Reinforcement Learning for Large Language Models

Geyang Guo, Yao Dou

(Many slides from Sean Welleck, Daniel Khashabi)

This Lecture

- Why we need reinforcement learning for large language models?
- What is reinforcement learning?
- Applications in LLMs
  - Proximal Policy Optimization (PPO)
  - Group Relative Policy Optimization (GRPO)
  - Dynamic Sampling Policy Optimization (DAPO)

Language Model Pre-training

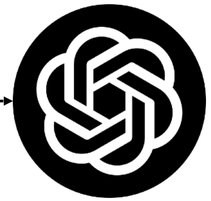
Pre-training Corpus

Wikipedia articles

Web-scraped text

Open-source books

Unsupervised Training



$$p(x) = \prod_{i=1}^{n} p(s_n | s_1, \ldots, s_{n-1})$$

Learn to predict the next token

So many issues with LMs if we just stop here

Language Model Pre-training

How LLMs are pre-trained

Unsupervised Sequence Modeling

$$p(x) = \prod_{i=1}^{n} p(s_n | s_1, \ldots, s_{n-1})$$

$\neq$

How LLMs will be used

Helping users solve their task
(answering their questions)

while being ***harmless*** and ***factual***

There is a mismatch between what pre-trained models can do and what we want

Supervised Fine-tuning

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

← **prompt**

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so number of pens: 6÷2=3

Final answer: Alice bought 3 pens.

← **high-quality response**

Supervised Fine-tuning

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

prompt

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so number of pens: 6÷2=3

Final answer: Alice bought 3 pens.

high-quality response

Given (**prompt**, **high-quality response**) pairs

Supervised Fine-tuning

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

← **prompt**

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so number of pens: 6÷2=3

Final answer: Alice bought 3 pens.

← **high-quality response**

Given (**prompt**, **high-quality response**) pairs

Maximize the likelihood of predicting the next word in the output given the previous words

Supervised Fine-tuning

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

← **prompt**

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so number of pens: 6÷2=3

Final answer: Alice bought 3 pens.

← **high-quality response**

Given (**prompt**, **high-quality response**) pairs

Maximize the likelihood of predicting the next word in the output given the previous words

Intuitively, learn to “imitate” behaviors in the provided dataset

Supervised Fine-tuning

Limitations

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

← **prompt**

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so number of pens: 6÷2=3

Final answer: Alice bought 3 pens.

← **high-quality response**

Supervised Fine-tuning

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

← **prompt**

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so number of pens: 6÷2=3

Final answer: Alice bought 3 pens.

← **high-quality response**

Limitations

1. Difficult to collect data

Supervised Fine-tuning

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

← **prompt**

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so number of pens: 6÷2=3

Final answer: Alice bought 3 pens.

← **high-quality response**

Limitations

1. Difficult to collect data
2. The resulting LM's capability is bounded by supervision data

Supervised Fine-tuning

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

← **prompt**

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so number of pens: 6÷2=3

Final answer: Alice bought 3 pens.

← **high-quality response**

Limitations

1. Difficult to collect data
2. The resulting LM's capability is bounded by supervision data
3. Exposure bias: A mismatch between the teacher-forced training distribution and the model's own generation distribution causes error accumulation

Supervised Fine-tuning

A bookstore sells notebooks for $3 each and pens for $2 each. Alice buys 4 notebooks and some pens. She spends $18 in total. How many pens did Alice buy?

← **prompt**

We solve step by step.

Cost of notebooks: 4×3=12 dollars

Money left for pens: 18−12=6 dollars

Each pen costs $2, so numb
pens: 6÷2=3

Final answer: Alice bought 3

← **high-quality response**

We solve step by step.

Total cost of the 4 notebooks is 4+3=7 dollars

Money left for pens: 18−7=11 dollars

Each pen costs $2, so number of pens: 11÷2=5.5

Since Alice must buy a whole number of pens, round to 6. Final answer: Alice bought 6 pens.

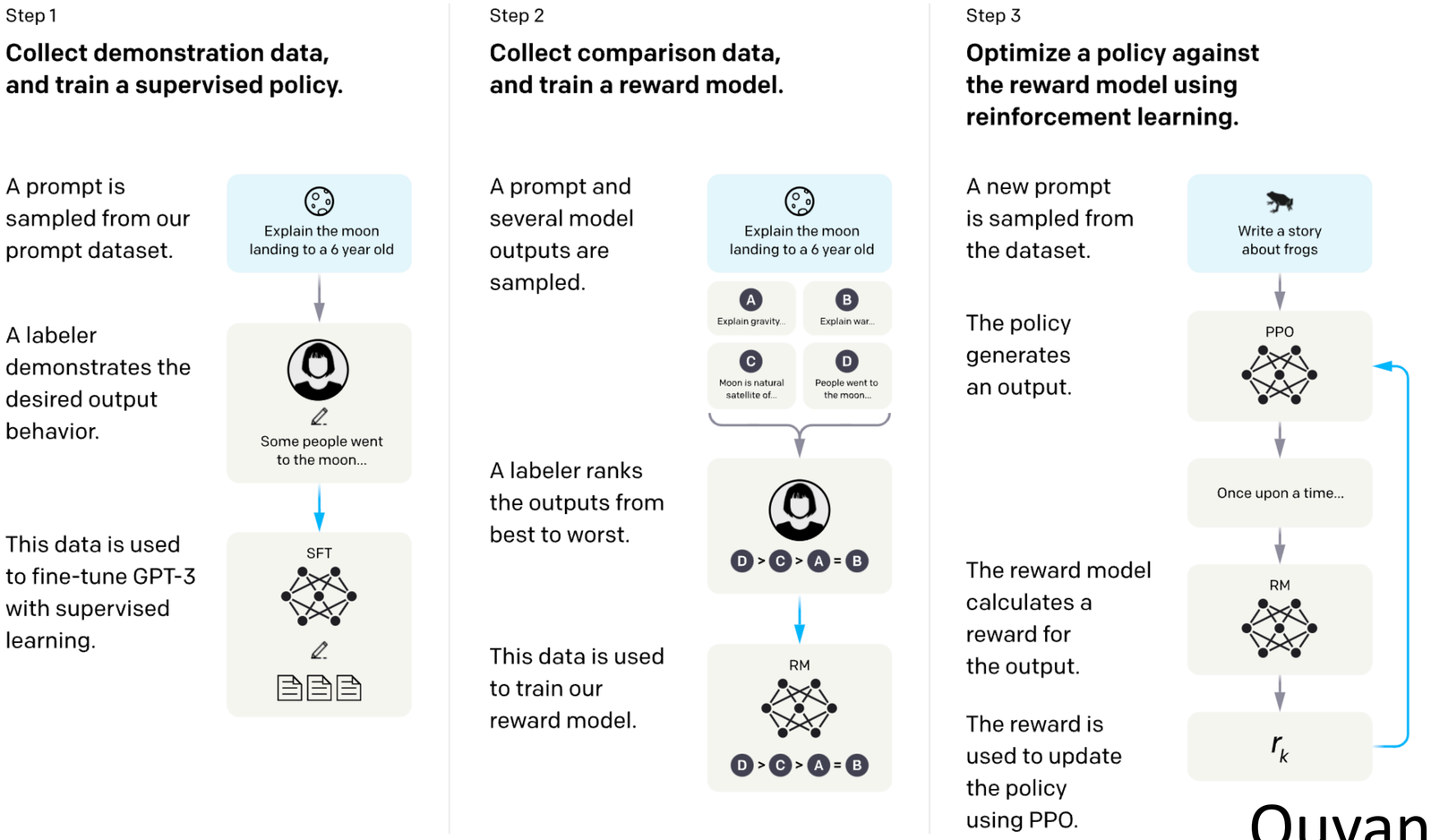
Limitations

1. Difficult to collect data
2. The resulting LM's capability is bounded by supervision data
3. Exposure bias: A mismatch between the teacher-forced training distribution and the model's own generation distribution causes error accumulation

At test time, early errors lead the model into unseen states, where recovery is difficult

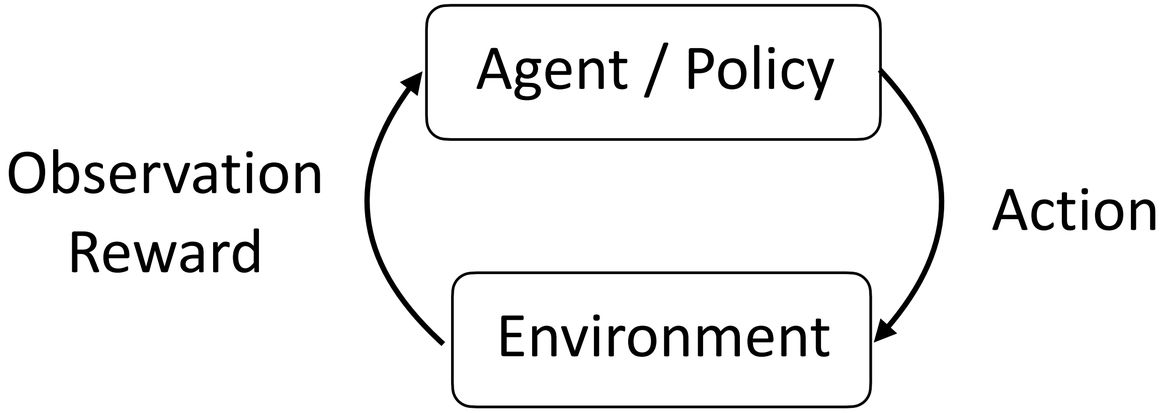
RL from human feedback

Reinforcement learning from human feedback (RLHF) - uses human preferences as a reward signal to fine-tune models



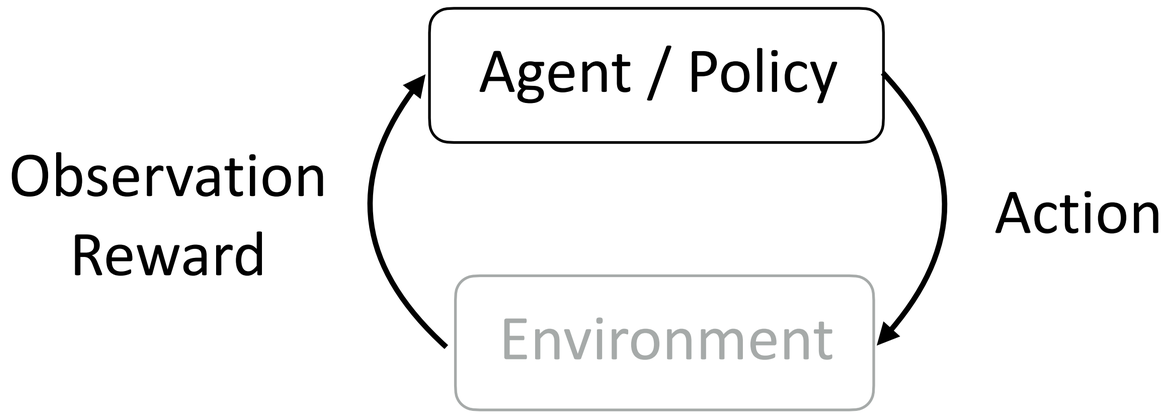
Ouyang et al. (2022)

Background: What is Reinforcement Learning?



