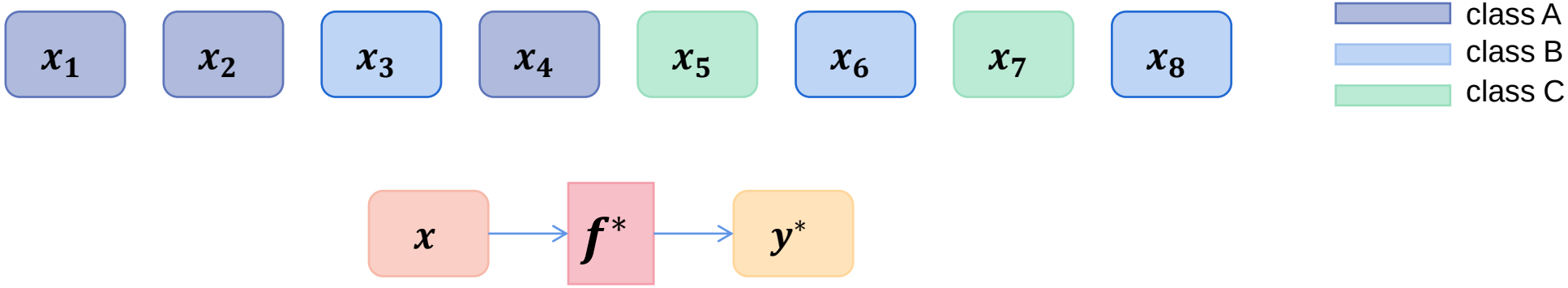
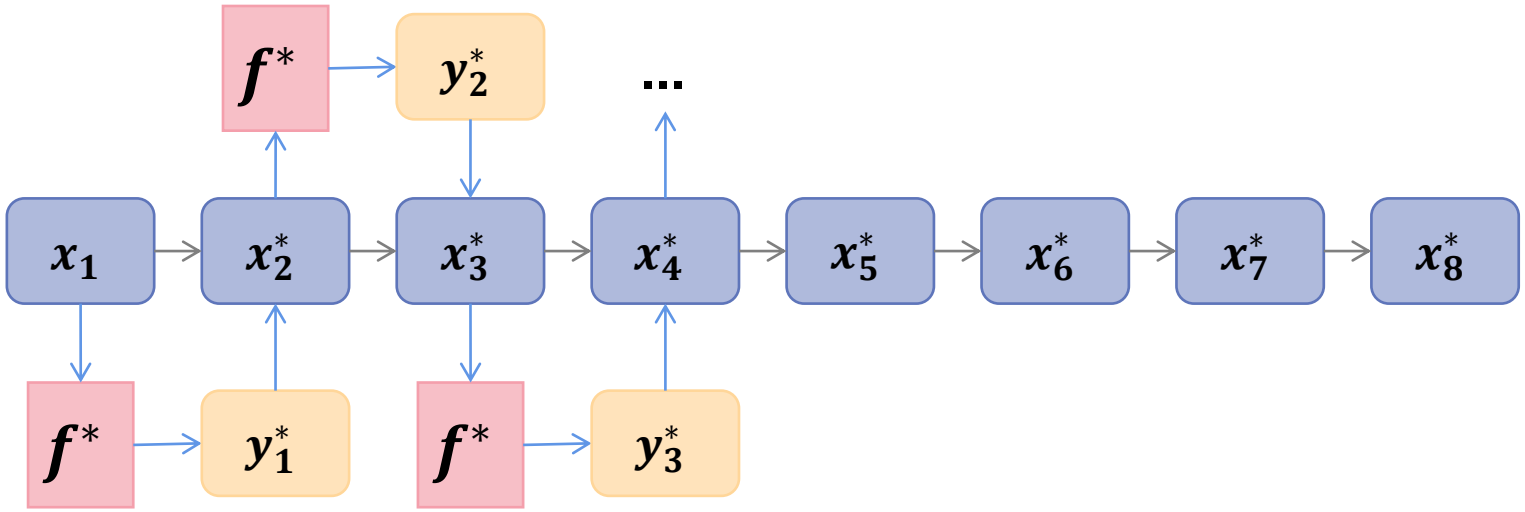
Bridging RL and Other ML Paradigms

Background

Supervised Learning



Reinforcement Learning



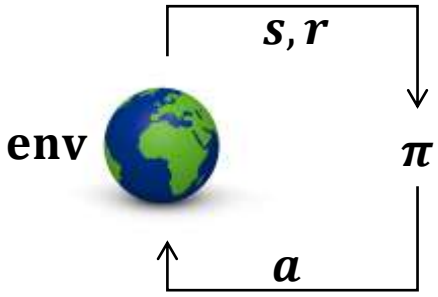
sequence decision

- $<x_1, x_2^*, x_3^*, \ldots>$
- $x_{t+1} = P(x_t, y_t)$

$x \rightarrow s$	state
$f \rightarrow \pi$	policy
$y \rightarrow a$	action
$R(s,a)$	reward

Background

Concept



- env → s_0
- π → $a_0 \sim \pi(\cdot|s_0)$
- env → s_1, r_0
- ...

MDP(Markov Decision Process)

Definition: (S, A, R, P, γ)

$s \in S \quad a \in A \quad r_t = R(s_t, a_t)$

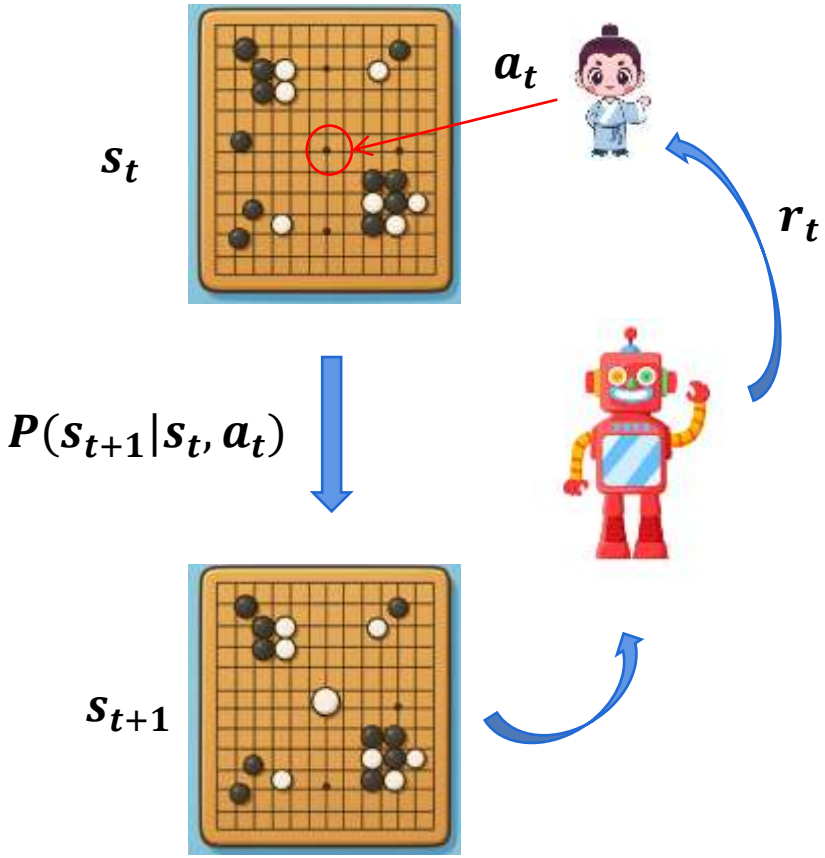
Markov property: $P(s_{t+1}|s_{<t}, a_{<t}, s_t, a_t) = P(s_{t+1}|s_t, a_t)$

