Failure Modes of Maximum Entropy RLHF

Anonymous ACL submission

Abstract

In this paper, we show that Simple Preference Optimization (SimPO) can be derived as Maximum Entropy Reinforcement Learning, providing a theoretical foundation for this reference-free method. Motivated by SimPO's strong performance in offline preference optimization, we investigate whether Maximum Entropy RL can achieve similar results in online RLHF settings. Our experiments find that Maximum Entropy RL consistently exhibits overoptimization and unstable KL dynamics, even at very low learning rates. Unlike KL-constrained methods that maintain stable training, entropy regularization fails to prevent reward hacking and appears to correlate with overoptimization. Lastly, we discuss possible explanations for why SimPO succeeds in offline settings while Maximum Entropy RL struggles in online scenarios. Our findings suggest that reference-free approaches may face distinct challenges when applied to online or offline preference learning.

1 Introduction

Aligning AI systems with human values is widely recognized as a central challenge in modern AI (Bengio et al., 2025; Russell, 2022). The dominant paradigm, Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2023; Stiennon et al., 2022; Ziegler et al., 2020; Bai et al., 2022; Ouyang et al., 2022), typically follows a three-stage pipeline: supervised fine-tuning, reward model training from preference data, and policy optimization with reinforcement learning under KL divergence regularization to constrain deviation from a reference model. While effective, this pipeline is computationally expensive and operationally complex, requiring separate reward models, substantial human annotation, and careful hyperparameter tuning to maintain stability.

These challenges have motivated direct alignment algorithms (DAAs) (Rafailov et al., 2024a), which bypass explicit reward modeling and online RL. Direct Preference Optimization (DPO) (Rafailov et al., 2024c) derives an analytical solution to a KL-regularized RL objective, expressing the reward implicitly as a function of the optimal policy and reducing preference learning to supervised optimization. More recently, Simple Preference Optimization (SimPO) (Meng et al., 2024) has demonstrated strong empirical performance while eliminating the reference model entirely, instead using length-normalized log-likelihoods and a fixed margin between preferred and dispreferred responses.

Despite its empirical success, SimPO has lacked a principled theoretical foundation comparable to that of reference-based methods such as DPO. In this work, we establish a connection between SimPO and Maximum Entropy Reinforcement Learning (Ziebart et al., 2008). We show that SimPO can be interpreted as a closed-form solution to a Maximum Entropy RL objective, providing a theoretical grounding analogous to DPO's relationship with KL-constrained RL and suggesting that reference-free optimization can naturally arise from entropy regularization under appropriate conditions.

This perspective raises an important empirical question: if SimPO corresponds to an offline Maximum Entropy solution, can online Maximum Entropy RL serve as a viable alternative to KL-constrained methods in RLHF? To investigate this, we compare Maximum Entropy RL and KL-constrained RL on the TL;DR summarization benchmark (Stiennon et al., 2022) using models from the Pythia suite (Biderman et al., 2023).

Our results reveal a clear asymmetry. While SimPO performs reliably in offline preference optimization, online Maximum Entropy RL frequently exhibits instability and overoptimization, even at conservative learning rates. We observe that increases in entropy often correlate with these insta-

bilities, indicating that entropy regularization alone does not reliably prevent reward hacking and may, in some cases, exacerbate it. We hypothesize that SimPO benefits from implicit stabilizing mechanisms—such as dataset constraints and target margins—that partially substitute for the regularization provided by a reference model, whereas these protections are absent in online Maximum Entropy RL.

Our contributions are threefold. First, we provide a theoretical interpretation of SimPO as a Maximum Entropy Reinforcement Learning solution, situating it within established RL frameworks. Second, we empirically demonstrate that directly applying Maximum Entropy RL in online RLHF settings can lead to instability and overoptimization, highlighting limitations of entropy regularization in isolation. Third, we offer insight into why SimPO succeeds offline despite these challenges, emphasizing the role of implicit regularization through data constraints and margin-based objectives. Together, these findings clarify the relationship between entropy-based methods and preference optimization, and point to the need for additional regularization mechanisms for robust reference-free alignment in online settings.

2 Background

In this section, we review the relevant background topics, while additional related work is provided in Appendix A.

2.1 Canonical RLHF

We reiterate the standart RLHF pipeline as outlined in (Ziegler et al., 2020) and subsequent works (Stiennon et al., 2022; Bai et al., 2022; Ouyang et al., 2022). It consists of three main stages: (1) Supervised Fine-Tuning (SFT), (2) Reward Modeling, and (3) RL Optimization.

SFT: A pre-trained LM is fine-tuned on task-specific high-quality data via supervised learning to obtain the initial policy π^{SFT}.

Reward Modeling: Prompts x are sampled, and π^{SFT} generates answer pairs (y_1, y_2). Human annotators indicate preferences $y_w \succ y_l \mid x$, assumed to reflect a latent reward function $r(x, y)$. A common approach is to model preferences with the Bradley-Terry (BT) model (Bradley and Terry, 1952):

$$p(y_1 \succ y_2 \mid x) = \frac{e^{r^*(x,y_1)}}{e^{r^*(x,y_1)} + e^{r^*(x,y_2)}} \quad (1)$$

Given a dataset $\mathcal{D} = x^{(i)}, y_w^{(i)}, y_l^{(i)}$, we learn a reward model r_ϕ by minimizing the binary classification loss:

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

where σ is the sigmoid function. In practice, r_ϕ is initialized from π^{SFT} with a linear head, and reward outputs are normalized for stability.

RL Fine-Tuning: Finally, the policy π_θ is optimized using the learned reward, constrained by a KL term to stay close to the reference policy $\pi_{\text{ref}} = \pi^{\text{SFT}}$:

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

This prevents overoptimization and distributional shift. In practice, this objective is optimized with PPO (Schulman et al., 2017), using a reward defined as $r(x, y) = r_\phi(x, y) - \beta(\log(\pi_\theta(y \mid x)) - \log(\pi_{\text{ref}}(y \mid x)))$.

2.2 Direct Preference Optimization

Direct Preference Optimization (DPO) (Rafailov et al., 2024c) has become a popular method for preference-based tuning. Unlike traditional approaches that train a separate reward model, DPO defines the reward directly in terms of the optimized policy:

$$r(x, y) = \beta \log \frac{\pi_\theta(y \mid x)}{\pi_{\text{ref}}(y \mid x)} + \beta \log Z(x),$$

Here, π_θ is the current policy, π_{ref} is a reference (often the SFT model), and $Z(x)$ is a normalization term. DPO incorporates this reward into the Bradley-Terry (Bradley and Terry, 1952) framework, where preference probabilities are given by:$p(y_w \succ y_l \mid x) = \sigma\left(r(x, y_w) - r(x, y_l)\right)$. This leads to the following objective, computed over preference triplets (x, y_w, y_l):

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \beta \log \frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right)\right]$$

By modeling preferences directly through policy ratios, DPO removes the need for an explicit reward model while remaining grounded in a probabilistic preference framework.

2.3 Simple Preference Optimization

Simple Preference Optimization (SimPO) (Meng et al., 2024) is a reference-free method for preference-based fine-tuning that aligns the reward used in training with the likelihood used at inference. Unlike DPO, SimPO eliminates the need for a reference policy by defining the reward as the length-normalized log-likelihood of the model output:

$$r_{\text{SimPO}}(x, y) = \frac{\beta}{|y|} \log \pi_\theta(y \mid x) = \frac{\beta}{|y|} \sum_{i=1}^{|y|} \log \pi_\theta(y_i \mid x, y_{<i})$$

This formulation ensures that the reward ranking $r(x, y_w) > r(x, y_l)$ aligns with the generation-time likelihood ranking $p_\theta(y_w \mid x) > p_\theta(y_l \mid x)$, which is often violated in DPO. SimPO also introduces a target margin $\gamma > 0$ into the Bradley-Terry model to encourage separation between preferred and dispreferred responses:

$$p(y_w \succ y_l \mid x) = \sigma\left(r(x, y_w) - r(x, y_l) - \gamma\right)$$

This leads to the SimPO training objective:

$$\mathcal{L}_{\text{SimPO}}(\pi_\theta) = -\mathbb{E}\left[\log \sigma\left(\frac{\beta}{|y_w|} \log \pi_\theta(y_w \mid x) - \frac{\beta}{|y_l|} \log \pi_\theta(y_l \mid x) - \gamma\right)\right]$$

3 SimPO is the Maximum Entropy RL

SimPO is a widely used preference alignment method, appreciated for its strong empirical performance and simplicity due to its reference-free objective. However, it lacks a theoretical foundation, unlike reference-based approaches such as DPO, which is derived from a KL-constrained RL objective. Recent work (Liu et al., 2024) made the important observation that posterior probability rewards correspond to Maximum Entropy RL in their analysis of reference policies. Building on this insight, we establish the connection between this Maximum Entropy RL formulation and SimPO, showing that SimPO can be understood as Maximum Entropy RL with the addition of length normalization and target margins from preference learning.

3.1 Maximum Entropy RL

Maximum Entropy Reinforcement Learning (MaxEnt RL) augments the standard RL objective with an entropy term, encouraging policies that align with the soft value function (Ziebart et al., 2008; Toussaint, 2009; Rawlik et al., 2013; Fox et al., 2015; O'Donoghue et al., 2016; Abdolmaleki et al., 2018; Haarnoja et al., 2018; Mazoure et al., 2020; Han and Sung, 2021; Zhang et al., 2025). It is deeply connected to probabilistic inference (Toussaint, 2009; Rawlik et al., 2013; Levine, 2018) and supported by both stochastic inference (Ziebart, 2010; Eysenbach and Levine, 2021) and game-theoretic foundations (Grünwald and Dawid, 2004; Ziebart et al., 2010; Han and Sung, 2021; Kim and Sung, 2023). MaxEnt is often favored for promoting exploration (Haarnoja et al., 2018; Hazan et al., 2019), smoothing optimization (Ahmed et al., 2019), and enabling robust decision-making (Eysenbach and Levine, 2021).