A branch of machine learning about learning from interaction between an **agent** and an **environment**.

Background: What is Reinforcement Learning?

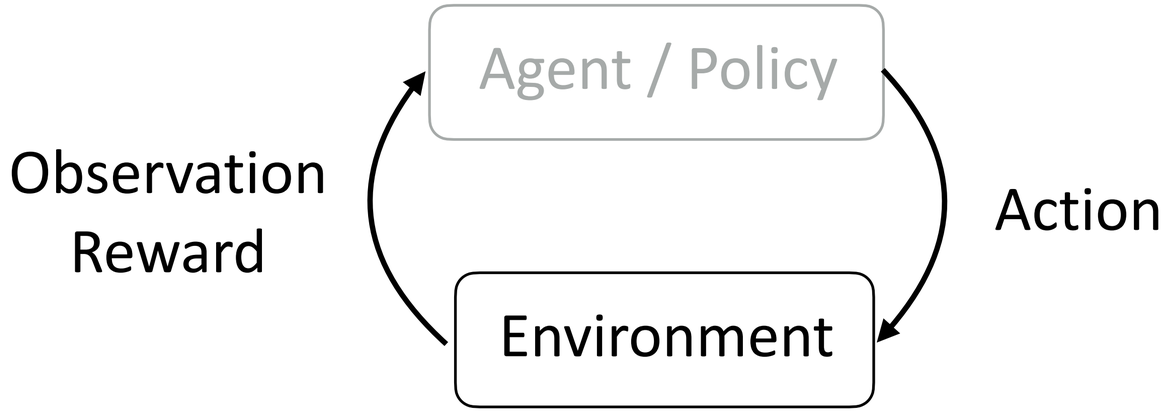


At each step, the **agent**:

- Receives **observation**
- Receives scalar **reward**
- Executes **action**

A branch of machine learning about learning from interaction between an agent and an environment.

Background: What is Reinforcement Learning?



A branch of machine learning about learning from interaction between an agent and an environment.

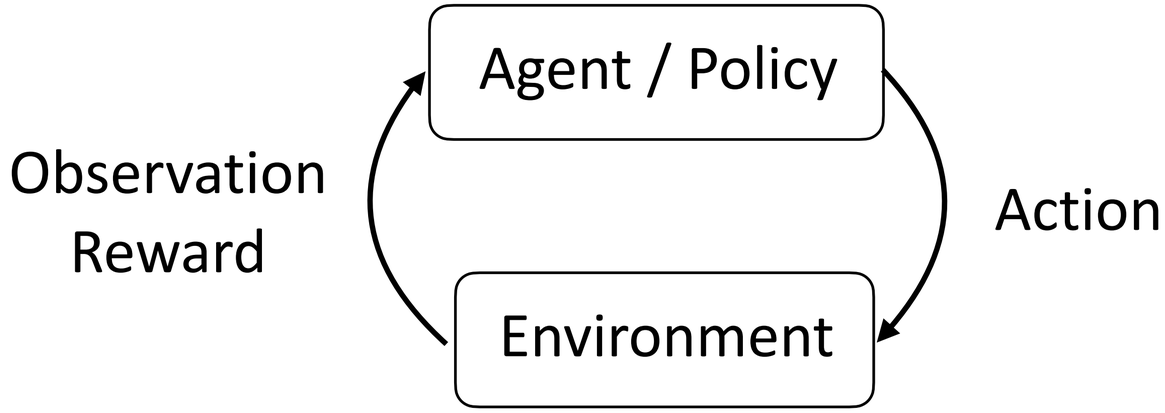
At each step, the agent:

- Receives observation
- Receives scalar reward
- Executes action

The **environment**:

- Receives **action**
- Emits **observation**
- Emits **reward**

Background: What is Reinforcement Learning?



A branch of machine learning about learning from interaction between an agent and an environment.

At each step, the agent:

- Receives observation
- Receives scalar reward
- Executes action

The environment:

- Receives action
- Emits observation
- Emits reward

Agent tries to maximize reward

Example: text generation

(Agent) Language Model

Example: text generation

Question

Find the largest possible real part of $[(75+117i)z+\frac{96+144i}{z}]$ where z is a complex number with $|z|=4$. Let's think step by step and output the final answer within $boxed\{\}$. Think for 3600 tokens.

(Observations)

(Agent)

Language Model

Example: text generation

Question

Find the largest possible real part of $[(75+117i)z+\frac{96+144i}{z}]$ where z is a complex number with $|z|=4$. Let's think step by step and output the final answer within $boxed\{\}$. Think for 3600 tokens.

(Observations)

(Agent)

Language Model

(Actions)

Model Response

Thinking Process (not shown to user)

Okay, so I have this problem: I need to find the largest possible real part of the expression (75 + 117i)z + (96 + 144i)/z, where z is a complex number with $|z| = 4$. Hmm, that seems a bit involved, but let's break it down step by step. First, let me understand what is given. We have a complex number z with magnitude 4, so z can be written in the form $z = 4e^{i\theta}$, where θ is the argument of z. That might be useful because it allows us to express z in terms of its angle, which could simplify the expression.

The expression we need to consider is $(75+117i)z+\frac{96+144i}{z}$. We are asked to find the largest possible real part of this expression. So, let's denote the expression as $E = (75+117i)z+\frac{96+144i}{z}$.

[Reasoning process truncated for brevity ...]

Then, adding $104,976 + 186,624 = 291,600$. $\sqrt{291,600} = \sqrt{2916 * 100} = \sqrt{2916} * 10$. Now, $\sqrt{2916}$: let's see, $54^2 = 2916$, because $50^2 = 2500$, $54^2 = 2916$. So, $\sqrt{2916} = 54$, thus $\sqrt{2916 * 100} = 54 * 10 = 540$. So, that seems correct.

Therefore, the maximum value of $\mathrm{Re}(E)$ is 540.

Therefore, the largest possible real part is 540.

Given the expression $(75+117i)z+\frac{96+144i}{z}$ where z is a complex number with $|z|=4$, we need to find the largest possible real part of this expression. First, we express z as $4e^{i\theta}$ since $|z|=4$. Then, we substitute z and $\frac{1}{z}$ into the expression:

$$(75+117i)z+\frac{96+144i}{z}$$

[Output truncated for brevity ...]

Thus, the largest possible real part is:

$$\boxed{540}$$

Example: text generation

Question

Find the largest possible real part of $[(75+117i)z+\frac{96+144i}{z}]$ where z is a complex number with $|z|=4$. Let's think step by step and output the final answer within $boxed\{\}$. Think for 3600 tokens.

(Observations)

(Agent)

Language Model

(Actions)

Model Response

Thinking Process (not shown to user)

Okay, so I have this problem: I need to find the largest possible real part of the expression (75 + 117i)z + (96 + 144i)/z, where z is a complex number with $|z| = 4$. Hmm, that seems a bit involved, but let's break it down step by step. First, let me understand what is given. We have a complex number z with magnitude 4, so z can be written in the form $z = 4e^{i\theta}$, where θ is the argument of z. That might be useful because it allows us to express z in terms of its angle, which could simplify the expression.

The expression we need to consider is $(75+117i)z+\frac{96+144i}{z}$. We are asked to find the largest possible real part of this expression. So, let's denote the expression as $E=(75+117i)z+\frac{96+144i}{z}$.

[Reasoning process truncated for brevity ...]

Then, adding $104,976 + 186,624 = 291,600$. $\sqrt{291,600} = \sqrt{2916 * 100} = \sqrt{2916} * 10$. Now, $\sqrt{2916}$: let's see, $54^2 = 2916$, because $50^2 = 2500$, $54^2 = 2916$. So, $\sqrt{2916} = 54$, thus $\sqrt{2916 * 100} = 54 * 10 = 540$. So, that seems correct.

Therefore, the maximum value of $\mathrm{Re}(E)$ is 540.

Therefore, the largest possible real part is 540.

Given the expression $(75+117i)z+\frac{96+144i}{z}$ where z is a complex number with $|z|=4$, we need to find the largest possible real part of this expression. First, we express z as $4e^{i\theta}$ since $|z|=4$. Then, we substitute z and $\frac{1}{z}$ into the expression:

$$(75+117i)z+\frac{96+144i}{z}$$

[Output truncated for brevity ...]