Policy objective:

$$J(\pi)$$

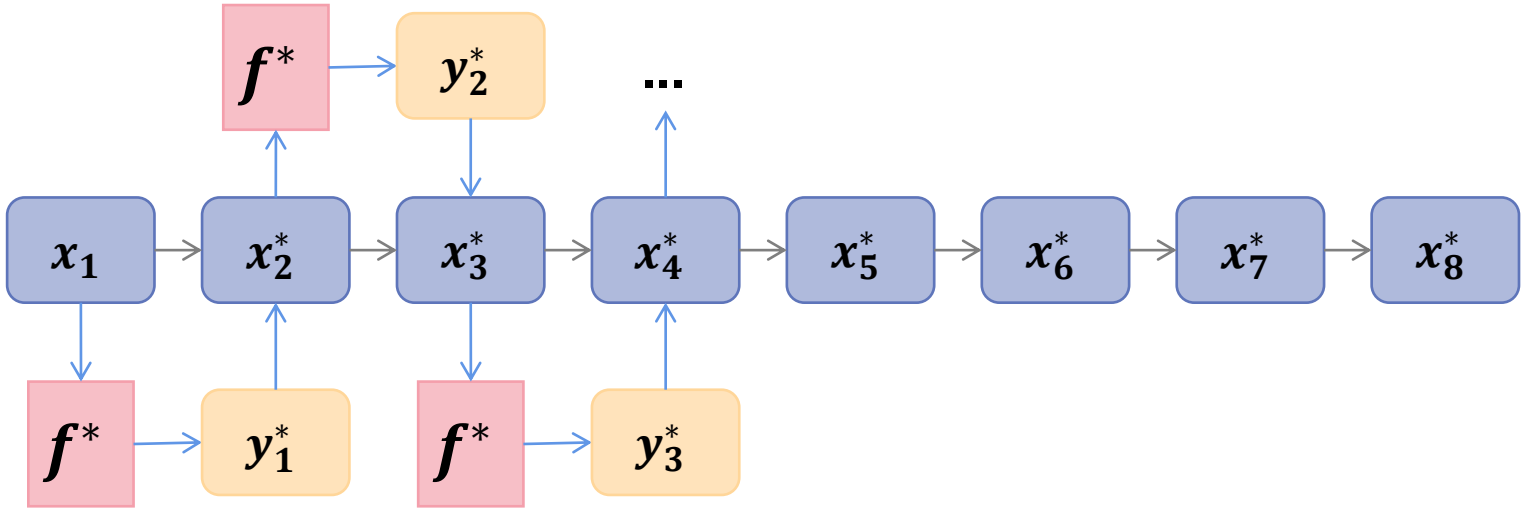
$$= \mathbb{E}_{a_0 \sim \pi(\cdot|s_0)}\left[r(s_0, a_0) + \gamma \mathbb{E}_{s_1 \sim P(\cdot|s_0, a_0)}\left[\mathbb{E}_{a_1 \sim \pi(\cdot|s_1)}[r(s_1, a_1) + \cdots]\right]\right]$$

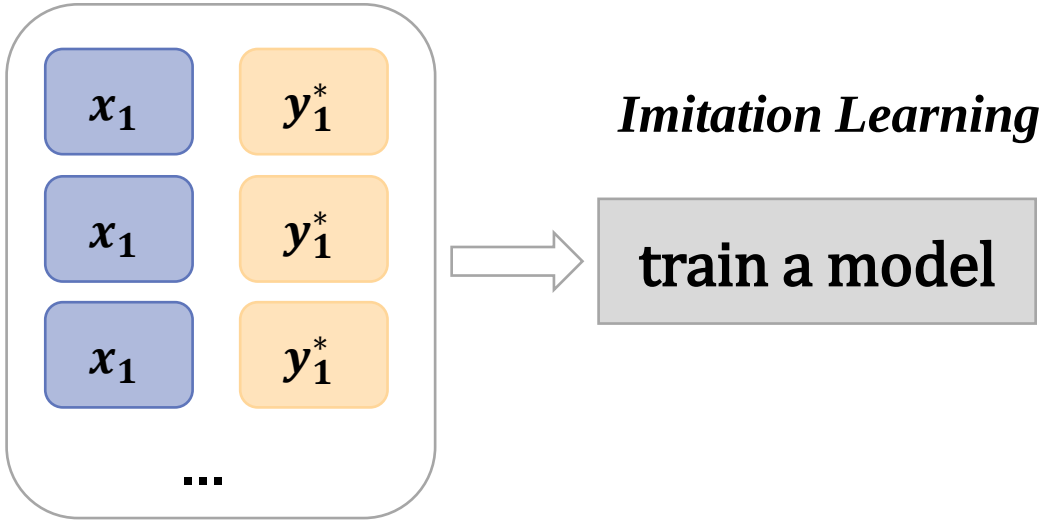
Example



Background

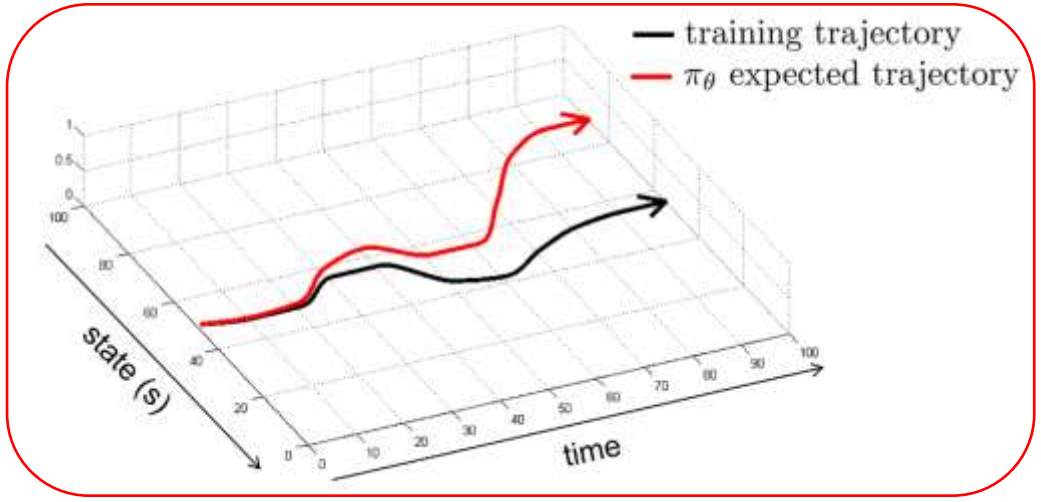
Reinforcement Learning





Imitation Learning

train a model



compounding errors

Reinforcement Learning-Guided Semi-Supervised Learning

Marzi Heidari **Hanping Zhang** **Yuhong Guo**
School of Computer Science, Carleton Univerity, Ottawa, Canada