The general form of the Maximum Entropy Reinforcement Learning (MaxEnt RL) objective can be written as

$$\pi^\star = \arg\max_\pi \; \mathbb{E}_{\tau \sim p^\pi(\tau)} \left[\sum_{t=1}^{T} r(\mathbf{s_t}, \mathbf{a_t}) + \alpha \, \mathcal{H}_\pi[\mathbf{a_t} \mid \mathbf{s_t}]\right]$$

where $\tau = (\mathbf{s_1}, \mathbf{a_1}, \mathbf{s_2}, \mathbf{a_2}, \ldots, \mathbf{s_T}, \mathbf{a_T})$ is a trajectory sampled under policy π, and $p^\pi(\tau) = p_1(\mathbf{s_1}) \prod_{t=1}^{T} \pi(\mathbf{a_t} \mid \mathbf{s_t}) \, p(\mathbf{s_{t+1}} \mid \mathbf{s_t}, \mathbf{a_t})$ denotes the trajectory distribution induced by π. The term $\mathcal{H}_\pi[\mathbf{a_t} \mid \mathbf{s_t}] = -\int \pi(\mathbf{a_t} \mid \mathbf{s_t}) \log \pi(\mathbf{a_t} \mid \mathbf{s_t}) \, d\mathbf{a_t}$ represents the conditional entropy of the policy at each time step, and the temperature coefficient α controls the trade-off between reward maximization and policy stochasticity.

3.2 SimPO from Maximum Entropy RL

RLHF is commonly modeled as a contextual bandit problem, though some approaches treat it as a token-level MDP (Rafailov et al., 2024b; Xie et al., 2024). In this work, we adopt the contextual bandit view (Elwood et al., 2023), under which the maximum entropy formulation aligns with KL-constrained objectives. The resulting objective is given as follows.

$$\max_\pi \mathbb{E}_{x \sim D, y \sim \pi}[r(x, y)] + \alpha D_{\mathcal{H}}[\pi(y|x)] \quad (2)$$

It is straightforward to show that optimal policy of the equation 2 (proof in Appendix C) is as follows:

$$\pi_r(y|x) = \frac{1}{Z(x)} \exp\left(\frac{1}{\alpha} r(x,y)\right) \tag{3}$$

Following the analytical approach used in DPO's derivation, we can rearrange this optimal policy equation to express the reward function in terms of the policy:

$$r(x,y) = \alpha \log \pi_r(y|x) + \alpha \log Z(x) \tag{4}$$

Now, applying this reparameterization to the Bradley-Terry preference model. For the ground-truth reward r^* and corresponding optimal policy π^*, the preference probability becomes:

$$p^*(y_1 \succ y_2|x) = \sigma(r^*(x,y_1) - r^*(x,y_2)) \tag{5}$$

Substituting our reparameterization:

$$\begin{aligned} &p^*(y_1 \succ y_2|x) \\ &\quad = \sigma\big(\alpha \log \pi^*(y_1|x) + \alpha \log Z(x) \\ &\qquad - \alpha \log \pi^*(y_2|x) - \alpha \log Z(x)\big) \\ &\quad = \sigma\left(\alpha \log \pi^*(y_1|x) - \alpha \log \pi^*(y_2|x)\right) \end{aligned}$$

Crucially, the partition function $Z(x)$ cancels out, eliminating the need to compute it explicitly. This establishes a direct connection between the Bradley–Terry preference model and the optimal policy induced by Maximum Entropy RL. To recover the SimPO training objective used in practice, we introduce two additional components that are motivated by empirical and structural considerations in preference learning rather than by the Maximum Entropy RL derivation itself.

First, we apply length normalization by scaling the temperature with the response length. This is necessary in SimPO because, unlike KL-constrained methods such as DPO where the reference policy induces implicit regularization through cancellation of log-probabilities, SimPO optimizes raw log-likelihoods and is therefore prone to severe length exploitation without explicit normalization. Second, inspired by ψPO (Azar et al., 2023), we incorporate a target reward margin $\gamma > 0$ as a fixed substitute for the adaptive margin implicitly provided by the reference policy in DPO. These additions shape the optimization landscape and improve empirical stability, but do not alter the underlying connection between SimPO and Maximum Entropy RL established above. This leads to the SimPO objective for a parametric policy π_θ:

$$\begin{aligned} \mathcal{L}_{\text{SimPO}}(\pi_\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Bigg[\log \sigma\Bigg(\\ \frac{\alpha}{|y_w|} \log \pi_\theta(y_w|x) \\ - \frac{\alpha}{|y_l|} \log \pi_\theta(y_l|x) - \gamma\Bigg)\Bigg] \end{aligned}$$

This derivation reveals that SimPO is equivalent to Maximum Entropy RL under the contextual bandit formulation with addition of length normalization and target margin augmentation, making explicit the theoretical connection that underlies SimPO's design.

Theoretical Guarantees. Following the same theoretical framework as DPO, SimPO inherits analogous guarantees regarding representational completeness, equivalence class preservation, and consistency under the Bradley-Terry preference model. The detailed proofs and formal statements of these properties are provided in Appendix C.

4 Maximum Entropy RLHF

Having established the theoretical connection between SimPO and Maximum Entropy RL, we now turn to the online RLHF setting. Our goal is to evaluate whether Maximum Entropy RL can perform comparably to its KL-constrained counterpart when applied directly to preference optimization.

4.1 Experimental Setup and Methodology

We conduct experiments using 1B, 2.8B, and 6.9B parameter models from the Pythia suite (Biderman et al., 2023), trained with RLOO (Ahmadian et al., 2024) on the TL;DR summarization dataset (Stiennon et al., 2022). All experiments follow the RLHF training recipe described by Huang et al. (2024) and are implemented using the TRL library (von Werra et al., 2020). Model alignment is evaluated using simulated preference win rates computed with GPT-4o-mini (OpenAI et al., 2024) as a proxy evaluator, measured against reference summaries for TL;DR using greedy decoding unless otherwise stated. Our experimental protocol closely follows the setup of Rafailov et al. (2024a), enabling direct comparison with prior work.

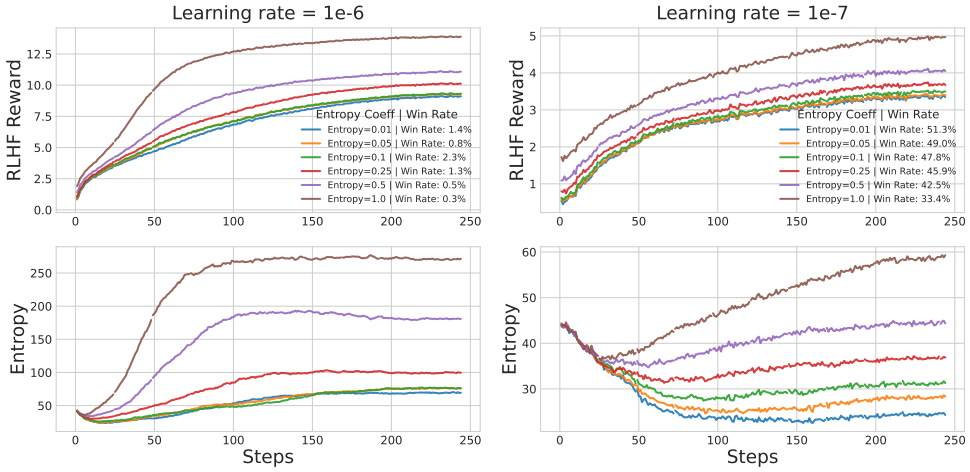

Figure 1: RLHF reward and entropy bonus during training for Pythia 1B with different entropy coefficients at learning rates 1e-6 (left) and 1e-7 (right). Win rates are reported in the legend for each entropy bonus coefficient setting.

Our aim is to study overoptimization phenomena in online RLHF, which are difficult to quantify reliably in more complex training setups involving larger models, diverse tasks, or heterogeneous reward signals. The TL;DR benchmark with Pythia models provides a controlled setting in which overoptimization can be measured consistently through preference win rates against fixed reference summaries, allowing systematic comparison across optimization objectives and model scales.

We adopt RLOO as a critic-free alternative to the standard RLHF pipeline while optimizing equivalent reward objectives. In the KL-constrained formulation, the reward is defined as

$$r(x,y) = r_\phi(x,y) - \beta\Big(\log \pi_\theta(y|x) - \log \pi_{\text{ref}}(y|x)\Big),$$

whereas in the Maximum Entropy formulation, the reward takes the form

$$r(x,y) = r_\phi(x,y) - \alpha \log \pi_\theta(y|x). \qquad (6)$$

When applying Maximum Entropy RL to sequence generation, the entropy term can be trivially increased by producing longer responses. Unlike KL-constrained objectives, which include an explicit reference policy that naturally counteracts such length effects, the reference-free formulation lacks an inherent mechanism to penalize verbosity. We therefore employ length normalization in the entropy term to ensure that the reward contribution is comparable across responses of different lengths and to prevent systematic length exploitation during optimization. This yields the reward

$$r(x,y) = r_\phi(x,y) - \frac{\beta}{|y|} \log \pi_\theta(y|x). \qquad (7)$$

We also explored extending our evaluation to additional preference datasets such as Anthropic-HH (Bai et al., 2022). In this setting, we attempted to reproduce reported baselines using both the official DPO implementation and the TRL library. Across implementations, we consistently observed that DPO failed to outperform the SFT baseline, with performance often degrading during training. This behavior exhibited substantial variability across runs and implementations, making it difficult to establish stable and reproducible reference baselines for systematic analysis. We therefore focus our empirical study on the TL;DR benchmark, which provides more consistent training dynamics suitable for controlled investigation.

4.2 Results and Analysis

Online Maximum Entropy RLHF across model scales. We evaluate Maximum Entropy RL in an online RLHF setting using Pythia models at three scales: 1B, 2.8B, and 6.9B parameters. Learning rates are chosen based on prior work, with 1×10^{-6} corresponding to a regime where KL-constrained RLHF performs well, and smaller learning rates following common practice in reference-free methods. Entropy coefficients are selected via grid search.

Across all model scales, Maximum Entropy RL is highly sensitive to the learning rate. At 1×10^{-6},

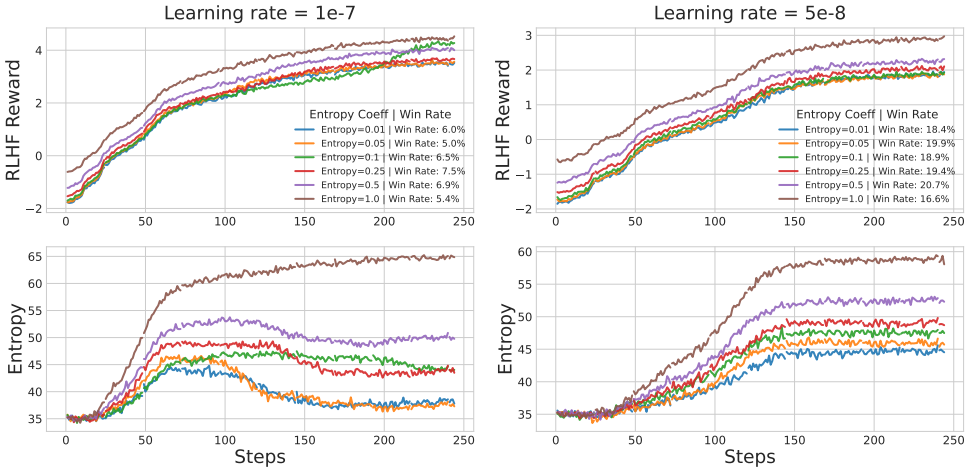

Figure 2: RLHF reward and entropy bonus during training for Pythia 2.8B with different entropy bonus coefficients at learning rates 1e-7 (left) and 5e-8 (right). Win rates are reported in the legend for each entropy bonus coefficient setting.

all models rapidly enter overoptimized regimes regardless of the entropy coefficient. Lower learning rates improve stability, but do not fully eliminate this behavior.

