Alignment as Distribution Learning: Your Preference Model is Explicitly a Language Model

Jihun Yun [* 1] **Juno Kim** [* 2] **Jongho Park** [1 2] **Junhyuck Kim** [1] **Jongha Jon Ryu** [3] **Jaewoong Cho** [1] **Kwang-Sung Jun** [4]

Abstract

Alignment via reinforcement learning from human feedback (RLHF) has become the dominant paradigm for controlling the quality of outputs from large language models (LLMs). However, when viewed as 'loss + regularization,' the standard RLHF objective lacks theoretical justification and incentivizes degenerate, deterministic solutions, an issue that variants such as Direct Policy Optimization (DPO) also inherit. In this paper, we rethink alignment by framing it as *distribution learning* from pairwise preference feedback by explicitly modeling how information about the target language model bleeds through the preference data. This explicit modeling leads us to propose three principled learning objectives: preference maximum likelihood estimation, preference distillation, and reverse KL minimization. We theoretically show that all three approaches enjoy strong non-asymptotic $O(1/n)$ convergence to the target language model, naturally avoiding degeneracy and reward overfitting. Finally, we empirically demonstrate that our distribution learning framework, especially preference distillation, consistently outperforms or matches the performances of RLHF and DPO across various tasks and models.

1. Introduction

Alignment refers to the task of controlling the quality of responses (e.g., helpfulness and harmlessness) generated from large language models (LLMs) via human preferences (Bai et al., 2022; Ouyang et al., 2022) and has become the de facto final step in LLM training. The first method introduced for alignment is Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Stiennon et al., 2020), which trains a reward model R from pairwise preferences and then optimizes a policy π (i.e., language model) that maximizes the reward via reinforcement learning (RL):

$$\max_{\pi} \ \mathbb{E}_{x\sim\mathcal{D},a\sim\pi(x)}\left[R(x,a)\right] - \beta\,\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi(x)\|\pi_0(x))\right] \tag{1}$$

where $\mathcal{D}$ is the prompt distribution, $\pi(x)$ is the policy π's distribution over responses to a prompt x, $\mathrm{KL}(p\|q)$ is the Kullback-Leibler divergence from p to q, $\beta > 0$ is a hyperparameter, and π_0 is a reference LLM resulting from the supervised fine-tuning phase.

The RLHF objective is central to various practical algorithms and has fundamentally shaped how researchers think about alignment. For example, DPO (Direct Policy Optimization) reformulates RLHF so the objective consists of simple likelihood terms rather than relying on on-policy responses (Rafailov et al., 2023). ΨPO extends RLHF by generalizing $R(x,a)$ to a Ψ-transformation of the preference probability (Azar et al., 2024). Even theoretical analyses of alignment algorithms often treat the RLHF objective or its variants as the ultimate learning-theoretic goal, aiming to establish convergence guarantees for its solution (Xiong et al., 2024; Huang et al., 2025; Xie et al., 2025b). Certainly, the RLHF objective has proven useful, and one may argue that it is a sensible objective.

However, the justification of the RLHF objective remains unclear from a learning-theoretic perspective. When viewed as a standard machine learning objective of the form 'loss + regularizer,' the loss part is simply the negative reward. Consequently, the loss desires to drive the language model towards a degenerate solution, collapsing into a deterministic mapping rather than a proper distribution; only the KL regularizer is what prevents such degeneracy. In addition, DPO, a reformulation of RLHF, also introduces an objective whose optimal solution can be degenerate (Song et al., 2024; Fisch et al., 2025).

Our contributions. In this paper, we depart from blindly taking the RLHF objective as the ultimate goal and instead

[*]Equal contribution [1]KRAFTON [2]UC Berkeley [3]MIT [4]University of Arizona. Correspondence to: Jihun Yun <jihuny@krafton.com>, Juno Kim <junokim@berkeley.edu>, Kwang-Sung Jun <kjun@cs.arizona.edu>.

ICML 2025 Workshop on Models of Human Feedback for AI Alignment.

Table 1. Summary of our proposed methods and theoretical guarantees. In each section, we draw parallels to existing approaches such as DPO and REBEL (Gao et al., 2024).

Distribution Learning	Related to	Reward Model	Requires RL training	Objective	Theoretical Guarantee
Preference MLE (Sec. 3)	DPO	Not Used	No	Eq. 4	Forward KL (Thm. 4)
Preference distillation (Sec. 4)	REBEL	Required	No	Eq. 11	Forward KL (Thm. 6)
Reverse KL (Sec. 5)	RLHF	Required	Yes	Eq. 16	Reverse KL (Thm. 7)

propose a fresh **distribution learning** perspective based on statistical principles, which we found to be largely underexplored in existing works. We assume that there exists a target 'oracle' language model π^* and explicitly model how information about π^* bleeds through preference feedback. Specifically, we model the probability of response a being preferred over b as follows:

$$\mathbb{P}(a \succ b \mid x) = \frac{\pi^*(a \mid x)^\gamma}{\pi^*(a \mid x)^\gamma + \pi^*(b \mid x)^\gamma} \tag{2}$$

for some $\gamma > 0$, which is a Bradley-Terry model (Bradley & Terry, 1952) with preference score proportional to tilted response likelihood. This assumption says that the preference model is *explicitly* a language model. This is in stark contrast to DPO which starts from the RLHF formulation and leverages the all-policy assumption to realize that there is a *secret* relationship between the reward model (or preference model) and the language model (Rafailov et al., 2023).

This simple modeling assumption naturally leads to various training objectives whose solutions provably converge to π^* with respect to metrics such as KL divergence. Specifically, we propose three algorithms summarized in Table 1 and described as follows:

- **PMLE** (Preference Maximum Likelihood Estimate; Section 3): This objective maximizes the likelihood of the preference model (2), subject to reverse KL regularization w.r.t. a reference policy π_0. Similarly to DPO, it is relatively straightforward to optimize. We provide a theoretical guarantee on the forward KL: $\mathbb{E}_x[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))] \le O(1/n)$ where n is the training set size.

- **Preference distillation** (Section 4): By directly estimating the expected preference from a learned reward model, the MLE can be rewritten as distilling the preference distribution into a language model. Unlike existing reward distillation (Fisch et al., 2025; Gao et al., 2024), this formulation is explicitly derived from the Bradley-Terry model (2) and also enjoys an $O(1/n)$ convergence guarantee on the forward KL.

- **Reverse KL** (Section 5): Since our goal is distribution learning, it is natural to optimize the so-called reverse KL divergence: $\mathbb{E}_x[\mathrm{KL}(\hat{\pi}(x)\|\pi^*(x)]$. Although π^* is unknown, its unnormalized form can be estimated from (2) with a shallow network, which amounts to learning a reward model in RLHF. Plugging in our estimate of π^* in the reverse KL along with a KL regularizer ends up being a generalization of the RLHF objective that has an additional entropy term, effectively smoothing the prior π_0. Via an $O(1/n)$ reverse KL error bound, our framework offers a learning-theoretic grounding for RLHF.

All our theoretical guarantees are non-asymptotic and first-of-its-kind for learning a distribution from pairwise feedback, to the best of our knowledge (cf. Dumoulin et al., 2023). We complement our theory with experiments showing that our methods consistently outperform baseline winrates in TL;DR summarization and generate more preferred responses in general chat experiments, confirming the practical utility of our distribution learning viewpoint.

We defer a detailed discussion of related work to Appendix A; however closely related work is cited and discussed throughout the main content.

2. Preliminaries

Alignment as distribution learning. Let $\mathcal{X}, \mathcal{A}$ be the set of prompts and responses, respectively, and let $\mathcal{D} \in \Delta(\mathcal{X})$ be a fixed distribution over prompts. We define a language model (LM) as a function or policy $\pi : \mathcal{X} \to \Delta(\mathcal{A})$ determining a collection of conditional (i.e., contextual) distributions $\pi(\cdot \mid x)$, which we also denote more simply as $\pi(x)$.[1] We view alignment as learning these distributions from pairwise preference feedback, drawn from a model explicitly depending on π^*, the ideal (target) LM we wish to learn. Hence given a class of language models Π, our ultimate goal is to find $\hat{\pi} \in \Pi$ that is as close as possible to π^* with respect to a suitable measure of distance between distributions, such as KL divergence.

An explicit preference model. Let μ be the LM used for generating responses to be preference-labeled; this could

[1] This definition can also cover unconditional distributions by introducing a member in $\mathcal{X}$ as a null prompt.

be a reference LLM or simply an existing dataset. We are given a preference dataset $D_n = \{(x, a^+, a^-)\}$ of n independent samples where $x \sim \mathcal{D}$ is a prompt, a^+ is a preferred response, and a^- is a dispreferred response. We assume that, given x, the pair (a^+, a^-) is sampled by drawing $a, b \sim \mu(x)$ independently and then sampling a preference from $\mathbb{P}_*(a \succ b \mid x) := \mathbb{P}_{\pi^*}(a \succ b \mid x)$ where

$$\mathbb{P}_\pi(a \succ b \mid x) := \frac{\pi(a \mid x)^\gamma}{\pi(a \mid x)^\gamma + \pi(b \mid x)^\gamma}, \tag{3}$$

followed by setting $(a^+, a^-) = (a, b)$ if a is preferred over b and $(a^+, a^-) = (b, a)$ otherwise. The value of γ determines the extent to which differences in the response probabilities under policy π are accentuated or attenuated. In practice, γ is a hyperparameter typically set as $0 < \gamma < 1$.

Theoretical setup. To present our learning-theoretic guarantees, we introduce the following notations. We call $R_\pi(x, a) := \gamma \ln \pi(a \mid x)$ the *reward* induced by $\pi \in \Pi$.[2] The centered reward is defined as $\bar{R}_\pi(x, a) := R_\pi(x, a) - \mathbb{E}_{a \sim \mu(x)}[R_\pi(x, a) \mid x]$. As with $\mathbb{P}_*$, we write $R_* := R_{\pi^*}$ and $\bar{R}_* := \bar{R}_{\pi^*}$. Finally, we denote $\Delta \bar{R}_\pi := \bar{R}_\pi - \bar{R}_*$.

Our main assumptions are as follows:

Assumption 1 (Realizability)**.** $\pi^* \in \Pi$ for a finite policy class Π.

Assumption 2 (Boundedness)**.** There exists $R > 0$ such that $|\bar{R}_\pi(x, a)| \le \gamma R$ for all $\pi \in \Pi$.

Since the responses (a^+, a^-) in the data are sampled from $\mu(x)$ rather than $\pi^*(x)$, the alignment problem is an instance of offline learning where there is a distribution shift between the data that we observe versus the target distribution that we aim to have guarantees on. It is thus necessary to introduce a coverage assumption between μ and the policy class Π, which is well-studied in the offline reinforcement learning literature (Agarwal et al., 2019). In particular, we use the following generalized coverage coefficient (Xie et al., 2021; Agarwal et al., 2025).