NeurIPS 2024

Method

MDP for Semi-Supervised Learning

policy: $\pi_\theta(\cdot) = P_\theta(\cdot)$

- **state:** $X^l \quad X^u$
- **action:** $Y^l \quad Y^u$

one-armed bandit: No need for $\boldsymbol{\gamma}$ and $\boldsymbol{P}$

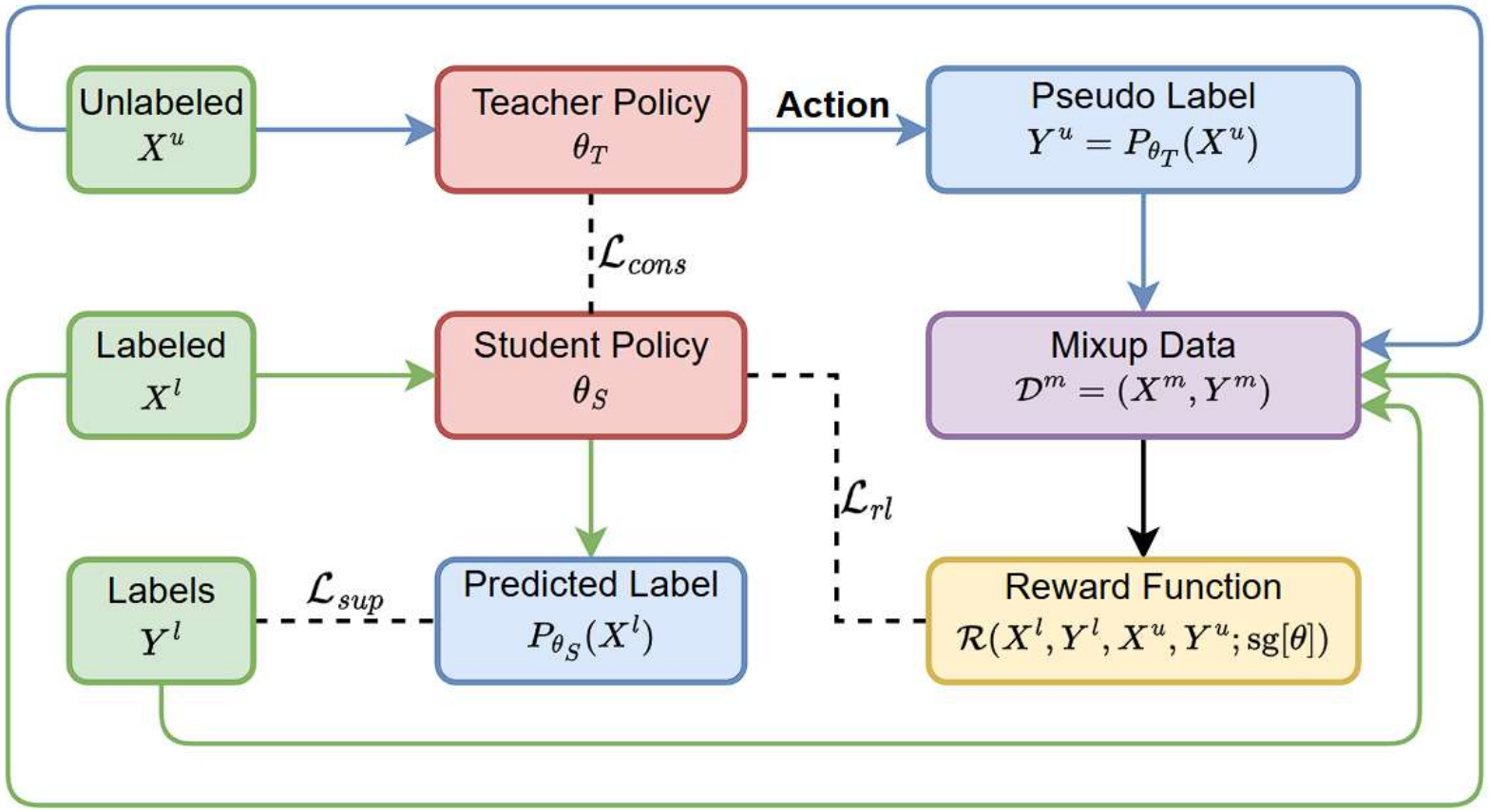
reward function: ***data mixup***

$$\mathcal{D}^l \longrightarrow \widetilde{\mathcal{D}}^l \qquad r = \lceil \frac{N^u}{N^l} \rceil$$

$$x_i^{\mathrm{m}} = \mu\, x_i^u + (1-\mu)\, x_i^l, \qquad \mathbf{y}_i^{\mathrm{m}} = \mu\, \mathbf{y}_i^u + (1-\mu)\, \mathbf{y}_i^l$$

$$\mathcal{R}(s, a; \mathrm{sg}[\theta]) = \mathcal{R}(X^l, Y^l, X^u, Y^u; \mathrm{sg}[\theta]) = -\frac{1}{C \cdot N^{\mathrm{m}}} \sum_{i=1}^{N^{\mathrm{m}}} ||P_\theta(x_i^{\mathrm{m}}) - \mathbf{y}_i^{\mathrm{m}}||_2^2$$

Method



$$\begin{aligned}\mathcal{L}_{\mathrm{rl}} &= -\mathbb{E}_{\mathbf{y}_i^u \sim \pi_\theta} \mathrm{KL}(\mathbf{e}, \mathbf{y}_i^u) \mathcal{R}(s, a; \mathrm{sg}[\theta]) \\ &= -\mathbb{E}_{x_i^u \in \mathcal{D}_u} \mathrm{KL}(\mathbf{e}, P_\theta(x_i^u)) \mathcal{R}(s, a; \mathrm{sg}[\theta])\end{aligned}$$

$$\mathcal{L}^{\mathrm{sup}} = \mathbb{E}_{(x^l, \mathbf{y}^l) \in \mathcal{D}^l} \left[\ell_{CE} \left(P_{\theta_S}(x^l), \mathbf{y}^l \right) \right]$$

$$\mathcal{L}^{\mathrm{cons}} = \mathbb{E}_{x^u \in \mathcal{D}^u} \left[\ell_{\mathrm{KL}} \left(P_{\theta_S}(x^u), P_{\theta_T}(x^u) \right) \right]$$

$$\mathcal{L}(\theta_S) = \mathcal{L}_{\mathrm{rl}} + \lambda_1 \mathcal{L}_{\mathrm{sup}} + \lambda_2 \mathcal{L}_{\mathrm{cons}}$$

Experiments

Table 1: Performance of RLGSSL and state-of-the-art SSL algorithms with the CNN-13 network. We report the average test errors and the standard deviations of 5 trials.

Dataset	CIFAR-10			CIFAR-100	
Number of Labeled Samples	1000	2000	4000	4000	10000
Supervised	$39.95_{(0.75)}$	$27.67_{(0.12)}$	$20.42_{(0.21)}$	$58.31_{(0.89)}$	$44.56_{(0.30)}$
Supervised + MixUp [38]	$31.83_{(0.65)}$	$24.22_{(0.15)}$	$17.37_{(0.35)}$	$54.87_{(0.07)}$	$40.97_{(0.47)}$
Π-model [6]	$28.74_{(0.48)}$	$17.57_{(0.44)}$	$12.36_{(0.17)}$	$55.39_{(0.55)}$	$38.06_{(0.37)}$
Temp-ensemble [6]	$25.15_{(1.46)}$	$15.78_{(0.44)}$	$11.90_{(0.25)}$	-	$38.65_{(0.51)}$
Mean Teacher[8]	$21.55_{(0.53)}$	$15.73_{(0.31)}$	$12.31_{(0.28)}$	$45.36_{(0.49)}$	$35.96_{(0.77)}$
VAT [5]	$18.12_{(0.82)}$	$13.93_{(0.33)}$	$11.10_{(0.24)}$	-	-
SNTG [10]	$18.41_{(0.52)}$	$13.64_{(0.32)}$	$10.93_{(0.14)}$	-	$37.97_{(0.29)}$
Learning to Reweight [42]	$11.74_{(0.12)}$	-	$9.44_{(0.17)}$	$46.62_{(0.29)}$	$37.31_{(0.47)}$
MT + Fast SWA [9]	15.58	11.02	9.05	-	$33.62_{(0.54)}$
ICT [11]	$12.44_{(0.57)}$	$8.69_{(0.15)}$	$7.18_{(0.24)}$	$40.07_{(0.38)}$	$32.24_{(0.16)}$
RLGSSL (Ours)	$\mathbf{9.15}_{(0.57)}$	$\mathbf{6.90}_{(0.11)}$	$\mathbf{6.11}_{(0.10)}$	$\mathbf{36.92}_{(0.45)}$	$\mathbf{29.12}_{(0.20)}$

Table 2: Performance of RLGSSL and state-of-the-art SSL algorithms with the CNN-13 network. We report the average test errors and the standard deviations of 5 trials.

	VAT [5]	Π-model [6]	Temp-ensemble [6]	MT [8]	ICT [11]	SNTG [10]	RLGSSL (Ours)
SVHN/500	-	$6.65_{(0.53)}$	$5.12_{(0.13)}$	$4.18_{(0.27)}$	$4.23_{(0.15)}$	$3.99_{(0.24)}$	$\mathbf{3.12}_{(0.07)}$
SVHN/1000	$5.42_{(0.00)}$	$4.82_{(0.17)}$	$4.42_{(0.16)}$	$3.95_{(0.19)}$	$3.89_{(0.04)}$	$3.86_{(0.27)}$	$\mathbf{3.05}_{(0.04)}$

Experiments

Main Results

Table 1: Performance of RLGSSL and state-of-the-art SSL algorithms with the CNN-13 network. We report the average test errors and the standard deviations of 5 trials.