For Pythia 1B, reducing the learning rate to 1×10^{-7} yields stable training and improved win rates relative to the SFT baseline (Figure 1). However, these gains are not driven by entropy regularization: similar performance is achieved even when the entropy coefficient is set to zero. Moreover, stable runs exhibit decaying entropy bonuses, while overoptimized runs show increasing entropy bonuses, suggesting that entropy correlates with reward hacking rather than preventing it.

Scaling behavior. For Pythia 2.8B, we restrict experiments to learning rates of 1×10^{-7} and 5×10^{-8}, as higher learning rates consistently caused rapid overoptimization. Despite these conservative settings, all Maximum Entropy variants overoptimize and fail to outperform the SFT baseline (Figure 2). In contrast, KL-constrained RLHF remains stable and effective at a higher learning rate of 1×10^{-6}.

Interestingly, Pythia 6.9B does not exhibit behavior intermediate between smaller and larger models; instead, its training dynamics more closely resemble those of Pythia 1B than Pythia 2.8B. Training remains stable at a learning rate of 1×10^{-7} (Figure 11), but increasing the learning rate to 1×10^{-6} (Figure 10) leads to overoptimization, mirroring the instability observed in smaller models. However, KL-constrained RLHF remains stable for Pythia 6.9B while consuming only a small KL budget (Figure 12), indicating that effective KL budgets vary across model scales and are not reliably controlled by reference-free objectives.

KL budget and optimization dynamics. To better understand these failures, we analyze KL divergence during training. Well-tuned KL-constrained RLHF exhibits an initial growth phase followed by slow KL increase, keeping the policy close to the reference model. This behavior is sensitive to the KL coefficient, but provides a reliable mechanism for controlling optimization.

In contrast, Maximum Entropy RL exhibits fragile KL behavior across model scales. For Pythia 1B and 6.9B, KL divergence grows approximately linearly without immediate collapse, whereas for Pythia 2.8B similar KL magnitudes correspond to severe overoptimization. This indicates that effective KL budgets are inherently model-dependent, and that reference-free objectives lack a mechanism to infer or enforce an appropriate optimization budget. As a result, their success is not predictable across model scales.

KL update magnitudes and optimization stability. To disentangle whether overoptimization arises from the objective or from policy optimization dynamics, we examine the KL divergence between consecutive policy updates. Since RLOO implements policy optimization using PPO, one might attribute instability in Maximum Entropy RL to PPO-specific issues, such as poor ratio control or clipping behavior.

However, Figures 3, 10, 11 show that this explanation is insufficient. In KL-constrained runs, PPO reliably maintains bounded KL updates between

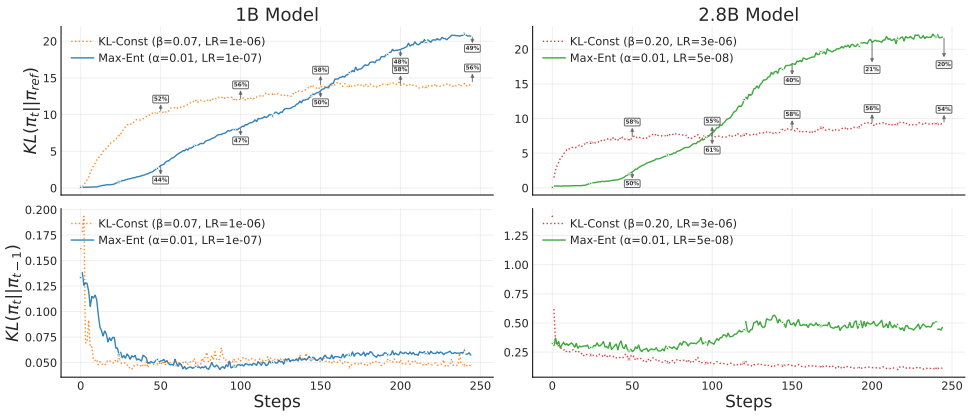

Figure 3: KL divergence metrics for KL-constrained and Maximum Entropy RL across training. Top row shows KL divergence between the current policy and the reference policy ($KL(\pi_t\|\pi_{\text{ref}})$) for Pythia 1B (left) and 2.8B (right). Bottom row shows KL divergence between consecutive policy iterations ($KL(\pi_t\|\pi_{t-1})$).

successive policies, even in regimes that eventually overoptimize when the KL coefficient is relaxed. In contrast, Maximum Entropy runs consistently exhibit increasing KL drift between updates, including in runs that appear well-behaved early in training. This effect intensifies over time despite using substantially lower learning rates than those required for stable KL-constrained optimization.

These observations suggest that the instability is not merely algorithmic, but objective-driven. Without an explicit reference policy anchoring the optimization, reward gradients become sharper and induce more aggressive parameter updates, leading to compounding KL drift. Tightening PPO clipping alone is insufficient to prevent this behavior: for Pythia 2.8B, we experimented with clipping ranges as small as 10^{-4}, yet still observed increasing KL drift and eventual overoptimization. This indicates that reference-free objectives lack the structural safeguards needed to control update magnitudes in online RLHF.

4.2.1 Minimum Entropy RL

Motivated by the link between maximum entropy and overoptimization and recent work showing entropy minimization can serve as an effective reward signal for LLM reasoning (Agarwal et al., 2025), we adopt an unconventional strategy: minimizing entropy to discourage excessively high-entropy which we expect to prevent overoptimization.

Our experiments reveal that Minimum Entropy RL prevents overoptimization and achieves competitive performance with Pythia-1B even at the same learning rate used by KL-constrained RL, under which Maximum Entropy collapses. Yet, with Pythia-2.8B, entropy minimization proves unstable: it is either too conservative, stalling learning, or too loose, leading to overoptimization. While entropy minimization succeeds as a standalone reward for reasoning, combining it with preference-based rewards appears to create optimization instabilities. Reducing the learning rate might offer some improvement, but Minimum Entropy is not a one-to-one substitute for KL, which remains more dynamic and adaptive. Lastly, this underscores that reference-free methods break down once they move outside a healthy KL budget, limiting their reliability.

4.3 Offline Maximum Entropy RLHF (SimPO)

Although Maximum Entropy RL fails to provide sufficient regularization in online settings, its closed-form offline solution, SimPO, is both effective and empirically competitive. This effectiveness cannot be attributed solely to conservative optimization: for example, one SimPO configuration for Llama 3 (Grattafiori et al., 2024) uses a higher learning rate than DPO. Nevertheless, maintaining a sufficiently low learning rate remains critical for controlling the KL divergence, which appears to be a universal requirement across alignment methods (Gao et al., 2022; Rafailov et al., 2024a).

A common argument for offline stability is that all training samples remain in-distribution, removing the need for explicit out-of-distribution regularization. However, as noted by Azar et al. (2023); Rafailov et al. (2024a), the reward model can effectively drift out of distribution during optimization, leading to sub-epoch overoptimization. This suggests that stability depends not only on data

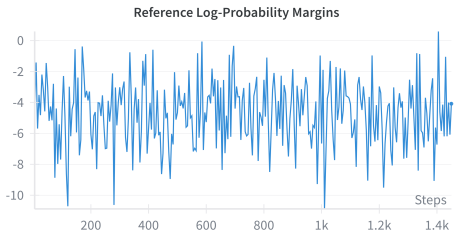

Figure 4: Batch average of $\log\left(\frac{\pi_{\text{ref}}(y_w|x)}{\pi_{\text{ref}}(y_l|x)}\right)$ during DPO training on Pythia 1B.

coverage, but also on the structure of the reward objective itself. To address this, Huang et al. (2025) propose replacing KL regularization with χ^2 regularization to inject pessimism directly into the reward model, reporting improved stability across multiple epochs. We were unable to reproduce these results, leaving open the question of whether χ^2 regularization reliably implements pessimism. In this context, SimPO's performance is particularly intriguing, not because it enforces pessimism, but because it achieves strong results without relying on a reference model.

To better understand this behavior, consider the pairwise reward used in DPO:

$$r(y_w|x) - r(y_l|x) = \beta\Bigg(\log\frac{\pi(y_w|x)}{\pi(y_l|x)} - \log\frac{\pi_{\text{ref}}(y_w|x)}{\pi_{\text{ref}}(y_l|x)}\Bigg),$$

where the second term reflects the contribution of the reference model. Because both y_w and y_l are sampled from the reference distribution, this term is typically negative but small, acting as an *adaptive regularizer*. This behavior parallels the role of the target margin in SimPO, with the key distinction that SimPO employs a fixed margin rather than a reference-dependent one (Ahrabian et al., 2025).

To examine this connection empirically, we visualize the reference log-probability margins

$$\log\left(\frac{\pi_{\text{ref}}(y_w|x)}{\pi_{\text{ref}}(y_l|x)}\right)$$

during DPO training with Pythia 1B in Figure 4. We observe that these margins lie within a relatively narrow range, consistent with the fixed margins commonly used in reference-free methods such as SimPO. This suggests that, in offline settings, reference models may primarily serve to provide a bounded margin signal rather than strong distributional control.

Several caveats are worth noting. Large margins combined with high learning rates can drive aggressive separation, exacerbating reward overoptimization (Rafailov et al., 2024a). We observe extreme likelihood decreases indicative of such behavior, consistent with prior analyses of direct alignment algorithms. While reference models may implicitly adapt margins across samples—providing smaller corrections for easy examples and larger ones for harder cases—fixed margins lack this adaptivity and can amplify pathological updates under aggressive optimization. These findings suggest that reference models are neither necessary nor sufficient to prevent overoptimization in offline preference optimization. Instead, stability emerges from the interaction between learning rate, margin choice, and model capacity. We analyze these dynamics in more detail in Appendix B.

