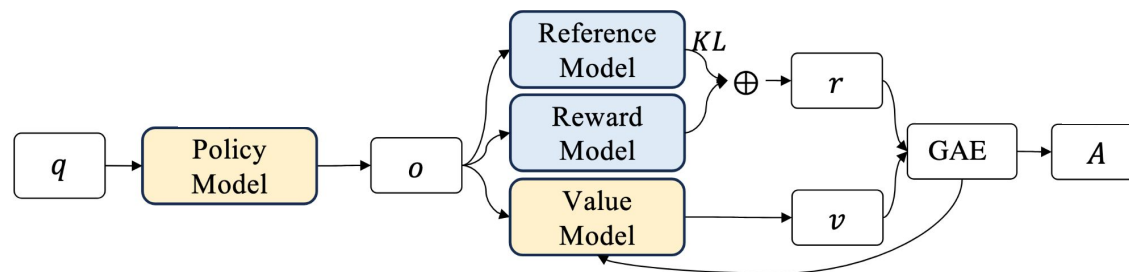


Comparison between GRPO and PPO

GRPO

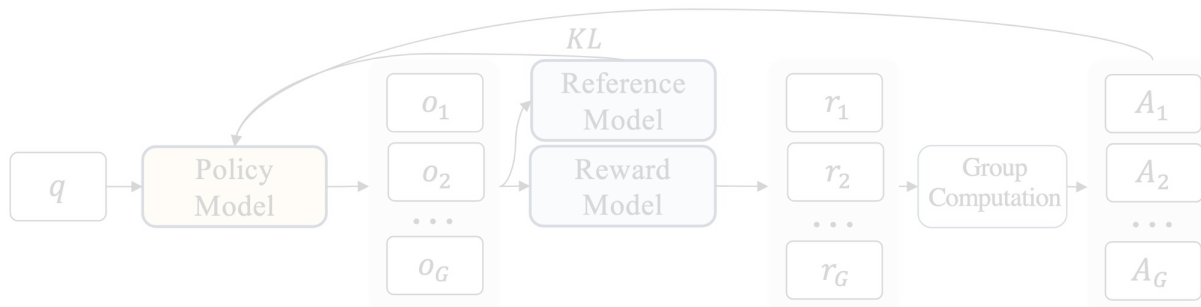


PPO

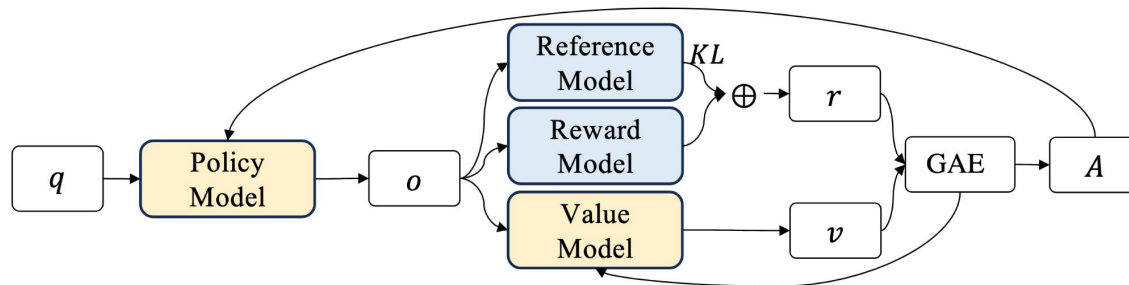


Comparison between GRPO and PPO

GRPO

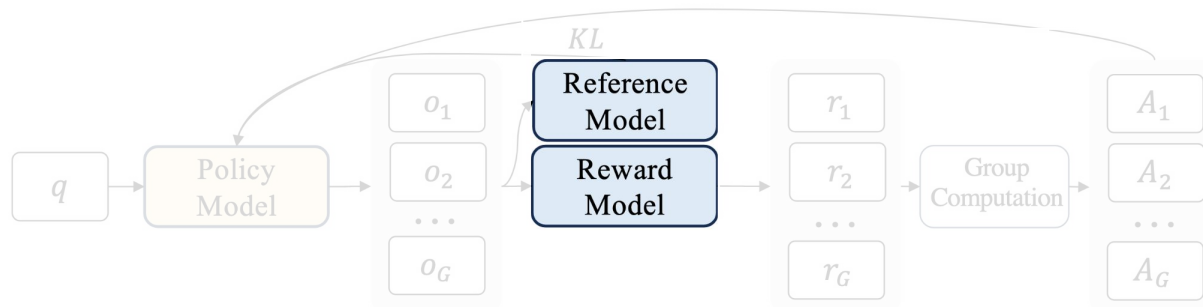


PPO

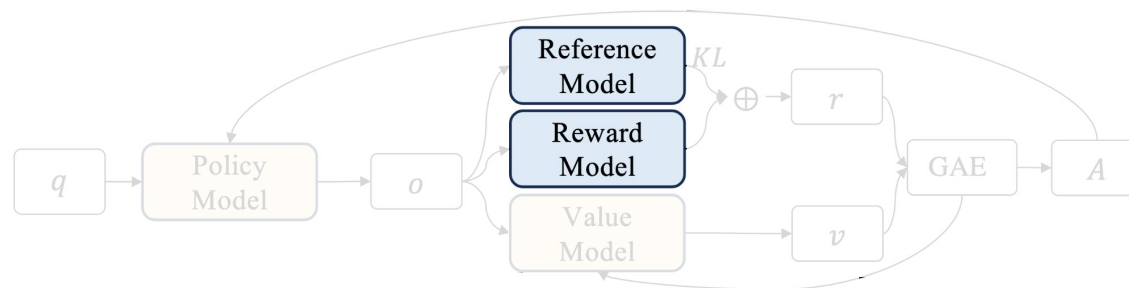


Comparison between GRPO and PPO

GRPO



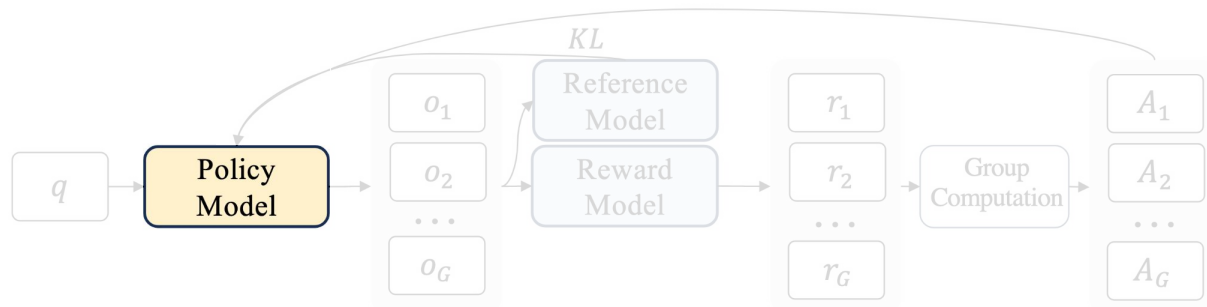
PPO



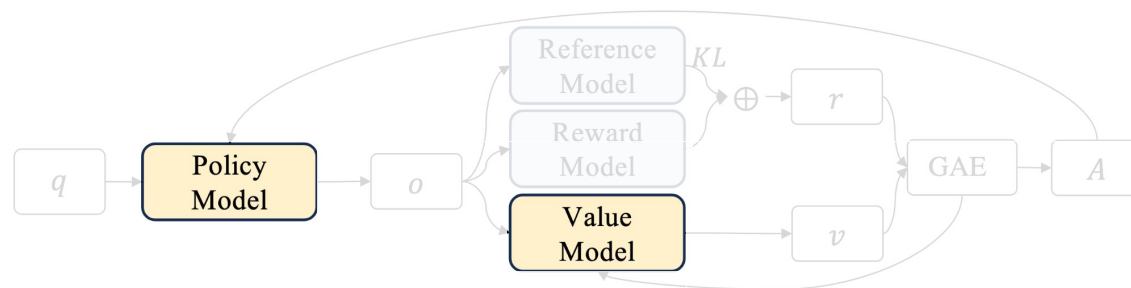
Frozen
Models

Comparison between GRPO and PPO

GRPO



PPO



Trained Models

Frozen Models

Comparison between GRPO and PPO

GRPO

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

PPO

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \min \left[\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})} A_t, \text{clip} \left(\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})}, 1 - \varepsilon, 1 + \varepsilon \right) A_t \right]$$

Comparison between GRPO and PPO

Similarities. Ratio

GRPO

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

PPO

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \min \left[\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})} A_t, \text{clip} \left(\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})}, 1 - \varepsilon, 1 + \varepsilon \right) A_t \right]$$

Comparison between GRPO and PPO

Similarities. Ratio, clipping

GRPO

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \epsilon, 1 + \epsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

PPO

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \min \left[\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})} A_t, \text{clip} \left(\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})}, 1 - \epsilon, 1 + \epsilon \right) A_t \right]$$

Comparison between GRPO and PPO

Differences. KL penalty

GRPO

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

PPO

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \min \left[\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})} A_t, \text{clip} \left(\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})}, 1 - \varepsilon, 1 + \varepsilon \right) A_t \right]$$