Dataset	CIFAR-10			CIFAR-100	
Number of Labeled Samples	1000	2000	4000	4000	10000
Supervised	$39.95_{(0.75)}$	$27.67_{(0.12)}$	$20.42_{(0.21)}$	$58.31_{(0.89)}$	$44.56_{(0.30)}$
Supervised + MixUp [38]	$31.83_{(0.65)}$	$24.22_{(0.15)}$	$17.37_{(0.35)}$	$54.87_{(0.07)}$	$40.97_{(0.47)}$
Π-model [6]	$28.74_{(0.48)}$	$17.57_{(0.44)}$	$12.36_{(0.17)}$	$55.39_{(0.55)}$	$38.06_{(0.37)}$
Temp-ensemble [6]	$25.15_{(1.46)}$	$15.78_{(0.44)}$	$11.90_{(0.25)}$	-	$38.65_{(0.51)}$
Mean Teacher[8]	$21.55_{(0.53)}$	$15.73_{(0.31)}$	$12.31_{(0.28)}$	$45.36_{(0.49)}$	$35.96_{(0.77)}$
VAT [5]	$18.12_{(0.82)}$	$13.93_{(0.33)}$	$11.10_{(0.24)}$	-	-
SNTG [10]	$18.41_{(0.52)}$	$13.64_{(0.32)}$	$10.93_{(0.14)}$	-	$37.97_{(0.29)}$
Learning to Reweight [42]	$11.74_{(0.12)}$	-	$9.44_{(0.17)}$	$46.62_{(0.29)}$	$37.31_{(0.47)}$
MT + Fast SWA [9]	15.58	11.02	9.05	-	$33.62_{(0.54)}$
ICT [11]	$12.44_{(0.57)}$	$8.69_{(0.15)}$	$7.18_{(0.24)}$	$40.07_{(0.38)}$	$32.24_{(0.16)}$
RLGSSL (Ours)	$\mathbf{9.15}_{(0.57)}$	$\mathbf{6.90}_{(0.11)}$	$\mathbf{6.11}_{(0.10)}$	$\mathbf{36.92}_{(0.45)}$	$\mathbf{29.12}_{(0.20)}$

Table 2: Performance of RLGSSL and state-of-the-art SSL algorithms with the CNN-13 network. We report the average test errors and the standard deviations of 5 trials.

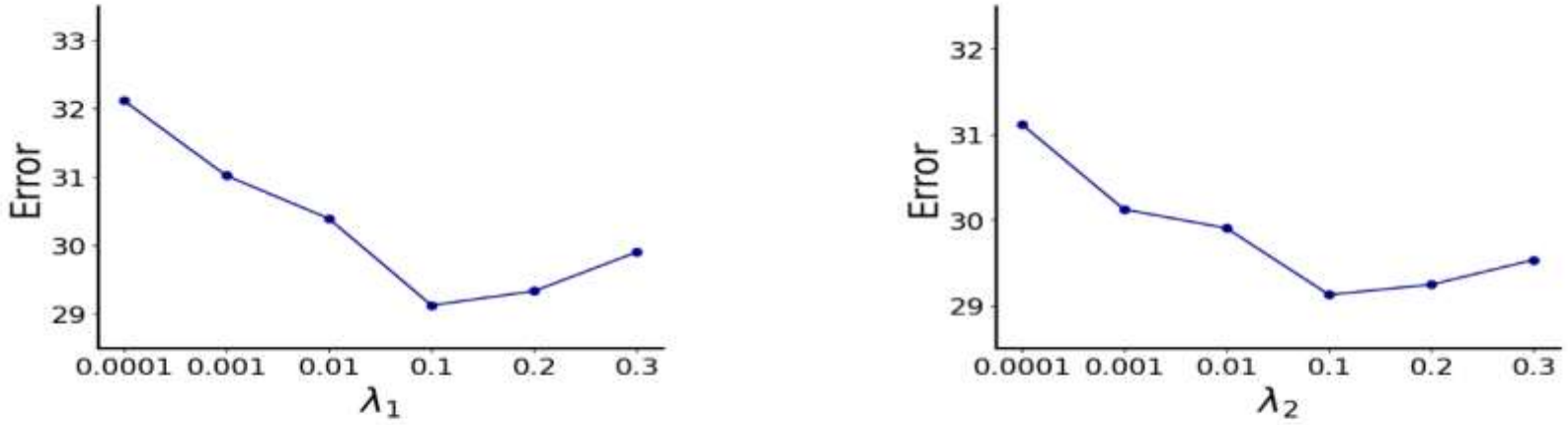
	VAT [5]	Π-model [6]	Temp-ensemble [6]	MT [8]	ICT [11]	SNTG [10]	RLGSSL (Ours)
SVHN/500	-	$6.65_{(0.53)}$	$5.12_{(0.13)}$	$4.18_{(0.27)}$	$4.23_{(0.15)}$	$3.99_{(0.24)}$	$\mathbf{3.12}_{(0.07)}$
SVHN/1000	$5.42_{(0.00)}$	$4.82_{(0.17)}$	$4.42_{(0.16)}$	$3.95_{(0.19)}$	$3.89_{(0.04)}$	$3.86_{(0.27)}$	$\mathbf{3.05}_{(0.04)}$

Experiments

Ablation Study

Table 4: Ablation study results. We report the test errors on CIFAR-100 with 10000, and 4000 labels on CNN-13 backbone 5 trials.

	RLGSSL	−w/o $\mathcal{L}_{\text{rl}}$	−w/o $\mathcal{L}_{\text{sup}}$	−w/o $\mathcal{L}_{\text{cons}}$	−w/o EMA	−w/o mixup
CIFAR-100/4000	$\mathbf{36.92}_{(0.45)}$	$44.92_{(0.55)}$	$39.52_{(0.58)}$	$38.78_{(0.48)}$	$43.12_{(0.52)}$	$40.12_{(0.51)}$
CIFAR-100/10000	$\mathbf{29.12}_{(0.20)}$	$33.12_{(0.52)}$	$32.67_{(0.45)}$	$31.48_{(0.32)}$	$32.84_{(0.45)}$	$31.48_{(0.32)}$
	RLGSSL	$\mathcal{R} = 1$	$\mathcal{R} : \mu = 0$	$\mathcal{R}(L_2 \to \text{KL})$	$\mathcal{R}(L_2 \to \text{JS})$	$\mathcal{R}$: w/o sg[θ]
CIFAR-100/4000	$\mathbf{36.92}_{(0.45)}$	$39.52_{(0.63)}$	$39.54_{(0.33)}$	$38.02_{(0.42)}$	$39.52_{(0.45)}$	$40.62_{(0.55)}$
CIFAR-100/10000	$\mathbf{29.12}_{(0.20)}$	$31.25_{(0.62)}$	$32.37_{(0.57)}$	$31.12_{(0.52)}$	$31.39_{(0.68)}$	$32.12_{(0.62)}$



(a) λ_1 (b) λ_2

Figure 2: Sensitivity analysis for four hyperparameters λ_1 and λ_2 CIFAR-100 using 10000 labeled samples (a) λ_1, (b) λ_2.