Definition 3 (Generalized coverage coefficient)**.** For a policy class Π', we denote by $C_{\Pi'} > 0$ the smallest constant satisfying for every $\pi \in \Pi'$,

$$\mathbb{E}_{x \sim \mathcal{D}, a \sim \pi^*(x)} \left[\Delta \bar{R}_\pi(x, a)^2 \right] \le C_{\Pi'} \, \mathbb{E}_{x \sim \mathcal{D}, a \sim \mu(x)} \left[\Delta \bar{R}_\pi(x, a)^2 \right].$$

The generalized coverage coefficient improves upon the naive all-policy ℓ_∞-concentrability assumption $\sup_{\pi \in \Pi} \max_{x,a} \frac{\pi(a|x)}{\mu(a|x)} \le C'$ (Munos, 2003) because the former can be bounded even if the latter is infinite, depending on $\mathcal{D}$ and the reward class $\Delta \bar{R}$.[3]

3. Preference Maximum Likelihood Estimation Approach

We begin by introducing a maximum likelihood-based objective that can be directly derived from treating alignment as distribution learning from pairwise feedback. Suppose that we are given a dataset of n independent pairwise preferences $D_n = \{(x, a^+, a^-)\}$ as described in Section 2. We wish to estimate π^* by finding a policy $\hat{\pi}$ that maximizes the likelihood of observed pairwise preferences under the Bradley-Terry preference assumption (3). Concretely, the negative log-likelihood for each pair (x, a^+, a^-) under a candidate policy π is:

$$-\ln \mathbb{P}_\pi(a^+ \succ a^- \mid x) = -\ln \left[\sigma\left(\gamma \ln \frac{\pi(a^+ \mid x)}{\pi(a^- \mid x)} \right) \right],$$

where $\sigma(z) = 1/(1 + \exp(-z))$ is the logistic sigmoid. Summing over all preference pairs yields

$$\mathcal{L}_{\mathsf{PMLE}}(\pi) = \frac{1}{n} \sum_{(x,a^+,a^-) \in D_n} -\ln \left[\sigma\left(\gamma \ln \frac{\pi(a^+ \mid x)}{\pi(a^- \mid x)} \right) \right]. \tag{4}$$

By minimizing $\mathcal{L}_{\mathsf{PMLE}}$, we encourage π to place higher probability on response a^+ relative to a^-. Note that in practice, we rarely learn a policy π from scratch; instead, we typically optimize a reasonably performant pretrained and fine-tuned model, referred to as the *reference policy* π_0. Thus, it is natural to introduce a KL penalty that keeps π close to π_0 for alignment: $\beta \cdot \mathrm{KL}(\pi(x) \| \pi_0(x))$. Putting everything together, our **PMLE** (preference maximum likelihood estimation) objective for distribution learning is

$$\mathcal{L}_{\mathsf{PMLE},\beta}(\pi) = \frac{1}{n} \sum_{(x,a^+,a^-) \in D_n} -\ln \left[\sigma\left(\gamma \ln \frac{\pi(a^+ \mid x)}{\pi(a^- \mid x)} \right) \right] + \beta \, \mathrm{KL}(\pi(x) \| \pi_0(x)). \tag{5}$$

Remark. Recall that DPO (Rafailov et al., 2023) minimizes the objective

$$\mathcal{L}_{\mathsf{DPO}}(\pi) = \frac{1}{n} \sum_{(x,a^+,a^-) \in D_n} -\ln \left[\sigma\left(\gamma \ln \frac{\pi(a^+ \mid x)}{\pi(a^- \mid x)} - \gamma \ln \frac{\pi_0(a^+ \mid x)}{\pi_0(a^- \mid x)} \right) \right]. \tag{6}$$

[2]There is no actual reward in the PMLE scheme; we just call this reward for convenience.

[3]Leveraging pessimism (Gabbianelli et al., 2024; Huang et al., 2025; Zhan et al., 2022) may further improve the coverage coefficient to a single concentrability coefficient that relies on π^* rather than the policy class Π, which is beyond the scope of our work.

Compared to (5), the DPO objective does not have an explicit regularizer, which could lead to undesirable behaviors if the policy class Π is sufficiently expressive. Specifically, Fisch et al. (2025) have proven that DPO may converge to a degenerate distribution. Furthermore, Song et al. (2024) have shown that DPO relies on a strong coverage assumption: if π_0 does not fully cover the relevant distribution, DPO can produce out-of-distribution responses, making its reward estimates inaccurate. Unlike RLHF, which is constrained by a KL term to stay within the support of π_0, DPO can assign non-zero probability to responses that π_0 would never select, undermining performance guarantees. In contrast, our PMLE objective (5) incorporates an explicit KL term that effectively circumvents the aforementioned pitfalls.

Convergence guarantee. Under the assumptions stated in Section 2, we show the following bound on the forward KL. Throughout the paper, constants depending only on R are hidden.

Theorem 4. *The PMLE estimate* $\hat{\pi} = \arg\min_{\pi\in\Pi} \mathcal{L}_{\mathsf{PMLE}}(\pi)$ *satisfies with probability at least* $1-\delta$,

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right] \lesssim \frac{C_\Pi}{\gamma^2}\cdot\frac{\ln(|\Pi|/\delta)}{n}. \quad (7)$$

The proof, provided in Appendix B.2, is inspired in part by Agarwal et al. (2025, proof of Theorem 3.6), but we leverage Schulman's trick (Schulman, 2020) followed by a quadratic approximation to obtain a $1/n$ rate rather than $1/\sqrt{n}$ that would be obtained when directly following their proof. Also note that the left-hand side of (7) is equivalent to the KL divergence between the induced *joint* distributions on $\mathcal{X}\times\mathcal{A}$: $\mathrm{KL}(\mathcal{D}(x)\pi^*(a\mid x)\|\mathcal{D}(x)\hat{\pi}(a\mid x))$.

We assume the regularizer $\beta = 0$ here and for all theoretical guarantees in the sequel for simplicity and to demonstrate that the objective derived from purely considering preference feedback via (3) already suffices to learn the true distribution π^*. Nonetheless, we posit that starting from a well-aligned π_0 can result in improved convergence guarantees by mitigating the dependency of constants on R, which we leave to future work.

In the following section, we shift our focus to a distribution learning perspective on algorithms that require explicit reward modeling.

4. Preference Distillation Approach

Since the popularization of RLHF, the use of reward modeling has become popular in the research community and resulted in various extensions (Christiano et al., 2017). While the main role of the reward model in the RLHF objective (1) is to view alignment as an RL problem, recent studies have attempted to use the reward model for supervised learning losses, i.e., objectives that do not require RL to solve (Guo et al., 2024; Fisch et al., 2025). These efforts can be seen as *distilling information* from the reward model as pointed out by Fisch et al. (2025). The main benefit of these methods is that they can avoid RL algorithms, which are typically slow to converge. While reward model training is an extra burden to perform compared to purely likelihood-based methods such as DPO or our PMLE, the compute cost for doing so is typically quite low because it usually suffices to train a shallow network on top of an existing LLM's frozen torso.

Reward model. Due to our preference model (3), learning a reward model $R:\mathcal{X}\times\mathcal{A}\to\mathbb{R}$ is equivalent to learning a language model π and then setting $R(x,a)=\gamma\ln\pi(a\mid x)$ up to an additive constant. Conversely, given a reward model $R(x,a)$, we can estimate an LM by

$$\pi(a\mid x)\propto\exp(\gamma^{-1}R(x,a)),\quad \forall x\in\mathcal{X}. \quad (8)$$

Note that this is a model from which sampling is computationally intractable in general. Formally, we assume that we are given a reward model class $\mathcal{R}$ of rewards $R:\mathcal{X}\times\mathcal{A}\to\mathbb{R}$ and learn:

$$\hat{R} = \underset{R\in\mathcal{R}}{\arg\min}\ \frac{1}{n}\sum_{(x,a^+,a^-)\in D_n} -\ln\sigma(R(x,a^+)-R(x,a^-)). \quad (9)$$

This is equivalent to the PMLE objective under (8) but with the constraint $R\in\mathcal{R}$. To train a reward model, we recommend regularization strategies such as ℓ_2 regularization or early stopping.

Preference distillation. One popular method for distilling rewards is the REBEL algorithm (Gao et al., 2024). Motivated by the characterization of the RLHF solution under the all-policy assumption (Rafailov et al., 2023), REBEL aims to extract information from relative reward values of paired responses, enforcing the condition $\ln\frac{\pi(a^+|x)/\pi_0(a^+|x)}{\pi(a^-|x)/\pi_0(a^-|x)} \approx \eta(\hat{R}(x,a^+)-\hat{R}(x,a^-))$ by optimizing a squared loss of the form

$$\frac{1}{n}\sum_{(x,a^+,a^-)\in D_n}\left(\ln\frac{\pi(a^+\mid x)/\pi_0(a^+\mid x)}{\pi(a^-\mid x)/\pi_0(a^-\mid x)} - \eta(\hat{R}(x,a^+)-\hat{R}(x,a^-))\right)^2 \quad (10)$$

where $\eta>0$ controls the strength of the reward signals. In our modeling assumption, the reward model can be seen as a shifted version of $\gamma\ln\pi^*(a\mid x)$, so we could optimize (10) without the π_0 terms, replacing η by γ^{-1}. However, the use of squared loss in (10) is not well justified from a statistical perspective, and it is unclear if squared loss should be preferred over any other loss, e.g., absolute loss.

What is then the appropriate error measure? Our framework tells us that learning a reward model amounts to learning a preference model. In other words, we have trained a *preference simulator*: a non-generative language model estimate $\widetilde{\pi}(a \mid x) \propto \exp(\gamma^{-1}\hat{R}(x,a))$ from which preference can be sampled for any pair of responses as $y \sim \mathrm{Bernoulli}(\mathbb{P}_{\widetilde{\pi}}(a \succ b \mid x))$. Plugging this into PMLE would yield a natural distribution learning objective. However, this process introduces additional randomness which can hinder optimization. Instead, observe that we can evaluate the expectation of the PMLE objective and replace the discrete label y with the *expected* preference

$$\begin{aligned}\mathbb{P}_{\widetilde{\pi}}(a^+ \succ a^- \mid x) &= \frac{\widetilde{\pi}(a^+ \mid x)^\gamma}{\widetilde{\pi}(a^+ \mid x)^\gamma + \widetilde{\pi}(a^- \mid x)^\gamma} \\ &= \sigma(\hat{R}(x,a^+) - \hat{R}(x,a^-)).\end{aligned}$$

Then, minimizing the log-loss with respect to this synthetic preference leads to:

$$\begin{aligned}&\mathcal{L}_{\mathsf{Distill}}(\pi) \\ &:= \frac{1}{n}\sum_{(x,a^+,a^-)\in D_n} -\mathbb{P}_{\widetilde{\pi}}(a^+ \succ a^- \mid x)\ln \mathbb{P}_{\pi}(a^+ \succ a^- \mid x) \\ &\quad - \mathbb{P}_{\widetilde{\pi}}(a^- \succ a^+ \mid x)\ln \mathbb{P}_{\pi}(a^- \succ a^+ \mid x). \qquad (11)\end{aligned}$$

This also corresponds to minimizing the summed KL divergence between the binary preference distributions, $\mathrm{Bernoulli}(\mathbb{P}_{\widetilde{\pi}}(a^+ \succ a^- \mid x))$ and $\mathrm{Bernoulli}(\mathbb{P}_{\pi}(a^+ \succ a^- \mid x))$. As in the PMLE approach (Section 3), in practice we add a KL regularizer and optimize:

$$\mathcal{L}_{\mathsf{Distill},\beta}(\pi) := \mathcal{L}_{\mathsf{Distill}}(\pi) + \beta\,\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi(x)\|\pi_0(x))\right]. \qquad (12)$$

We remark that the data for reward model training (9) and preference distillation (11) can come from the same dataset or different datasets: our theoretical analysis is easily adapted to both scenarios.

Convergence guarantee. The family of (non-generative) language models induced by the reward model class $\mathcal{R}$ is defined as

$$\begin{aligned}\mathcal{P}_\gamma(\mathcal{R}) := \big\{&\pi : \pi(a \mid x) \propto \exp(\gamma^{-1}R(x,a)), \\ &\forall a, x \in \mathcal{X} \text{ for some } R \in \mathcal{R}\big\}.\end{aligned}$$

Assumption 5. The reward-induced LM class $\mathcal{P}_\gamma(\mathcal{R}) \subseteq \Pi$.

This assumption is related to the generator-verifier gap, which informally states that verifying whether a given answer is correct or not is easier than generating a correct answer (Li et al., 2024; West et al., 2024). Such a gap implies that $\mathcal{R}$ is easier to learn than Π from a learning-theoretic perspective ($|\mathcal{R}| \ll |\Pi|$), and is speculated to hold for LLMs in practice (Swamy et al., 2025). Assumption 5 can also be justified by the fact that the reward model is often built on top of the supervised fined-tuned model's (frozen) torso. Denoting by $C_{\mathcal{R}} := C_{\mathcal{P}_\gamma(\mathcal{R})}$ the generalized coverage coefficient of the induced subclass, under Assumption 5 it holds that $C_{\mathcal{R}} \leq C_\Pi$.

Theorem 6. *The preference distillation estimate $\hat{\pi} = \arg\min_{\pi\in\Pi}\mathcal{L}_{\mathsf{Distill}}(\pi)$ satisfies with probability at least $1-\delta$,*

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right] \lesssim \frac{C_\Pi}{\gamma^2}\cdot\frac{\ln(|\Pi|/\delta)}{n}. \qquad (13)$$

This results in the same rate as PMLE (7). The proof is provided in Appendix B.3.