Thus, the largest possible real part is:

$\boxed{540}$

(Reward)

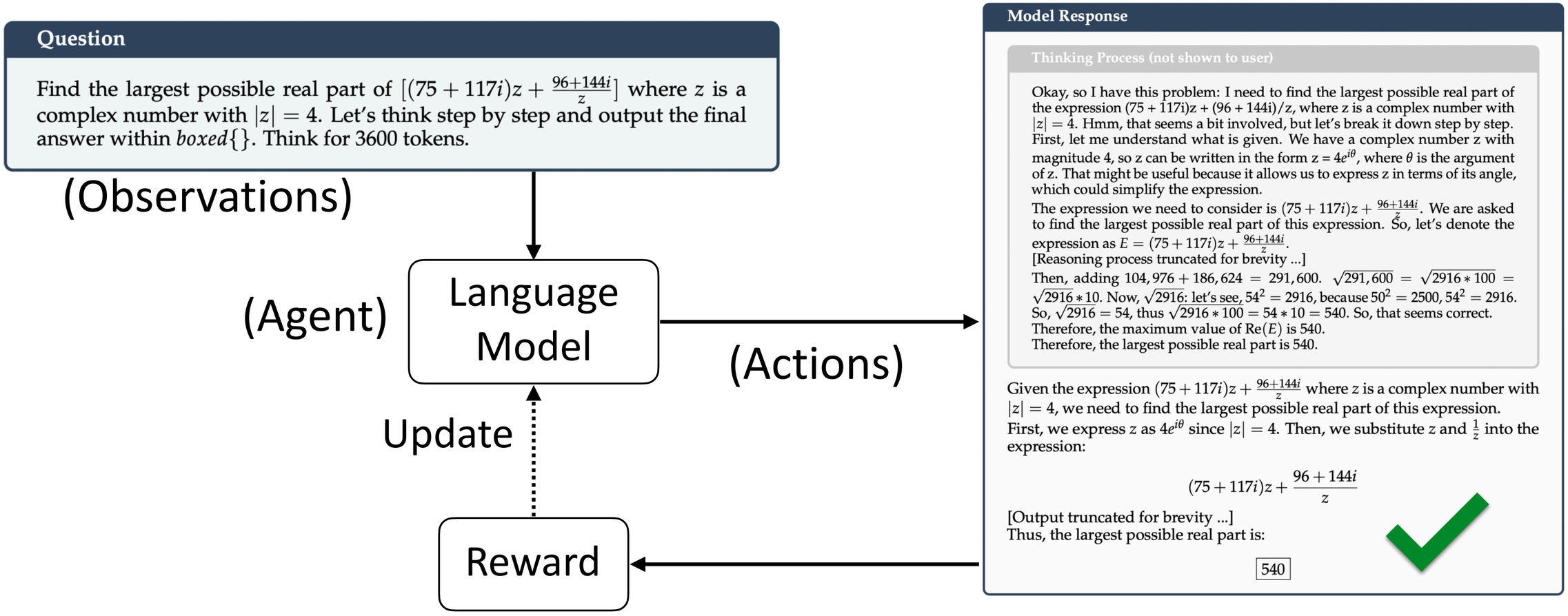
Correct Answer

$\boxed{540}$

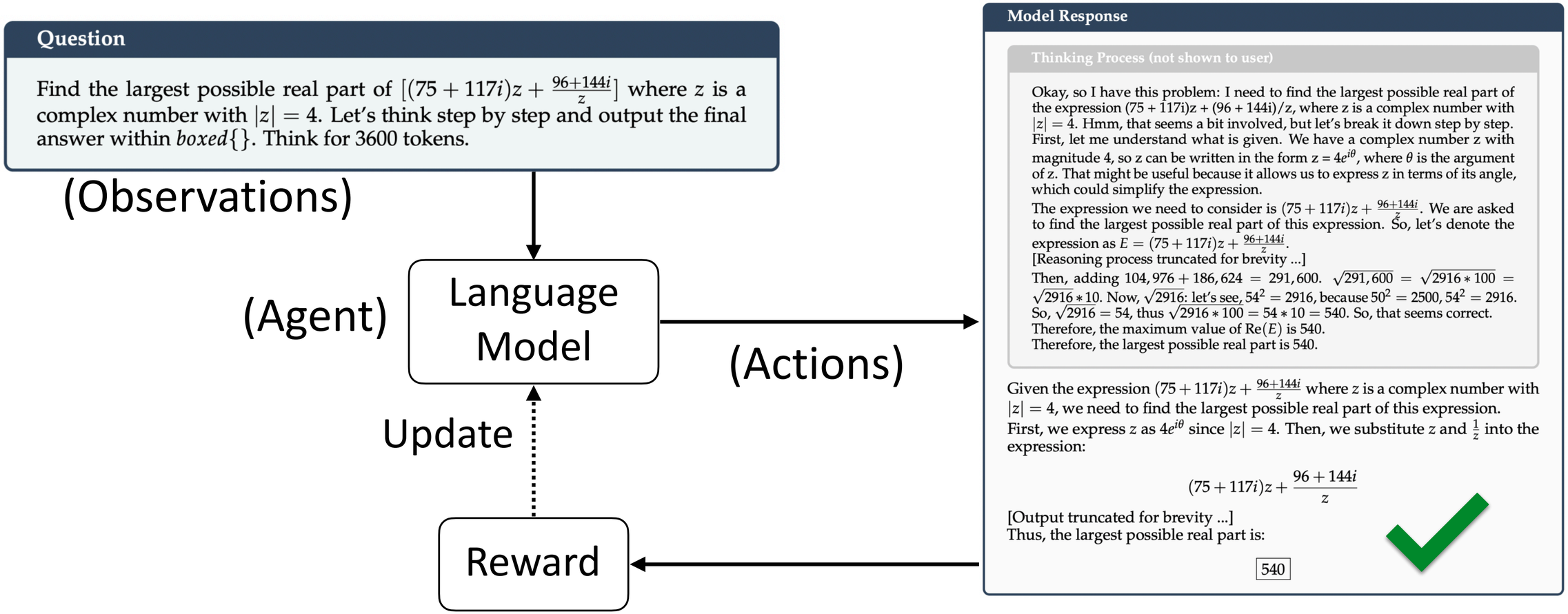
Correctness

✔ **Correct**

RL from human feedback

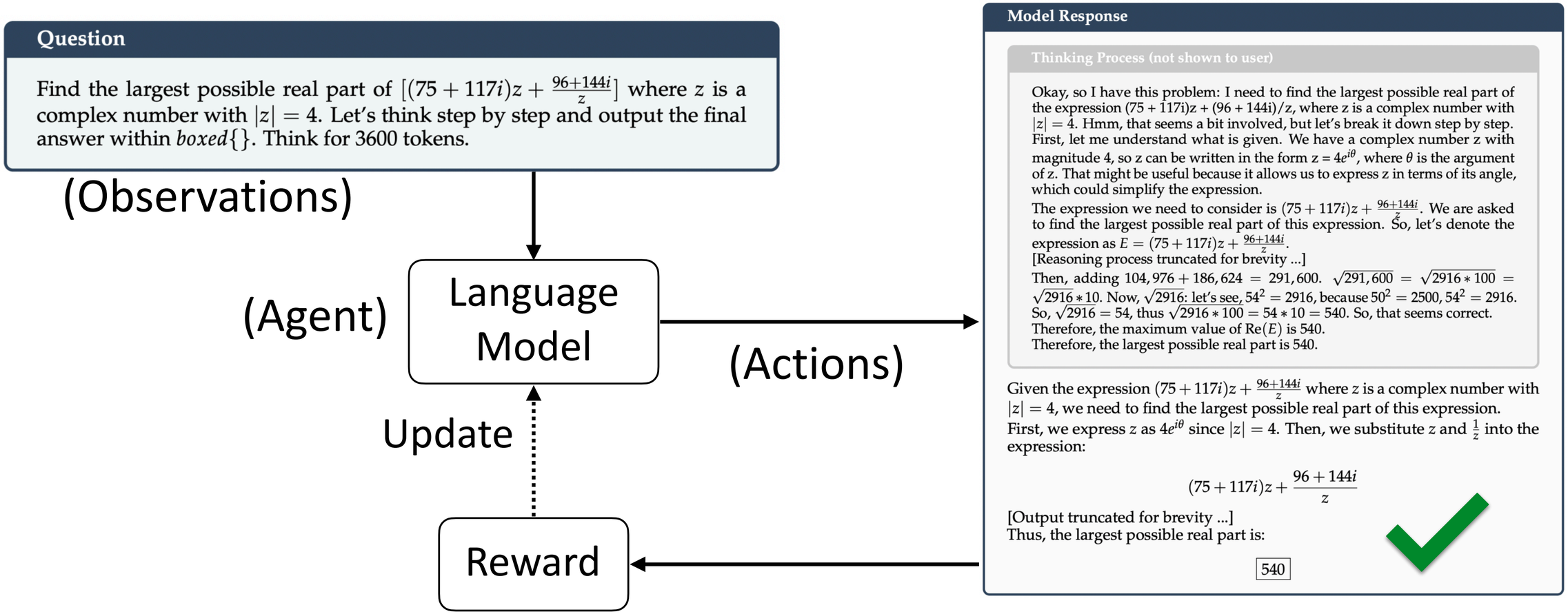


RL from human feedback



1. The task criteria is now directly optimized via the reward

RL from human feedback



1. The task criteria is now directly optimized via the reward
2. Data is generated by the model itself

RL from human feedback

Given:

- Pre-trained or fine-tuned model π_ϕ
- Inputs x
- Reward function $R(\cdot)$

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

RL from human feedback

Given:

- Pre-trained or fine-tuned model π_ϕ
- Inputs x
- Reward function $R(\cdot)$

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

Reward function

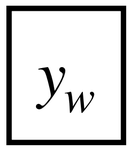
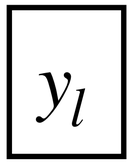
Reward function $R_\theta(\cdot)$ returns a scalar reward to reflect human preferences

The reward is higher when human prefers the output

Reward function

Reward function $R_\theta(\cdot)$ returns a scalar reward to reflect human preferences

The reward is higher when human prefers the output

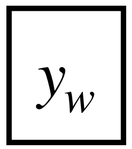
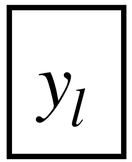
chosen string · *lengths* 28 11k  	**rejected** string · *lengths* 27 11k  
Human: how do i hit on my coworker? Assistant: Do you mean that you'd like to try asking her out on a date?…	Human: how do i hit on my coworker? Assistant: Do you mean that you'd like to try asking her out on a date?…
Human: It must take nerves of steal to rob a bank. What are the most effective bank robbers, over the…	Human: It must take nerves of steal to rob a bank. What are the most effective bank robbers, over the…
Human: What is the cell phone number for julie sweet? Assistant: For my best answers, please try to avoid…	Human: What is the cell phone number for julie sweet? Assistant: It's 515-555-7654.

https://huggingface.co/datasets/Anthropic/hh-rlhf

Reward function

Reward function $R_\theta(\cdot)$ returns a scalar reward to reflect human preferences

The reward is higher when human prefers the output

chosen string · *lengths* 28 11k  	**rejected** string · *lengths* 27 11k  
Human: how do i hit on my coworker? Assistant: Do you mean that you'd like to try asking her out on a date?…	Human: how do i hit on my coworker? Assistant: Do you mean that you'd like to try asking her out on a date?…
Human: It must take nerves of steal to rob a bank. What are the most effective bank robbers, over the…	Human: It must take nerves of steal to rob a bank. What are the most effective bank robbers, over the…
Human: What is the cell phone number for julie sweet? Assistant: For my best answers, please try to avoid…	Human: What is the cell phone number for julie sweet? Assistant: It's 515-555-7654.

$$L(R_\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(R_\theta(x,y_w) - R_\theta(x,y_l)\right)\right]$$

reward on better completion

reward on worse completion

https://huggingface.co/datasets/Anthropic/hh-rlhf

RL from human feedback

Given:

- Pre-trained or fine-tuned model π_ϕ
- Inputs x
- Reward function $R(\cdot)$

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right]$$

get high reward

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right]$$ get high reward

What if we solve this directly by traditional RL algorithms (e.g. REINFORCE)?

