Contrastive Preference Learning: Learning From Human Feedback Without RL

Goal: Reinforcement Learning from human feedback (RLHF) without traditional reinforcement learning techniques.

1. **Motivation and problem with traditional RLHF**

* Traditional RLHF methods rely on two phases: learning a reward function from human feedback and then optimizing it using reinforcement learning (RL). However, this two-phase approach assumes that human preferences are based on rewards, which may not be accurate. Instead, human preferences are better modeled by regret or advantage, which measures how far a behavior deviates from the optimal policy.
* Existing RL methods, like policy gradients and dynamic programming, face optimization challenges, especially in high-dimensional problems, limiting their application.

1. **Contrastive Preference Learning (CPL)**

* CPL introduces a new approach that avoids the need for RL by directly learning policies from human feedback using the regret-based preference model. This model assumes that human preferences are based on the regret of actions compared to the optimal policy.
* CPL is derived using the principle of maximum entropy, which leads to a simple contrastive learning objective. This objective uses preferences to directly optimize the policy, bypassing the need to learn a reward function.

1. **Key Formula**

The regret-based preference model is formulated as:

Where is the optimal advantage function, is the discount factor, and are behavior segments, and the human prefers + over -.

The CPL objective for optimizing the policy is:

1. **Benefits of CPL**

* **Fully off-policy**: CPL can work with any offline data without requiring online interactions.
* **Scalability**: Since it uses supervised learning objectives, CPL scales better to high-dimensional problems like those involving image inputs.
* **Simplified Optimization**: CPL avoids RL-specific optimization challenges, making it simpler and more efficient.

1. **Experiments**

The paper demonstrates CPL's effectiveness on tasks involving high-dimensional data and sequential decision-making. It shows that CPL can match or surpass the performance of traditional RL-based methods while being faster and more efficient.

Direct Preference Optimization: You Language Model is Secretly a Reward Model

1. **Motivation and Problems with Traditional RLHF**

* Traditional Reinforcement Learning from Human Feedback (RLHF) methods typically involve training a reward model based on human feedback and optimizing a language model (LM) to maximize this reward using reinforcement learning (RL).
* This process is complex, computationally expensive, and can result in unstable training due to the need for fine-tuning, hyperparameter tuning, and maintaining a balance between the LM's reward and reference policy.

1. **Direct Preference Optimization (DPO)**

* DPO offers a new approach that directly optimizes a language model to align with human preferences, bypassing the need for explicit reward modeling and reinforcement learning.
* Instead of relying on RL, DPO uses a simple classification objective based on human preferences to adjust the LM’s outputs. It directly optimizes the policy using human preference data by reparametrizing the reward model into a preference model.

1. **Key Fomula**

The core DPO objective is based on maximizing the likelihood of human preferences between pairs of model outputs:

Where is the model being optimized, is the reference policy (pre-trained model), and are human-preferred and dis-preferred outputs, controls the strength of regularization.

1. **Key features**

* DPO uses human preferences over model completions to fine-tune the model directly, optimizing the likelihood that the preferred output is chosen over the dispreferred one.
* DPO is highly stable, simple to implement, and avoids the complexities of RL (e.g., no need for sampling from the LM during training or performing complex reward function updates).

1. **Advantages of DPO**

* **Stability**: DPO avoids many instability issues inherent in RL-based methods.
* **Computational Efficiency**: It requires less computational overhead because it does not need to maintain and update reward models.
* **Better Performance**: DPO matches or exceeds the performance of RLHF-based methods, such as Proximal Policy Optimization (PPO), in tasks like sentiment control, summarization, and dialogue generation.

1. **Experiments**

* Sentiment Generation: It was tested on IMDb reviews to generate positive sentiment, where it outperformed PPO-based RLHF in optimizing reward while keeping KL-divergence to the reference policy low.
* Summarization: DPO achieved better performance in summarization tasks compared to RLHF-based models, as measured by win rates using GPT-4 as an evaluator.
* Dialogue Generation: It performed similarly or better than PPO and baseline methods in producing high-quality responses in single-turn dialogue tasks.