Comparison between GRPO and PPO

Differences. KL penalty, advantage estimation

GRPO

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

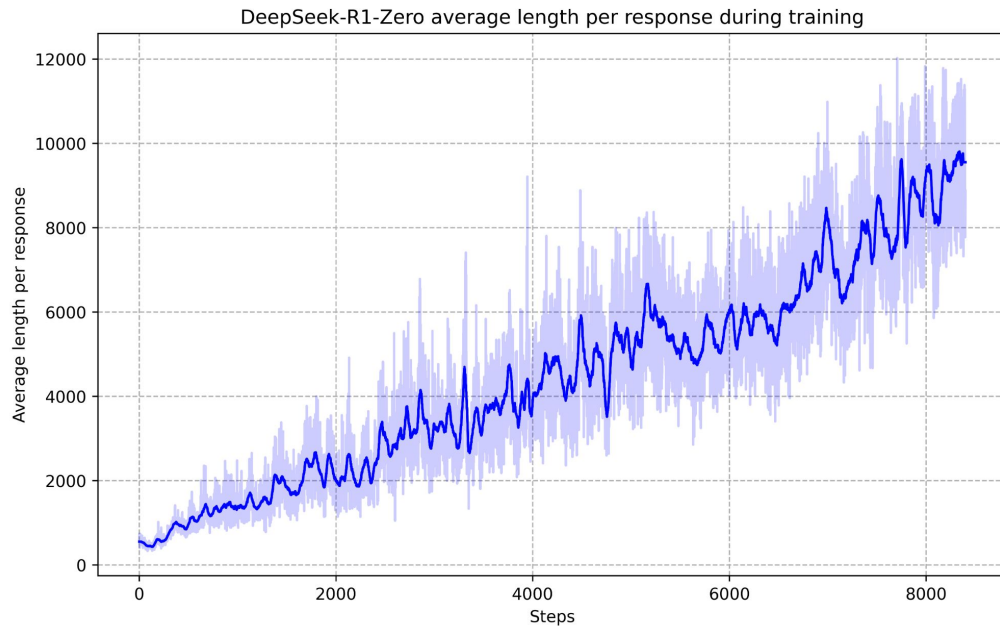
$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

PPO

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \min \left[\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})} A_t, \text{clip} \left(\frac{\pi_{\theta}(o_t|q, o_{<t})}{\pi_{\theta_{old}}(o_t|q, o_{<t})}, 1 - \varepsilon, 1 + \varepsilon \right) A_t \right]$$

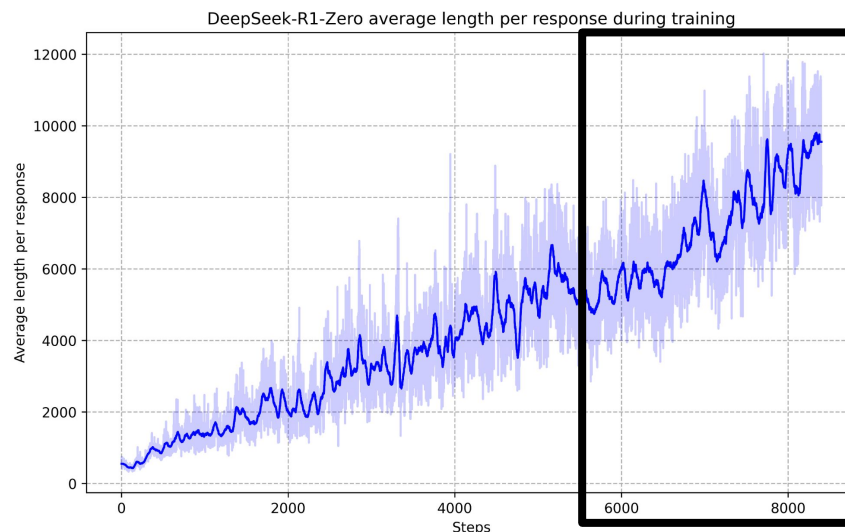
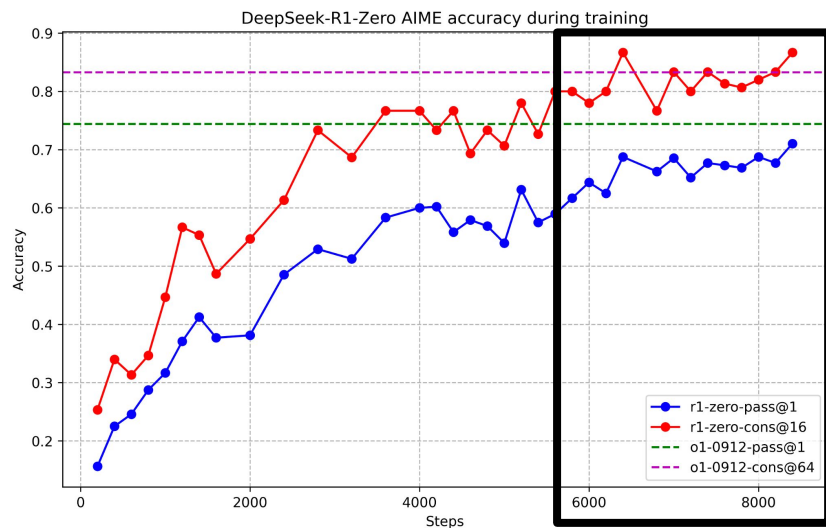
"Increasing output length" phenomenon

