Reinforcement Learning with Human Feedback for Aligning Large Language Models

Suhas Palwai and Stuart Powers

November 8, 2024

Outline

What is Reinforcement Learning (RL)?

- RL is a framework where an **agent** learns to make decisions by interacting with an **environment**.
- The goal is to maximize cumulative **rewards**.
- Applications include robotics, game playing, and natural language processing.

Key Components of RL

- **Agent**: Learns and makes decisions.
- **Environment**: The system the agent interacts with.
- **State Space** ($\mathcal{S}$): All possible states $s \in \mathcal{S}$.
- **Action Space** ($\mathcal{A}$): All possible actions $a \in \mathcal{A}$.
- **Transition Dynamics** ($P(s'|s, a)$)
  - Inputs: Current state s, action a, next state s'.
  - Output: Probability of transitioning to s'.
- **Reward Function** ($R(s, a)$)
  - Inputs: State s, action a.
  - Output: Reward $r = R(s, a)$.

Policy and Objective

- **Policy** $(\pi_\theta(a|s))$
  - Maps states to action probabilities.
  - Defines the agent's behavior.
- **Trajectory** $\tau = (s_0, a_0, s_1, a_1, \ldots, s_T)$
- **Probability of Trajectory**

$$P(\tau|\pi_\theta) = \rho_0(s_0) \prod_{t=0}^{T-1} \pi_\theta(a_t|s_t) P(s_{t+1}|s_t, a_t)$$

- **Return along Trajectory**

$$R(\tau) = \sum_{t=0}^{T-1} \gamma^t R(s_t, a_t)$$

- **Objective Function**

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

Central Optimization Problem

$$\pi_\theta^* = \arg\max_{\pi_\theta} J(\pi_\theta)$$

- Find the optimal policy π_θ^* that maximizes expected return.

State-Value Function

- **Definition**:
$$V^{\pi_\theta}(s) = \mathbb{E}_{\tau \sim P(\cdot|\pi_\theta)} \left[R(\tau) \middle| s_0 = s \right]$$
- Expected return starting from state s and following policy π_θ.

Action-Value Function

- **Definition**:

$$Q^{\pi_\theta}(s, a) = \mathbb{E}_{\tau \sim P(\cdot|\pi_\theta)} \left[R(\tau) \middle| s_0 = s, a_0 = a \right]$$

- Expected return starting from state s, taking action a, then following π_θ.

Relationship between $V^{\pi_\theta}(s)$ and $Q^{\pi_\theta}(s, a)$

$$V^{\pi_\theta}(s) = \sum_{a \in \mathcal{A}} \pi_\theta(a|s) Q^{\pi_\theta}(s, a)$$

- The value of a state is the expected value of action-values under policy π_θ.

What is RLHF?

- Incorporates human feedback into the RL framework.
- Useful for tasks where rewards are hard to define.
- Aligns AI behavior with human values and preferences.

Policy Gradient Methods

- Directly adjust policy parameters θ to maximize expected return $J(\pi_\theta)$.
- **Policy Gradient Theorem**:

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{\pi_\theta}\left[\nabla_\theta \log \pi_\theta(a_t|s_t) G_t\right]$$

- High variance and instability can be issues.

Derivation of the Policy Gradient Theorem

- **Objective Function**:

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

- **Gradient of the Objective**:

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{\tau \sim P(\cdot|\pi_\theta)} \left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t|s_t) G_t \right]$$

where $G_t = \sum_{k=0}^{T-t-1} \gamma^k R(s_{t+k}, a_{t+k})$

Challenges with Policy Gradient Methods

- **High Variance**: Stochastic sampling leads to high variance in gradient estimates, making learning unstable.
- **Instability**: Large updates to policy parameters can cause drastic policy changes, potentially degrading performance.

Proximal Policy Optimization (PPO)

- Addresses instability in policy gradients.
- Introduces a clipped surrogate objective.
- Limits the magnitude of policy updates.

PPO Objective Function

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

- $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$
- $\hat{A}_t$ is the estimated advantage.
- ϵ controls the clipping range (e.g., $\epsilon = 0.2$).

Derivation of PPO Objective Function

- **Surrogate Objective**:

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

- **Clipping Mechanism**:

$$\text{clip}\left(r_t(\theta), 1-\epsilon, 1+\epsilon\right) = \begin{cases} 1-\epsilon & \text{if } r_t(\theta) < 1-\epsilon \\ r_t(\theta) & \text{if } 1-\epsilon \le r_t(\theta) \le 1+\epsilon \\ 1+\epsilon & \text{if } r_t(\theta) > 1+\epsilon \end{cases}$$

- **Purpose**:
  - Prevents $r_t(\theta)$ from deviating too much from 1.
  - Ensures updates are within a "trust region."

Intuition Behind PPO

- Prevents large policy updates that could destabilize training.
- Maintains policy within a "trust region."
- Balances exploration and exploitation.

What is DPO?

- Directly optimizes policy based on human preferences.
- Uses pairwise comparisons instead of scalar rewards.
- Effective for subjective tasks.

DPO Objective Function

$$L^{\text{DPO}}(\theta) = \sum_i \log \sigma \left(f_\theta(x_i, y_i^+) - f_\theta(x_i, y_i^-) \right)$$

- $f_\theta(x, y) = \log \pi_\theta(y|x)$
- $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function.
- Maximizes likelihood of preferred responses.

Derivation of DPO Objective Function

- **Probability of Preference**:

$$P_\theta(y_i^+ \succ y_i^- | x_i) = \frac{\exp(f_\theta(x_i, y_i^+))}{\exp(f_\theta(x_i, y_i^+)) + \exp(f_\theta(x_i, y_i^-))}$$

- **Log-Likelihood**:

$$L^{\text{DPO}}(\theta) = \sum_i \log \left(\frac{\exp(f_\theta(x_i, y_i^+))}{\exp(f_\theta(x_i, y_i^+)) + \exp(f_\theta(x_i, y_i^-))} \right)$$

- **Simplification**:

$$L^{\text{DPO}}(\theta) = \sum_i \left[f_\theta(x_i, y_i^+) - \log \left(\exp(f_\theta(x_i, y_i^+)) + \exp(f_\theta(x_i, y_i^-)) \right) \right]$$

- **Using Sigmoid Function**:

$$L^{\text{DPO}}(\theta) = \sum_i \log \sigma \left(f_\theta(x_i, y_i^+) - f_\theta(x_i, y_i^-) \right)$$

Intuition Behind DPO

- Directly aligns model outputs with human preferences.
- Simplifies training without explicit reward functions.
- Captures nuanced human judgments.

Key Takeaways

- RL is about maximizing expected cumulative rewards.
- RLHF incorporates human feedback for alignment.
- PPO provides stable policy updates.
- DPO directly optimizes for human preferences.
- Both methods enhance LLM alignment with human values.

Questions

Thank you!
Questions?