Bridging Supervised Learning and Reinforcement Learning in Math Reasoning

Huayu Chen[1,2] **Kaiwen Zheng**[1,2] **Qinsheng Zhang**[2] **Ganqu Cui**[1] **Yin Cui**[2]

Haotian Ye[2,3] **Tsung-Yi Lin**[2] **Ming-Yu Liu**[2] **Jun Zhu**[1†] **Haoxiang Wang**[2]

[1]Tsinghua University [2]NVIDIA [3]Stanford University

https://research.nvidia.com/labs/dir/Negative-aware-Fine-Tuning

RL for LLMs

MDP: (S, A, R, P, γ) ⟹ Language-augmented MDP: (V, S, A, R, P, γ) where $\gamma = 1$

V: vocabulary

A specific token: $w \in V$

S: $S \subset V^M$ $\quad s = (w_1, w_2, w_3, \ldots, w_M)$ $\quad s_0$: prompt

A: $S \subset V^N$ $\quad a = (w_1, w_2, \ldots, w_N)$

R: $r = R(s, a)$

P: $S \times V \to S$ $\quad s_{i+1} = (s_i, w_i) = (s_0, w_{1:i+1})$ auto-regressive paradigm

Token-level policy: $\pi(w_i | s_0, w_{1:i-1})$

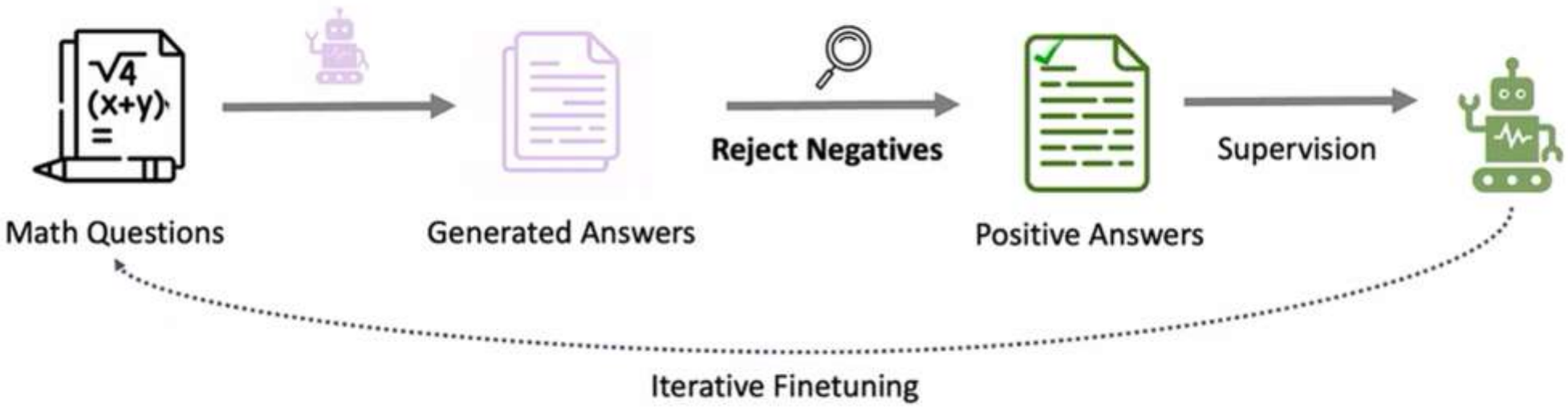
Sentence-level policy: $\pi(a|s_0) = \prod_{i=1}^{N} \pi(w_i | s_0, w_{1:i-1})$

Policy objective:

$$J(\pi) = \mathbb{E}_{s_0 \sim D}\left[\mathbb{E}_{a \sim \pi(\cdot|s_0)}[r(s_0, a)]\right]$$

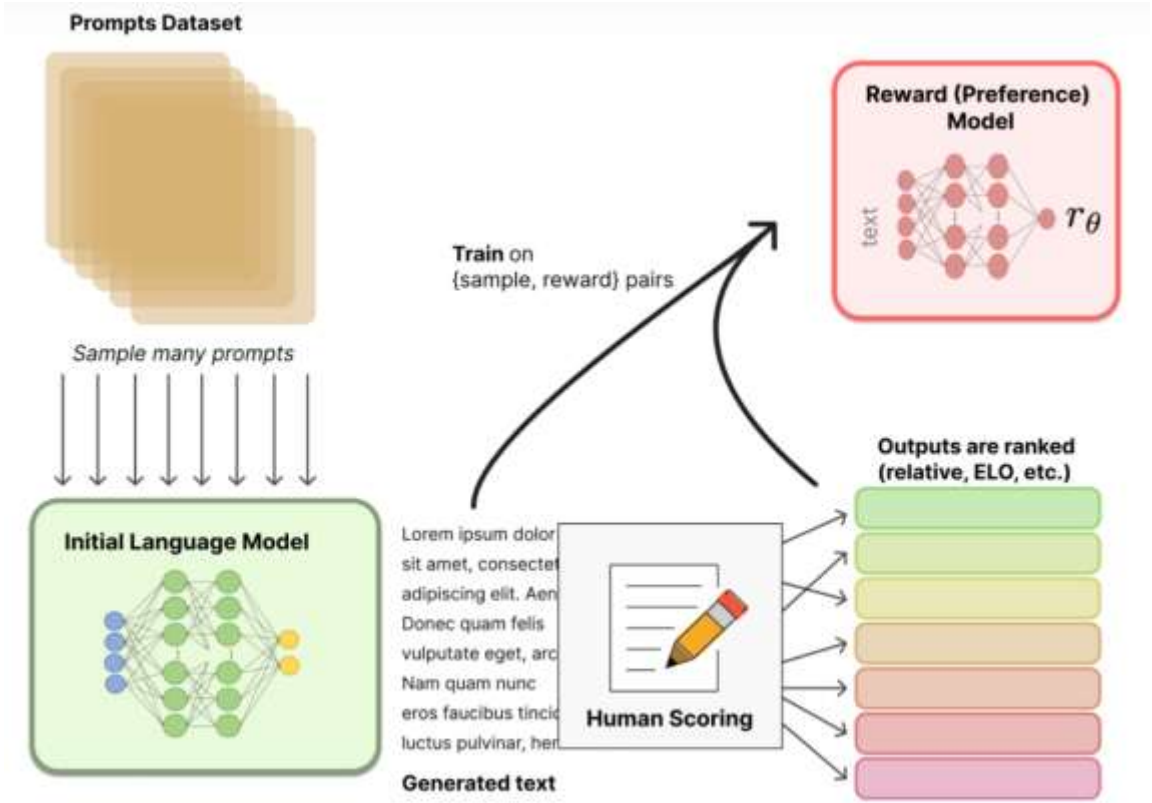
Motivation

Rejection sampleing Fine-Tuning (RFT)



How to leverage negative feedbacks?

Why use RL?



Prompt: s
Candidate answers: $\boldsymbol{a_1, a_2, a_3, a_4}$
Quality rank: $\boldsymbol{a_1 > a_2 > a_3 > a_4}$

How can we optmize the model?

Supervised learning: $\mathbf{max}\ \mathbf{log}\, \boldsymbol{p}(\boldsymbol{a_1}|\boldsymbol{s})$

⇩

Data waste & poor generalization

A solution:

for each answers: $\mathbf{max}\ \boldsymbol{R_i} \cdot \mathbf{log}\, \boldsymbol{p}(\boldsymbol{a_i}|\boldsymbol{s})$

gradient: $\boldsymbol{R_i} \cdot \boldsymbol{\nabla_\theta} \mathbf{log}\, \boldsymbol{p_\theta}(\boldsymbol{a_i}|\boldsymbol{s})$

↓

✓ **The policy gradient of RL**

Connection between SL and RL

Supervised Learning

$$\max_{\theta} \mathbb{E}_{\boldsymbol{a}\sim\pi(\boldsymbol{a}|\boldsymbol{q})} \log \pi_\theta(\boldsymbol{a}|\boldsymbol{q}) \Leftrightarrow \min_{\theta} D_{\mathrm{KL}}\left[\pi(\boldsymbol{a}|\boldsymbol{q}) \| \pi_\theta(\boldsymbol{a}|\boldsymbol{q})\right]$$