5. Reverse KL Minimization Approach

Our two proposed methods both maximize a preference likelihood and ultimately enjoy a guarantee on the forward KL divergence $\mathbb{E}_x[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))]$. However, it is also plausible to aim to minimize the reverse KL divergence $\mathbb{E}_x[\mathrm{KL}(\hat{\pi}(x)\|\pi^*(x))]$ to learn the distribution π^*. Reverse KL has the well-known 'mode seeking' behavior as opposed to 'mode covering' behavior of the forward KL. This mode-seeking behavior tends to find distributions that generate realistic content in image generation and has been preferred in image generative models (Goodfellow et al., 2014; Mao et al., 2019).

In this section, we explore the reverse KL formulation for alignment under our modeling assumption (3), which turns out to be a generalization of the original RLHF framework (1) (Stiennon et al., 2020; Ouyang et al., 2022). Directly minimizing the reverse KL w.r.t. the target LM π^* would yield:

$$\begin{aligned}\hat{\pi} &= \arg\min_{\pi\in\Pi}\mathbb{E}_{x\sim\mathcal{D}}\big[\mathrm{KL}(\pi(x)\|\pi^*(x))\big] \\ &= \arg\min_{\pi\in\Pi}\mathbb{E}_{x\sim\mathcal{D}}\Big[\mathbb{E}_{a\sim\pi(x)}\left[-\ln\pi^*(a \mid x)\right] + H(\pi(x))\Big], \qquad (14)\end{aligned}$$

where $H(\pi(x))$ is the Shannon entropy of $\pi(x)$. However, this requires rewards of the form $-\ln\pi^*$, which is the very object we are trying to estimate. To solve this issue, we propose to find a plugin estimator from a surrogate class of language models that are easier to train but harder to sample from. Specifically, we determine $\widetilde{\pi} = \arg\min_{\pi\in\mathcal{P}_\gamma(\mathcal{R})}\mathcal{L}_{\mathsf{PMLE}}(\pi)$ (with a suitable regularization), which is equivalent to obtaining $\hat{R}$ via (9) followed by setting $\widetilde{\pi}(a \mid x) \propto \exp(\gamma^{-1}\hat{R}(x,a))$ as before. Then we can plug in our learned $\widetilde{\pi}$ to π^* in the objective (14) to arrive at

$$\hat{\pi} = \arg\min_{\pi\in\Pi} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_{a\sim\pi(x)}\left[-\gamma^{-1}\hat{R}(x,a)\right] - H(\pi(x))\right]. \tag{15}$$

The normalization constant, which is prohibitive to compute in practice, naturally disappears as we only require relative rewards for optimization. Lastly, we again add a KL regularizer w.r.t. π_0:

$$\mathcal{L}_{\mathsf{RKL},\beta}(\pi) := \frac{1}{n}\sum_{(x,\cdot,\cdot)\in D_n} -\mathbb{E}_{a\sim\pi(x)}\left[\hat{R}(x,a)\right] - \gamma H(\pi(x)) + \beta\,\mathrm{KL}(\pi(x)\|\pi_0(x)) \tag{16}$$

where β and γ take the role of hyperparameters controlling the relative weights of the policy entropy and KL regularizer, respectively. In practice, as in standard RLHF pipelines (Ouyang et al., 2022; Bai et al., 2022), one first fits a reward model $\hat{R}$ to approximate the underlying true reward R^* from pairwise preferences, then applies an RL algorithm (e.g., PPO (Schulman et al., 2017)) to maximize $\hat{R}$ with a KL penalty followed by an additional entropy regularizer.

Relation to RLHF. When $\gamma = 0$, the objective above exactly coincides with (1), implying that our reverse KL objective is a generalization of the RLHF objective. Conversely, under our preference assumption (3), RLHF itself can be interpreted as minimizing a reverse KL term in the population limit. In this sense, our derivation can be viewed as providing *theoretical justification* for the RLHF objective (1) which has been widely as the gold standard for alignment (Stiennon et al., 2020; Ouyang et al., 2022; Bai et al., 2022; Rafailov et al., 2023), *while also providing a minor correction.* Note that such a connection to RLHF may not be surprising given that the max entropy RL objective can be seen as reverse KL minimization (Ziebart, 2010).

Prior smoothing. A key distinction from the standard RLHF objective lies in how our formulation balances reward maximization with *prior smoothing*. For an explicit comparison, we illustrate the effect of the additional entropy term for a toy alignment problem. Consider learning a K-categorical distribution on the simplex $\Delta_K = \{\mathbf{p} \in \mathbb{R}^d_{\geq 0} : \sum_{k=1}^K p_k = 1\}$, which can be viewed as a contextless language model with response length one and a vocabulary size of K. Suppose we are given a fixed vector $\mathbf{p}_0 \in \Delta_K$ as the reference model and a learned reward function $\hat{\mathbf{r}} = (r_1, \cdots, r_K)$. In the standard RLHF approach (1) with temperature $\beta+\gamma$, the optimal policy is given for all $k \in [K]$ by $\hat{p}_k^{\mathsf{RLHF}} \propto p_{0,k}\exp(\frac{r_k}{\beta+\gamma})$. In contrast, our reverse KL objective (16) can be rearranged as

$$\begin{aligned}\mathcal{L}_{\mathsf{RKL},\beta}(\mathbf{p}) &= -\mathbf{p}\cdot\hat{\mathbf{r}} - \gamma H(\mathbf{p}) + \beta\,\mathrm{KL}(\mathbf{p}\|\mathbf{p}_0)\\ &= -\mathbf{p}\cdot\hat{\mathbf{r}} + (\beta+\gamma)\,\mathrm{KL}(\mathbf{p}\|\mathbf{p}_0^\alpha) + \text{const.}\end{aligned}$$

where $\alpha := \frac{\beta}{\beta+\gamma}$, resulting in the policy $\hat{p}_k \propto p_{0,k}^\alpha \exp(\frac{r_k}{\beta+\gamma})$. The additional exponent $\alpha \in (0,1)$ acts to smooth the prior from $\mathbf{p}_0$ to $\mathbf{p}_0^\alpha$, allocating relatively more mass to actions with low initial probability. This boosts exploration especially for actions which were unlikely under the base policy, so that the estimated policy $\hat{\mathbf{p}}$ would not be too close to a degenerate distribution even if $\mathbf{p}_0$ is.

Convergence guarantee. With the objective $\mathcal{L}_{\mathsf{RKL}}$, we are indeed able to obtain a convergence guarantee for the reverse KL:

Theorem 7. *The reverse KL estimate* $\hat{\pi} = \arg\min_{\pi\in\Pi}\mathcal{L}_{\mathsf{RKL},0}(\pi)$ *with satisfies with probability at least* $1-\delta$*,*

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\hat{\pi}(x)\|\pi^*(x))\right] \lesssim \frac{\ln(|\Pi|/\delta)}{n} + \frac{C_{\mathcal{R}}}{\gamma^2}\cdot\frac{\ln(|\mathcal{R}|/\delta)}{n}. \tag{17}$$

See Appendix B.4 for the proof.

Why does reverse KL attain a better bound? Note that the reverse KL formulation results in an improved upper bound that depends on the coverage coefficient of $\mathcal{P}_\gamma(\mathcal{R})$ (17) rather than Π (13); in particular, under Assumption 5, $\ln|\Pi|$ and $C_{\mathcal{R}}\ln|\mathcal{R}|$ may both be much smaller than $C_\Pi \ln|\Pi|$.

Astute readers may wonder: How can reverse KL avoid C_Π while preference distillation does not, even though they both leverage the reward model? The reason is not because we bound the reverse KL instead of forward KL; we show that the forward and reverse KL error may be compared (up to a constant exponential in R) in Proposition 14 in the appendix. The true reason is that the *policy learning step* of preference distillation still relies on the responses $(x, a^+, a^-) \in D_n$ sampled from μ, unlike reverse KL which uses the responses from μ for the lightweight *reward modeling step* only.

Intractability of forward KL. An alternative is to directly optimize the forward KL:

$$\arg\min_{\pi\in\Pi}\textstyle\sum_{(x,\cdot,\cdot)\sim D_n}\mathrm{KL}(\widetilde{\pi}(x)\|\pi(x)).$$

Here, we are not using a^+ and a^-, so the dependence on μ disappears and we will not pay for C_Π, similarly to Theorem 7. But how do we compute the forward KL? Direct computation is untenable due to the sheer size of $\mathcal{A}$ in language models. Instead, one may attempt to sample from $\widetilde{\pi}(x)$ and perform stochastic optimization; however, such a sampling is not feasible because we only have access to the unnormalized version $\exp(\gamma^{-1}\hat{R}(x,\cdot))$. Another attempt is to use the fact that

$$\mathrm{KL}(\widetilde{\pi}(x)\|\pi(x)) = \mathbb{E}_{a\sim\pi(x)}\left[\frac{\widetilde{\pi}(a\mid x)}{\pi(a\mid x)}\ln\frac{\widetilde{\pi}(a\mid x)}{\pi(a\mid x)}\right].$$

While we do not have to sample from $\widetilde{\pi}(x)$, we now have to evaluate the value of $\widetilde{\pi}(a \mid x)$, which, again, is intractable.

6. Experiments

In this section, we present empirical results demonstrating that alignment via distribution learning yields strong performance in practice. Specifically, we validate our proposed methods by systematically comparing them against their well-established baselines, including DPO, RLHF, and REBEL, on a range of language tasks.

6.1. TL;DR Summarization

We focus on the TL;DR summarization task (Stiennon et al., 2020), largely adhering to the training procedure detailed by Gao et al. (2024); Song et al. (2024).

Setup. We use Pythia-1.4B and Pythia-2.8B (Biderman et al., 2023). We initially perform a single epoch of supervised fine-tuning (SFT), using human reference responses as target labels. Following this, we obtain the reward model by fine-tuning the SFT model with a regression head for one epoch on the preference training dataset. We train DPO and MLE on preference datasets annotated with prompt-specific preference labels, whereas the online RL approaches use the dataset containing only human reference responses. All models are initialized with the SFT model prior to alignment, and RLHF and reverse KL minimization are optimized using PPO.

Evaluation. For each algorithm, we measure both the reward score assigned by our learned reward model and the KL divergence from the reference model, $\mathrm{KL}(\pi \| \pi_0)$. To evaluate the quality of model-generated responses, we use GPT-4 to compare them against human reference responses, calculating the win-rate. This win-rate is computed over 600 randomly selected samples, which corresponds to roughly 10% of the test set. We describe additional details and experiments in Appendix C.

Results. For both Pythia-1.4B and Pythia-2.8B in Table 2, our distribution-learning objectives – PMLE, preference distillation, and reverse KL – mostly outperform their respective baselines in terms of win-rate, which serves as the most direct measure of language model quality on downstream tasks. The only exception is PMLE on the 2.8B model, where it performs slightly worse than DPO. Additionally, since PMLE implements a KL regularizer with online data, it achieves much lower KL divergence from π_0 compared to DPO, which solely relies on the offline dataset; this finding aligns with the results reported by Song et al. (2024). As for RLHF and REBEL, both methods use the same KL penalty for each experiment, naturally leading to similar $\mathrm{KL}(\pi \| \pi_0)$ values. Notably, compared to REBEL, preference distillation achieves comparable RM scores and KL divergence, while exhibiting a substantially higher win-rate. Overall, our experiments demonstrate that the algorithms derived from our modeling assumption (2) can match or exceed existing methods on practical tasks.

6.2. General Chat

Prior studies (Noukhovitch et al., 2023; Lin et al., 2024) suggest that aligning LLM with RLHF can incur an *alignment tax*, where models forget some of their pretrained capabilities, resulting in performance degradation on standard benchmarks. We hypothesize that our distribution learning framework mitigates this issue more effectively than conventional reward maximization. To this end, we examine the performance of our method across multiple language tasks when trained on a more general chat dataset while maintaining the downstream task performance. In particular, we compare our best-performing approach from the previous section, preference distillation, against its counterpart algorithm, REBEL.

Setup. We train LLaMA-3-8B-Instruct (Grattafiori et al., 2024) as our base model on the UltraFeedBack dataset (Cui et al., 2023), using Eurus-RM-7B (Yuan et al., 2024) as the reward model. We provide further experimental details such as hyperparameter settings in Appendix C.

Evaluation. Building on earlier work, we measure the alignment tax, i.e., the extent of performance deterioration using the Open LLM leaderboard (Beeching et al., 2023) as metrics, a widely adopted criteria for LLM evaluation. In particular, we focus on MMLU (Hendrycks et al., 2021), GSM8K (Cobbe et al., 2021), ARC challenge (Clark et al., 2018), Winogrande (Sakaguchi et al., 2021), TruthfulQA (Lin et al., 2022), and HellaSwag (Zellers et al., 2019) as done in Gao et al. (2024); Chen et al. (2025); Xie et al. (2025a).