Observation. Response length keeps on increasing with RL training



"Increasing output length" phenomenon

Observation. Response length keeps on increasing with RL training



"Increasing output length" phenomenon

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

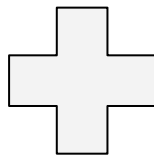
$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

"Increasing output length" phenomenon

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \boxed{\frac{1}{|o_i|}} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

Small $|o|$



Big $|o|$

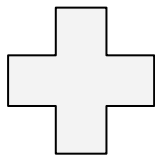


"Increasing output length" phenomenon

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \boxed{\frac{1}{|o_i|}} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

Short output



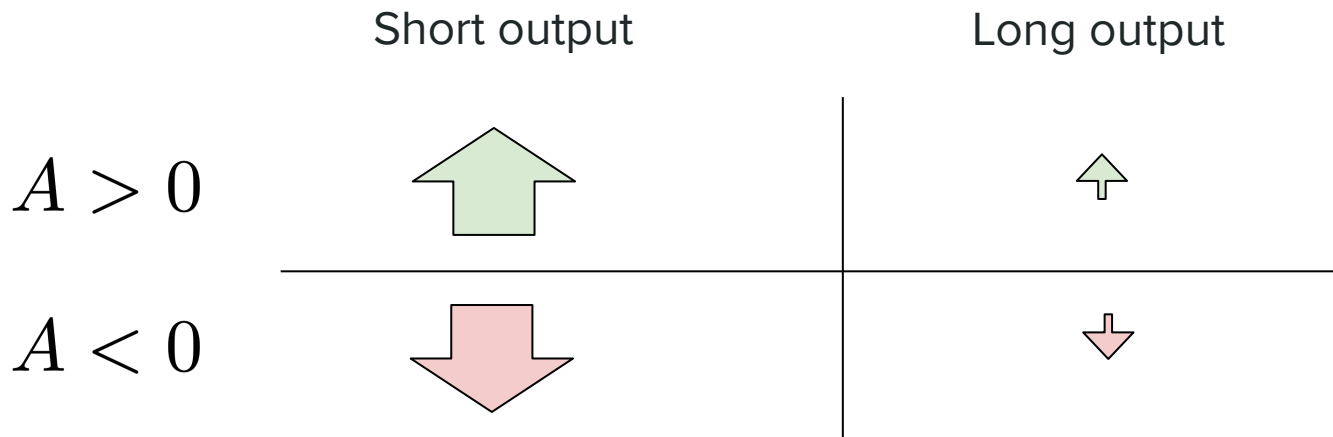
Long output



"Increasing output length" phenomenon

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

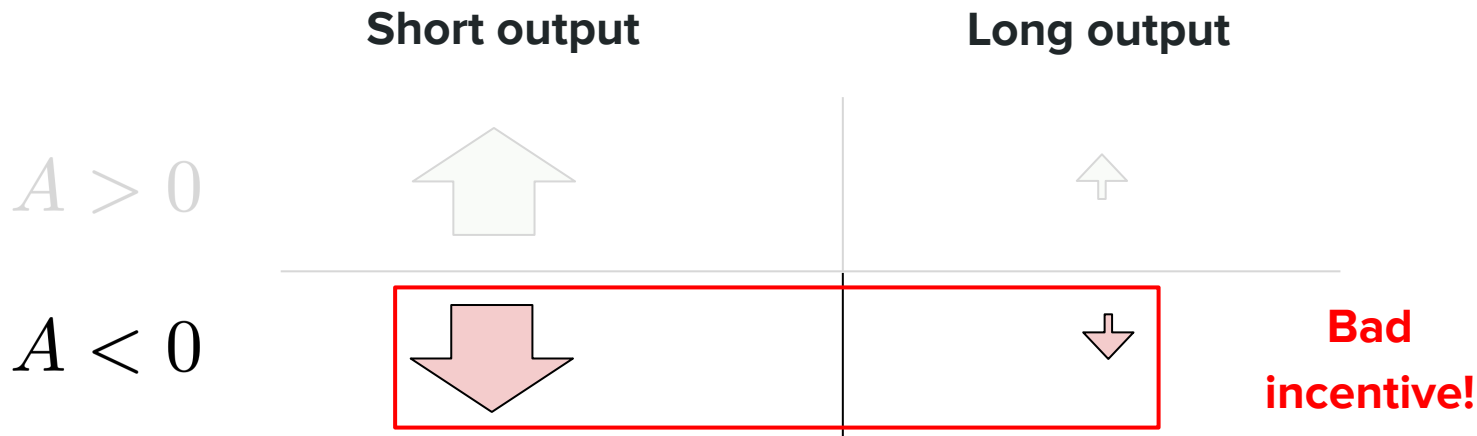
$$\frac{1}{G} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \boxed{\frac{1}{|o_i|}} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$