Reinforcement Learning

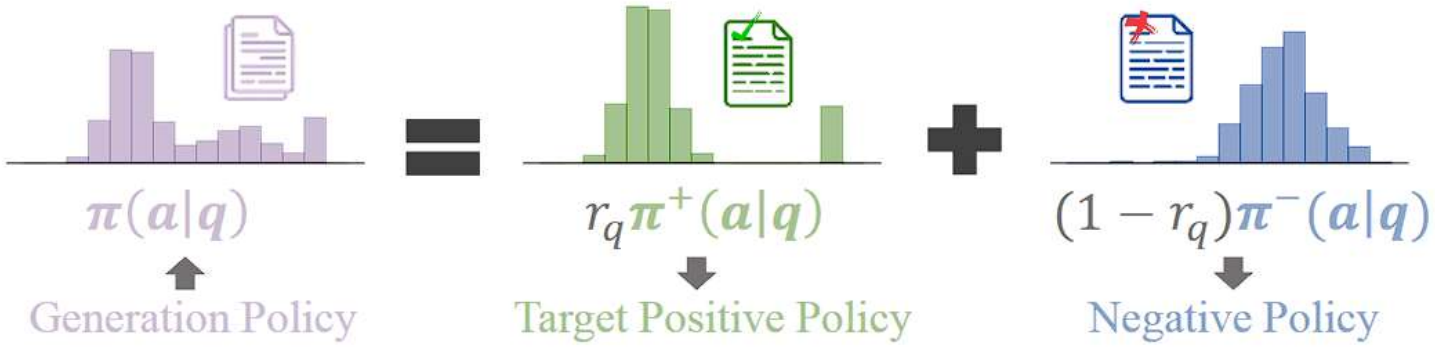
$$\max_{\theta} J(\theta) := \mathbb{E}_{\boldsymbol{a}\sim\pi_\theta(\boldsymbol{a}|\boldsymbol{q})} r(\boldsymbol{q},\boldsymbol{a})$$

$$\nabla_\theta J(\theta) = \mathbb{E}_{\boldsymbol{a}\sim\pi_\theta(\boldsymbol{a}|\boldsymbol{q})} \nabla_\theta \left[r(\boldsymbol{q},\boldsymbol{a}) \log \pi_\theta(\boldsymbol{a}|\boldsymbol{q})\right]$$

on-policy constraint

leverage of negative feedbacks

Method



target policy

$$\pi^+(a|q) := \pi(a|q, r{=}1) = \frac{\pi(a|q)p(r{=}1|q,a)}{\sum_A \pi(a|q)p(r{=}1|q,a)}$$

$$\pi^-(a|q) := \pi(a|q, r{=}0) = \frac{\pi(a|q)[1-p(r{=}1|q,a)]}{\sum_A \pi(a|q)[1-p(r{=}1|q,a)]}$$

$$r_q\pi^+(a|q) + [1-r_q]\,\pi^-(a|q) = \pi(a|q)$$

$$\pi_\theta^-(a|q) := \frac{\pi(a|q) - r_q\pi_\theta^+(a|q)}{1-r_q}$$

Method

Theorem 3.1 (Policy Optimization with Negative Answers). *Consider the maximum-likelihood objective for training the implicit negative policy* π_θ^-*:*

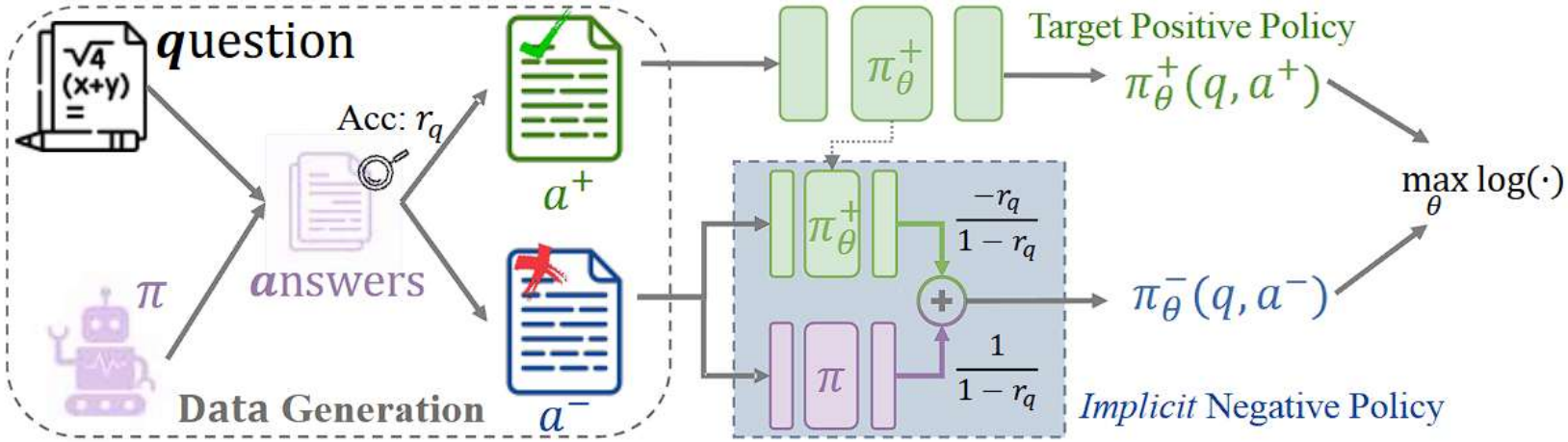
$$\max_\theta \mathbb{E}_{p(\boldsymbol{q})\pi^-(\boldsymbol{a}|\boldsymbol{q})}\left[\log \pi_\theta^-(\boldsymbol{a}|\boldsymbol{q})\right] \Leftrightarrow \min_\theta \left[-\mathbb{E}_{(\boldsymbol{q},\boldsymbol{a})\sim\mathcal{D}^-} \log \frac{\pi(\boldsymbol{a}|\boldsymbol{q}) - r_{\boldsymbol{q}}\pi_\theta^+(\boldsymbol{a}|\boldsymbol{q})}{1 - r_{\boldsymbol{q}}}\right] \tag{8}$$

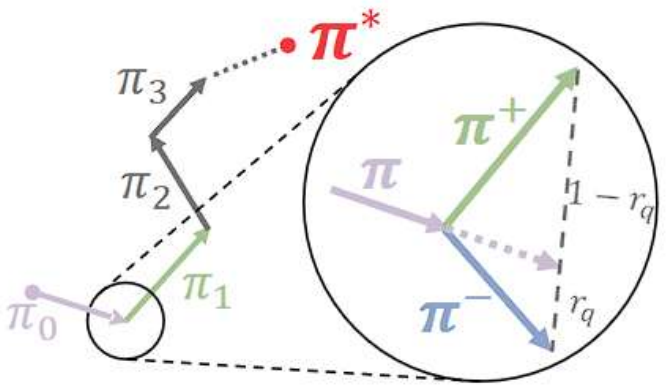
Assuming unlimited data and model capacity, the optimal solution for solving Eq. 8 is

$$\forall \boldsymbol{q}, \boldsymbol{a}: \quad \pi_\theta^+(\boldsymbol{a}|\boldsymbol{q}) = \pi^+(\boldsymbol{a}|\boldsymbol{q})$$

To further utilize positive data,

$$\mathcal{L}^{\text{NFT}}_{(\boldsymbol{a},\boldsymbol{q},r)\sim\mathcal{D}}(\theta) = r\left[-\log \frac{\pi_\theta^+(\boldsymbol{a}|\boldsymbol{q})}{\pi(\boldsymbol{a}|\boldsymbol{q})}\right] + (1-r)\left[-\log \frac{1 - r_{\boldsymbol{q}}\frac{\pi_\theta^+(\boldsymbol{a}|\boldsymbol{q})}{\pi(\boldsymbol{a}|\boldsymbol{q})}}{1 - r_{\boldsymbol{q}}}\right]$$