5 Conclusion

This work establishes a theoretical foundation for SimPO by connecting it to Maximum Entropy Reinforcement Learning, while revealing a striking asymmetry between offline and online performance. Although SimPO excels in offline preference optimization, our empirical investigation shows that online Maximum Entropy RL suffers from instability and overoptimization, with entropy regularization paradoxically correlating with rather than preventing reward hacking. These findings highlight that reference-free approaches, while appealing for their simplicity, may face fundamental limitations in online training scenarios, and suggest that SimPO's success stems from implicit stabilizing factors such as dataset constraints and target margins that approximate the regularization benefits of reference models.

6 Limitations

Our experiments are conducted in a controlled setting using the TL;DR benchmark and Pythia models to ensure stable and reproducible analysis of overoptimization and KL dynamics, which limits its coverage of other tasks, reward structures, and model families. Results may not directly transfer to instruction-following, dialogue, or reasoning-heavy settings. We rely on GPT-4o-mini as a simulated preference evaluator rather than human judgments, which enables consistent large-scale comparison

but may not fully reflect human preferences. Finally, instability and sensitivity to hyperparameters across datasets and implementations constrained the breadth of our empirical evaluation, reflecting a broader reproducibility challenge in online preference optimization.

7 Ethical considerations

This work focuses on the theoretical and empirical analysis of reinforcement learning objectives for aligning large language models. All experiments were conducted on publicly available preference datasets, and no personally identifiable or sensitive information was used. Our results are intended to improve the understanding of alignment methods and do not involve deployment of models in real-world settings. Nevertheless, as with all research on large language models, advances in alignment can have dual-use implications: while they may contribute to safer and more reliable AI systems, they could also lower barriers to developing more capable models that might be misused. We encourage responsible use and further investigation into the societal impacts of alignment research.

8 The Use of Large Language Models

All text was initially drafted by the authors, after which Large Language Models were employed to refine phrasing and enhance clarity of expression.

References

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. 2018. Maximum a posteriori policy optimisation. *arXiv preprint arXiv:1806.06920*.

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. 2025. The unreasonable effectiveness of entropy minimization in llm reasoning. *Preprint*, arXiv:2505.15134.

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *Preprint*, arXiv:2402.14740.

Zafarali Ahmed, Nicolas Le Roux, Mohammad Norouzi, and Dale Schuurmans. 2019. Understanding the impact of entropy on policy optimization. In *International conference on machine learning*, pages 151–160. PMLR.

Kian Ahrabian, Xihui Lin, Barun Patra, Vishrav Chaudhary, Alon Benhaim, Jay Pujara, and Xia Song. 2025. A practical analysis of human alignment with *po. *Preprint*, arXiv:2407.15229.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete problems in ai safety. *Preprint*, arXiv:1606.06565.

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. 2023. A general theoretical paradigm to understand learning from human preferences. *Preprint*, arXiv:2310.12036.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, and 12 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *Preprint*, arXiv:2204.05862.

Yoshua Bengio, Tegan Maharaj, Luke Ong, Stuart Russell, Dawn Song, Max Tegmark, Lan Xue, Ya-Qin Zhang, Stephen Casper, Wan Sie Lee, Sören Mindermann, Vanessa Wilfred, Vidhisha Balachandran, Fazl Barez, Michael Belinsky, Imane Bello, Malo Bourgon, Mark Brakel, Siméon Campos, and 69 others. 2025. The singapore consensus on global ai safety research priorities. *Preprint*, arXiv:2506.20702.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. *Preprint*, arXiv:2304.01373.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2023. Deep reinforcement learning from human preferences. *Preprint*, arXiv:1706.03741.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. Ultrafeedback: Boosting language models with scaled ai feedback. *Preprint*, arXiv:2310.01377.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. 2023. Raft: Reward ranked finetuning for generative foundation model alignment. *Preprint*, arXiv:2304.06767.

Adam Elwood, Marco Leonardi, Ashraf Mohamed, and Alessandro Rozza. 2023. Maximum entropy exploration in contextual bandits with neural networks and energy based models. *Entropy*, 25(2):188.

Benjamin Eysenbach and Sergey Levine. 2021. Maximum entropy rl (provably) solves some robust rl problems. *arXiv preprint arXiv:2103.06257*.

Roy Fox, Ari Pakman, and Naftali Tishby. 2015. Taming the noise in reinforcement learning via soft updates. *arXiv preprint arXiv:1512.08562*.

Leo Gao, John Schulman, and Jacob Hilton. 2022. Scaling laws for reward model overoptimization. *Preprint*, arXiv:2210.10760.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Peter D Grünwald and A Philip Dawid. 2004. Game theory, maximum entropy, minimum discrepancy and robust bayesian decision theory.

Aman Gupta, Shao Tang, Qingquan Song, Sirou Zhu, Jiwoo Hong, Ankan Saha, Viral Gupta, Noah Lee, Eunki Kim, Siyu Zhu, Parag Agrawal, Natesh Pillai, and S. Sathiya Keerthi. 2025. Alphapo: Reward shape matters for llm alignment. *Preprint*, arXiv:2501.03884.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1861–1870, Stockholmsmässan, Stockholm Sweden. PMLR.

Dylan Hadfield-Menell, Smitha Milli, Pieter Abbeel, Stuart Russell, and Anca Dragan. 2020. Inverse reward design. *Preprint*, arXiv:1711.02827.

Seungyul Han and Youngchul Sung. 2021. A max-min entropy framework for reinforcement learning. *Advances in Neural Information Processing Systems*, 34:25732–25745.

Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. 2019. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pages 2681–2691. PMLR.

Jiwoo Hong, Noah Lee, and James Thorne. 2024. Orpo: Monolithic preference optimization without reference model. *Preprint*, arXiv:2403.07691.

Audrey Huang, Wenhao Zhan, Tengyang Xie, Jason D. Lee, Wen Sun, Akshay Krishnamurthy, and Dylan J. Foster. 2025. Correcting the mythos of kl-regularization: Direct alignment without overoptimization via chi-squared preference optimization. *Preprint*, arXiv:2407.13399.

Shengyi Huang, Michael Noukhovitch, Arian Hosseini, Kashif Rasul, Weixun Wang, and Lewis Tunstall. 2024. The n+ implementation details of rlhf with ppo: A case study on tl;dr summarization. *Preprint*, arXiv:2403.17031.

Woojun Kim and Youngchul Sung. 2023. An adaptive entropy-regularization framework for multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 16829–16852. PMLR.

Sergey Levine. 2018. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*.

Qi Liu, Jingqing Ruan, Hao Li, Haodong Zhao, Desheng Wang, Jiansong Chen, Wan Guanglu, Xunliang Cai, Zhi Zheng, and Tong Xu. 2025. Amopo: Adaptive multi-objective preference optimization without reward models and reference models. *Preprint*, arXiv:2506.07165.

Yixin Liu, Pengfei Liu, and Arman Cohan. 2024. Understanding reference policies in direct preference optimization. *Preprint*, arXiv:2407.13709.

Bogdan Mazoure, Thang Doan, Audrey Durand, Joelle Pineau, and R Devon Hjelm. 2020. Leveraging exploration in off-policy algorithms via normalizing flows. In *Conference on Robot Learning*, pages 430–444. PMLR.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. *Preprint*, arXiv:2405.14734.

Brendan O'Donoghue, Remi Munos, Koray Kavukcuoglu, and Volodymyr Mnih. 2016. Combining policy gradient and q-learning. *arXiv preprint arXiv:1611.01626*.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Alexander Pan, Kush Bhatia, and Jacob Steinhardt. 2022. The effects of reward misspecification: Mapping and mitigating misaligned models. *Preprint*, arXiv:2201.03544.

R. L. Plackett. 1975. The analysis of permutations. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 24(2):193–202.

Rafael Rafailov, Yaswanth Chittepu, Ryan Park, Harshit Sikchi, Joey Hejna, Bradley Knox, Chelsea Finn, and Scott Niekum. 2024a. Scaling laws for reward model overoptimization in direct alignment algorithms. *Preprint*, arXiv:2406.02900.

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. 2024b. From r to q^*: Your language model is secretly a q-function. *Preprint*, arXiv:2404.12358.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024c. Direct preference optimization: Your language model is secretly a reward model. *Preprint*, arXiv:2305.18290.

Konrad Rawlik, Marc Toussaint, and Sethu Vijayakumar. 2013. On stochastic optimal control and reinforcement learning by approximate inference.

Stuart Russell. 2022. *Artificial Intelligence and the Problem of Control*, pages 19–24.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *Preprint*, arXiv:1707.06347.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. 2025. Defining and characterizing reward hacking. *Preprint*, arXiv:2209.13085.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. 2022. Learning to summarize from human feedback. *Preprint*, arXiv:2009.01325.

Jie Sun, Junkang Wu, Jiancan Wu, Zhibo Zhu, Xingyu Lu, Jun Zhou, Lintao Ma, and Xiang Wang. 2025. Robust preference optimization via dynamic target margins. *Preprint*, arXiv:2506.03690.

Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement Learning: An Introduction*, second edition. The MIT Press.

Marc Toussaint. 2009. Robot trajectory optimization using approximate inference. In *Proceedings of the 26th annual international conference on machine learning*, pages 1049–1056.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. 2020. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`.

Tengyang Xie, Dylan J. Foster, Akshay Krishnamurthy, Corby Rosset, Ahmed Awadallah, and Alexander Rakhlin. 2024. Exploratory preference optimization: Harnessing implicit q*-approximation for sample-efficient rlhf. *Preprint*, arXiv:2405.21046.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024a. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation. *Preprint*, arXiv:2401.08417.

Jing Xu, Andrew Lee, Sainbayar Sukhbaatar, and Jason Weston. 2024b. Some things are more cringe than others: Iterative preference optimization with the pairwise cringe loss. *Preprint*, arXiv:2312.16682.

Hee Suk Yoon, Eunseop Yoon, Mark Hasegawa-Johnson, Sungwoong Kim, and Chang D. Yoo. 2025. Confpo: Exploiting policy model confidence for critical token selection in preference optimization. *Preprint*, arXiv:2506.08712.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback without tears. *Preprint*, arXiv:2304.05302.

Ruipeng Zhang, Ya-Chien Chang, and Sicun Gao. 2025. When maximum entropy misleads policy optimization. *Preprint*, arXiv:2506.05615.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. 2023. Slic-hf: Sequence likelihood calibration with human feedback. *Preprint*, arXiv:2305.10425.

Brian D Ziebart. 2010. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. Carnegie Mellon University.

Brian D Ziebart, Drew Bagnell, and Anind K Dey. 2010. Maximum causal entropy correlated equilibria for markov games. In *Workshops at the Twenty-Fourth AAAI Conference on Artificial Intelligence*.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, and 1 others. 2008. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2020. Fine-tuning language models from human preferences. *Preprint*, arXiv:1909.08593.