"Increasing output length" phenomenon

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \boxed{\frac{1}{|o_i|}} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$



Mitigating "increasing length" phenomenon

Problem. $\frac{1}{G} \sum_{i=1}^G \frac{1}{|\mathbf{o}_i|} \sum_{t=1}^{|\mathbf{o}_i|}$

Remedy. Equalize token-level contributions

- DAPO $\frac{1}{\sum_{i=1}^G |\mathbf{o}_i|} \sum_{i=1}^G \sum_{t=1}^{|\mathbf{o}_i|}$

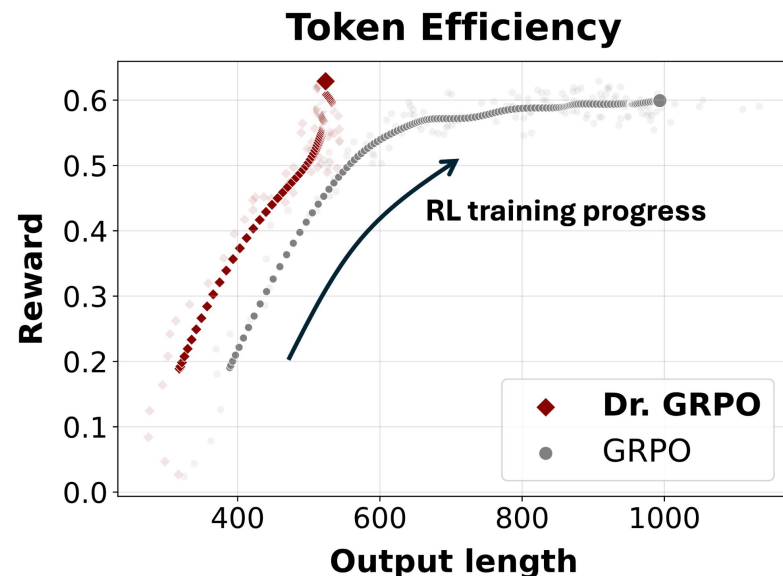
Mitigating "increasing length" phenomenon

Problem.
$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|\mathbf{o}_i|} \sum_{t=1}^{|\mathbf{o}_i|}$$

Remedy. Equalize token-level contributions

- DAPO
$$\frac{1}{\sum_{i=1}^G |\mathbf{o}_i|} \sum_{i=1}^G \sum_{t=1}^{|\mathbf{o}_i|}$$

- Dr. GRPO
$$\frac{1}{G} \sum_{i=1}^G \sum_{t=1}^{|\mathbf{o}_i|}$$



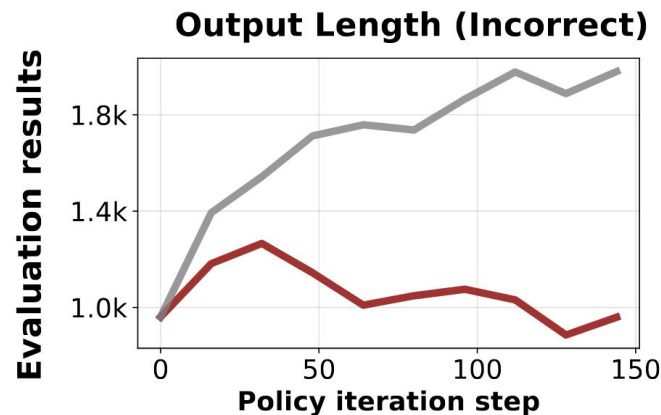
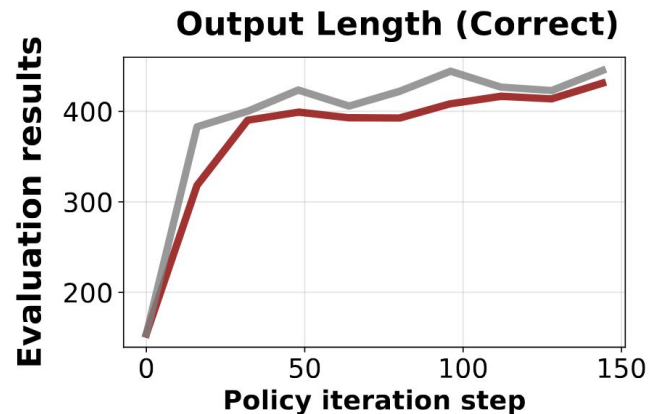
Mitigating "increasing length" phenomenon