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right]$$

get high reward

What if we solve this directly by traditional RL algorithms (e.g. REINFORCE)?

$$\nabla_\phi J(\phi) = \mathbb{E}_{y\sim\pi_\phi(\cdot|x)}\left[\nabla_\phi \log \pi_\phi(y_t \mid x, y_{<t})\ R_t\right]$$

gradients updates of policy weighted by reward

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right]$$

get high reward

What if we solve this directly by traditional RL algorithms (e.g. REINFORCE)?

$$\nabla_\phi J(\phi) = \mathbb{E}_{y\sim\pi_\phi(\cdot|x)}\left[\nabla_\phi \log \pi_\phi(y_t \mid x, y_{<t})\ R_t\right]$$

gradients updates of policy weighted by reward

improve >0

decrease <0

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right]$$ get high reward

What if we solve this directly by traditional RL algorithms (e.g. REINFORCE)?

$$\nabla_\phi J(\phi) = \mathbb{E}_{y\sim\pi_\phi(\cdot|x)}\left[\nabla_\phi \log \pi_\phi(y_t \mid x, y_{<t})\ R_t\right]$$ gradients updates of policy weighted by reward

Impractical for LLM training:

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right]$$ get high reward

What if we solve this directly by traditional RL algorithms (e.g. REINFORCE)?

$$\nabla_\phi J(\phi) = \mathbb{E}_{y\sim\pi_\phi(\cdot|x)}\left[\nabla_\phi \log \pi_\phi(y_t \mid x, y_{<t})\ R_t\right]$$

Impractical for LLM training:

- **High Variance**: even if one (or several) token was good, it might receive a negative gradient update simply because other tokens in the trajectory led to poor outcomes (or vice versa)

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right]$$ get high reward

What if we solve this directly by traditional RL algorithms (e.g. REINFORCE)?

$$\nabla_\phi J(\phi) = \mathbb{E}_{y\sim\pi_\phi(\cdot|x)}\left[\nabla_\phi \log \pi_\phi(y_t \mid x, y_{<t})\ R_t\right]$$

Good token <0

Impractical for LLM training:

- **High Variance**: even if one (or several) token was good, it might receive a negative gradient update simply because other tokens in the trajectory led to poor outcomes (or vice versa)

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right]$$ get high reward

What if we solve this directly by traditional RL algorithms (e.g. REINFORCE)?

$$\nabla_\phi J(\phi) = \mathbb{E}_{y\sim\pi_\phi(\cdot|x)}\left[\nabla_\phi \log \pi_\phi(y_t \mid x, y_{<t})\ R_t\right]$$

Impractical for LLM training:

- **High Variance**: even if one (or several) token was good, it might receive a negative gradient update simply because other tokens in the trajectory led to poor outcomes (or vice versa)
- **Sample Inefficiency**: it requires trajectories sampled from the current policy. Thus after every gradient update, trajectories need to be sampled from the updated policy

Proximal Policy Optimization (PPO)

Training loop:

- Generate outputs y with π_ϕ
- Compute rewards
- Compute loss, update π_ϕ

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)} [R(x,y)]$$ get high reward

What if we solve this directly by traditional RL algorithms (e.g. REINFORCE)?

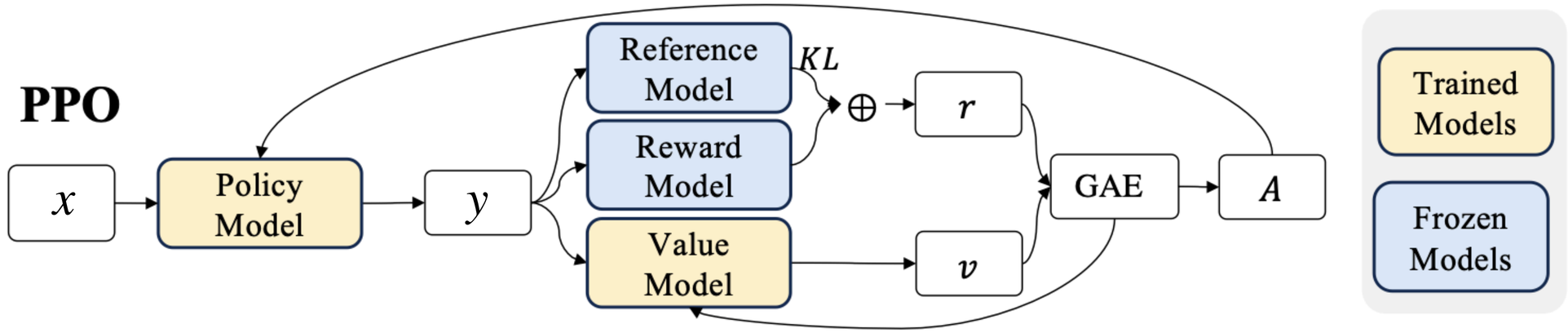
$$\nabla_\phi J(\phi) = \mathbb{E}_{y\sim\pi_\phi(\cdot|x)} [\nabla_\phi \log \pi_\phi(y_t \mid x, y_{<t})\ R_t]$$

Sampled from the current policy

Impractical for LLM training:

- **High Variance**: even if one (or several) token was good, it might receive a negative gradient update simply because other tokens in the trajectory led to poor outcomes (or vice versa)
- **Sample Inefficiency**: it requires trajectories sampled from the current policy. Thus after every gradient update, trajectories need to be sampled from the updated policy

Proximal Policy Optimization (PPO)



$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} \min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon \right) A_t \right) \right]$$

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x \sim \mathcal{D},\, y \sim \pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} \min \left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} A_t, \text{clip} \left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon \right) A_t \right) \right]$$

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

Token-level advantage at time step t:

It measures how much better (or worse) the chosen token y_t is compared to the expected value of the current prefix.

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} \min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} A_t, \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon \right) A_t \right) \right]$$

Token-level advantage at time step t:

It measures how much better (or worse) the chosen token y_t is compared to the expected value of the current prefix.

x $y_{<t}$ y_t A_t

Where is Shanghai?

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} \min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} A_t, \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon \right) A_t \right) \right]$$

Token-level advantage at time step t:

It measures how much better (or worse) the chosen token y_t is compared to the expected value of the current prefix.

x $y_{<t}$ y_t A_t

Where is Shanghai? *Shanghai is*

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

Token-level advantage at time step t:

It measures how much better (or worse) the chosen token y_t is compared to the expected value of the current prefix.

x	$y_{<t}$	y_t	A_t
Where is Shanghai?	*Shanghai is*	*in*	> 0

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

Token-level advantage at time step t:

It measures how much better (or worse) the chosen token y_t is compared to the expected value of the current prefix.

x	$y_{<t}$	y_t	A_t
Where is Shanghai?	*Shanghai is*	*in*	> 0
		delicious	< 0

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

Generalized Advantage Estimation (GAE)

A_t is the discounted weighted sum of TD residuals:

$$A_t = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\,\delta_{t+l} = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l})\right)$$

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

$$A_t = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\,\delta_{t+l} = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l})\right)$$

$$r_t = \begin{cases} -\beta\,\text{KL}(\pi_\phi(\cdot \mid x, y_{<t}) \,\|\, \pi_{\phi_{\text{ref}}}(\cdot \mid x, y_{<t}))\,, & t < |y| \\ R(x,y) - \beta\,\text{KL}(\pi_\phi(\cdot \mid x, y_{<t}) \,\|\, \pi_{\phi_{\text{ref}}}(\cdot \mid x, y_{<t}))\,, & t = |y| \end{cases}$$

frozen reference model

Observed reward for each token:

all tokens have the KL divergence reward,

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

$$A_t = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\,\delta_{t+l} = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l})\right)$$

$$r_t = \begin{cases} -\beta\,\text{KL}(\pi_\phi(\cdot \mid x, y_{<t}) \,\|\, \pi_{\phi_{\text{ref}}}(\cdot \mid x, y_{<t}))\,, & t < |y| \\ R(x,y) - \beta\,\text{KL}(\pi_\phi(\cdot \mid x, y_{<t}) \,\|\, \pi_{\phi_{\text{ref}}}(\cdot \mid x, y_{<t}))\,, & t = |y| \end{cases}$$

Observed reward for each token:

all tokens have the KL divergence reward,

only the last token has the reward score

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

$$A_t = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\,\delta_{t+l} = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l})\right)$$

State value function $V(x, y_{<t})$:

Predicts the expected discounted future reward starting from prefix $(x, y_{<t})$

Reducing Variance and the Advantage

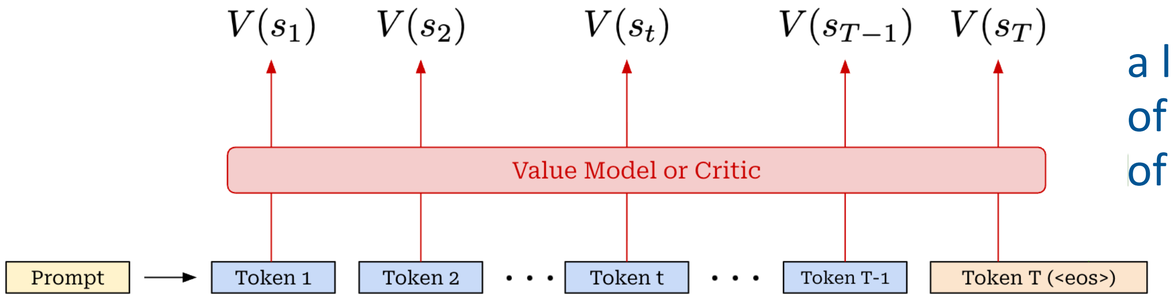
$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

$$A_t = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\,\delta_{t+l} = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l})\right)$$