Illustrating Reinforcement Learning from Human Feedback (RLHF)

1. **Motivation**

* Language models (LMs) are increasingly capable of generating text based on human input, but what constitutes "good" text is often subjective and context-dependent.
* Traditional training methods use metrics like BLEU and ROUGE to evaluate language models. However, these metrics fall short in capturing human preferences, particularly in tasks requiring creativity, truthfulness, or executability, such as writing code or stories.
* Reinforcement Learning from Human Feedback (RLHF) aims to align models with complex human values by directly optimizing language models using feedback from human evaluators, as seen in models like ChatGPT.

1. **RLHF Process**

Pretraining the Language Model (LM):

* RLHF starts with a pretrained LM, such as GPT-3 or other large models like Anthropic’s Gopher. These LMs are trained on diverse corpora to generate general text.
* This pretrained model serves as the foundation, with fine-tuning to ensure responsiveness to varied instructions.

Training the Reward Model (RM):

* The RM, also called the preference model, learns to score model outputs based on human preferences.
* Humans evaluate generated text by ranking it rather than assigning scalar scores directly. Ranking helps mitigate noisy and uncalibrated human judgments.
* Example: Human annotators compare outputs from two models on the same prompt, and an Elo rating system assigns relative rewards to the outputs.

Fine-tuning the LM with Reinforcement Learning:

* Proximal Policy Optimization (PPO), an RL algorithm, is used to fine-tune the LM. The goal is to maximize the reward model’s score while constraining the policy shift from the original pretrained model.
* The KL divergence penalty helps prevent the model from deviating too much from the pretrained model to maintain coherence and avoid reward hacking (where the model exploits the reward function without improving text quality).

1. **Key Formula**

The reward function in RLHF can be described as:

Where is the reward from the human preference model, is the penalty term based on the Kullback-Leibler divergence between the current model’s output and pretrained model’s outout and is a scaling parameter that controls the influence of the KL penalty.

1. **Training Dynamics**

* During the RL phase, the LM generates text from a given prompt, and the text is evaluated by the reward model. The output is adjusted through the PPO algorithm, updating the LM’s parameters to maximize the reward while minimizing excessive deviation from the original model.
* Fine-tuning such large models is resource-intensive, and to reduce costs, techniques like Low-Rank Adaptation (LoRA) are used to freeze some model parameters during training.
* The balance between how much of the model to fine-tune and how much to freeze remains an open research question.

RLHF lit. review #1 and missing pieces in RLHF

1. **Motivation**

* This paper is the first in a series exploring the literature on Reinforcement Learning from Human Feedback (RLHF), focusing on gaps in research compared to industry practices.
* RLHF has become central across the AI industry, with companies like Anthropic integrating it into their models through approaches like Constitutional AI. However, RLHF is often built on unstable foundations, especially the reinforcement learning (RL) component, which currently uses less data and computational resources than the rest of the generative AI pipeline.

1. **Big-Picture Challenges in RLHF**

The field of RLHF faces two possible future directions:

* RLHF could be integrated into the pretraining process, potentially without traditional RL algorithms.
* A new form of RL could emerge, focusing on fine-tuning models based on implicit user feedback for continuous learning.

Key challenges for RL in RLHF include:

* **Exploration**: RLHF doesn't explicitly incentivize exploring new capabilities or learning from new data. The process lacks multi-action trajectories, which limits RL’s potential to assign long-term credit to behaviors.
* **Computation**: RL methods are seen as inefficient, not due to poor information use, but because they aren't designed to handle large-scale data in the way modern machine learning does. RL algorithms like **Proximal Policy Optimization (PPO)** discard a lot of valuable data after each batch.

1. **Research fields in RLHF**

Direct Preference Optimization (DPO): This method avoids traditional RL, instead optimizing models directly from human preferences. DPO is seen as a potential future for integrating RL-like ideas without full RL algorithms.

New Optimizers:

* Pairwise Proximal Policy Optimization (P3O): A new technique that compares the rewards of two texts directly, improving efficiency over traditional PPO.
* Advantage Model and Selective Rehearsal: A method that regularizes reward models more effectively, improving PPO's data sourcing.

1. **Model Challenges**

* **Overoptimization**: There is concern about reward models being over-optimized, leading to models gaming the reward rather than improving their capabilities. Research is focusing on methods like reward model ensembles to mitigate this.
* **Inconsistency**: Reward models can assign inconsistent scores to similar outputs with different word orders, a major issue that researchers are beginning to address through benchmarking.