Results. Table 3 presents the results on academic benchmarks. Preference distillation exhibits a similar alignment tax compared to REBEL while achieving higher reward scores (last column). This show that preference distillation can generate more preferred responses with similar alignment tax. This finding lends further support to our preference assumption (3) in conjunction with our discussion indicating that a learned reward model's score can serve as an indirect proxy for the underlying preference distribution (Section 4). We also include evaluations on MT-Bench and AlpacaEval 2.0 in Appendix C.

7. Conclusion

In this paper, we have explored the significance of making clear assumptions about the target model π^* and its rela-

Table 2. **Results on TL;DR dataset with Pythia 1.4B and 2.8B**. Win-rate is evaluated by GPT-4 and reward model (RM) score evaluated by the trained reward model. Our proposed methods mostly outperform their counterparts, while preference distillation proves the most competitive across both models in terms of win-rate.

Model size	Algorithm	Win-rate (↑)	RM score (↑)	KL($\pi \| \pi_0$)(↓)
1.4B	DPO	45.0%	1.03	33.46
	PMLE	**46.0%**	**1.12**	**19.60**
	REBEL	59.5%	2.60	**31.67**
	Preference distillation	**62.5%**	**2.61**	33.31
	RLHF	60.0%	**2.74**	24.41
	Reverse KL	**61.5%**	2.73	**23.91**
2.8B	DPO	**50.6%**	**1.83**	65.01
	PMLE	48.6%	1.51	**30.40**
	REBEL	70.1%	**1.85**	**18.34**
	Preference distillation	**75.8%**	1.82	19.44
	RLHF	74.0%	**1.83**	15.64
	Reverse KL	**74.6%**	1.82	**15.50**

Table 3. **Alignment Tax**. Performance comparison across academic benchmarks.

Model	**MMLU** (5-shot)	**GSM8K** (5-shot)	**ARC** (25-shot)	**WINOG** (5-shot)	**TRUTH** (0-shot)	**HELLA** (10-shot)	**Avg.**	RM Score
LLaMA-3-8B-Instruct	65.8	75.3	62.0	75.5	51.7	78.7	68.1	-
REBEL-LLaMA-3	65.6	76.5	61.9	75.6	51.4	78.6	68.2	2610
Distill-LLaMA-3	65.7	76.5	62.2	75.3	51.5	78.7	**68.3**	**2697**

tionship to observed preferences, a perspective we found underexplored in existing literature. By formulating alignment as distribution learning based on our explicit modeling assumption (3), we naturally derived three novel alignment methods: PMLE, preference distillation, and reverse KL. We have shown that these approaches correct and generalize existing methods in a principled manner, and demonstrated strong convergence guarantees and empirical performance.

Our work opens several important directions for future research. First, it would be interesting to compare mode-seeking and mode-covering objectives in terms of response quality across various domains. Second, alternative metrics such as Jensen-Shannon divergence and Wasserstein distance warrant exploration under our framework, potentially yielding more novel algorithms with strong theoretical guarantees. Finally, the constants in our upper bounds could be improved; in particular, the exponential dependence on R might be removed by incorporating the KL regularizer and assuming π_0 is sufficiently close to π^*.

References

Agarwal, A., Jiang, N., Kakade, S. M., and Sun, W. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, 2019.

Agarwal, A., Dann, C., and Marinov, T. V. Design considerations in offline preference-based rl. *arXiv preprint arXiv:2502.06861*, 2025.

Amini, A., Vieira, T., and Cotterell, R. Direct preference optimization with an offset. *CoRR*, abs/2402.10571, 2024.

Azar, M. G., Guo, Z. D., Piot, B., Munos, R., Rowland, M., Valko, M., and Calandriello, D. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Machine Learning*, 2024.

Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Beeching, E., Fourrier, C., Habib, N., Han, S., Lambert, N., Rajani, N., Sanseviero, O., Tunstall, L., and Wolf, T. Open llm leaderboard (2023-2024). https://huggingface.co/spaces/open-llm-leaderboard-old/open_llm_leaderboard, 2023.

Biderman, S., Schoelkopf, H., Anthony, Q. G., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pp. 2397–2430. PMLR, 2023.

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

Cen, S., Mei, J., Goshvadi, K., Dai, H., Yang, T., Yang, S., Schuurmans, D., Chi, Y., and Dai, B. Value-incentivized preference optimization: A unified approach to online and offline RLHF. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=SQnitDuow6.

Chen, M., Chen, Y., Sun, W., and Zhang, X. Avoiding exp(rmax) scaling in rlhf through preference-based exploration. *arXiv preprint arXiv:2502.00666*, 2025.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., and Tafjord, O. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Cui, G., Yuan, L., Ding, N., Yao, G., Zhu, W., Ni, Y., Xie, G., Liu, Z., and Sun, M. Ultrafeedback: Boosting language models with high-quality feedback. *CoRR*, 2023.

Dubois, Y., Liang, P., and Hashimoto, T. Length-controlled alpacaeval: A simple debiasing of automatic evaluators. In *First Conference on Language Modeling*, 2024.

Dumoulin, V., Johnson, D. D., Castro, P. S., Larochelle, H., and Dauphin, Y. A density estimation perspective on learning from pairwise human preferences. *arXiv preprint arXiv:2311.14115*, 2023.

Ethayarajh, K., Xu, W., Muennighoff, N., Jurafsky, D., and Kiela, D. Model alignment as prospect theoretic optimization. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Fisch, A., Eisenstein, J., Zayats, V., Agarwal, A., Beirami, A., Nagpal, C., Shaw, P., and Berant, J. Robust preference optimization through reward model distillation. *Transactions on Machine Learning Research (TMLR)*, 2025. ISSN 2835-8856.

Foster, D. J. and Krishnamurthy, A. Efficient first-order contextual bandits: Prediction, allocation, and triangular discrimination. *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 18907–18919, 2021.

Foster, D. J., Mhammedi, Z., and Rohatgi, D. Is a good foundation necessary for efficient reinforcement learning? the computational role of the base model in exploration, 2025. URL https://arxiv.org/abs/2503.07453.

Gabbianelli, G., Neu, G., and Papini, M. Importance-weighted offline learning done right. In *International Conference on Algorithmic Learning Theory (ALT)*, pp. 614–634. PMLR, 2024.

Gao, Z., Chang, J., Zhan, W., Oertell, O., Swamy, G., Brantley, K., Joachims, T., Bagnell, D., Lee, J. D., and Sun, W. Rebel: Reinforcement learning via regressing relative rewards. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Guo, S., Zhang, B., Liu, T., Liu, T., Khalman, M., Llinares, F., Rame, A., Mesnard, T., Zhao, Y., Piot, B., et al. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*, 2024.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Hong, J., Lee, N., and Thorne, J. ORPO: monolithic preference optimization without reference model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024.

Hu, E. J., yelong shen, Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Huang, A., Zhan, W., Xie, T., Lee, J. D., Sun, W., Krishnamurthy, A., and Foster, D. J. Correcting the mythos of KL-regularization: Direct alignment without overoptimization via chi-squared preference optimization. In *International Conference on Learning Representations*, 2025.

Huang, S., Noukhovitch, M., Hosseini, A., Rasul, K., Wang, W., and Tunstall, L. The n+ implementation details of RLHF with PPO: A case study on TL;DR summarization. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=kHO2ZTa8e3.

Kaufmann, T., Weng, P., Bengs, V., and Hüllermeier, E. A survey of reinforcement learning from human feedback, 2024. URL https://arxiv.org/abs/2312.14925.

Lee, K., Liu, H., Ryu, M., Watkins, O., Du, Y., Boutilier, C., Abbeel, P., Ghavamzadeh, M., and Gu, S. S. Aligning text-to-image models using human feedback, 2023. URL https://arxiv.org/abs/2302.12192.

Li, X. L., Shrivastava, V., Li, S., Hashimoto, T., and Liang, P. Benchmarking and improving generator-validator consistency of language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Liang, Y., He, J., Li, G., Li, P., Klimovskiy, A., Carolan, N., Sun, J., Pont-Tuset, J., Young, S., Yang, F., Ke, J., Dvijotham, K. D., Collins, K. M., Luo, Y., Li, Y., Kohlhoff, K. J., Ramachandran, D., and Navalpakkam, V. Rich human feedback for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19401–19411, June 2024.

Lin, S., Hilton, J., and Evans, O. TruthfulQA: Measuring how models mimic human falsehoods. In Muresan, S., Nakov, P., and Villavicencio, A. (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL https://aclanthology.org/2022.acl-long.229/.

Lin, Y., Lin, H., Xiong, W., Diao, S., Liu, J., Zhang, J., Pan, R., Wang, H., Hu, W., Zhang, H., Dong, H., Pi, R., Zhao, H., Jiang, N., Ji, H., Yao, Y., and Zhang, T. Mitigating the alignment tax of RLHF. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 580–606, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.35. URL https://aclanthology.org/2024.emnlp-main.35/.

Liu, J., Zhou, Z., Liu, J., Bu, X., Yang, C., Zhong, H.-S., and Ouyang, W. Iterative length-regularized direct preference optimization: A case study on improving 7b language models to gpt-4 level, 2024. URL https://arxiv.org/abs/2406.11817.

Mao, Q., Lee, H.-Y., Tseng, H.-Y., Ma, S., and Yang, M.-H. Mode seeking generative adversarial networks for diverse image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1429–1437, 2019.

Meng, Y., Xia, M., and Chen, D. Simpo: Simple preference optimization with a reference-free reward. *Advances in Neural Information Processing Systems (NeurIPS)*, 37: 124198–124235, 2024.

Munos, R. Error bounds for approximate policy iteration. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2003.

Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., Hesse, C., Jain, S., Kosaraju, V., Saunders, W., Jiang, X., Cobbe, K., Eloundou, T., Krueger, G., Button, K., Knight, M., Chess, B., and Schulman, J. WebGPT: Browser-assisted question-answering with human feedback, 2022. URL https://arxiv.org/abs/2112.09332.

Nath, A., Jung, C., Seefried, E., and Krishnaswamy, N. Simultaneous reward distillation and preference learning: Get you a language model who can do both, 2025. URL https://arxiv.org/abs/2410.08458.

Noukhovitch, M., Lavoie, S., Strub, F., and Courville, A. C. Language model alignment with elastic reset. *Advances in Neural Information Processing Systems*, 36:3439–3461, 2023.

OpenAI. ChatGPT: Optimizing Language Models for Dialogue. https://openai.com/blog/chatgpt, 2022. Accessed: 2025-05-19.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Park, R., Rafailov, R., Ermon, S., and Finn, C. Disentangling length from quality in direct preference optimization. In *Findings of the Association for Computational Linguistics (ACL)*, 2024.

Qi, B., Li, P., Li, F., Gao, J., Zhang, K., and Zhou, B. Online dpo: Online direct preference optimization with fast-slow chasing, 2024. URL https://arxiv.org/abs/2406.05534.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:53728–53741, 2023.

Sakaguchi, K., Bras, R. L., Bhagavatula, C., and Choi, Y. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Schulman, J., 2020. URL http://joschu.net/blog/kl-approx.html.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Song, Y., Swamy, G., Singh, A., Bagnell, J., and Sun, W. The importance of online data: Understanding preference fine-tuning via coverage. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:12243–12270, 2024.

Stiennon, N., Ouyang, L., Wu, J., Ziegler, D., Lowe, R., Voss, C., Radford, A., Amodei, D., and Christiano, P. F. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Swamy, G., Choudhury, S., Sun, W., Wu, Z. S., and Bagnell, J. A. All roads lead to likelihood: The value of reinforcement learning in fine-tuning, 2025.

Tang, Y., Guo, Z. D., Zheng, Z., Calandriello, D., Munos, R., Rowland, M., Richemond, P. H., Valko, M., Avila Pires, B., and Piot, B. Generalized preference optimization: A unified approach to offline alignment. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Tunstall, L., Beeching, E. E., Lambert, N., Rajani, N., Rasul, K., Belkada, Y., Huang, S., Werra, L. V., Fourrier, C., Habib, N., Sarrazin, N., Sanseviero, O., Rush, A. M., and Wolf, T. Zephyr: Direct distillation of LM alignment. In *First Conference on Language Modeling*, 2024.

van de Geer, S. *Empirical Processes in M-Estimation*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2009.