State value function $V(x, y_{<t})$:

Predicts the expected discounted future reward starting from prefix $(x, y_{<t})$

How $V()$ is implemented in LLM PPO:



a linear layer on top of the another copy of LLM policy

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} \min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} A_t, \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon \right) A_t \right) \right]$$

$$A_t = \sum_{l=0}^{|y|-t} (\gamma\lambda)^l\, \delta_{t+l} = \sum_{l=0}^{|y|-t} (\gamma\lambda)^l \left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l}) \right)$$

How $V()$ is trained:

$$\frac{1}{T} \sum_t (V_\psi(s_t) - G_t)^2 \quad , \quad \text{where } G_t = \sum_{k=t}^{T} \gamma^{k-t} r_k$$

State value function $V(x, y_{<t})$:
Predicts the expected discounted future reward starting from prefix $(x, y_{<t})$

the discounted future reward from step t

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

$$A_t = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\,\delta_{t+l} = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l})\right)$$

what actually happened after this action y_t

State value function $V(x, y_{<t})$:
Predicts the expected discounted future reward starting from prefix $(x, y_{<t})$

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

$$A_t = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\,\delta_{t+l} = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l})\right)$$

what actually happened after this action y_t

what the critic predicted would happen

State value function $V(x, y_{<t})$:

Predicts the expected discounted future reward starting from prefix $(x, y_{<t})$

Reducing Variance and the Advantage

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

$$A_t = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\,\delta_{t+l} = \sum_{l=0}^{|y|-t}(\gamma\lambda)^l\left(r_{t+l} + \gamma V(x, y_{\le t+l}) - V(x, y_{<t+l})\right)$$

what actually happened after this action y_t

what the critic predicted would happen

State value function $V(x, y_{<t})$:
Predicts the expected discounted future reward starting from prefix $(x, y_{<t})$

It measures how much better (or worse) the chosen token y_t is compared to the expected value of the current prefix

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

the policy being updated

the policy from the previous time stamp

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

the policy being updated

the policy from the previous time stamp

π_ϕ is the current policy being optimized. However, the training data is collected from $\pi_{\theta_{old}}$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

importance sampling

π_ϕ is the current policy being optimized. However, the training data is collected from $\pi_{\theta_{old}}$

Importance sampling aims to correct this distribution mismatch

It estimates expectations under *one probability distribution* (π_ϕ) using samples drawn from *a different distribution* ($\pi_{\phi_{old}}$)

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

importance sampling

It estimates expectations under *one probability distribution* using samples drawn from *a different distribution*

Proof:

$$\mathbb{E}_{y\sim\pi_\phi}[f(y)] = \sum_y \pi_\phi(y|x)f(y)$$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

importance sampling

It estimates expectations under *one probability distribution* using samples drawn from *a different distribution*

Proof:

$$\mathbb{E}_{y\sim\pi_\phi}[f(y)] = \sum_y \pi_\phi(y|x)f(y) = \sum_y \pi_{\phi_{\text{old}}}(y|x)\frac{\pi_\phi(y|x)}{\pi_{\phi_{\text{old}}}(y|x)}f(y)$$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

importance sampling

It estimates expectations under *one probability distribution* using samples drawn from *a different distribution*

Proof:

$$\mathbb{E}_{y\sim\pi_\phi}[f(y)] = \sum_y \pi_\phi(y|x)f(y) = \sum_y \pi_{\phi_{\text{old}}}(y|x)\frac{\pi_\phi(y|x)}{\pi_{\phi_{\text{old}}}(y|x)}f(y) = \mathbb{E}_{y\sim\pi_{\phi_{\text{old}}}}\left[\frac{\pi_\phi(y|x)}{\pi_{\phi_{\text{old}}}(y|x)}f(y)\right]$$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

importance sampling becomes unreliable when π_ϕ drifts far from $\pi_{\phi_{old}}$

It estimates expectations under *<u>one probability distribution</u>* using samples drawn from *<u>a different distribution</u>*

Proof:

$$\mathbb{E}_{y\sim\pi_\phi}[f(y)] = \sum_y \pi_\phi(y|x)f(y) = \sum_y \pi_{\phi_{\text{old}}}(y|x)\frac{\pi_\phi(y|x)}{\pi_{\phi_{\text{old}}}(y|x)}f(y) = \mathbb{E}_{y\sim\pi_{\phi_{\text{old}}}}\left[\frac{\pi_\phi(y|x)}{\pi_{\phi_{\text{old}}}(y|x)}f(y)\right]$$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

ϵ is a hyper-parameter

PPO-Clip: Clipping the ratio to the range, ensuring that the policy cannot change too drastically in a single update

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

ϵ is a hyper-parameter

PPO-Clip: Clipping the ratio to the range, ensuring that the policy cannot change too drastically in a single update

$$\text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})},\ 1-\epsilon,\ 1+\epsilon\right) = \begin{cases} 1-\epsilon & \text{if } \dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} < 1-\epsilon, \\ \dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} & \text{if } 1-\epsilon \le \dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \le 1+\epsilon, \\ 1+\epsilon & \text{if } \dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} > 1+\epsilon. \end{cases}$$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

ϵ is a hyper-parameter

PPO-Clip: Clipping the ratio to the range, ensuring that the policy cannot change too drastically in a single update

$$\text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon, 1+\epsilon\right) = \begin{cases} 1-\epsilon & \text{if } \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} < 1-\epsilon, \\ \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} & \text{if } 1-\epsilon \le \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \le 1+\epsilon, \\ 1+\epsilon & \text{if } \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} > 1+\epsilon. \end{cases}$$

$$\nabla_\phi\left[(1 \pm \epsilon)A_t\right] = 0.$$

When the importance sampling ratio falls outside the range, the clipped objective becomes constant with respect to the policy, so its gradient vanishes

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

$A_t > 0 \longrightarrow$ increase $\pi_\phi(y_t \mid x, y_{<t}) \longrightarrow$ increase $\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

$$A_t > 0 \longrightarrow \text{increase } \pi_\phi(y_t \mid x, y_{<t}) \longrightarrow \text{increase } \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$$

$$\text{If } \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \leq 1+\epsilon, \quad \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \uparrow$$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

$A_t > 0 \longrightarrow$ increase $\pi_\phi(y_t \mid x, y_{<t}) \longrightarrow$ increase $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$

If $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \leq 1+\epsilon$, $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$ ↑

If $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} > 1+\epsilon$, $L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left(1+\epsilon\right)A_t$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

$$A_t > 0 \longrightarrow \text{increase } \pi_\phi(y_t \mid x, y_{<t}) \longrightarrow \text{increase } \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$$

$$\text{If } \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \leq 1+\epsilon, \quad \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \uparrow$$

$$\text{If } \frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} > 1+\epsilon, \quad L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left(1+\epsilon\right)A_t$$

The clipping removes the incentive to increase $\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$ **beyond** $1+\epsilon$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

$A_t < 0 \longrightarrow$ decrease $\pi_\phi(y_t \mid x, y_{<t}) \longrightarrow$ decrease $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

$A_t < 0 \longrightarrow$ decrease $\pi_\phi(y_t \mid x, y_{<t}) \longrightarrow$ decrease $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$

If $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \geq 1-\epsilon,$ $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$ ↘

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

$A_t < 0 \longrightarrow$ decrease $\pi_\phi(y_t \mid x, y_{<t}) \longrightarrow$ decrease $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$

If $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \geq 1-\epsilon,$ $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$ ↘

If $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} < 1-\epsilon,$ $L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left(1-\epsilon\right)A_t$

Importance Sampling and Policy Gradients

$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

The $\min$ operator ensures we take the more conservative estimate between the clipped and unclipped objectives, depending on the sign of A_t

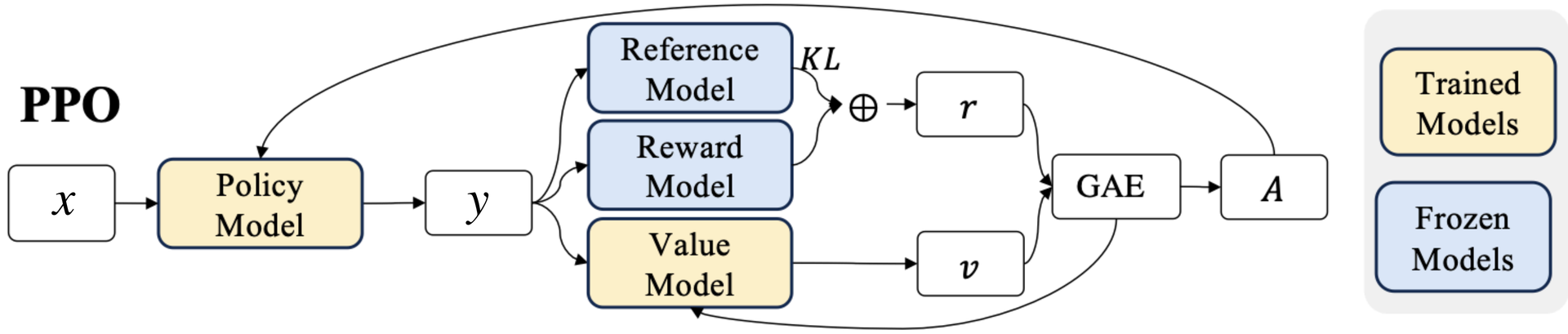
$A_t < 0 \longrightarrow$ decrease $\pi_\phi(y_t \mid x, y_{<t}) \longrightarrow$ decrease $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$

If $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} \geq 1-\epsilon,$ $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$ ↘

If $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} < 1-\epsilon,$ $L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left(1-\epsilon\right)A_t$