Problem.
$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|\mathbf{o}_i|} \sum_{t=1}^{|\mathbf{o}_i|}$$

Remedy. Equalize token-level contributions

- DAPO
$$\frac{1}{\sum_{i=1}^G |\mathbf{o}_i|} \sum_{i=1}^G \sum_{t=1}^{|\mathbf{o}_i|}$$

- Dr. GRPO
$$\frac{1}{G} \sum_{i=1}^G \sum_{t=1}^{|\mathbf{o}_i|}$$



Exploration of other adjustments

- Bias linked to **level of difficulty**

$$\hat{A}_{i,t} = \frac{R(\mathbf{q}, \mathbf{o}_i) - \text{mean}(\{R(\mathbf{q}, \mathbf{o}_1), \dots, R(\mathbf{q}, \mathbf{o}_G)\})}{\text{std}(\{R(\mathbf{q}, \mathbf{o}_1), \dots, R(\mathbf{q}, \mathbf{o}_G)\})}$$

Exploration of other adjustments

- Bias linked to **level of difficulty**

$$\hat{A}_{i,t} = \frac{R(\mathbf{q}, \mathbf{o}_i) - \text{mean}(\{R(\mathbf{q}, \mathbf{o}_1), \dots, R(\mathbf{q}, \mathbf{o}_G)\})}{\text{std}(\{R(\mathbf{q}, \mathbf{o}_1), \dots, R(\mathbf{q}, \mathbf{o}_G)\})}$$

- Encourage **diversity**

$$\text{clip}(\rho_{i,t}(\theta), 1 - \epsilon, 1 + \epsilon) \longrightarrow \text{clip}(\rho_{i,t}(\theta), 1 - \epsilon_{\text{low}}, 1 + \epsilon_{\text{high}})$$

Exploration of other adjustments

- Bias linked to **level of difficulty**

$$\hat{A}_{i,t} = \frac{R(\mathbf{q}, \mathbf{o}_i) - \text{mean}(\{R(\mathbf{q}, \mathbf{o}_1), \dots, R(\mathbf{q}, \mathbf{o}_G)\})}{\text{std}(\{R(\mathbf{q}, \mathbf{o}_1), \dots, R(\mathbf{q}, \mathbf{o}_G)\})}$$

- Encourage **diversity**

$$\text{clip}(\rho_{i,t}(\theta), 1 - \epsilon, 1 + \epsilon) \longrightarrow \text{clip}(\rho_{i,t}(\theta), 1 - \epsilon_{\text{low}}, 1 + \epsilon_{\text{high}})$$

Exploration of other adjustments

- Bias linked to **level of difficulty**

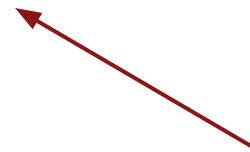
$$\hat{A}_{i,t} = \frac{R(\mathbf{q}, \mathbf{o}_i) - \text{mean}(\{R(\mathbf{q}, \mathbf{o}_1), \dots, R(\mathbf{q}, \mathbf{o}_G)\})}{\text{std}(\{R(\mathbf{q}, \mathbf{o}_1), \dots, R(\mathbf{q}, \mathbf{o}_G)\})}$$

- Encourage **diversity**

$$\text{clip}(\rho_{i,t}(\theta), 1 - \epsilon, 1 + \epsilon) \longrightarrow \text{clip}(\rho_{i,t}(\theta), 1 - \epsilon_{\text{low}}, 1 + \epsilon_{\text{high}})$$

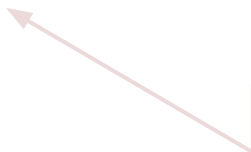
...among others!

Challenges addressed by GRPO...

- 1 **Gradients** are **noisy** 
 - 2 **No explicit token-level credit** assignment
 - 3 **No constraint on magnitude of policy updates** 
- Compute **advantages**
with **group average**
- Use **clipping** and/or
KL penalty

...with a core limitation

- 1 Gradients are noisy 

Compute **advantages** with **group average**
- 2 **No explicit token-level **credit** assignment**  
- 3 No constraint on magnitude of policy updates 

Use **clipping** and/or **KL penalty**



Transformers & Large Language Models

Reinforcement learning

Reward design

Policy gradients

PPO

GRPO

On-policy distillation

GRPO: credit system

Prompt

Teddy is in a sealed box. The box moves to the kitchen. Where is Teddy?

