Seminar Presentation:

DPO Meets PPO: Reinforced Token Optimization for RLHF

Ravindra Mina (25AI60R02)

M.Tech 1st year
Department of AI
IIT Kharagpur

Topic Assigned By:

Prof. Prabhat Kumar Mishra

Agenda

- Introduction
- Preliminaries
- RLHF as MDP
- RTO Algorithm
- Experiments
- Ablations & Efficiency
- Conclusion & References

Introduction

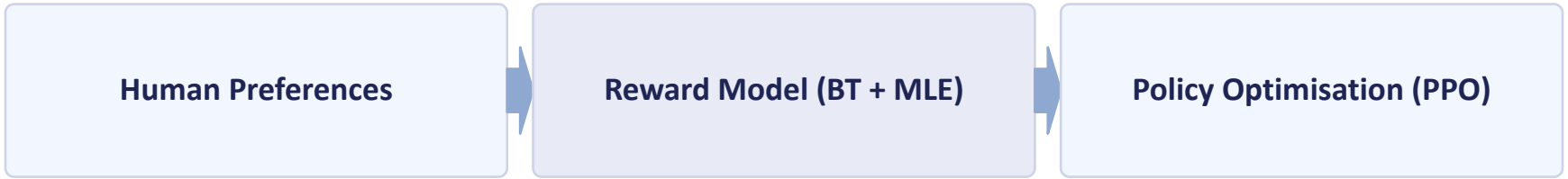
- RLHF aligns large language models with human values, enabling safer and more helpful responses.
- Classical RLHF pipeline: reward model from preference data + policy optimisation via PPO.
- PPO struggles with unstable and sample-inefficient training in open implementations.
- Mismatch: RLHF treated as bandit (sentence-level rewards) while PPO expects multi-step MDPs.

Motivation & Contributions

- Bandit formulation yields sparse sentence-level rewards and poor credit assignment.
- PPO expects dense token-level feedback but current RLHF assigns reward only to the last token.
- Need for principled token-wise formulation to improve stability and sample efficiency.

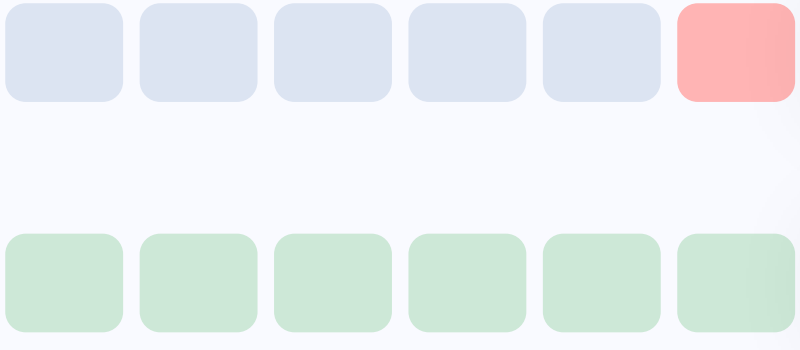
- Model RLHF as an MDP to enable token-wise reward signals.
- Propose RTO: combines DPO token-level rewards with PPO for RLHF.
- Theoretical analysis shows token-level MDPs reduce sample complexity from O(A^H) to O(A^{ξ+1}).
- Empirically, RTO outperforms PPO and other baselines by 7.5 and 4.1 points on major benchmarks.

Preliminaries: RLHF Pipeline



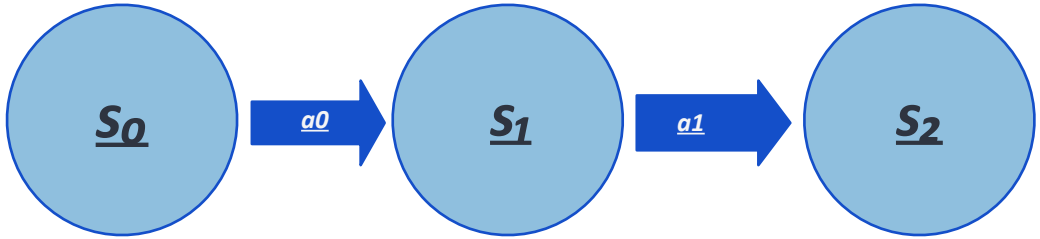
- Collect preference data: pairs of prompts and ranked responses.
- Train reward model via MLE using the Bradley–Terry (BT) model.
- Optimise policy with KL-regularised PPO to stay close to the reference model.