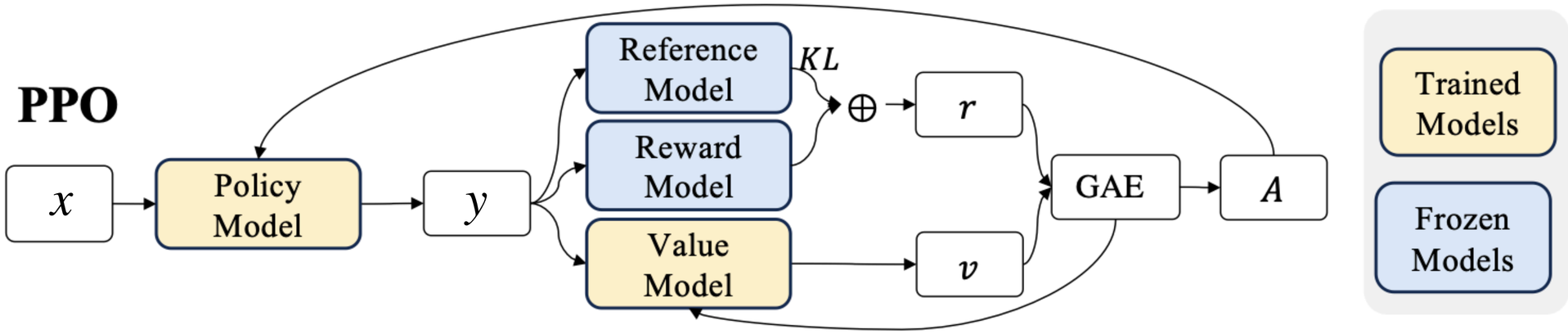
The clipping removes the incentive to decrease $\dfrac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}$ **beyond** $1-\epsilon$

Limitations of PPO



$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} \min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon \right) A_t \right) \right]$$

Limitations of PPO

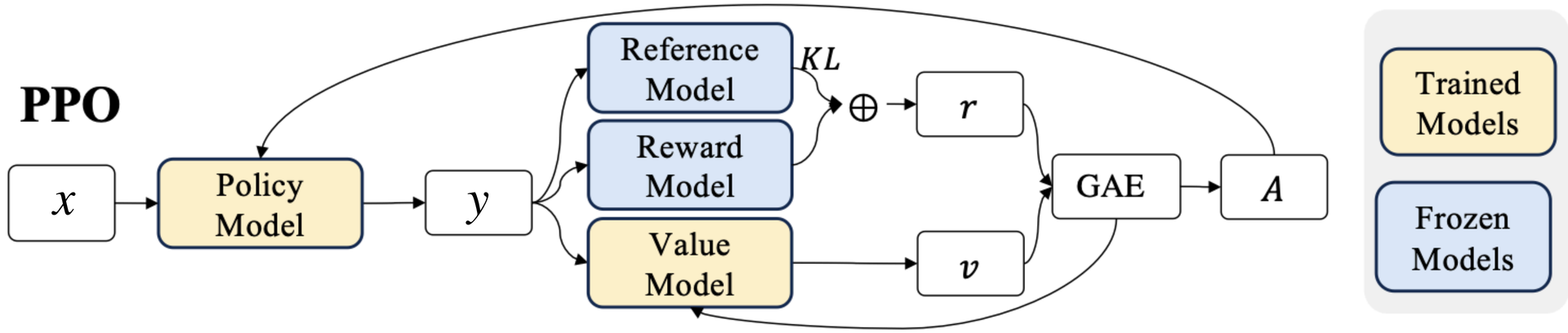


$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

$$\text{objective}(\phi) = \max_{\pi_\phi}\ \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right] \ -\ \beta\,\mathbb{D}_{\text{KL}}(\pi_\phi(y \mid x) \,\|\, \pi_{\text{ref}}(y \mid x))$$

get high reward — stay close to reference model

Limitations of PPO



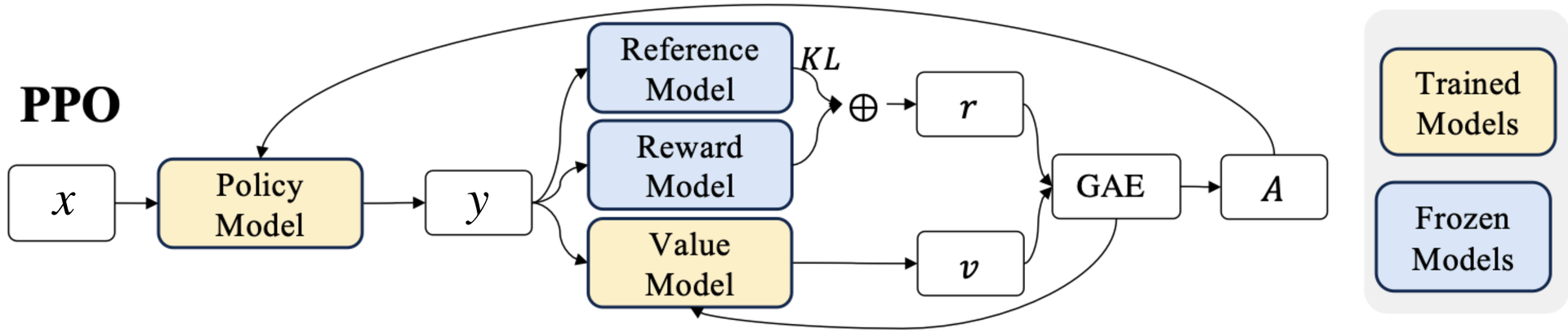
$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} \min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})} A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon \right) A_t \right) \right]$$

$$\text{objective}(\phi) = \max_{\pi_\phi} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)} \left[R(x,y) \right] - \beta\, \mathbb{D}_{\text{KL}}(\pi_\phi(y \mid x) \,\|\, \pi_{\text{ref}}(y \mid x))$$

get high reward — stay close to reference model

Training LLMs with PPO needs to coordinate four models to work together, i.e., a policy model, a reward model, a reference model, and a value model

Limitations of PPO



$$L(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\phi_{\text{old}}}(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|}\min\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}A_t,\ \text{clip}\left(\frac{\pi_\phi(y_t \mid x, y_{<t})}{\pi_{\phi_{\text{old}}}(y_t \mid x, y_{<t})}, 1-\epsilon,\ 1+\epsilon\right)A_t\right)\right]$$

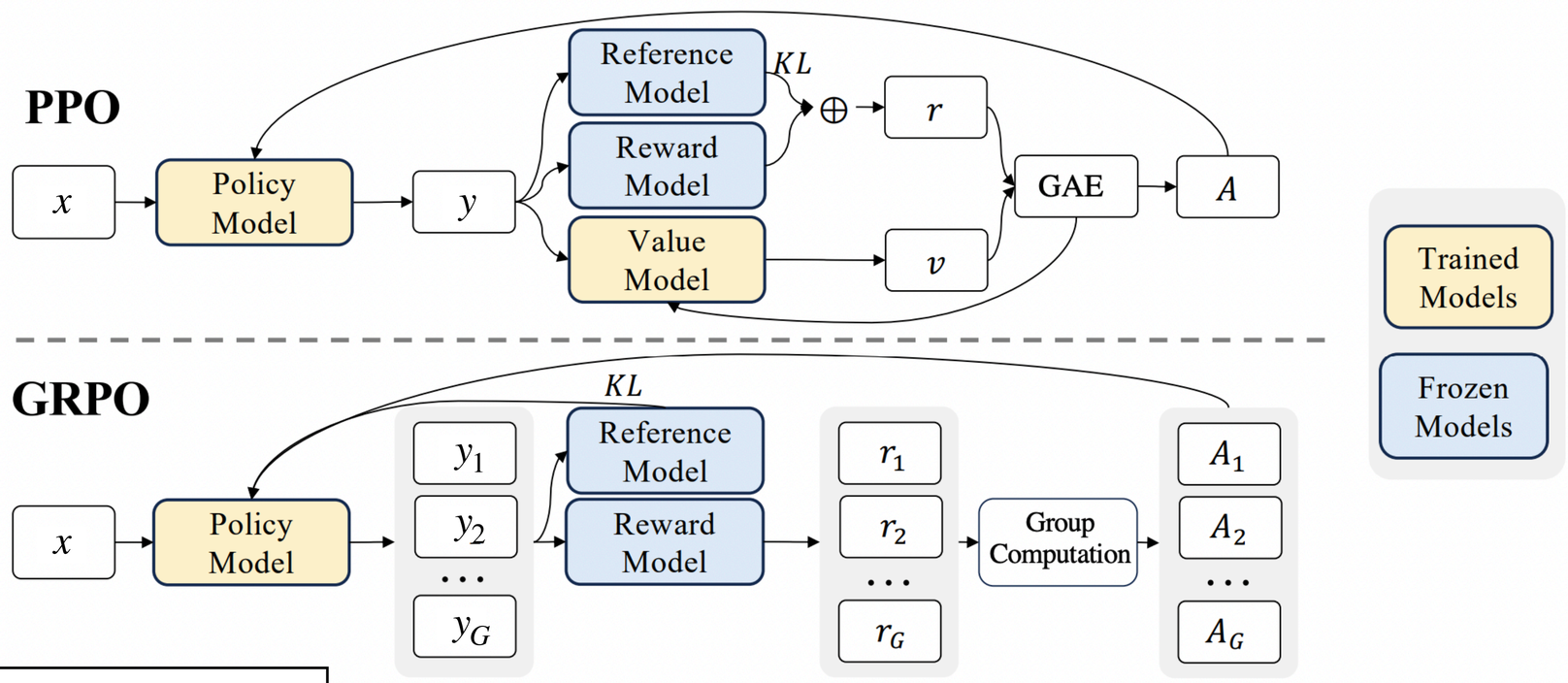
$$\text{objective}(\phi) = \max_{\pi_\phi}\ \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\phi(\cdot|x)}\left[R(x,y)\right] \ -\ \beta\,\mathbb{D}_{\text{KL}}(\pi_\phi(y \mid x) \,\|\, \pi_{\text{ref}}(y \mid x))$$

get high reward — stay close to reference model

Training LLMs with PPO needs to coordinate four models to work together, i.e., a policy model, a reward model, a reference model, and a **value model**