A Additional Related Work

Reference-free Alignment. While early methods like RRHF (Yuan et al., 2023) and RAFT (Dong et al., 2023) still relied on external reward models for ranking, they revealed that

complex RL dynamics were unnecessary. SLiC-HF (Zhao et al., 2023) showed that sequence likelihood calibration could directly incorporate human feedback without explicit reward modeling. ORPO (Hong et al., 2024) made the key insight that odds ratios could replace probability ratios, enabling monolithic training without reference model drift. CPO (Xu et al., 2024a) and SimPO (Meng et al., 2024) both recognized that sequence probabilities themselves encode preference signals. SimPO can be seen as CPO's length-normalized variant with zero behavior cloning, but this seemingly minor change eliminates the need for hyperparameter tuning of the BC coefficient. The Cringe Loss (Xu et al., 2024b) explored iterative self-improvement through token-level soft margins rather than sequence-level optimization. The proliferation of SimPO variants (AlphaPO's (Gupta et al., 2025) reward shaping, γPO's adaptive margins (Sun et al., 2025), AMoPO's (Liu et al., 2025) multi-objective extension, ConfPO's (Yoon et al., 2025) token-level refinement) demonstrates the flexibility of SimPO's reward formulation while addressing specific optimization challenges.

Overoptimization in Preference Learning Reward hacking (Skalse et al., 2025) is a long-standing problem in reinforcement learning (Sutton and Barto, 2018) where policies achieve high rewards but fail to meet the actual objective (Amodei et al., 2016; Hadfield-Menell et al., 2020; Pan et al., 2022). In language model alignment, this manifests as models learning to generate outputs that score highly on proxy metrics while being of poor actual quality. This overoptimization phenomenon was first systematically studied in traditional RLHF (Christiano et al., 2023; Stiennon et al., 2022; Gao et al., 2022; Ouyang et al., 2022), where optimizing imperfect proxy reward models leads to qualitatively worse outputs, including overly wordy responses and hallucinated information.

Direct alignment algorithms like DPO (Rafailov et al., 2024c) were designed to bypass RL training by parameterizing rewards directly in terms of the policy, but they introduce their own form of overoptimization. Azar et al. (2023) show that DPO's unbounded log-odds transformation leads to severely overfitted implicit rewards, losing the regularization benefits of standard RLHF's explicit reward modeling. They propose IPO using bounded Ψ functions to address this issue. However, Rafailov et al. (2024a) demonstrate that even IPO, despite its theoretical guarantees against overoptimization, still exhibits similar degradation patterns to DPO and RLHF at higher KL budgets and across different model scales, suggesting that overoptimization in direct alignment algorithms may be a more fundamental issue than initially anticipated. More recently, Huang et al. (2025) propose χ^2-Preference Optimization (χPO), which replaces DPO's logarithmic link function with χ^2-divergence regularization to implement pessimism under uncertainty, providing theoretical guarantees against overoptimization based on single-policy concentrability.

B Margins and Overoptimization

It has been shown that methods such as SimPO can achieve performance comparable to DPO even with a target margin of $\gamma = 0$, as demonstrated in the original SimPO paper. This suggests that offline methods do not necessarily require reference models when operating within the safe KL region, and that introducing margins generally improves performance across benchmarks. This effect arises from both model capabilities and dataset coverage: larger models are less prone to common overfitting behaviors and can extract more meaningful signals during optimization, rather than engaging in reward hacking, a phenomenon observed in both online and offline preference optimization (Gao et al., 2022; Rafailov et al., 2024c). Consequently, the influence of the reference model is minimal and can often be neglected. However, this behavior is contingent on the task being sufficiently challenging and the model being strong enough to avoid overoptimization. To validate this observation, we train Pythia-1B on TL;DR using SimPO across different margin values (γ) and learning rates, in a setting where the model is relatively weaker and the task is easier compared to standard chat datasets such as UltraFeedback (Cui et al., 2024) used in SimPO.

We first consider a learning rate of 1×10^{-6}, which is known to be effective for DPO, DPO metrics in Figure 9. In this setting, all SimPO models exhibit overoptimization regardless of the γ hyperparameter, SimPO metrics in Figure 8 . Although reward definitions differ and direct comparison of losses or other training metrics is challenging, log-probabilities of samples remain comparable. We observe the characteristic extreme likelihood decreases, which correlate with overoptimization; this

pattern is present in DAAs and, as we show, also occurs in online methods. Increasing the margin exacerbates this issue, as optimization aggressively seeks high separation, naturally resulting in overoptimization.

Reference-free methods like SimPO are particularly susceptible because they lack prior knowledge about sample difficulty, treating all samples equally. Some samples are inherently harder and should receive more attention, a behavior that could be partially captured by negative reference contributions in pairwise preferences. When a hardcoded margin pushes the model to satisfy strict separation objectives, it can amplify pathological behaviors during training.

However, when using a relatively low learning rate that allows for gradual updates, SimPO performs significantly better, metrics in Figure 7 and win rates in Figure 6 . In this regime, it emerges as a strong preference optimization method: an appropriate margin encourages the model to learn and optimize meaningful signals. Therefore, reference-free models require extra safeguards against overoptimization. Controlling the learning rate can act as an anchor, keeping updates within meaningful distributional shifts, although these models can still experience the overoptimization patterns observed in DAAs.

C Mathematical Derivations for Maximum Entropy RL

C.1 Deriving the Optimum of the Entropy-Regularized Reward Maximization Objective

In this appendix, we will derive the optimal policy for Maximum Entropy RL. Analogously to the KL-constrained case (Rafailov et al., 2024c), we optimize the following objective:

$$\max_{\pi} \mathbb{E}_{x\sim\mathcal{D},y\sim\pi}\big[r(x,y)\big] + \alpha\mathcal{H}\big[\pi(y|x)\big]$$

under any reward function $r(x,y)$ and a general non-parametric policy class, where $\mathcal{H}[\pi(y|x)] = -\mathbb{E}_{y\sim\pi(y|x)}[\log\pi(y|x)]$ is the entropy of the policy. We now have:

$$\begin{aligned}
&\max_{\pi} \mathbb{E}_{x\sim\mathcal{D},y\sim\pi}\big[r(x,y)\big] + \alpha\mathcal{H}\big[\pi(y|x)\big] \\
&= \max_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[r(x,y) - \alpha\log\pi(y|x)\right] \\
&= \min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[\log\pi(y|x) - \frac{1}{\alpha}r(x,y)\right] \\
&= \min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[\log\frac{\pi(y|x)}{\frac{1}{Z(x)}\exp\left(\frac{1}{\alpha}r(x,y)\right)}\right. \\
&\qquad\qquad \left. - \log Z(x)\right]
\end{aligned}$$

where we have partition function:

$$Z(x) = \sum_{y}\exp\left(\frac{1}{\alpha}r(x,y)\right).$$

Note that the partition function is a function of only x and the reward function r, but does not depend on the policy π. We can now define

$$\pi^*(y|x) = \frac{1}{Z(x)}\exp\left(\frac{1}{\alpha}r(x,y)\right),$$

which is a valid probability distribution as $\pi^*(y|x) \geq 0$ for all y and $\sum_y \pi^*(y|x) = 1$. Since $Z(x)$ is not a function of y, we can then re-organize the final objective in Eq. C.1 as:

$$\begin{aligned}
&\min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_{y\sim\pi(y|x)}\left[\log\frac{\pi(y|x)}{\pi^*(y|x)}\right]\right. \\
&\qquad\qquad \left. - \log Z(x)\right] = \\
&\min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\left[D_{\mathrm{KL}}(\pi(y|x)\|\pi^*(y|x)) - \log Z(x)\right]
\end{aligned}$$

Since $Z(x)$ is independent of π, the minimum is attained by the policy that minimizes the first KL term. By Gibbs' inequality, the KL divergence reaches its minimum value of zero if and only if the two distributions are identical. Therefore, this yields the optimal solution:

$$\pi(y|x) = \pi^*(y|x) = \frac{1}{Z(x)} \exp\left(\frac{1}{\alpha} r(x, y)\right) \quad (8)$$

for all $x \in \mathcal{D}$. This completes the derivation.

C.2 Deriving the SimPO Objective Under the Bradley-Terry Model

It is straightforward to derive the SimPO objective under the Bradley-Terry preference model as we have

$$p^*(y_1 \succ y_2|x) = \frac{\exp\left(r^*(x, y_1)\right)}{\exp\left(r^*(x, y_1)\right) + \exp\left(r^*(x, y_2)\right)} \quad (9)$$

We can express the (unavailable) ground-truth reward through its corresponding optimal policy:

$$r^*(x, y) = \alpha \log \pi^*(y|x) + \alpha \log Z(x) \quad (10)$$

Substituting Eq. C.2 into Eq. C.2 we obtain:

$$\begin{aligned} &p^*(y_1 \succ y_2|x) \\ &= \frac{\exp(\alpha \log \pi^*(y_1|x))}{\sum_{i\in\{1,2\}} \exp(\alpha \log \pi^*(y_i|x))} \\ &= \frac{1}{1 + \exp\left(\alpha \log \frac{\pi^*(y_2|x)}{\pi^*(y_1|x)}\right)} \\ &= \sigma\left(\alpha \log \pi^*(y_1|x) - \alpha \log \pi^*(y_2|x)\right) \end{aligned}$$

The last line is the per-instance loss for SimPO, without target margin γ and length normalization.