Output	Step 1	Step 2	Answer
O_1	Teddy stays in the box	The box moves to the kitchen	Kitchen
O_2	Teddy climbs out of the box	Teddy walks to the kitchen	Kitchen
O_3	Teddy stays in the box	The box stays in the room	Room
O_4	Teddy climbs out of the box	Teddy stays in the room	Room

GRPO: credit system

Prompt

Teddy is in a sealed box. The box moves to the kitchen. Where is Teddy?

Output	Step 1	Step 2	Answer
O_1	Teddy stays in the box	The box moves to the kitchen	Kitchen
O_2	Teddy climbs out of the box	Teddy walks to the kitchen	Kitchen
O_3	Teddy stays in the box	The box stays in the room	Room
O_4	Teddy climbs out of the box	Teddy stays in the room	Room

What we're missing out on

Prompt

Teddy is in a sealed box. The box moves to the kitchen. Where is Teddy?

Output	Step 1	Step 2	Answer
O_1	Teddy stays in the box	The box moves to the kitchen	Kitchen
O_2	Teddy climbs out of the box	Teddy walks to the kitchen	Kitchen
O_3	Teddy stays in the box	The box stays in the room	Room
O_4	Teddy climbs out of the box	Teddy stays in the room	Room

Consequences of GRPO

Immediate. Every token in an output gets the same advantage.

noisy

Long-term. GRPO learns good steps only on average, over many groups.

slow

Switch to "Process Supervision"?

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

Problems.

- Need for step-level labeling

Switch to "Process Supervision"?

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

Problems.

- Need for step-level labeling
- Each reasoning step may carry a similar ambiguity

Switch to "Process Supervision"?

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left\{ \min \left[\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip} \left(\frac{\pi_{\theta}(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL} [\pi_{\theta} || \pi_{ref}] \right\}$$

Problems.

- Need for step-level labeling
- Each reasoning step may carry a similar ambiguity
- Need for dedicated reward model training

Wish list recap

We want:

- On-policy training
- Explicit credit assignment on each token

We don't want:

- Any additional training

Proposal

- Train on the model's generations
- Loss on next token prediction
- Use an already available teacher

KL divergence

$$\text{KL}(P||Q) = \sum_i p_i \log \left(\frac{p_i}{q_i} \right)$$

In a world governed by P,
how bad is it to approximate it with Q?

Reverse KL

Formula.

$$\text{KL}(S||T) = \sum_i s_i \log \left(\frac{s_i}{t_i} \right)$$

Reverse KL

Formula.

$$\text{KL}(S||T) = \sum_i \overset{\text{low}}{\boxed{s_i}} \log \left(\frac{\boxed{s_i}}{\underset{\text{high}}{\boxed{t_i}}} \right)$$



Properties.

- Forgiving about student misses on non-zero teacher regions

Reverse KL

Formula.

$$\text{KL}(S||T) = \sum_i s_i \log \left(\frac{s_i}{t_i} \right)$$

high
low



Properties.

- Forgiving about student misses on non-zero teacher regions
- Heavily penalizes the student for putting probability where the teacher puts little

"mode seeking"

On-policy distillation

Prompt

Teddy is in a sealed box. The box moves to the kitchen. Where is Teddy?

Output	Step 1	Step 2	Answer
O_1	Teddy stays in the box	The box moves to the kitchen	Kitchen
O_2	Teddy climbs out of the box	Teddy walks to the kitchen	Kitchen
O_3	Teddy stays in the box	The box stays in the room	Room
O_4	Teddy climbs out of the box	Teddy stays in the room	Room

Remark

$$p_S(x_{t+1} | \boxed{x_1, x_2, \dots, x_t})$$

context

Behavior.

- Token prediction typically penalized at the moment it starts going wrong
- Rest of the sentence usually fine since next-token prediction assumes context

Discussion

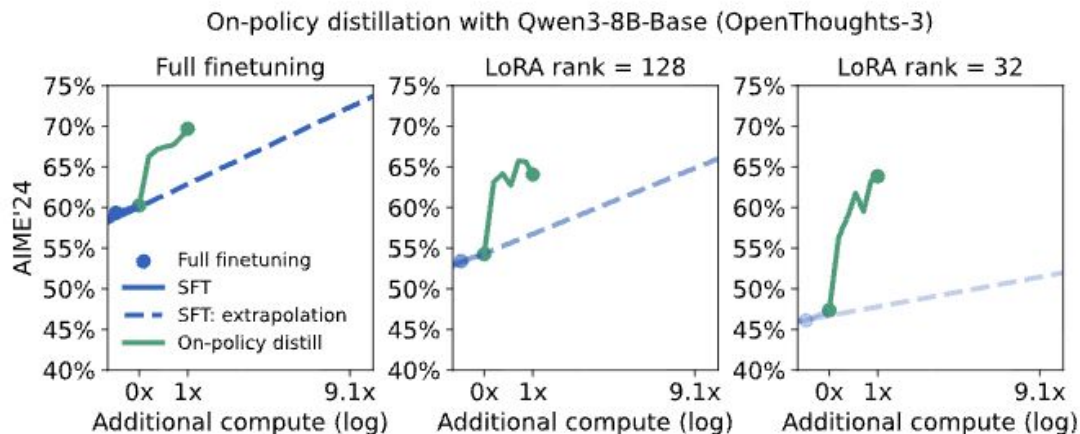
Pros.