Wallace, B., Dang, M., Rafailov, R., Zhou, L., Lou, A., Purushwalkam, S., Ermon, S., Xiong, C., Joty, S., and Naik, N. Diffusion model alignment using direct preference optimization, 2023. URL https://arxiv.org/abs/2311.12908.

West, P., Lu, X., Dziri, N., Brahman, F., Li, L., Hwang, J. D., Jiang, L., Fisher, J., Ravichander, A., Chandu, K., et al. The generative ai paradox:"what it can create, it may not understand". In *The Twelfth International Conference on Learning Representations*, 2024.

Xiao, T., Yuan, Y., Zhu, H., Li, M., and Honavar, V. G. Cal-dpo: Calibrated direct preference optimization for language model alignment. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

Xie, T., Cheng, C.-A., Jiang, N., Mineiro, P., and Agarwal, A. Bellman-consistent pessimism for offline reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Xie, T., Foster, D. J., Krishnamurthy, A., Rosset, C., Awadallah, A. H., and Rakhlin, A. Exploratory preference optimization: Harnessing implicit q*-approximation for sample-efficient RLHF. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=QYigQ6gXNw.

Xie, T., Foster, D. J., Krishnamurthy, A., Rosset, C., Awadallah, A. H., and Rakhlin, A. Exploratory preference optimization: Harnessing implicit q*-approximation for sample-efficient RLHF. In *International Conference on Learning Representations*, 2025b.

Xiong, W., Dong, H., Ye, C., Wang, Z., Zhong, H., Ji, H., Jiang, N., and Zhang, T. Iterative preference learning from human feedback: Bridging theory and practice for RLHF under KL-constraint. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Yuan, H., Yuan, Z., Tan, C., Wang, W., Huang, S., and Huang, F. Rrhf: Rank responses to align language models with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Yuan, L., Cui, G., Wang, H., Ding, N., Wang, X., Deng, J., Shan, B., Chen, H., Xie, R., Lin, Y., Liu, Z., Zhou, B., Peng, H., Liu, Z., and Sun, M. Advancing LLM reasoning generalists with preference trees. In *AI for Math Workshop @ ICML 2024*, 2024. URL https://openreview.net/forum?id=2Y1iiCqM5y.

Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., and Choi, Y. HellaSwag: Can a machine really finish your sentence? In Korhonen, A., Traum, D., and Màrquez,

L. (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Zhan, W., Huang, B., Huang, A., Jiang, N., and Lee, J. Offline reinforcement learning with realizability and single-policy concentrability. In *Proceedings of the Conference on Learning Theory (COLT)*, 2022.

Zhan, W., Uehara, M., Kallus, N., Lee, J. D., and Sun, W. Provable offline preference-based reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=tVMPfEGT2w.

Zhang, S., Yu, D., Sharma, H., Zhong, H., Liu, Z., Yang, Z., Wang, S., Hassan, H., and Wang, Z. Self-exploring language models: Active preference elicitation for online alignment, 2024. URL https://arxiv.org/abs/2405.19332.

Zhang, T. From ε-entropy to kl-entropy: Analysis of minimum information complexity density estimation. *The Annals of Statistics*, 34, 2007.

Zhang, Y., Wang, L., Fang, M., Du, Y., Huang, C., Wang, J., Lin, Q., Pechenizkiy, M., Zhang, D., Rajmohan, S., and Zhang, Q. Distill not only data but also rewards: Can smaller language models surpass larger ones?, 2025. URL https://arxiv.org/abs/2502.19557.

Zhao, Y., Joshi, R., Liu, T., Khalman, M., Saleh, M., and Liu, P. J. Slic-hf: Sequence likelihood calibration with human feedback. *CoRR*, abs/2305.10425, 2023a.

Zhao, Y., Khalman, M., Joshi, R., Narayan, S., Saleh, M., and Liu, P. J. Calibrating sequence likelihood improves conditional language generation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023b.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36: 46595–46623, 2023.

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

Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., Christiano, P., and Irving, G. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Appendix

Table of Contents

A. Related Work

Preference optimization with RL. A widely adopted paradigm in preference optimization is Reinforcement Learning from Human Feedback (RLHF). In this framework, one first trains a reward model–effectively serving as a classifier–on a preference dataset collected from human annotators, and subsequently leverages the learned reward model to run RL algorithms such as PPO (Christiano et al., 2017; Ziegler et al., 2019). RLHF and its variants have been instrumental in training prominent LLMs such as ChatGPT (OpenAI, 2022), and have achieved remarkable success across diverse applications such as text summarization, question answering, instruction following, and text-to-image generation (Stiennon et al., 2020; Nakano et al., 2022; Ouyang et al., 2022; Lee et al., 2023; Liang et al., 2024). We guide the interested reader to Kaufmann et al. (2024) for a recent dedicated survey on RLHF.

Without RL and without a reward model. Direct Preference Optimization (DPO) dispenses with an explicit reward model by treating the log-ratio of each preference pair as a training signal and directly training the policy with a single contrastive cross-entropy loss (Rafailov et al., 2023). Such an RL-free objective was shown to match PPO-based RLHF without requiring a reward model, value network, or on-policy sampling, and has led to variants such as distilled DPO (Tunstall et al., 2024), Cal-DPO (Xiao et al., 2024), diffusion DPO (Wallace et al., 2023), ΨPO (Azar et al., 2024), SLiC/SLiC-HF (Zhao et al., 2023b;a), GPO (Tang et al., 2024), χPO (Huang et al., 2025), R-DPO (Park et al., 2024), ODPO (Amini et al., 2024), SimPO (Meng et al., 2024), RRHF (Yuan et al., 2023), KTO (Ethayarajh et al., 2024), ORPO (Hong et al., 2024), and many more.

At the same time, such direct optimization from preference labels has been noted to underperform along some dimensions compared to conventional RLHF. One challenge stems from relying exclusively on an offline dataset, which can induce out-of-distribution responses. This is likely due to insufficient on-policy interaction during training (Song et al., 2024). Some hybrid approaches have been proposed to overcome this issue: iterative DPO performs iterative training with labeled online preferences (Liu et al., 2024), HyPO combines offline data for preference optimization and online data for KL regularization (Song et al., 2024), and online DPO utilizes fast and slow chasing to simulate competition (Qi et al., 2024).

Without RL but with a reward model. Another prominent method of preference optimization is reward distillation. This line of work aims to distill information on a reward model's preferences directly into the policy. As discussed in Section 4, the REBEL objective (Gao et al., 2024) regresses the log-ratio of the likelihoods of two responses on the reward difference using a simple squared-loss objective, which is repeated with batches of on-policy responses. Reward distillation from Fisch et al. (2025) can be seen as a simplified version of REBEL where we only use the responses from the preference dataset. DRDO learns a reward model and policy in one pass by jointly matching oracle rewards while also learning human preferences (Nath et al., 2025). Finally, Zhang et al. (2025) develops an LLM distillation pipeline to distill both data and rewards.

Theoretical analyses of preference optimization. Zhan et al. (2024) studies offline preference-based RL with an MLE-based reward model similar to ours, but only obtains guarantees in terms of maximizing the policy value. Xie et al. (2025a) proposes an exploratory version of DPO which is shown to achieves $\widetilde{O}(\sqrt{T})$ regret with a favorable coverage parameter. Zhang et al. (2024) proposes an online direct alignment algorithm which also attains $\widetilde{O}(\sqrt{T})$ regret. Xiong et al. (2024) derives regret bounds for online and offline versions of RLHF under a linearly parametrized reward model; see also Foster et al. (2025) for a theoretical analysis of RL with linear-softmax policies. Cen et al. (2025) introduces VPO, a value-regularized DPO-type objective for both online and offline RLHF, and also prove regret bounds under linear rewards. The χPO algorithm is shown to attain optimal sample complexity, also in terms of regret, under a weaker single-policy concentrability (Huang et al., 2025).

The work of Agarwal et al. (2025) is most relevant to our paper, especially PMLE (Section 3): they develop a theoretical analysis of offline RLHF variants that minimize DPO-type objectives, and show a forward KL bound w.r.t. an optimal policy π^*. However, this formulation is not due to a distribution learning viewpoint but merely a byproduct of their strong realizability assumption (Assumption 3.2). Moreover, their upper bound has a square-root dependence on the excess risk $\varepsilon = L(\pi) - L(\pi^*)$, which when applied to our framework yields a statistical rate of $1/\sqrt{n}$. In contrast, we obtain an improved rate of $1/n$ with a more careful analysis in Appendix B.

B. Theoretical Guarantees

B.1. Auxiliary Lemmas

We require the following basic results.

Lemma 8. *For all $a, b \in \mathbb{R}$ it holds that $|\sigma(a) - \sigma(b)| \geq \frac{1}{2}e^{-(|a|\vee|b|)}|a-b|$.*

Proof. Recall that $\sigma(z) = 1/(1+\exp(-z))$ is the logistic sigmoid. σ' is symmetric, so that

$$\sigma'(z) = \sigma'(|z|) = \frac{1}{1+e^{|z|}}\frac{1}{1+e^{-|z|}} \geq \frac{1}{1+e^{|z|}} \geq \frac{1}{2}e^{-|z|}.$$

It suffices to assume $b > a$ due to symmetry. Then,

$$\sigma(b) - \sigma(a) = \int_a^b \sigma'(z)\,\mathrm{d}z \geq \int_a^b \frac{1}{2}e^{-|z|}\,\mathrm{d}z \geq \frac{1}{2}e^{-(|a|\vee|b|)}(b-a)$$

as desired. □

The next two lemmas will allow us to convert between the expectation of the log ratio (i.e., KL divergence) and squared log ratio. Let us define the auxiliary function

$$\psi(z) := \frac{z-1-\ln z}{(\ln z)^2}.$$

Lemma 9. *For $r_{\max} > 1$, it holds for all $r \in (0, r_{\max}]$ that*

$$r - 1 - \ln r \leq (\frac{1}{2} \vee \psi(r_{\max}))(\ln r)^2 \leq \frac{r_{\max}}{(\ln r_{\max})^2}(\ln r)^2.$$

Proof. Define the auxiliary function

$$f(r) := \frac{1}{2}(\ln r)^2 - (r - 1 - \ln r).$$

For $r \in (0,1)$, it holds that $f(1) = 0$ and $f'(r) = \frac{\ln r - r + 1}{r} < 0$. Thus, $f(r) > 0$ which implies

$$r - 1 - \ln r \leq \frac{1}{2}(\ln r)^2.$$

For $r \in [1, r_{\max}]$, it is easily checked that ψ is nondecreasing on $(1, \infty)$ and thus

$$\frac{r - 1 - \ln r}{(\ln r)^2} = \psi(r) \le \psi(r_{\max}) \le \frac{r_{\max}}{(\ln r_{\max})^2},$$

as was to be shown. □

Lemma 10. *For $r_{\min} > 0$, it holds for all $r \in [r_{\min}, \infty)$ that*

$$r - 1 - \ln r \ge \frac{1}{e(\ln r_{\min}^{-1} \vee 1)} (\ln r)^2.$$

Proof. The function ψ defined in Lemma 9 extends to a nondecreasing continuous function on $(0, \infty)$ by setting $\psi(1) := \frac{1}{2}$. When $r \ge e^{-1}$, it follows that $\psi(r) \ge \psi(e^{-1}) = e^{-1}$.

When $r_{\min} \le r < e^{-1}$, we use the fact that $\ln r \le \frac{1}{1-e^{-1}}(r-1)$ to bound

$$\psi(r) \ge \frac{(1 - e^{-1}) \ln r - \ln r}{(\ln r)^2} = \frac{1}{e \ln r^{-1}} \ge \frac{1}{e \ln r_{\min}^{-1}}.$$

□

Lemma 11 (Symmetrization inequality)**.** *Let $D_n, \tilde{D}_n$ be two datasets of n i.i.d. samples, $C(\pi, D_n)$ be any functional of a policy π and dataset D_n, and $\hat{\pi} := \hat{\pi}(D_n)$ be any estimator computed from D_n. Then with probability $1 - \delta$, it holds that*

$$-\log \mathbb{E}_{\tilde{D}_n}[\exp(C(\hat{\pi}, \tilde{D}_n))] \le -C(\hat{\pi}, D_n) + \ln(|\Pi|/\delta).$$

Proof. This is shown for example in the proof of Theorem 6 in Foster & Krishnamurthy (2021). □

B.2. Proofs for Section 3

The following convergence bound for maximum likelihood estimators is mostly classical (Zhang, 2007; van de Geer, 2009); for completeness, we provide a brief proof following Theorem 6 of Foster & Krishnamurthy (2021).