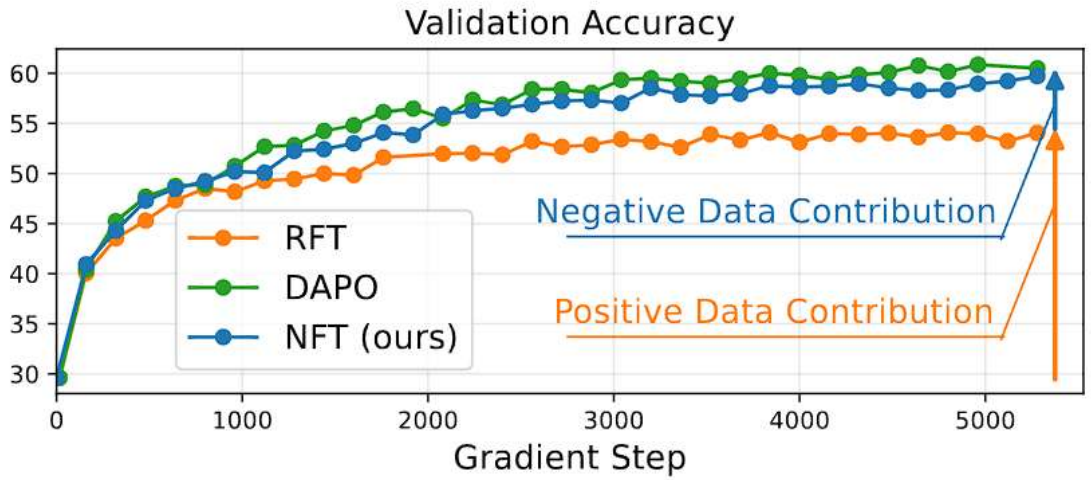


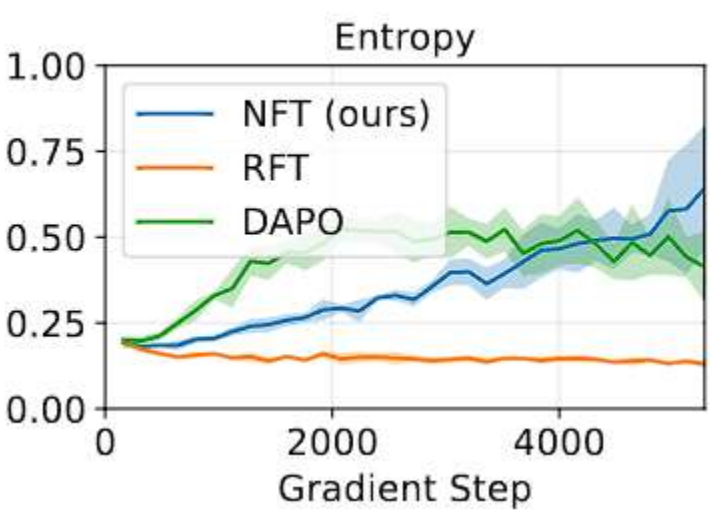
Experiments

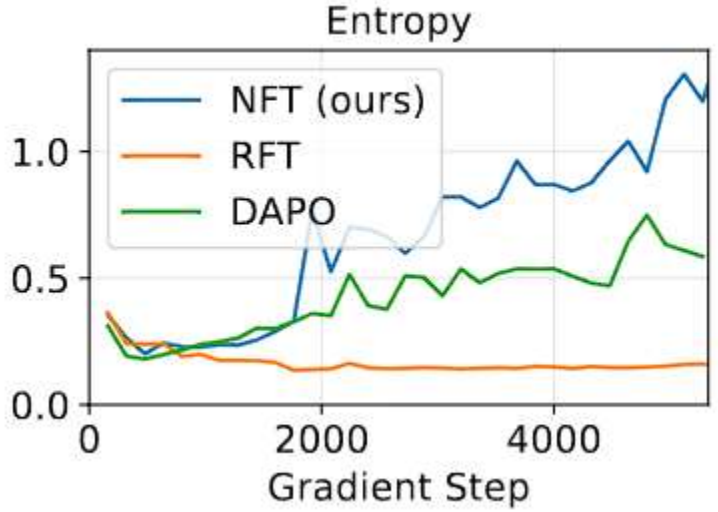
Table 1: NFT performs competitively compared with other algorithms. We report `avg@32` for AIME24, AIME25, and AMC23 and `avg@1` for others. Numbers within 1 % of the max are bolded.

Model	AIME24	MATH500	AIME25	AMC23	Olympiad	Minerva	Average
Qwen2.5-Math-7B	13.3	69.0	5.5	45.8	34.7	21.3	31.6
Preference fine-tuning							
+ DPO	29.8	79.8	13.8	83.2	48.0	39.0	48.9
Reinforcement fine-tuning							
+ GRPO	30.2	80.4	17.1	79.5	**51.8**	38.2	49.5
+ Dr. GRPO	31.8	**83.4**	15.7	80.2	49.6	38.2	49.8
+ DAPO	**33.1**	81.6	**18.7**	85.0	49.9	39.3	**51.2**
Supervised fine-tuning							
+ RFT	**33.7**	79.8	13.4	79.7	44.3	38.6	48.3
+ **NFT**	32.0	**83.2**	**18.3**	**88.5**	47.3	**40.8**	**51.7**
Qwen2.5-32B	4.1	68.6	1.0	45.0	31.1	27.9	29.6
+ DAPO	**44.1**	**89.2**	**33.4**	90.9	**54.1**	47.5	**59.9**
+ RFT	29.9	86.2	19.1	92.4	45.3	44.1	52.8
+ **NFT**	37.8	**88.4**	31.5	**93.8**	**55.0**	**48.9**	**59.2**