- Works well in practice

Discussion

Pros.

- Works well in practice
- Compute-efficient



Discussion

Pros.

- Works well in practice
- Compute-efficient
- No reward model

Discussion

Pros.

- Works well in practice
- Compute-efficient
- No reward model

Limitations.

- You need a teacher to learn from in the first place

Discussion

Pros.

- Works well in practice
- Compute-efficient
- No reward model

Limitations.

- You need a teacher to learn from in the first place
- Target performance is bounded by teacher model's

Discussion

Pros.

- Works well in practice
- Compute-efficient
- No reward model

Limitations.

- You need a teacher to learn from in the first place
- Target performance is bounded by teacher model's

Extension: self-distillation

Idea. Keep same student model as a teacher, but with a hint.

Prompt

Teddy is in a sealed box. The box moves to the kitchen. Where is Teddy?

Hint

Teddy remains in the box throughout the process, which means he ends up in the kitchen.

Now solve it yourself:

Discussion

Pros.

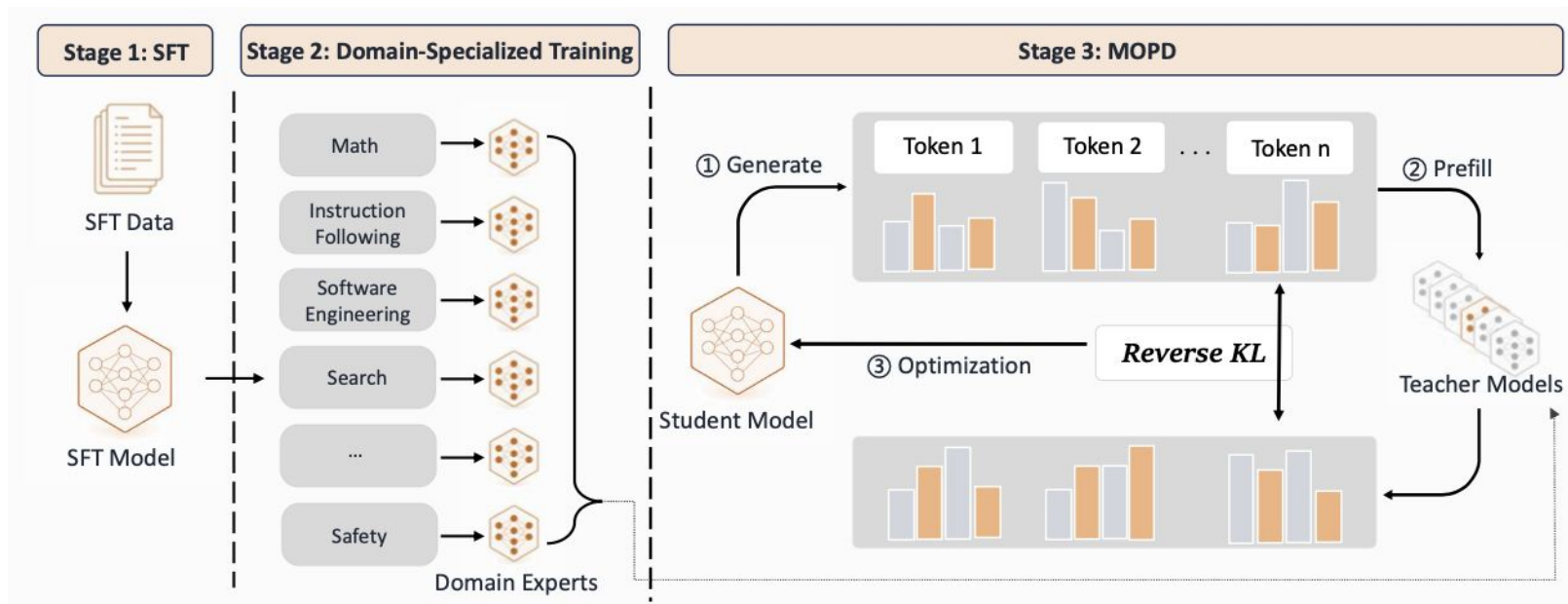
- Works well in practice
- Compute-efficient
- No reward model

Limitations.

- You need a teacher to learn from in the first place
- Target performance is bounded by teacher model's

Extension: multiple teachers

Proposal. Use a teacher that is expert in the associated domain.



Thank you for your attention!