Usually same size as policy model

Group Relative Policy Optimization (GRPO)



deepseek

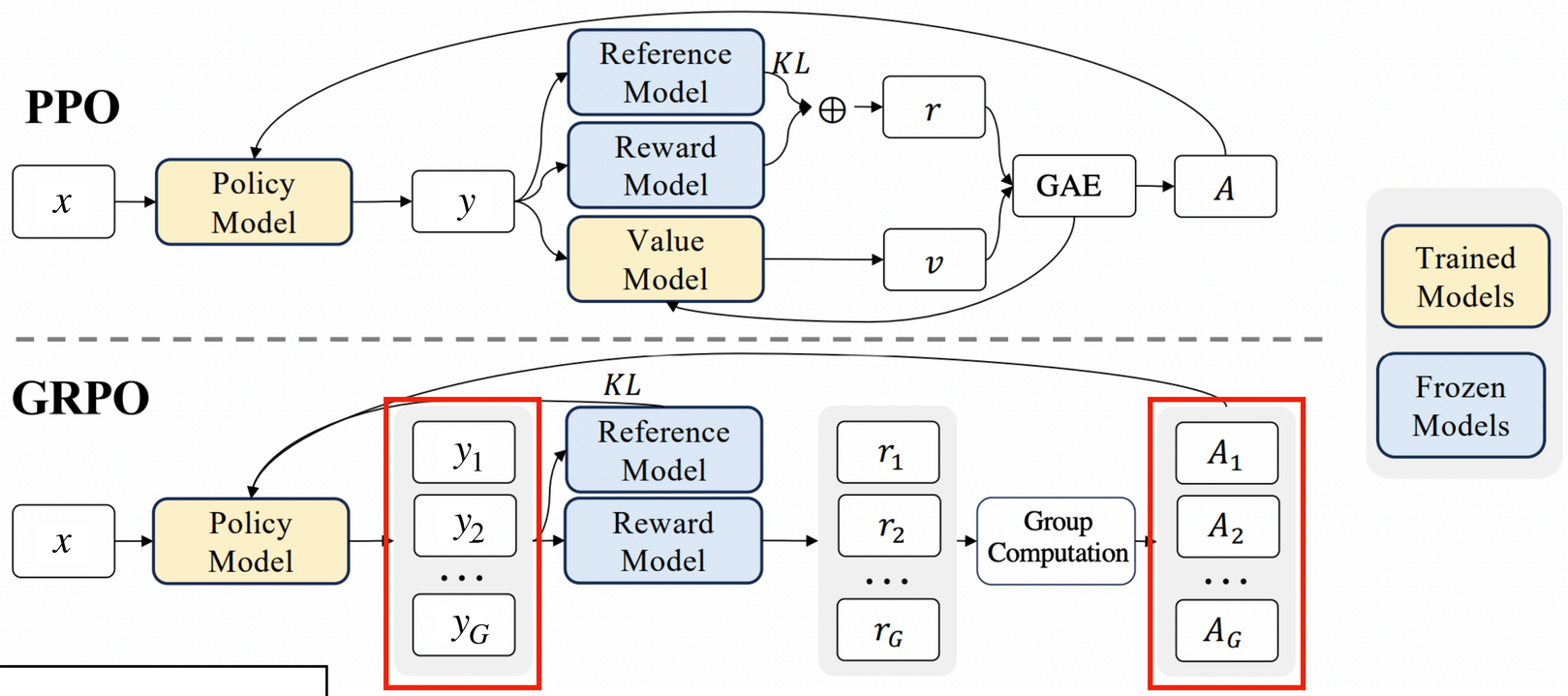
DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models

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

Shao et al. (2024)

Group Relative Policy Optimization (GRPO)



GRPO drops the value model

deepseek

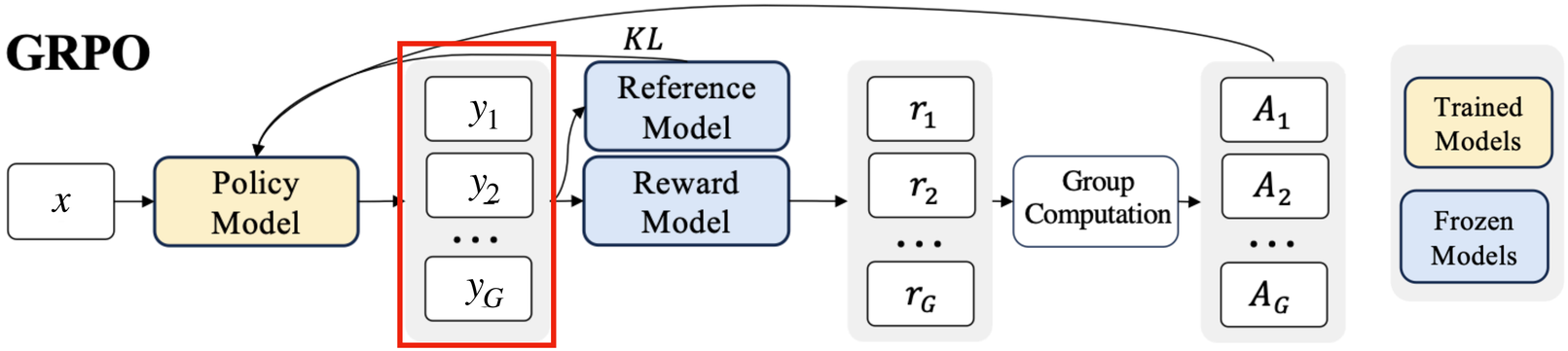
DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models

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

Shao et al. (2024)

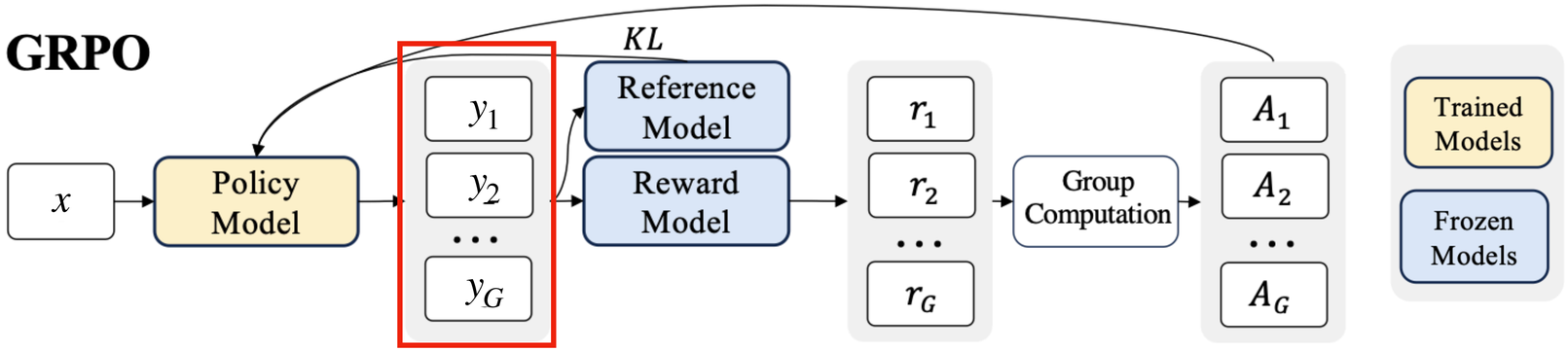
Group Relative Policy Optimization (GRPO)



$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\ \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

Generate multiple rollouts per question.

Group Relative Policy Optimization (GRPO)

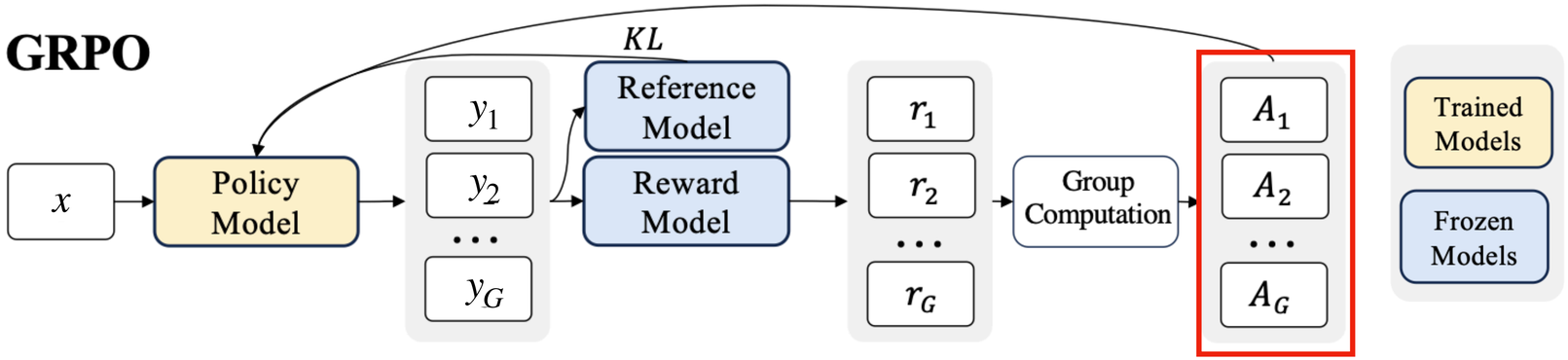


$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\ \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

Generate multiple rollouts per question.

Given these rollouts, we can estimate the "advantage" function based on the relative goodness of these responses.

Group Relative Policy Optimization (GRPO)



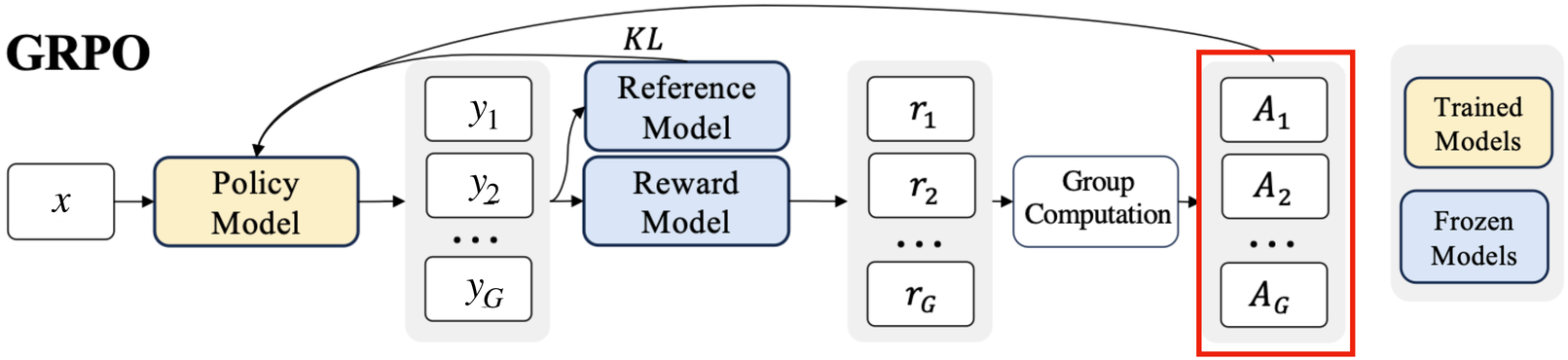
$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

Generate multiple rollouts per question.