Experiments

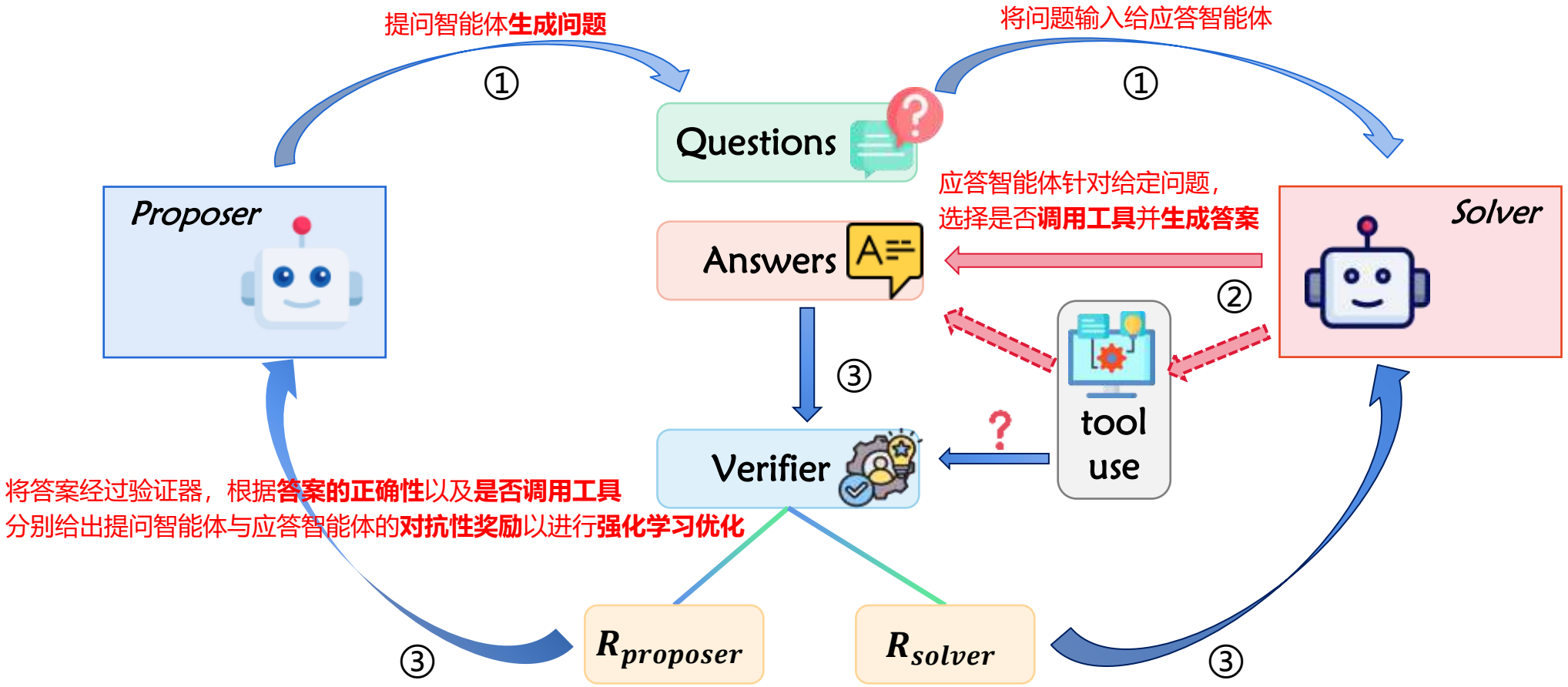






Negative feedback enhances exploration

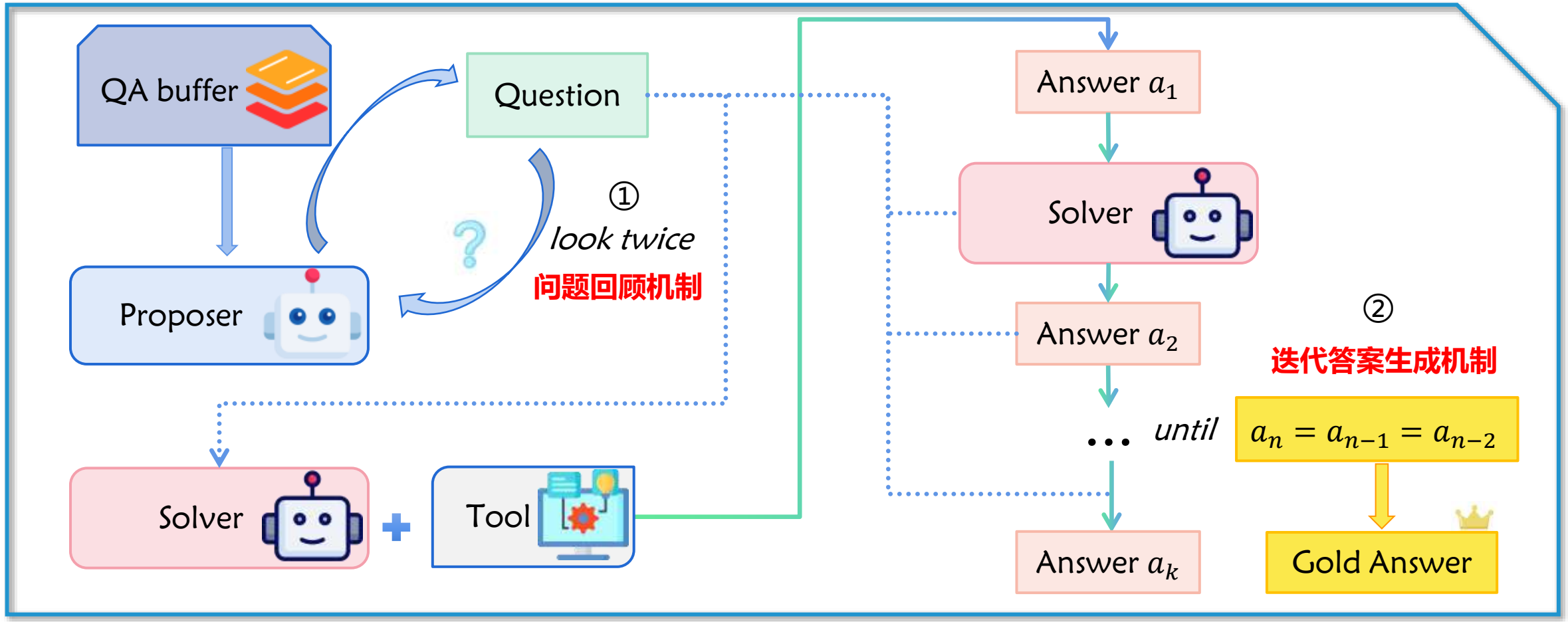
总体技术路线



✓ **该过程可形式化为一个Stackelberg博弈或迭代对抗训练框架**

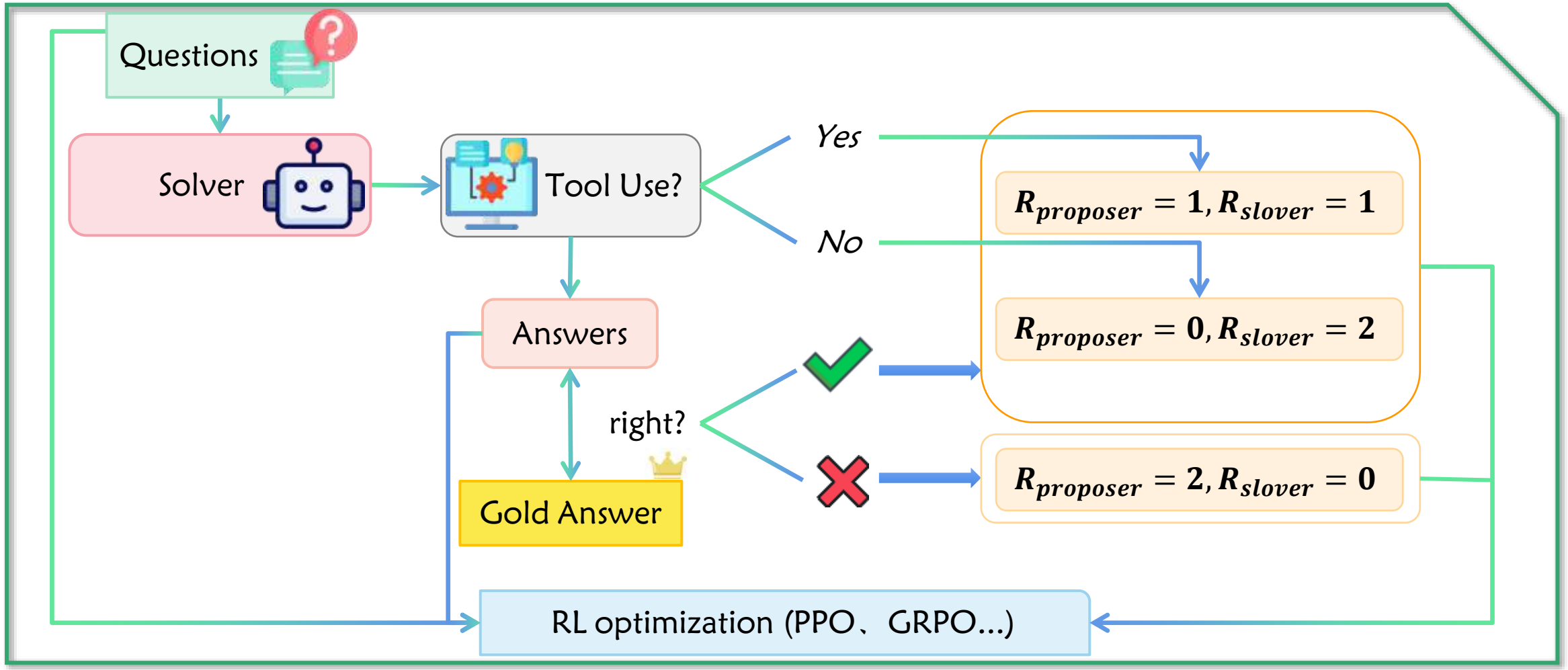
技术方案：自动化问题-答案生成

目的：自动化获得**有效的问题**以实现无人工参与的动态探测知识边界，并通过大模型对工具的反复调用获得**对应真实答案**，为后续可验证奖励提供参考，从而实现基于强化学习的模型能力优化。



技术方案：对抗性奖励设计

目的：通过基于**答案的正确性**与**是否调用工具**设计**对抗性奖励**，从而实现提问智能体与应答智能体的对抗博弈，鼓励提问智能体尽可能提出难度较高的问题，并且鼓励应答智能体尽可能通过自身知识解决问题。



Thanks