Proposition 12. *Let $\hat{\pi} = \arg\min_{\pi \in \Pi} \mathcal{L}_{\mathsf{PMLE}}(\pi)$ with $\beta = 0$. Then, with probability at least $1 - \delta$,*

$$\mathbb{E}_{x \sim \mathcal{D}, a, b \sim \mu(x)} \left[(\mathbb{P}_{\hat{\pi}}(a \succ b \mid x) - \mathbb{P}_*(a \succ b \mid x))^2 \right] \le \frac{4 \ln(|\Pi|/\delta)}{n}.$$

Proof. Recall that each preference pair (x, a^+, a^-) is collected by first sampling a, b independently from $\mu(x)$ and setting $(a^+, a^-) = (a, b)$ with probability $\mathbb{P}_*(a \succ b \mid x)$. In other words, for the indicator $y = 1_{\{a^+ = a\}}$ such that $\mathbb{P}(y = 1) = \mathbb{P}_*(a \succ b \mid x)$, we can write

$$\mathcal{L}_{\mathsf{PMLE}}(\pi) = \frac{1}{n} \sum_{(x,a,b) \in D_n} -y \ln \mathbb{P}_\pi(a \succ b \mid x) - (1 - y) \ln \mathbb{P}_\pi(b \succ a \mid x),$$

where we have abused notation to write the sum over (x, a, b) corresponding to each (x, a^+, a^-) as a sum over $(x, a, b) \in D_n$. Define the quantity

$$\begin{aligned} C(\pi, D_n) &= \frac{1}{2} \sum_{(x,a,b) \in D_n} y \ln \frac{\mathbb{P}_\pi(a \succ b \mid x)}{\mathbb{P}_*(a \succ b \mid x)} + (1 - y) \ln \frac{\mathbb{P}_\pi(b \succ a \mid x)}{\mathbb{P}_*(b \succ a \mid x)} \\ &= \frac{n}{2} (\mathcal{L}_{\mathsf{PMLE}}(\pi^*) - \mathcal{L}_{\mathsf{PMLE}}(\pi)) \end{aligned}$$

and $\hat{\pi}$ as the minimizer of $\mathcal{L}_{\mathsf{PMLE}}(\pi)$ for $\pi \in \Pi$. It follows from Lemma 11 that

$$-\log \mathbb{E}_{\tilde{D}_n}[\exp(C(\hat{\pi}, \tilde{D}_n))] \le -C(\hat{\pi}, D_n) + \ln(|\Pi|/\delta) \le \ln(|\Pi|/\delta)$$

and

$$\begin{aligned}
&-\log \mathbb{E}_{\tilde{D}_n}[\exp(C(\hat{\pi}, \tilde{D}_n))] \\
&= -n \log \mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)} \mathbb{E}_{y|a,b,x} \left[\left(\frac{\mathbb{P}_\pi(a \succ b \mid x)}{\mathbb{P}_*(a \succ b \mid x)}\right)^{y/2} \left(\frac{\mathbb{P}_\pi(b \succ a \mid x)}{\mathbb{P}_*(b \succ a \mid x)}\right)^{(1-y)/2} \right] \\
&= -n \log \mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)} \left[\sqrt{\mathbb{P}_\pi(a \succ b \mid x)\,\mathbb{P}_*(a \succ b \mid x)} + \sqrt{\mathbb{P}_\pi(b \succ a \mid x)\,\mathbb{P}_*(b \succ a \mid x)} \right].
\end{aligned}$$

Writing $p_\pi = \mathbb{P}_\pi(a \succ b \mid x)$ and $p_* = \mathbb{P}_*(a \succ b \mid x)$ for simplicity, we further have

$$\begin{aligned}
-\log \mathbb{E}\left[\sqrt{p_\pi p_*} + \sqrt{(1-p_\pi)(1-p_*)}\right] &\geq 1 - \mathbb{E}\left[\sqrt{p_\pi p_*} + \sqrt{(1-p_\pi)(1-p_*)}\right] \\
&= \mathbb{E}\left[\frac{1}{2}(\sqrt{p_\pi} - \sqrt{p_*})^2 + \frac{1}{2}(\sqrt{1-p_\pi} - \sqrt{1-p_*})^2\right] \\
&= \mathbb{E}\left[\frac{(p_\pi - p_*)^2}{2(\sqrt{p_\pi} + \sqrt{p_*})^2} + \frac{(p_\pi - p_*)^2}{2(\sqrt{1-p_\pi} + \sqrt{1-p_*})^2}\right] \\
&\geq \frac{1}{4}\,\mathbb{E}\left[(p_\pi - p_*)^2\right],
\end{aligned}$$

which yields the desired bound. □

Proof of Theorem 4. Our proof is partly inspired by Agarwal et al. (2025, proof of Theorem 3.6). The key difference is that their theorem relies on an assumption that the *population* loss of $\hat{\pi}$ is not too far away from that of π^*, which is rather strong. In contrast, our theorem provides an end-to-end guarantee. Furthermore, naively applying their theorem would result in an $1/\sqrt{n}$ rate rather than $1/n$. We obtain an improvement by applying Schulman's trick (Schulman, 2020) followed by Lemma 9. We elaborate more this later in Remark 13.

Using Lemma 8 with the fact

$$\left|\gamma \ln \frac{\pi(a \mid x)}{\pi(b \mid x)}\right| = \left|\bar{R}(x,a) - \bar{R}(x,b)\right| \leq 2\gamma R\,,$$

we can lower bound

$$\begin{aligned}
&\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\mathbb{P}_{\hat{\pi}}(a \succ b \mid x) - \mathbb{P}_*(a \succ b \mid x))^2\right] \\
&\geq \frac{e^{-4\gamma R}}{4}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[\left(\gamma \ln \frac{\hat{\pi}(a \mid x)}{\hat{\pi}(b \mid x)} - \gamma \ln \frac{\pi^*(a \mid x)}{\pi^*(b \mid x)}\right)^2\right] \\
&= \frac{e^{-4\gamma R}}{4}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\Delta\bar{R}_{\hat{\pi}}(x,a) - \Delta\bar{R}_{\hat{\pi}}(x,b))^2\right] \\
&= \frac{e^{-4\gamma R}}{2}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right]
\end{aligned}$$

where the last inequality is due to $\mathbb{E}[(X-Y)^2] = 2\,\mathbb{E}[(X - \mathbb{E}[X])^2]$ when X and Y and i.i.d. Thus, using Proposition 12, the difference in centered reward satisfies

$$\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right] \leq 8e^{4\gamma R} \cdot \frac{\ln(|\Pi|/\delta)}{n}. \tag{18}$$

Define the normalizing factor

$$Z_\pi(x) := \sum_{a\in\mathcal{A}} \exp\left(\frac{1}{\gamma}\bar{R}_\pi(x,a)\right) = \exp\left(-\frac{1}{\gamma}\,\mathbb{E}_{a\sim\mu(x)}[R_\pi(x,a) \mid x]\right), \quad Z_* := Z_{\pi^*}$$

so that $\pi(a \mid x) = Z_\pi(x)^{-1}\exp(\gamma^{-1}\bar{R}_\pi(x,a))$. Due to Assumption 2, for all $\pi \in \Pi, x \in \mathcal{X}$ it holds that $|\mathcal{A}|e^{-R} \leq Z_\pi(x) \leq |\mathcal{A}|e^R$, so that

$$0 < \frac{\hat{\pi}(a \mid x)}{\pi^*(a \mid x)} = \frac{Z_*(x)}{Z_{\hat{\pi}}(x)} \exp\left(\frac{1}{\gamma}\Delta\bar{R}_{\hat{\pi}}(x,a)\right) \leq e^{4R}. \tag{19}$$

Then, we bound the KL divergence between $\pi^*, \hat{\pi}$ using Schulman's trick (Schulman, 2020) followed by Lemma 9:

$$\begin{aligned}\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right] &= \mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\ln\frac{\pi^*(a\mid x)}{\hat{\pi}(a\mid x)}\right] \\ &= \mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)} - 1 - \ln\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)}\right] \\ &\le (\frac{1}{2}\vee\psi(e^{4R}))\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\ln\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right]. \end{aligned} \tag{20}$$

Extracting the normalization constants, we further have that

$$\begin{aligned}&\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\ln\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right] \\ &\le \mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[2\left(\ln\frac{\hat{\pi}(a\mid x)Z_{\hat{\pi}}(x)}{\pi^*(a\mid x)Z_*(x)}\right)^2 + 2\left(\ln\frac{Z_*(x)}{Z_{\hat{\pi}}(x)}\right)^2\right] \\ &= \frac{2}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right] + 2\,\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\ln\frac{Z_*(x)}{Z_{\hat{\pi}}(x)}\right)^2\right].\end{aligned}$$

Using Definition 3 and (18), the first term is bounded as

$$\frac{2}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right] \le \frac{2C_\Pi}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\mu(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right] \le \frac{16C_\Pi e^{4\gamma R}}{\gamma^2}\cdot\frac{\ln(|\Pi|/\delta)}{n}.$$

For the second term, we first characterize an upper and lower bound on $\ln\frac{Z_{\hat{\pi}}(x)}{Z_*(x)}$. Using

$$1 = \mathbb{E}_{a\sim\pi^*(x)}\left[\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)}\right] = \frac{Z_*(x)}{Z_{\hat{\pi}}(x)}\,\mathbb{E}_{a\sim\pi^*(x)}\left[\exp\left(\frac{1}{\gamma}\Delta\bar{R}_{\hat{\pi}}(x,a)\right)\right],$$

we have

$$\ln\frac{Z_{\hat{\pi}}(x)}{Z_*(x)} = \ln\mathbb{E}_{a\sim\pi^*(x)}\left[\exp\left(\frac{1}{\gamma}\Delta\bar{R}_{\hat{\pi}}(x,a)\right)\right] \ge \frac{1}{\gamma}\,\mathbb{E}_{a\sim\pi^*(x)}[\Delta\bar{R}_{\hat{\pi}}(x,a)]$$

where the last inequality is by Jensen's inequality. Moreover, using the inequality $e^x \le 1 + x + \frac{e^A}{2}x^2$ valid for all $x \in (-\infty, A]$, we have

$$\begin{aligned}\ln\frac{Z_{\hat{\pi}}(x)}{Z_*(x)} &= \ln\mathbb{E}_{a\sim\pi^*(x)}\left[\exp\left(\frac{1}{\gamma}\Delta\bar{R}_{\hat{\pi}}(x,a)\right)\right] \\ &\le \mathbb{E}_{a\sim\pi^*(x)}\left[\exp\left(\frac{1}{\gamma}\Delta\bar{R}_{\hat{\pi}}(x,a)\right)\right] - 1 \\ &\le \frac{1}{\gamma}\,\mathbb{E}_{a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)\right] + \frac{e^{2R}}{2\gamma^2}\,\mathbb{E}_{a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right].\end{aligned}$$

Thus, we have

$$\left|\ln\frac{Z_{\hat{\pi}}(x)}{Z_*(x)}\right| \le \left|\frac{1}{\gamma}\,\mathbb{E}_{a\sim\pi^*(x)}[\Delta\bar{R}_{\hat{\pi}}(x,a)]\right| + \frac{e^{2R}}{2\gamma^2}\,\mathbb{E}_{a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right],$$

which implies, using $\forall x, y \in \mathbb{R}, (x+y)^2 \le 2x^2 + 2y^2$,

$$\begin{aligned}&\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\ln\frac{Z_*(x)}{Z_{\hat{\pi}}(x)}\right)^2\right] \\ &\le \frac{2}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\mathbb{E}_{a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)\right]\right)^2\right] + \frac{e^{4R}}{2\gamma^4}\,\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\mathbb{E}_{a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right]\right)^2\right]\end{aligned}$$

$$\le \frac{2R^2 e^{4R} + 2}{\gamma^2} \mathbb{E}_{x\sim\mathcal{D}, a\sim\pi^*(x)} \left[\Delta \bar{R}_{\hat{\pi}}(x,a)^2\right] \qquad \text{(Jensen's inequality; Assumption 2)}$$

$$\le \frac{16 C_\Pi (R^2 e^{4R} + 1) e^{4\gamma R}}{\gamma^2} \cdot \frac{\ln(|\Pi|/\delta)}{n} . \qquad \text{(by (18))}$$

Putting everything together, we conclude:

$$\begin{aligned}
&\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right] \\
&\le (\frac{1}{2} \vee \psi(e^{4R}))\left(\frac{16 C_\Pi e^{4\gamma R}}{\gamma^2} \cdot \frac{\ln(|\Pi|/\delta)}{n} + \frac{32 C_\Pi (R^2 e^{4R}+1) e^{4\gamma R}}{\gamma^2} \cdot \frac{\ln(|\Pi|/\delta)}{n}\right) \\
&= (\frac{1}{2} \vee \psi(e^{4R}))\frac{16(2R^2 e^{4R}+3) C_\Pi e^{4\gamma R}}{\gamma^2} \cdot \frac{\ln(|\Pi|/\delta)}{n}.
\end{aligned}$$

We remark that by Lemma 9, the $\frac{1}{2} \vee \psi(e^{4R})$ term is further bounded above by $\frac{e^{4R}}{16R^2}$.