Reward Assignment: Sparse vs Dense



- Sparse PPO uses r_{MLE} only at final token; other tokens receive zero reward.
- Dense token-level reward improves credit assignment and learning efficiency.

RLHF as Markov Decision Process



RL: sequential decision making

- States: environment description
- Actions: agent choices
- Rewards: feedback signals
- Policy: map states to actions for maximum return

MDP: states, actions, rewards, transitions & discount factor

Preference Model & Value Function

- **Bradley–Terry preference model (token-wise):**

$$\sigma\left(\sum_{h=1}^{H} r(s_h^1, a_h^1) - \sum_{h=1}^{H} r(s_h^2, a_h^2)\right)$$

- **Token-wise rewards replace sentence-level rewards in BT model.**

Regularised Value Function:

$$V_\beta^\pi(s) = \mathbb{E}_\pi\left[\sum_{h=1}^{H}\left(r(s_h, a_h) - \beta \log \frac{\pi(a_h|s_h)}{\pi_{\text{ref}}(a_h|s_h)}\right)\right]$$

Optimal policy:

$$\pi_\beta^*(a \mid s) = \exp\left(\frac{Q_\beta^*(s,a) - V_\beta^*(s)}{\beta}\right)$$

Sample Complexity & Advantage

Sentence-level Bandit

Complexity: O(A^H)

Requires exploring all possible sequences to find optimum.

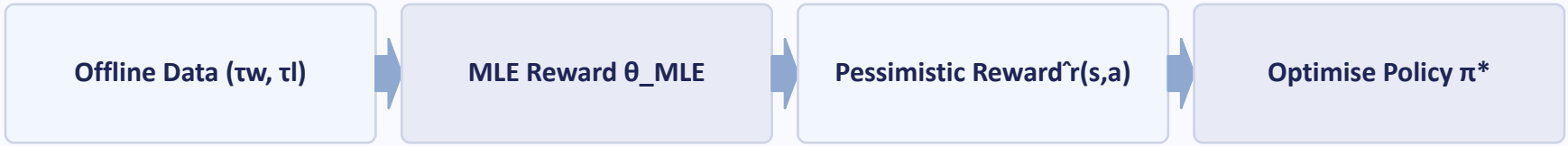
Token-level MDP

Complexity: O(A^{ξ+1})

Only a handful of promising nodes need to be explored.

Token-level MDP drastically reduces sample complexity when ξ ≪ H.

Reinforced Token Optimization



- Compute reward parameters via MLE on preference data.
- Construct pessimistic token-wise reward $\hat{r}(s,a)$ to mitigate overestimation.
- Find policy bπ that maximises the regularised value $V_{\beta}(s;\hat{r})$.

Practical RTO Implementation

Token-wise reward r_RTO(x,y_{1:h})