C.3 Deriving the SimPO Objective Under the Plackett-Luce Model

The Plackett-Luce model (Plackett, 1975) extends the Bradley-Terry model from pairwise comparisons to full rankings. As in the Bradley-Terry framework, the probability of selecting an option is assumed to be proportional to the value of an underlying latent reward function. In our setting, given a prompt x and a collection of K candidate answers $y_1, \ldots, y_K$, the user produces a permutation $\tau : [K] \to [K]$ that represents their ranking of the answers. Under the Plackett-Luce model, the probability of such a ranking is defined as follows. Let $r_k = r^*(x, y_{\tau(k)})$. Then:

$$p^*(\tau|y_1, \ldots, y_K, x) = \prod_{k=1}^{K} \frac{\exp(r_k)}{\sum_{j=k}^{K} \exp(r_j)} \quad (11)$$

Observe that when $K = 2$, Equation 11 simplifies to the Bradley-Terry model. For the general Plackett-Luce model, however, we can still leverage the reward parameterization by substituting the reward function expressed in terms of its optimal policy. As in Appendix C.2, the normalization constant $Z(x)$ cancels out. Let $s_k = \alpha \log \pi^*(y_{\tau(k)}|x)$. Then:

$$p^*(\tau|y_1, \ldots, y_K, x) = \prod_{k=1}^{K} \frac{\exp(s_k)}{\sum_{j=k}^{K} \exp(s_j)} \quad (12)$$

Similarly to the approach for standard DPO, if we have access to a dataset $\mathcal{D} = \{\tau^{(i)}, y_1^{(i)}, \ldots, y_K^{(i)}, x^{(i)}\}_{i=1}^N$ of prompts and user-specified rankings, we can use a parameterized model and optimize this objective with maximum-likelihood. Let $s_k^\theta = \alpha \log \pi_\theta(y_{\tau(k)}|x)$. Then:

$$\begin{aligned} &\mathcal{L}_{\text{SimPO}}(\pi_\theta) \\ &= -\mathbb{E}_{\mathcal{D}}\left[\sum_{k=1}^{K} \log \frac{\exp(s_k^\theta)}{\sum_{j=k}^{K} \exp(s_j^\theta)}\right] \end{aligned}$$

C.4 Deriving the Gradient of the SimPO Objective

In this section we derive the gradient of the SimPO objective:

$$\begin{aligned} &\nabla_\theta \mathcal{L}_{\text{SimPO}}(\pi_\theta) \\ &= -\nabla_\theta \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \\ &\left[\log \sigma\left(\alpha \log \pi_\theta(y_w|x) - \alpha \log \pi_\theta(y_l|x)\right)\right] \end{aligned}$$

We can rewrite the RHS of Equation C.4 as

$$\nabla_\theta \mathcal{L}_{\text{SimPO}}(\pi_\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\frac{\sigma'(u)}{\sigma(u)} \nabla_\theta(u)\right],$$

where $u = \alpha \log \pi_\theta(y_w|x) - \alpha \log \pi_\theta(y_l|x)$.

Using the properties of sigmoid function $\sigma'(x) = \sigma(x)(1 - \sigma(x))$ and $\sigma(-x) = 1 - \sigma(x)$, we obtain the final gradient

$$\nabla_\theta \mathcal{L}_{\text{SimPO}}(\pi_\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \Bigg[\alpha\sigma\left(\alpha \log \pi_\theta(y_l|x) - \alpha \log \pi_\theta(y_w|x)\right) \Bigg[\nabla_\theta \log \pi(y_w \mid x) - \nabla_\theta \log \pi(y_l \mid x)\Bigg]\Bigg],$$

After using the reward substitution of $\hat{r}_\theta(x, y) = \alpha \log \pi_\theta(y \mid x)$ we obtain the final form of the gradient.

C.5 Proof of Lemma 1 and 2 from DPO for Maximum Entropy RL

In this section, we will prove the two lemmas from DPO for Maximum Entropy RL.

Lemma 1 (Lemma 1)**.** *Under the Plackett-Luce preference framework, and in particular the Bradley-Terry framework, two reward functions from the same equivalence class induce the same preference distribution.*

Proof. We say that two reward functions $r(x, y)$ and $r'(x, y)$ are from the same equivalence class if $r'(x, y) = r(x, y) + f(x)$ for some function f. We consider the general Plackett-Luce (with the Bradley-Terry model a special case for $K = 2$) and denote the probability distribution over rankings induced by a particular reward function $r(x, y)$ as p_r. For any prompt x, answers $y_1, \ldots, y_K$ and ranking τ we have: Let $r_k = r(x, y_{\tau(k)})$ and $r'_k = r'(x, y_{\tau(k)}) = r_k + f(x)$. Then: Let $r_k = r(x, y_{\tau(k)})$ and $r'_k = r_k + f(x)$. Then:

$$\begin{aligned}
&p_{r'}(\tau|y_1, \ldots, y_K, x) \\
&= \prod_{k=1}^{K} \frac{\exp(r'_k)}{\sum_{j=k}^{K} \exp(r'_j)} \\
&= \prod_{k=1}^{K} \frac{\exp(r_k + f(x))}{\sum_{j=k}^{K} \exp(r_j + f(x))} \\
&= \prod_{k=1}^{K} \frac{\exp(f(x)) \exp(r_k)}{\exp(f(x)) \sum_{j=k}^{K} \exp(r_j)} \\
&= \prod_{k=1}^{K} \frac{\exp(r_k)}{\sum_{j=k}^{K} \exp(r_j)} \\
&= p_r(\tau|y_1, \ldots, y_K, x)
\end{aligned}$$

which completes the proof. □

Lemma 2 (Lemma 2)**.** *Two reward functions from the same equivalence class induce the same optimal policy under the entropy-regularized RL problem.*

Proof. Let us consider two reward functions from the same class, such that $r'(x, y) = r(x, y) + f(x)$ and, let us denote as π_r and $\pi_{r'}$ the corresponding optimal policies. For all x, y, let $\rho = \frac{1}{\alpha} r(x, y)$, $\rho' = \frac{1}{\alpha} r'(x, y)$, and $\phi = \frac{1}{\alpha} f(x)$. Then:

$$\begin{aligned}
\pi_{r'}(y|x) &= \frac{\exp(\rho')}{\sum_y \exp(\rho')} \\
&= \frac{\exp(\rho + \phi)}{\sum_y \exp(\rho + \phi)} \\
&= \frac{\exp(\phi) \exp(\rho)}{\exp(\phi) \sum_y \exp(\rho)} \\
&= \frac{\exp(\rho)}{\sum_y \exp(\rho)} \\
&= \pi_r(y|x),
\end{aligned}$$

□

C.6 Proof of Theorem 1 from DPO for Maximum Entropy RL

In this section, we will eloborate on the results of the main theorem from DPO for Maximum Entropy RL.

Theorem 1 (Maximum Entropy Version)**.** *Assume we have a parameter $\alpha > 0$. All reward equivalence classes, as defined in the previous section, can be represented with the reparameterization $r(x, y) = \alpha \log \pi(y|x)$ for some model $\pi(y|x)$.*

Proof. Consider any reward function $r(x, y)$, which induces an optimal model $\pi_r(y|x)$ under the entropy-regularized RL problem, with solution given by the optimal policy derivation. We have:

$$r(x, y) = \alpha \log \pi_r(y|x) + \alpha \log Z(x)$$

where $Z(x) = \sum_y \exp\left(\frac{1}{\alpha} r(x, y)\right)$ (notice that $Z(x)$ also depends on the reward function r). Using the operator $r'(x, y) = f(r, \alpha)(x, y) = r(x, y) - \alpha \log Z(x)$, we see that this new reward function is within the equivalence class of r and, we have:

$$r'(x, y) = \alpha \log \pi_r(y|x)$$

which completes the proof. □

We can further expand on these results. We can see that if r and r' are two reward functions in the same class, then

$$f(r, \alpha)(x, y) = \alpha \log \pi_r(y|x) = \alpha \log \pi_{r'}(y|x) = f(r', \alpha)(x, y)$$

where the second equality follows from Lemma 2. We have proven that the operator f maps all reward functions from a particular equivalence class to the same reward function. Next, we show that for every equivalence class of reward functions, the reward function that has the reparameterization outlined in the main theorem is unique.

Proposition 1. *Assume we have a parameter $\alpha > 0$. Then every equivalence class of reward functions has a unique reward function $r(x, y)$, which can be reparameterized as $r(x, y) = \alpha \log \pi(y|x)$ for some model $\pi(y|x)$.*

Proof. We will proceed using proof by contradiction. Assume we have two reward functions from the same class, such that $r'(x, y) = r(x, y) + f(x)$. Moreover, assume that $r'(x, y) = \alpha \log \pi'(y|x)$ for some model $\pi'(y|x)$ and $r(x, y) = \alpha \log \pi(y|x)$ for some model $\pi(y|x)$, such that $\pi \neq \pi'$. We then have

$$\begin{aligned} r'(x, y) = r(x, y) + f(x) &= \alpha \log \pi(y|x) + f(x) \\ &= \alpha \log \pi(y|x) \exp(\frac{1}{\alpha} f(x)) \\ &= \alpha \log \pi'(y|x) \end{aligned}$$

for all prompts x and completions y. Then we must have $\pi(y|x) \exp(\frac{1}{\alpha} f(x)) = \pi'(y|x)$. Since these are distributions, summing over y on both sides, we obtain that $\exp(\frac{1}{\alpha} f(x)) = 1$ and since $\alpha > 0$, we must have $f(x) = 0$ for all x. Therefore $r(x, y) = r'(x, y)$. This completes the proof. □

We have now shown that every reward class has a unique reward function that can be represented as outlined in the main theorem, which is given by $f(r, \alpha)$ for any reward function in that class.

D Extra Figures

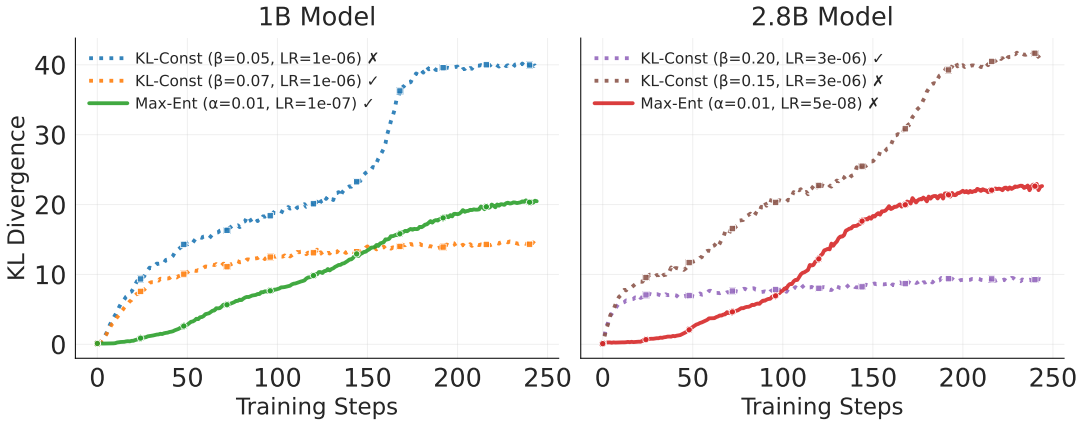

Figure 5: KL divergence evolution during training for 1B and 2.8B parameter models using different regularization methods. The left panel shows results for the 1B model and the right panel shows results for the 2.8B model. Each panel compares KL-Constrained and Maximum-Entropy approaches. Checkmarks (✓) indicate high win rate runs and crosses (×) indicate overoptimized runs.