Remark 13. One of our key novelty is (20). In Agarwal et al. (2025), they use Cauchy-Schwarz to derive

$$\mathbb{E}_{x\sim\mathcal{D}, a\sim\pi^*(x)}\left[\ln \frac{\pi^*(a \mid x)}{\hat{\pi}(a \mid x)}\right] \le \sqrt{\mathbb{E}_{x\sim\mathcal{D}, a\sim\pi^*(x)}\left[\left(\ln \frac{\pi^*(a \mid x)}{\hat{\pi}(a \mid x)}\right)^2\right]} ,$$

which introduces an extra square root compared to our derivation. Following their derivation naively would lead to a $1/\sqrt{n}$ rate instead of $1/n$.

□

B.3. Proofs for Section 4

Proof of Theorem 6. Up to constants, our distillation objective is equivalent to minimizing

$$\frac{1}{n} \sum_{(x,a^+,a^-)\in D_n} \mathrm{KL}\left(\mathrm{Bern}(\mathbb{P}_{\widetilde{\pi}}(a^+ \succ a^- \mid x)) \| \mathrm{Bern}(\mathbb{P}_{\pi}(a^+ \succ a^- \mid x))\right) ,$$

which can achieve zero loss since $\widetilde{\pi} \in \mathcal{P}_\gamma(\mathcal{R}) \subseteq \Pi$ is a valid solution. Thus, the solution $\hat{\pi}$ must satisfy

$$\mathbb{P}_{\widetilde{\pi}}(a \succ b \mid x) = \mathbb{P}_{\hat{\pi}}(a \succ b \mid x), \quad \forall (x, a, b) \in D_n$$

(recall that we use (a, b) to denote the independent unlabeled responses). Defining the set

$$\mathcal{K} := \left\{(\pi_1, \pi_2) \in \mathcal{P}_\gamma(\mathcal{R}) \times \Pi : \mathbb{E}_{x\sim\mathcal{D}, a,b\sim\mu(x)}\left[| \mathbb{P}_{\pi_1}(a \succ b \mid x) - \mathbb{P}_{\pi_2}(a \succ b \mid x)|\right] > \varepsilon\right\},$$

it follows that

$$\begin{aligned}
\mathbb{P}\left((\widetilde{\pi}, \hat{\pi}) \in \mathcal{K}\right) &= \sum_{(\pi_1,\pi_2)\in\mathcal{K}} \mathbb{P}(\widetilde{\pi} = \pi_1, \hat{\pi} = \pi_2) \\
&\le \sum_{(\pi_1,\pi_2)\in\mathcal{K}} \mathbb{P}\left(\mathbb{P}_{\pi_1}(a \succ b \mid x) = \mathbb{P}_{\pi_2}(a \succ b \mid x),\ \forall (x, a, b) \in D_n\right) \\
&= \sum_{(\pi_1,\pi_2)\in\mathcal{K}} \mathbb{P}\left(\mathbb{P}_{\pi_1}(a \succ b \mid x) = \mathbb{P}_{\pi_2}(a \succ b \mid x)\right)^n \\
&\le \sum_{(\pi_1,\pi_2)\in\mathcal{K}} \left(1 - \mathbb{E}\left[| \mathbb{P}_{\pi_1}(a \succ b \mid x) - \mathbb{P}_{\pi_2}(a \succ b \mid x)|\right]\right)^n \\
&\le \sum_{(\pi_1,\pi_2)\in\mathcal{K}} (1-\varepsilon)^n \\
&\le |\mathcal{K}|^2 \exp(-\varepsilon n).
\end{aligned}$$

Therefore $\mathbb{P}\left((\widetilde{\pi},\hat{\pi})\in\mathcal{K}\right)\leq|\Pi|^2\exp(-\varepsilon n)$, i.e.,

$$\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[\left|\,\mathbb{P}_{\widetilde{\pi}}(a\succ b\mid x)-\mathbb{P}_{\hat{\pi}}(a\succ b\mid x)\right|\right]\leq\frac{2\ln(|\Pi|/\delta)}{n}$$

with probability at least $1-\delta$, and so

$$\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\mathbb{P}_{\widetilde{\pi}}(a\succ b\mid x)-\mathbb{P}_{\hat{\pi}}(a\succ b\mid x))^2\right]\leq\frac{2\ln(|\Pi|/\delta)}{n}$$

as well. On the other hand, applying Proposition 12 to $\mathcal{P}_\gamma(\mathcal{R})$, we have

$$\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\mathbb{P}_{\widetilde{\pi}}(a\succ b\mid x)-\mathbb{P}_{*}(a\succ b\mid x))^2\right]\leq\frac{4\ln(|\mathcal{R}|/\delta)}{n}$$

with probability at least $1-\delta$. Hence by a union bound, it holds that, with probability at least $1-\delta$.

$$\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\mathbb{P}_{\hat{\pi}}(a\succ b\mid x)-\mathbb{P}_{*}(a\succ b\mid x))^2\right]\leq\frac{4\ln(2|\Pi|/\delta)+8\ln(2|\mathcal{R}|/\delta)}{n}\;.$$

Furthermore, by Lemma 8 it holds that

$$\begin{aligned}\left|\mathbb{P}_{\hat{\pi}}(a\succ b\mid x)-\mathbb{P}_{*}(a\succ b\mid x)\right| &= \left|\sigma\left(\gamma\ln\frac{\hat{\pi}(a\mid x)}{\hat{\pi}(b\mid x)}\right)-\sigma\left(\gamma\ln\frac{\pi^*(a\mid x)}{\pi^*(b\mid x)}\right)\right| \\ &\geq \frac{\gamma e^{-2\gamma R}}{2}\left|\ln\frac{\hat{\pi}(a\mid x)}{\hat{\pi}(b\mid x)}-\ln\frac{\pi^*(a\mid x)}{\pi^*(b\mid x)}\right| \\ &= \frac{e^{-2\gamma R}}{2}\left|\Delta\bar{R}_{\hat{\pi}}(x,a)-\Delta\bar{R}_{\hat{\pi}}(x,b)\right|,\end{aligned}$$

which implies that

$$\begin{aligned}\mathbb{E}_{x\sim\mathcal{D},a\sim\mu(x)}\left[(\Delta\bar{R}_{\hat{\pi}}(x,a))^2\right] &= \frac{1}{2}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\Delta\bar{R}_{\hat{\pi}}(x,a)-\Delta\bar{R}_{\hat{\pi}}(x,b))^2\right] \\ &\leq 4e^{4\gamma R}\cdot\frac{4\ln(2|\Pi|/\delta)+8\ln(2|\mathcal{R}|/\delta)}{n}. \end{aligned}\tag{21}$$

Finally as in the proof of Theorem 4, we combine the bounds

$$\begin{aligned}&\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\ln\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right] \\ &\leq\frac{2}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right]+2\,\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\ln\frac{Z_*(x)}{Z_{\hat{\pi}}(x)}\right)^2\right]\end{aligned}$$

and

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\ln\frac{Z_*(x)}{Z_{\hat{\pi}}(x)}\right)^2\right]\leq\frac{2R^2e^{4R}+2}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right]$$

along with (21) to conclude that

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right]\leq(\frac{1}{2}\vee\psi(e^{4R}))\frac{32(2R^2e^{4R}+3)C_\Pi e^{4\gamma R}}{\gamma^2}\cdot\frac{\ln(2|\Pi|/\delta)+2\ln(2|\mathcal{R}|/\delta)}{n}.$$

□

B.4. Proofs for Section 5

Proof of Theorem 7. The first step of the argument is similar to the proof of Theorem 6. Up to constants, the reverse KL objective is equivalent to

$$\hat{\pi} = \arg\min_{\pi\in\Pi} \frac{1}{n} \sum_{(x,\cdot,\cdot)\in D_n} \mathrm{KL}(\pi(x)\|\widetilde{\pi}(x)),$$

which achieves zero loss due to Assumption 5. Defining the set

$$K := \left\{(\pi_1,\pi_2) \in \Pi \times \mathcal{P}_\gamma(\mathcal{R}) : \mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi_1(x)\|\pi_2(x))\right] > \varepsilon\right\},$$

it follows that

$$\begin{aligned}\mathbb{P}\left(\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\hat{\pi}(x)\|\widetilde{\pi}(x))\right] > \varepsilon\right) &= \sum_{(\pi_1,\pi_2)\in K} \mathbb{P}(\hat{\pi}=\pi_1, \widetilde{\pi}=\pi_2) \\ &\le \sum_{(\pi_1,\pi_2)\in K} \mathbb{P}(\pi_1(x)=\pi_2(x)),\ \forall x\in D_n) \\ &= \sum_{(\pi_1,\pi_2)\in K} \mathbb{P}_{x\sim\mathcal{D}}(\pi_1(x)=\pi_2(x))^n.\end{aligned}$$

Note that for any $(\pi_1,\pi_2)\in K$, it holds that $\pi_1(a\mid x)/\pi_2(a\mid x) \le e^{4R}$ as in the proof of Theorem 4, so that

$$\mathrm{KL}(\pi_1(x)\|\pi_2(x)) \le \sup_{a\in\mathcal{A}} \ln\frac{\pi_1(a\mid x)}{\pi_2(a\mid x)} \le 4R\cdot 1_{\{\pi_1(x)\ne\pi_2(x)\}}, \quad \forall x\in\mathcal{X}.$$

This implies

$$\begin{aligned}\mathbb{P}_{x\sim\mathcal{D}}(\pi_1(x)=\pi_2(x)) &= 1-\mathbb{E}_{x\sim\mathcal{D}}[1_{\{\pi_1(x)\ne\pi_2(x)\}}] \\ &\le 1-\mathbb{E}_{x\sim\mathcal{D}}\left[\frac{\mathrm{KL}(\pi_1(x)\|\pi_2(x))}{4R}\right] \\ &\le 1-\frac{\varepsilon}{4R},\end{aligned}$$

and hence

$$\mathbb{P}\left(\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\hat{\pi}(x)\|\widetilde{\pi}(x))\right] > \varepsilon\right) \le |K|\left(1-\frac{\varepsilon}{4R}\right)^n \le |\Pi|^2\exp\left(-\frac{\varepsilon n}{4R}\right).$$

We now convert the reverse KL divergence involving $\widetilde{\pi}$, to that of π^*. By Lemma 9 it again holds

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\hat{\pi}(x)\|\pi^*(x))\right] \le (\frac{1}{2}\vee\psi(e^{4R}))\,\mathbb{E}_{x\sim\mathcal{D},a\sim\hat{\pi}(x)}\left[\left(\ln\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right].$$

We then bound each term of

$$\begin{aligned}&\mathbb{E}_{x\sim\mathcal{D},a\sim\hat{\pi}(x)}\left[\left(\ln\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right] \\ &\le 2\,\mathbb{E}_{x\sim\mathcal{D},a\sim\hat{\pi}(x)}\left[\left(\ln\frac{\hat{\pi}(a\mid x)}{\widetilde{\pi}(a\mid x)}\right)^2\right] + 2\,\mathbb{E}_{x\sim\mathcal{D},a\sim\hat{\pi}(x)}\left[\left(\ln\frac{\widetilde{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right].\end{aligned}$$