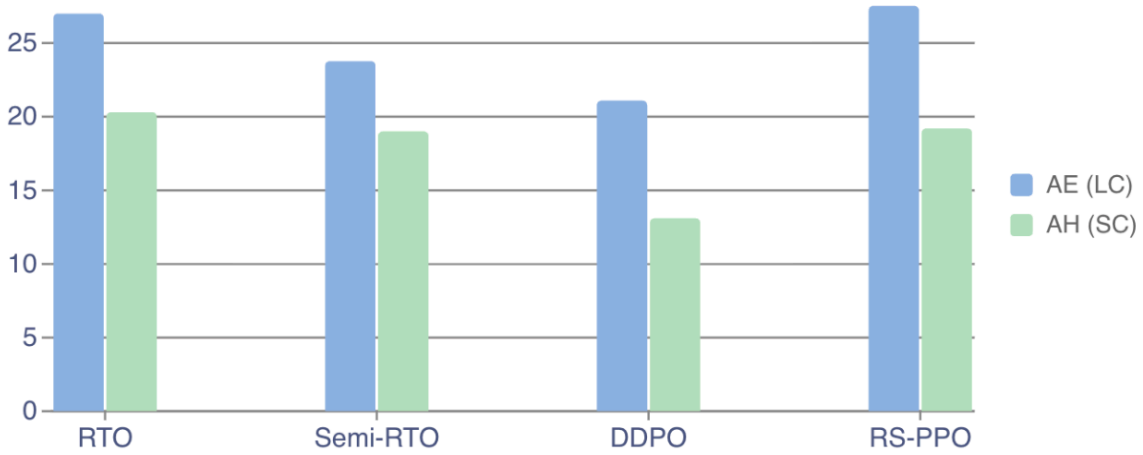
$$\begin{cases} \beta_1 \log \dfrac{\pi_{\text{dpo}}(y_h \mid x, y_{1:h-1})}{\pi_{\text{ref}}(y_h \mid x, y_{1:h-1})} - \beta_2 \log \dfrac{\pi(y_h \mid x, y_{1:h-1})}{\pi_{\text{ref}}(y_h \mid x, y_{1:h-1})}, & h < H, \\ \beta_1 \log \dfrac{\pi_{\text{dpo}}(y_h \mid x, y_{1:h-1})}{\pi_{\text{ref}}(y_h \mid x, y_{1:h-1})} - \beta_2 \log \dfrac{\pi(y_h \mid x, y_{1:h-1})}{\pi_{\text{ref}}(y_h \mid x, y_{1:h-1})} + \beta_3\, r_{\text{MLE}}(x, y_{1:H}), & h = H. \end{cases}$$

- Use DPO to approximate the optimal policy π*_β and compute implicit token-wise reward.
- Apply PPO updates on the dense reward r_RTO to train the policy.
- Hyper-parameters: set β3 = 1 (sentence reward), choose β2 as in PPO; β1 small to avoid dominating from DPO.

Benchmark Results

- **RTO achieves the highest scores on both benchmarks.**
- **Outperforms PPO by 7.5 (AE LC) and 4.1 (AH SC) points.**
- **Significant gains over direct preference methods like DPO and SimPO.**

Ablation Studies & Reward Shaping



- **Denser rewards (RTO) yield the best performance across metrics.**
- **Semi-RTO redistributes token rewards to delimiters, reducing effectiveness.**
- **Reward shaping via DPO (RS-PPO) surpasses DDPO, highlighting the role of DPO in shaping rather than direct reward.**

Sample Efficiency & Scaling

- **RTO matches PPO's performance using ~1/8 of data, then keeps improving.**
- **PPO saturates early and fails to benefit from additional data.**

Conclusion & Future Work

- RLHF as MDP: token-level formulation captures autoregressive nature of LLMs.
- RTO bridges DPO and PPO: uses implicit token reward from DPO to shape PPO training.
- Theoretical results: sample complexity O(A^{ξ+1}), sample efficiency guarantees.
- Empirical results: substantial gains over SFT, DPO, PPO and other baselines.
- Future directions: explore alternative token-wise reward learning beyond DPO and other RL algorithms for token rewards.

References

- Christiano et al. Deep RL from Human Preferences 2017
- Ziegler et al. Fine-Tuning Language Models from Human Preferences 2019
- Schulman et al. Proximal Policy Optimization (PPO) 2017
- Ouyang et al. Training Language Models to Follow Instructions 2022
- Rafailov et al. Direct Preference Optimization (DPO) 2023
- Park et al. Length-Controlled DPO (R-DPO) 2024
- Meng et al. Simplified Preference Optimization (SimPO) 2024
- Zeng et al. Token-wise DPO (TDPO) 2024
- Zhong et al. RTO meets PPO for RLHF (this work) 2025

Thank You!