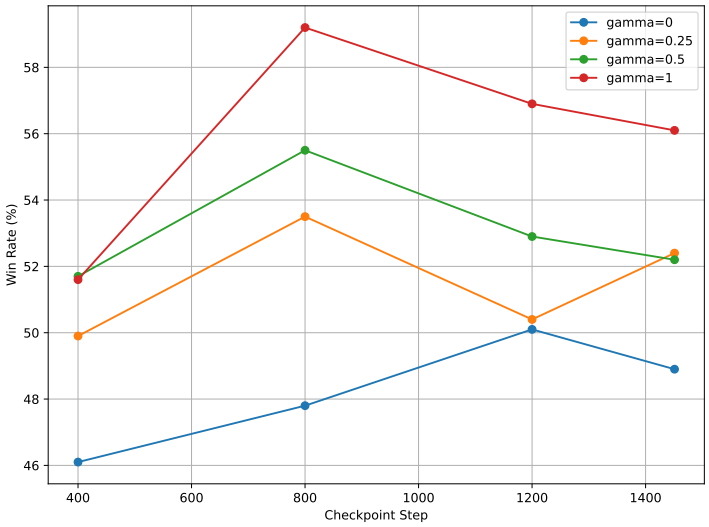

Figure 6: Win rate progression across training checkpoints for different values of the gamma hyperparameter. Results are for the Pythia-1B model trained with a learning rate of 2×10^{-7}.

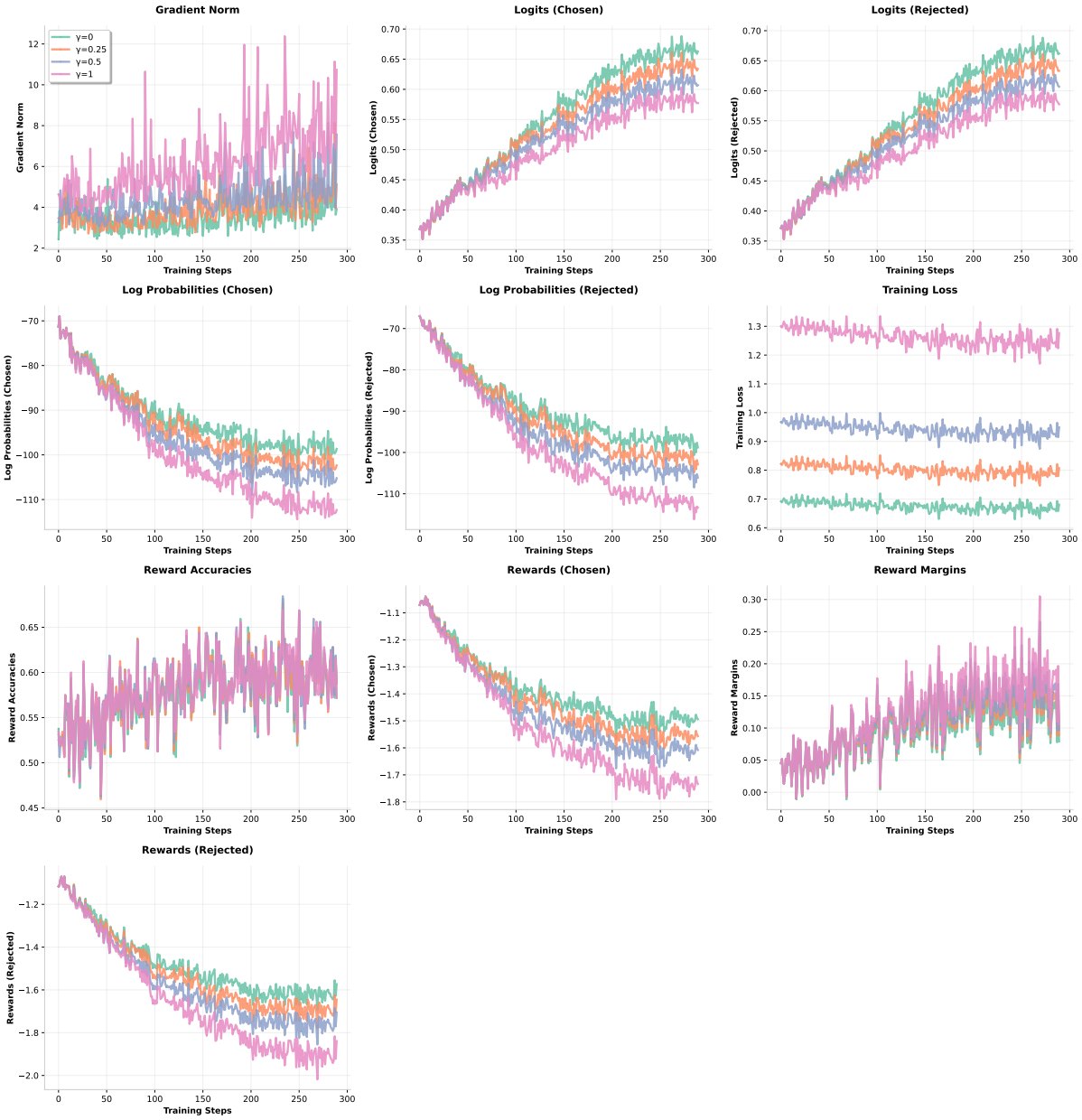

Figure 7: SimPO training metrics across different gamma values. Comparison of key training dynamics including loss, gradients, logits, and reward metrics for $\gamma \in \{0, 0.25, 0.5, 1.0\}$ using Pythia-1B with learning rate 2×10^{-7}.

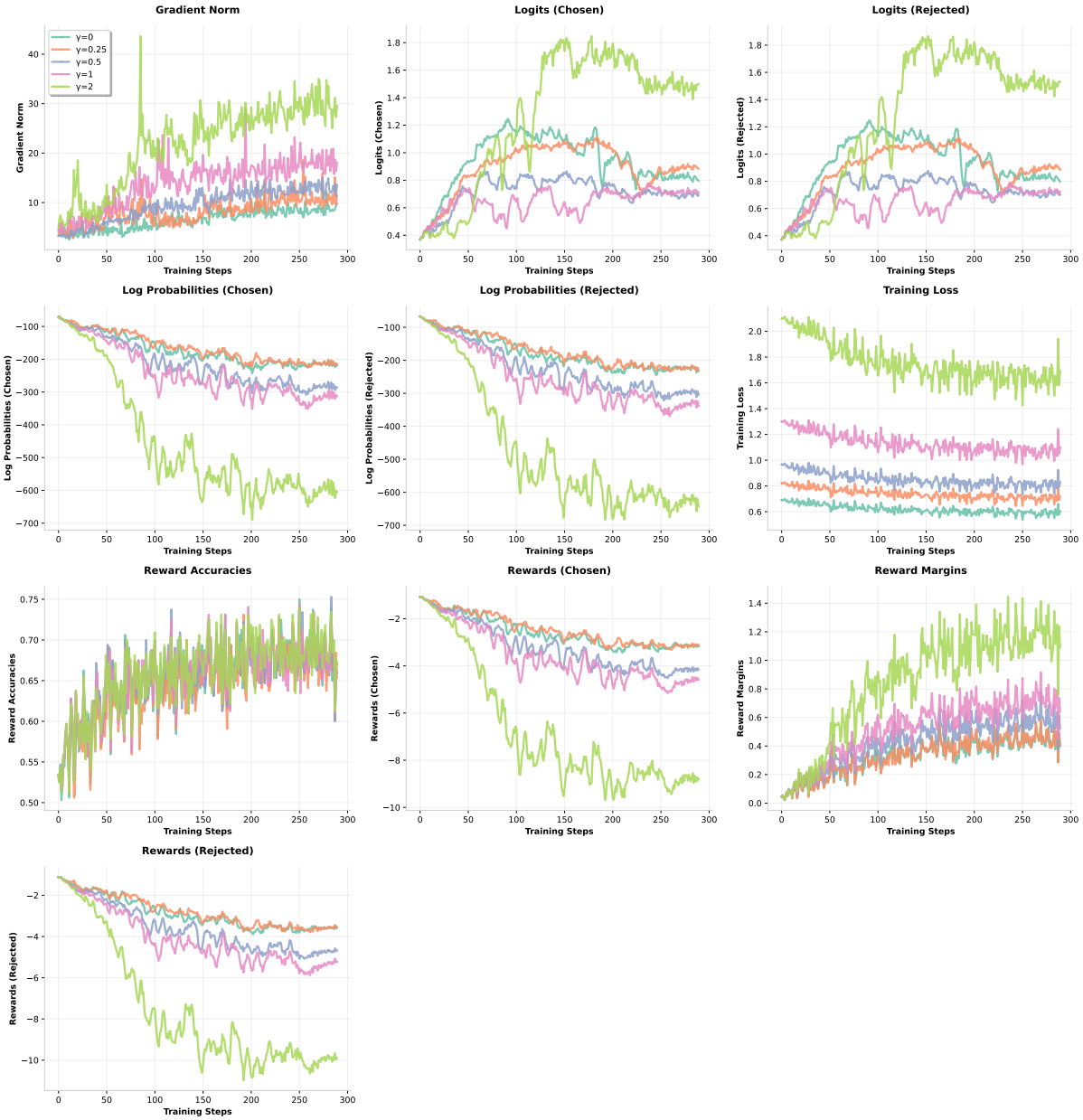

Figure 8: SimPO training metrics across different gamma values. Comparison of key training dynamics including loss, gradients, logits, and reward metrics for $\gamma \in \{0, 0.25, 0.5, 1.0, 2.0\}$ using Pythia-1B with learning rate 1×10^{-6}.