The first term can be bounded using Lemma 10:

$$\begin{aligned}\mathbb{E}_{x\sim\mathcal{D},a\sim\hat{\pi}(x)}\left[\left(\ln\frac{\hat{\pi}(a\mid x)}{\widetilde{\pi}(a\mid x)}\right)^2\right] &\le e(4R\vee 1)\,\mathbb{E}_{x\sim\mathcal{D},a\sim\hat{\pi}(x)}\left[\frac{\widetilde{\pi}(a\mid x)}{\hat{\pi}(a\mid x)} - 1 - \ln\frac{\widetilde{\pi}(a\mid x)}{\hat{\pi}(a\mid x)}\right] \\ &= e(4R\vee 1)\,\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\hat{\pi}(x)\|\widetilde{\pi}(x))\right].\end{aligned}$$

Furthermore, for the second term, using the same argument as (19),

$$\mathbb{E}_{x\sim\mathcal{D},a\sim\hat{\pi}(x)}\left[\left(\ln\frac{\widetilde{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right] \le e^{4R}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\ln\frac{\widetilde{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right]$$
$$\le \frac{16(2R^2e^{4R}+3)C_{\mathcal{R}}e^{(4\gamma+4)R}}{\gamma^2}\cdot\frac{\ln(|\mathcal{R}|/\delta)}{n},$$

where the last inequality is repeating the derivation of Theorem 4 for $\mathcal{P}_\gamma(\mathcal{R})$ instead of Π. Putting everything together, we conclude:

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\hat{\pi}(x)\|\pi^*(x))\right]$$
$$\le (\frac{1}{2}\vee\psi(e^{4R}))\left(16eR(4R\vee 1)\cdot\frac{\ln(2|\Pi|/\delta)}{n}+\frac{32(2R^2e^{4R}+3)C_{\mathcal{R}}e^{(4\gamma+4)R}}{\gamma^2}\cdot\frac{\ln(|\mathcal{R}|/\delta)}{n}\right).$$

□

Proposition 14. *It holds that*

$$\mathbb{E}_x[\mathrm{KL}(\hat{\pi}(x)\|\pi^*(x))] \le \frac{(4R\vee 1)e^{8R+1}}{R^2}\,\mathbb{E}_x[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))].$$

Proof of Proposition 14. Denote the ratio $r = \frac{\pi^*(a|x)}{\hat{\pi}(a|x)}$ for brevity. Using the same argument as (19), we have $r \in [e^{-4R}, e^{4R}]$. Then, we have

$$r-1-\ln r \overset{(a)}{\le} \frac{e^{4R}}{R^2}(\ln r)^2 = \frac{e^{4R}}{R^2}\left(\ln\frac{1}{r}\right)^2 \overset{(b)}{\le} \frac{e^{4R}}{R^2}e(4R\vee 1)\left(\frac{1}{r}-1-\ln\frac{1}{r}\right),$$

where (a) is by Lemma 9 and (b) is by Lemma 10 with r replaced by $1/r$. Thus,

$$\begin{aligned}\mathrm{KL}(\hat{\pi}(x)\|\pi^*(x)) &= \mathbb{E}_{a\sim\hat{\pi}(x)}[r-1-\ln r]\\ &\le \frac{e^{4R}}{R^2}e(4R\vee 1)\,\mathbb{E}_{a\sim\hat{\pi}(x)}\left[\frac{1}{r}-1-\ln\frac{1}{r}\right]\\ &\le \frac{e^{4R}}{R^2}e(4R\vee 1)e^{4R}\,\mathbb{E}_{a\sim\pi^*(x)}\left[\frac{1}{r}-1-\ln\frac{1}{r}\right]\\ &= \frac{e^{4R}}{R^2}e(4R\vee 1)e^{4R}\,\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x)).\end{aligned}$$

□

C. Experiment Details

In Section C.1, we provide implementation details on model card, hyperparameters, and compute resources on training and evaluating on the TL;DR dataset. In Section C.2, we provide details on our general chat experiments from Section 6 and also show additional results on MT-Bench and AlpacaEval 2.0.

C.1. TL;DR Summarization

Dataset. We use the TL;DR dataset that is widely used in related literature (Gao et al., 2024; Song et al., 2024; Huang et al., 2024), publicly available[4]. We summarize the dataset statistics in Table 4. Note that DPO and PMLE are trained on the preference dataset which has preference labels, and other algorithms evaluate the policy based on human references since they utilize the online responses.

Table 4. TL;DR dataset statistics.

Dataset	Train	Valid	Test
Human Reference	117K	64.5K	6.55K
Preference	92.9K	83.8K	N/A

Models. We use Pythia-1.4B[5] and Pythia-2.8B[6] (Biderman et al., 2023) as our pretrained models, using maximum context length 512 and maximum generation length up to 53 tokens. In order for training efficiency, we use LoRA (Low-Rank Adapter, Hu et al. (2022)) for alignment after full-parameter tuning the SFT model.

Implementations. We implement our three approaches (PMLE, reverse KL, preference distillation) on the top of a publicly available codebase[7]; preference distillation in particular is based on another publicly available code baseline[8]. For PMLE (Section 3), we implement the KL regularizer in (5) using the online responses described in Song et al. (2024). The DPO baseline takes about 3 hours and PMLE requires about 6 hours with 4 A100 40GB GPUs. Also, reverse KL and preference distillation, as well as their corresponding baselines RLHF and REBEL, takes about 2.5 days with 4 A100 40GB GPUs. Lastly, the win-rate is judged by GPT-4 using the `gpt-4` checkpoint (as of May 23rd, 2025).

Algorithm 1 Preference Distillation (Sec. 4)

1: **Input:** (Learned) reward $\hat{R}$, policy class Π, data distribution μ, learning rate η, training dataset of prompts $\{x_i\}_{i=1}^n$.
2: **Initialize:**
3: **for** $t = 0, 1, \ldots, T-1$ **do**
4: Sample two responses from $a_1, a_2 \sim \mu(\cdot \mid x)$ for a given prompt $x \sim \mathcal{D}$ for all $x \in \{x_i\}_{i=1}^n$.
5: Compute the probabilities with preference simulator by

$$\mathbb{P}_{\tilde{\pi}}(a_1 \succ a_2 \mid x) := \sigma\big(\hat{R}(x, a_1) - \hat{R}(x, a_2)\big)$$
$$\mathbb{P}_{\tilde{\pi}}(a_2 \succ a_1 \mid x) := \sigma\big(\hat{R}(x, a_2) - \hat{R}(x, a_1)\big)$$

6: Compute the preference distillation loss $\mathcal{L}_{\mathsf{Distill},\beta}(\pi)$ using (11) and (12).
7: $\pi_{t+1} \leftarrow \pi_t - \eta \nabla \mathcal{L}_{\mathsf{Distill},\beta}(\pi_t)$
8: **end for**

Pseudocode. Since the implementations of PMLE and reverse KL are straightforward from the corresponding DPO and RLHF baseline, we present the pseudocode for preference distillation for better understanding. As noted in Gao et al. (2024), the base distribution μ can also be π_t in our pseudocode (Algorithm 1). Following the baseline code implementations of REBEL, we also sample online responses from the distribution π_t.

[4] https://github.com/openai/summarize-from-feedback
[5] https://huggingface.co/EleutherAI/pythia-1.4b-deduped
[6] https://huggingface.co/EleutherAI/pythia-2.8b-deduped
[7] https://github.com/vwxyzjn/summarize_from_feedback_details
[8] https://github.com/ZhaolinGao/REBEL

Table 5. Hyperparameter configurations for TL;DR summarization tasks.

Setting	Parameters	
SFT & RM	batch size: 64 learning rate: 3e-6	schedule: cosine decay train epochs: 1
DPO	batch size: 64 learning rate: 3e-6 schedule: linear decay	train epochs: 1 β: 0.05
PMLE	batch size: 512 learning rate: 1e-6 schedule: linear decay	train epochs: 1 β: 1e-5 γ: 1e-2
REBEL	batch size: 512 learning rate: 3e-6 schedule: linear decay total episodes: 1e6	num epochs: 4 η: 1.0 kl coefficient: 0.05
Preference Distillation	batch size: 512 learning rate: 3e-6 schedule: linear decay total episodes: 1e6	num epochs: 4 γ: 0.1 kl coefficient: 0.05
RLHF (via PPO)	batch size: 512 learning rate: 3e-6 schedule: linear decay total episodes: 1e6 num epochs: 4	discount factor: 1 gae λ: 0.95 clip ratio: 0.2 value function coeff: 0.1 kl coefficient: 0.05
Reverse KL (Sec. 5)	batch size: 512 learning rate: 3e-6 schedule: linear decay	total episodes: 1e6 kl coefficient: 0.05 entropy coefficient: 0.01 or 0.001
LoRA Adapter Config	r: 1024 α: 2048	dropout: 0.0 bias: False
Generation Config	sampling: true top k: 0.0 top p: 1.0	min length: 53 max new tokens: 53 temperature: 0.01 (for DPO and PMLE) or 0.7 (others)

Hyperparameters. We adopt almost the same hyperparameters used in several studies (Huang et al., 2024; Gao et al., 2024; Song et al., 2024). For completeness, we summarize the hyperparameters used in our experiments in Table 5. Note that Gao et al. (2024) trains only a single epoch for RLHF and REBEL, but we cannot reproduce their results with just one epoch. Rather, following the implementation details (Huang et al., 2024), we consider the total episodes 10^6 which corresponds to roughly about 8.5 epochs. In this setting, we could reproduce the baseline results or obtain better results. Hence, reverse KL and preference distillation are also evaluated under this setup.

C.2. General Chat

Dataset and Models. In this experiment, we use the UltraFeedBack dataset (Cui et al., 2023), which is used in various baselines. We use LLaMA-3-8B-Instruct[9] as our base model and Eurus-RM-7B[10] as the reward model. One can use other public base models and reward models as well. Following Gao et al. (2024), we apply a length penalty Γ (for responses exceeding maximum response length) with a KL regularizer to the reward function: $r(x,a) = \hat{R}(x,y) - \zeta\big(\ln \pi_{\theta_t}(a \mid x) - \ln \pi_0(a \mid x)\big)$.

[9] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct
[10] https://huggingface.co/openbmb/Eurus-RM-7b

Implementations. As in the TL;DR experiments, our implementation for preference distillation is based on Gao et al. (2024), which is publicly available. The total training time for LLaMA-3-8B-Instruct takes around 7 days on 4 A100 GPUs. In order to evaluate the model quality using MT-bench (Zheng et al., 2023) and AlpacaEval 2.0 (Dubois et al., 2024), we use the public GitHub repositories.[11][12]

Hyperparameters. Similar to TL;DR experiments, we adopt the hyperparemeter configurations of Gao et al. (2024); for completeness, the full specification is presented in Table 6. Note that, due to the scale of experiments, we choose the best hyperparameter γ used in TL;DR experiments. Consequently, the reported performance of preference distillation might be conservative, as more fine-grained hyperparameter tuning could yield further performance gains.

Table 6. Hyperparameter configurations for general chat experiments.

Setting	Parameters	
REBEL	batch size: 32 learning rate: 1e-7 schedule: linear decay train epochs: 1 num epochs: 4 η: 1.0 ζ: 0.5 $\Gamma : -4$	
Preference distillation	batch size: 32 learning rate: 1e-7 schedule: linear decay train epochs: 1 num epochs: 4 β: 0.05 γ: 0.1 Γ: -4	
Generation Config	sampling: true top k: 0.0 top p: 1.0	min length: 1024 max new tokens: 1024 temperature: 0.5

Table 7. Quality analysis for general chat experiments.

Models	MT-Bench Average	AlpacaEval 2.0 LC Win-rate	AlpacaEval 2.0 Win-rate
LLaMA-3-8B-Instruct	8.10	30.50	30.50
LLaMA-3-8B-REBEL	**7.89**	31.25	31.68
LLaMA-3-8B-Distill	7.79	**32.59**	**33.04**

Additional Results. In addition to alignment tax in Section 6, we also include the general benchmark for evaluating LLMs: (i) MT-Bench (Zheng et al., 2023) and (ii) AlpacaEval 2.0 (Dubois et al., 2024) for quality analysis in Table 7. In Table 7, the MT-bench score of REBEL is slightly higher than that of preference distillation, but are very similar to each other. However, note that preference distillation is much better than REBEL baseline in terms of AlpacaEval 2.0 win-rate (including LC win-rate), which is known to have a higher Spearman correlation with Chatbot Arena (Dubois et al., 2024) than MT-bench. Taken together, these findings demonstrate the substantial promise of our preference distillation approach.

[11] https://github.com/lm-sys/FastChat/tree/main/fastchat/llm_judge

[12] https://github.com/tatsu-lab/alpaca_eval