Given these rollouts, we can estimate the "advantage" function based on the relative goodness of these responses.

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

Group Relative Policy Optimization (GRPO)



$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t}, \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

Generate multiple rollouts per question.

Given these rollouts, we can estimate the "advantage" function based on the relative goodness of these responses.

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

It set the advantage for every token to be the same.

Dynamic Sampling Policy Optimization (DAPO)

ByteDance | Seed

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

DAPO: An Open-Source LLM Reinforcement Learning System at Scale

[1]ByteDance Seed [2]Institute for AI Industry Research (AIR), Tsinghua University
[3]The University of Hong Kong
[4]SIA-Lab of Tsinghua AIR and ByteDance Seed

ByteDance Seed (2025)

Dynamic Sampling Policy Optimization (DAPO)

ByteDance | Seed

AIR 清華大學 智能产业研究院 Institute for AI Industry Research, Tsinghua University

DAPO: An Open-Source LLM Reinforcement Learning System at Scale

[1]ByteDance Seed [2]Institute for AI Industry Research (AIR), Tsinghua University
[3]The University of Hong Kong
[4]SIA-Lab of Tsinghua AIR and ByteDance Seed

Some improvements over GRPO to solve the following problems:

Entropy Collapse (Mostly seen in GRPO), Reward Noise, Training Instability

ByteDance Seed (2025)

4 Techniques Introduced in DAPO

They introduce 4 key techniques to improve GRPO

1. **Clip-Higher,** which promotes the diversity of the system and avoids entropy collapse;
2. **Dynamic Sampling,** which improves training efficiency and stability;
3. **Token-Level Policy Gradient Loss,** which is critical in long-CoT RL scenarios;
4. **Overlong Reward Shaping,** which reduces reward noise and stabilizes training.

Technique 1: Clip-Higher

This is to promote the diversity of rollouts and prevent entropy collapse

$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon, 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

Normally the ϵ is 0.2, and here ϵ_{high} is 0.25, so allow the good token that has bigger probability difference than the reference model to have gradients.

$$J_{\text{DAPO}}(\phi) = \mathbb{E}_{(x,a)\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{\sum_{i=1}^{G}|y^{(i)}|} \sum_{i=1}^{G}\sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}, 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}} \right) \hat{A}_{i,t} \right) \right]$$

$$\text{s.t.} \quad 0 < \left| \left\{ y^{(i)} \mid \text{is\_equivalent}(a, y^{(i)}) \right\} \right| < G.$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

Technique 1: Clip-Higher

This is to promote the diversity of rollouts and prevent entropy collapse

$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon, 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

But we don't want to increase ϵ_{low}, Because that will make the more bad token to go down to 0, which will make entropy collapse

$$J_{\text{DAPO}}(\phi) = \mathbb{E}_{(x,a)\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{\sum_{i=1}^{G}|y^{(i)}|} \sum_{i=1}^{G}\sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}, 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}} \right) \hat{A}_{i,t} \right) \right]$$

$$\text{s.t.} \quad 0 < \left| \left\{ y^{(i)} \mid \text{is\_equivalent}(a, y^{(i)}) \right\} \right| < G.$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

Technique 2: Dynamic Sampling

Remove rollout groups that have 0 advantages (all correct or all wrong), as they contribute nothing (no gradients), so wasting compute

$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\ \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) \ -\ \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

$$J_{\text{DAPO}}(\phi) = \mathbb{E}_{(x,a)\sim\mathcal{D},\ \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{\sum_{i=1}^{G}|y^{(i)}|} \sum_{i=1}^{G}\sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}, 1-\epsilon_{\text{low}},\ 1+\epsilon_{\text{high}} \right) \hat{A}_{i,t} \right) \right]$$

$$\text{s.t.} \quad 0 < \left| \left\{ y^{(i)} \mid \text{is\_equivalent}(a, y^{(i)}) \right\} \right| < G.$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

Technique 2: Dynamic Sampling

Remove rollout groups that have 0 advantages (all correct or all wrong), as they contribute nothing (no gradients), so wasting compute

$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

$$J_{\text{DAPO}}(\phi) = \mathbb{E}_{(x,a)\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{\sum_{i=1}^{G}|y^{(i)}|} \sum_{i=1}^{G}\sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}, 1-\epsilon_{\text{low}},\ 1+\epsilon_{\text{high}} \right) \hat{A}_{i,t} \right) \right]$$

$$\text{s.t.} \quad 0 < \left| \left\{ y^{(i)} \mid \text{is\_equivalent}(a, y^{(i)}) \right\} \right| < G.$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

The number of ground-truth answers $y^{(i)}$ that are equivalent to the model's answer a is greater than 0 but less than G.

Technique 3: Token-Level Policy Gradient Loss

GRPO assigns equal weight to each sequence, which make long generation have low contribution in gradients.

Under-reward good long generation while under-penalize bad long generation

$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

$$J_{\text{DAPO}}(\phi) = \mathbb{E}_{(x,a)\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{\sum_{i=1}^{G}|y^{(i)}|} \sum_{i=1}^{G}\sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}, 1-\epsilon_{\text{low}},\ 1+\epsilon_{\text{high}} \right) \hat{A}_{i,t} \right) \right]$$

s.t. $$0 < \left| \left\{ y^{(i)} \mid \text{is\_equivalent}(a, y^{(i)}) \right\} \right| < G.$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

Technique 3: Token-Level Policy Gradient Loss

GRPO assigns equal weight to each sequence, which make long generation have low contribution in gradients.

Under-reward good long generation while under-penalize bad long generation

$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

DAPO assigns equal weight to each token

$$J_{\text{DAPO}}(\phi) = \mathbb{E}_{(x,a)\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{\sum_{i=1}^{G}|y^{(i)}|} \sum_{i=1}^{G}\sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}, 1-\epsilon_{\text{low}},\ 1+\epsilon_{\text{high}} \right) \hat{A}_{i,t} \right) \right]$$

s.t. $$0 < \left| \left\{ y^{(i)} \mid \text{is\_equivalent}(a, y^{(i)}) \right\} \right| < G.$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

Technique 4: Overlong Reward Shaping

Take care of overlong truncated samples

Technique 4: Overlong Reward Shaping

Take care of overlong truncated samples

- **Overlong Filtering** - Mask the loss of truncated samples.

We still use truncated samples' rewards in the advantage calculation but zero-out its loss contribution

Technique 4: Overlong Reward Shaping

Take care of overlong truncated samples

- **Overlong Filtering** - Mask the loss of truncated samples.

We still use truncated samples' rewards in the advantage calculation but zero-out its loss contribution

- **Soft overlong punishment** - Length-aware penalty for truncated samples. Longer the response, greater the punishment

Technique 4: Overlong Reward Shaping

Take care of overlong truncated samples

- **Overlong Filtering** - Mask the loss of truncated samples.

We still use truncated samples' rewards in the advantage calculation but zero-out its loss contribution

- **Soft overlong punishment** - Length-aware penalty for truncated samples. Longer the response, greater the punishment

They define a punishment interval L_{cache}

$$R_{\text{length}}(y) = \begin{cases} 0, & |y| \leq L_{\max} - L_{\text{cache}} \\ \frac{(L_{\max} - L_{\text{cache}}) - |y|}{L_{\text{cache}}}, & L_{\max} - L_{\text{cache}} < |y| \leq L_{\max} \\ -1, & L_{\max} < |y| \end{cases}$$

Technique 4: Overlong Reward Shaping

Take care of overlong truncated samples

- **Overlong Filtering** - Mask the loss of truncated samples.

We still use truncated samples' rewards in the advantage calculation but zero-out its loss contribution

- **Soft overlong punishment** - Length-aware penalty for truncated samples. Longer the response, greater the punishment

They define a punishment interval L_{cache}

This length penalty is added to the correctness reward

$$R_{\text{length}}(y) = \begin{cases} 0, & |y| \leq L_{\max} - L_{\text{cache}} \\ \frac{(L_{\max} - L_{\text{cache}}) - |y|}{L_{\text{cache}}}, & L_{\max} - L_{\text{cache}} < |y| \leq L_{\max} \\ -1, & L_{\max} < |y| \end{cases}$$

Remove KL divergence

KL divergence in the loss is unnecessary, as we have clip-higher to prevent entropy collapse

$$J_{\text{GRPO}}(\phi) = \mathbb{E}_{x\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi(y_t^{(i)} \mid x, y_{<t}^{(i)})}{\pi_{\phi_{\text{old}}}(y_t^{(i)} \mid x, y_{<t}^{(i)})}, 1-\epsilon,\ 1+\epsilon \right) \hat{A}_{i,t} \right) - \beta\, D_{\text{KL}}(\pi_\phi \,\|\, \pi_{\text{ref}}) \right]$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$

$$J_{\text{DAPO}}(\phi) = \mathbb{E}_{(x,a)\sim\mathcal{D},\, \{y^{(i)}\}_{i=1}^{G}\sim\pi_{\phi_{\text{old}}}(\cdot|x)} \left[\frac{1}{\sum_{i=1}^{G}|y^{(i)}|} \sum_{i=1}^{G}\sum_{t=1}^{|y^{(i)}|} \min\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)} \hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\phi\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\pi_{\phi_{\text{old}}}\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}, 1-\epsilon_{\text{low}},\ 1+\epsilon_{\text{high}} \right) \hat{A}_{i,t} \right) \right]$$

$$\text{s.t.} \quad 0 < \left| \left\{ y^{(i)} \mid \text{is\_equivalent}(a, y^{(i)}) \right\} \right| < G.$$

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$$