Self-Play Preference Optimization for Language Model Alignment

1. **Motivation**

* Large language models (LLMs) have achieved impressive results in natural language processing tasks, but ensuring these models align with human preferences is challenging. The paper introduces **Self-Play Preference Optimization (SPPO),** a novel approach to fine-tuning language models through self-play. SPPO treats the problem of language model alignment as a **constant-sum two-player game** where each policy seeks to maximize its probability of being preferred over its competitor.
* Traditional RLHF methods rely on parametric models like the **Bradley-Terry (BT) model**, but these fall short of capturing the complexities of human preferences. Instead, SPPO approximates the Nash equilibrium through iterative policy updates, allowing it to more effectively align models with human preferences.

1. **SPPO Algorithm**

**SPPO** works by treating LLM alignment as a two-player game where the objective is to identify the Nash equilibrium policy, which consistently generates responses that are preferred over others. This is achieved through an iterative **multiplicative weight update** mechanism.

The SPPO update rule is:

Where is the current policy, is the probability that response is preferred over responses generated by the current policy , is the learning rate.

1. **Key Contributions**

* SPPO differs from methods like **Direct Preference Optimization (DPO)** and Identity **Preference Optimization (IPO)**, which focus on optimizing based on pairwise preference data. SPPO handles multi-round optimization and focuses on improving the log-likelihood of preferred responses while decreasing that of rejected responses.
* SPPO achieves superior performance over DPO and IPO because it uses a more efficient, self-play-based mechanism to identify Nash equilibrium policies, even when preference data is limited or noisy.

1. **Key Results**

The paper demonstrates SPPO’s performance using the **Mistral-7B-Instruct-v0.2** and **Llama-3-8B-Instruct** models, with SPPO significantly outperforming DPO and IPO across several benchmarks:

* **AlpacaEval 2.0**: SPPO achieves a state-of-the-art length-controlled win rate of 28.53% compared to 26.39% for DPO.
* **MT-Bench**: SPPO Iteration 3 consistently improves over earlier iterations, achieving the highest average score of 7.59.
* **Open LLM Leaderboard**: SPPO models demonstrate superior general performance across various LLM tasks, with improved results in areas like reasoning, role-playing, and coding.

1. **Optimization Model:**

The SPPO algorithm generates multiple responses for a given prompt, evaluates them using a preference model (such as PairRM), and updates the policy based on the estimated win rates of the responses.

The optimization problem is:

1. **Comparison with other models**

* SPPO is compared to other preference optimization techniques like DPO, IPO, and **Kahneman-Tversky Optimization (KTO).** While DPO and IPO focus on optimizing pairwise preferences, SPPO extends this by addressing the likelihood gap between winning and losing responses, ensuring better convergence to the Nash equilibrium.
* SPPO outperforms these methods in both preference alignment and generalist abilities across different tasks, such as reasoning, math, and dialogue generation.

Statistical Rejection Sampling Improves Preference Optimization

1. **Motivation**

* The paper addresses a key challenge in improving language model alignment with human preferences. Traditional methods, such as Reinforcement Learning from Human Feedback (RLHF), introduce significant complexity, including unstable training and the need for multiple models (reward model, policy model, etc.).
* New methods like Sequence Likelihood Calibration (SLiC) and Direct Preference Optimization (DPO) have emerged as alternatives that simplify the process by directly learning from preference data without reinforcement learning. However, both have limitations in generating preference pairs from the optimal policy. The absence of a reward model in DPO and limited sampling capabilities in SLiC make these approaches less effective.
* To overcome these limitations, the paper proposes a novel method called Statistical Rejection Sampling Optimization (RSO). RSO generates preference data more accurately from the estimated optimal policy using rejection sampling, enabling better optimization.

1. **Key Contributions**

* The paper introduces RSO, which leverages statistical rejection sampling to construct preference pairs from the optimal policy, using a pairwise reward-ranking model.
* The paper also provides a unified framework that enhances both SLiC and DPO by applying a preference modeling perspective to improve their loss functions.
* RSO consistently outperforms both SLiC and DPO in various experiments across tasks, as evaluated by both large language models (LLMs) and human raters.