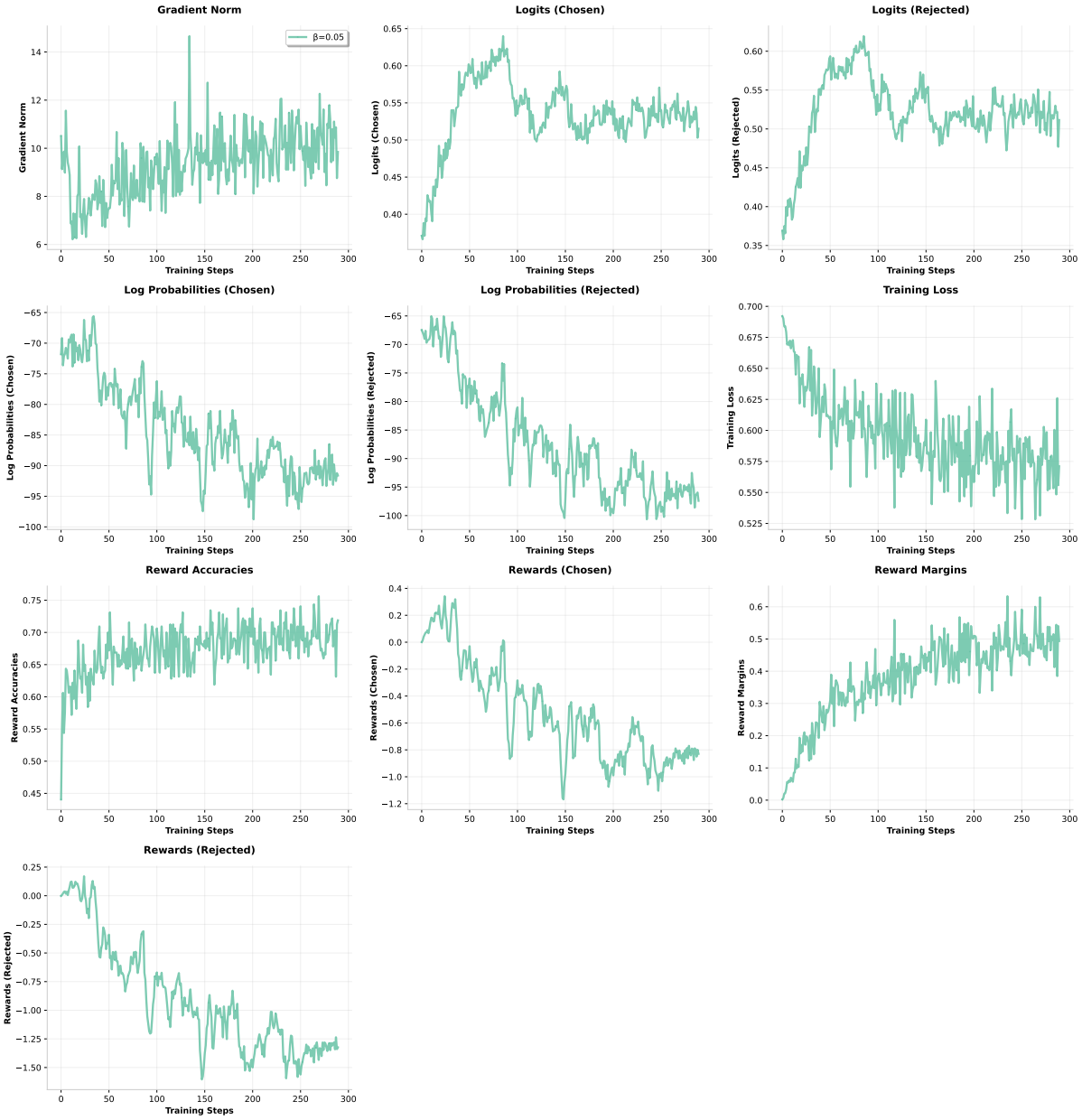

Figure 9: DPO training metrics with $\beta = 0.05$. Comparison of key training dynamics including loss, gradients, logits, and reward metrics, using Pythia-1B with learning rate 1×10^{-6}.

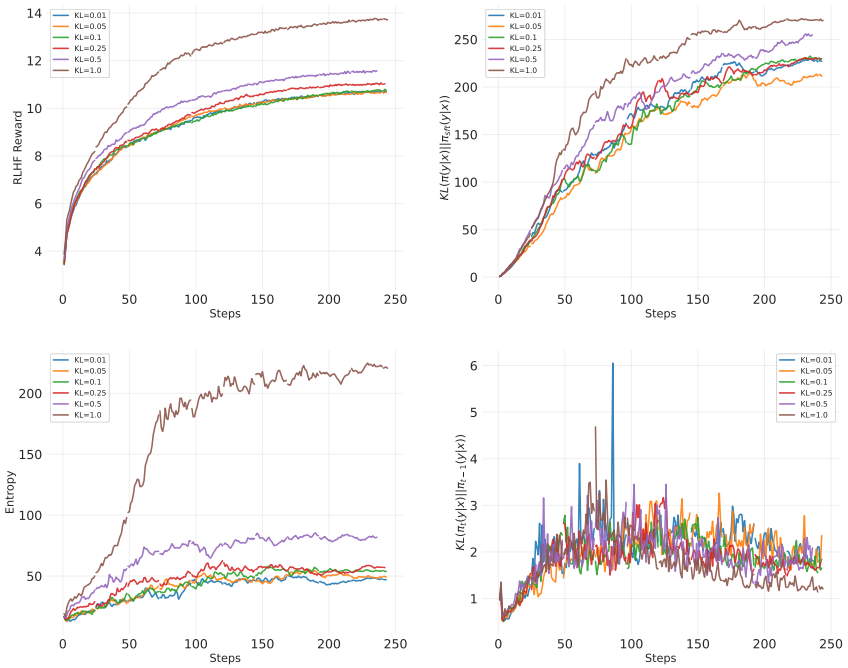

Figure 10: Reward dynamics and KL divergence metrics for entropy-regularized RL training across different entropy coefficients. Top-left panel shows reward progression (RLHF reward) over training steps for various entropy values. Top-right panel shows KL divergence between the current policy and the SFT reference policy ($KL(\pi_t||\pi_{\text{SFT}})$). Bottom-left panel tracks entropy reward across training steps. Bottom-right panel displays KL divergence between consecutive policy updates ($KL(\pi_t||\pi_{t-1})$). All plots are based on the Pythia-6.9B model trained with the learning rate of 1×10^{-6}.

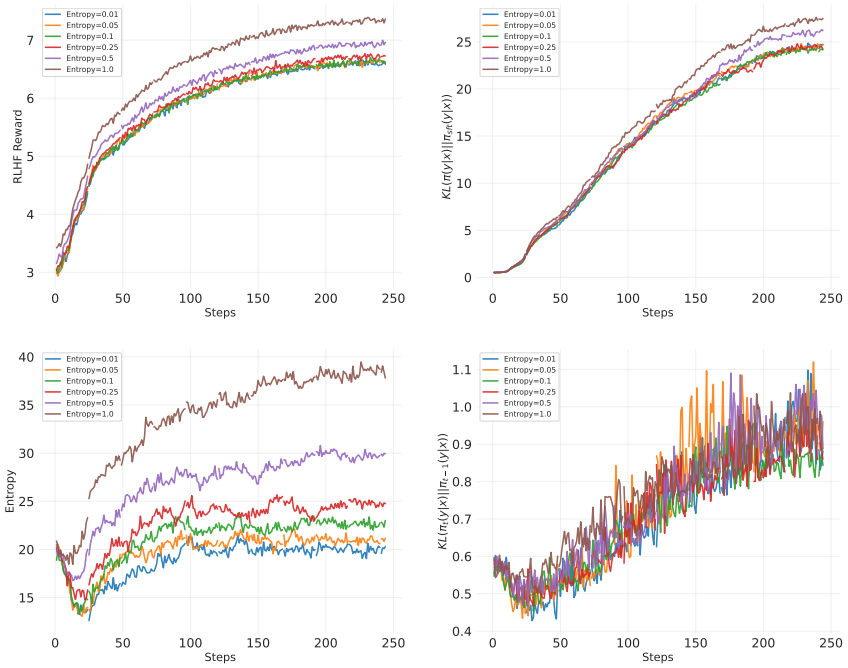

Figure 11: Reward dynamics and KL divergence metrics for entropy-regularized RL training across different entropy coefficients. Top-left panel shows reward progression (RLHF reward) over training steps for various entropy values. Top-right panel shows KL divergence between the current policy and the SFT reference policy ($KL(\pi_t||\pi_{\text{SFT}})$). Bottom-left panel tracks entropy reward across training steps. Bottom-right panel displays KL divergence between consecutive policy updates ($KL(\pi_t||\pi_{t-1})$). All plots are based on the Pythia-6.9B model trained with the learning rate of 1×10^{-7}.

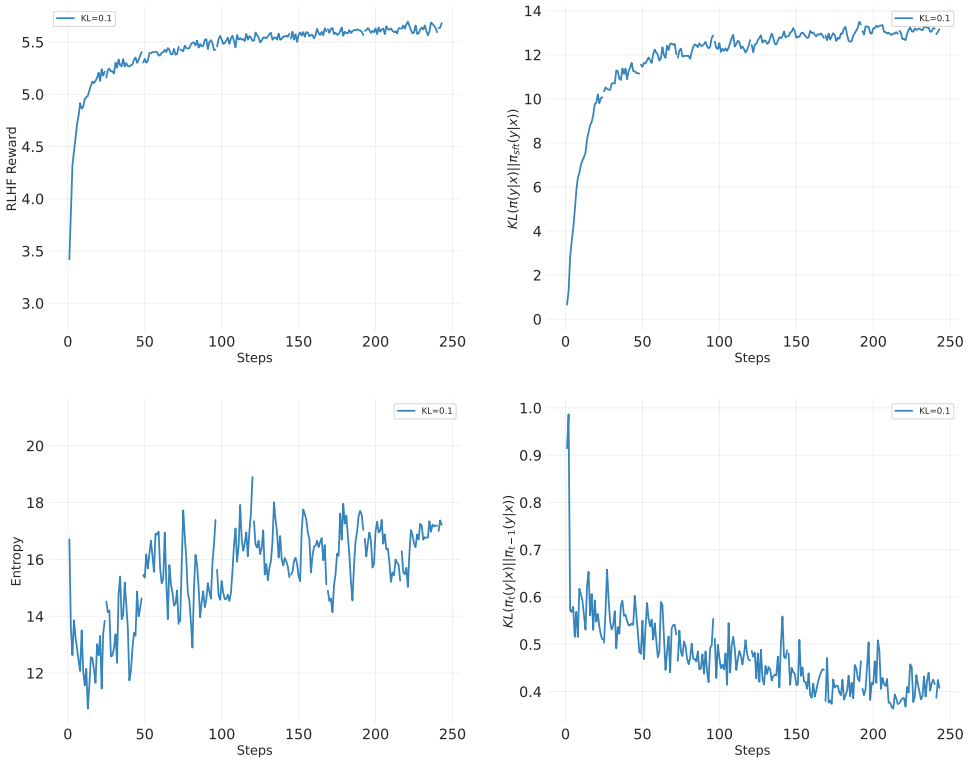

Figure 12: Training metrics for KL-constrained RL on the Pythia-6.9B model. Top panel shows the KL divergence between the policy and reference SFT policy ($KL(\pi_t||\pi_{\text{SFT}})$) over training steps. Middle panel displays the reward trajectory (RLHF reward). Bottom panel shows the KL divergence between consecutive policy updates ($KL(\pi_t||\pi_{t-1})$). All results correspond to a single training run with a fixed KL constraint.