BEYOND RLHF: A UNIFIED THEORETICAL FRAMEWORK OF ALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning from human feedback (RLHF) has become the dominant paradigm for aligning large language models (LLMs) to produce higher-quality responses. However, the standard training objective for RLHF lacks a learning-theoretic justification, and existing theories do not allow comparisons of the guarantees between various methods because different methods are often analyzed under different frameworks. Toward a unified framework of alignment, we ask under what assumptions we can justify existing algorithms or derive new ones. To this end, we reframe alignment as *distribution learning* from pairwise preferences, which makes a probabilistic assumption describing how preferences reveal information about the target LM. This leads us to derive three principled training objectives: preference maximum likelihood estimate, preference distillation, and reverse KL minimization. These can be viewed as corrections to existing objectives that, as we show, enjoy strong non-asymptotic $O(1/n)$ convergence to the target LM under our framework. Furthermore, our theory explains, for the first time, the empirical finding that on-policy objectives (e.g., RLHF) typically outperform likelihood-style objectives (e.g., DPO). Our empirical results indicate that our algorithms are competitive with strong baselines across several tasks and models.

1 INTRODUCTION

Alignment refers to the task of improving 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 $\hat{R}$ from pairwise preferences and then optimizes a language model (LM) or policy π that maximizes the reward via reinforcement learning:

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

where $\hat{R}$ is a reward model (RM) learned from preference data, $\mathcal{D}$ is the prompt distribution, $\pi(x)$ is the policy π's response distribution to prompt x, $\mathrm{KL}(p\|q) = \mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(p(x)\|q(x))\right]$ is the Kullback–Leibler divergence (between policies), π_0 is a supervised fine-tuned reference LM, and $\beta > 0$ is the regularization strength.

The RLHF objective is central to various practical algorithms and has fundamentally shaped how we think about alignment. For example, DPO (direct policy optimization) can be viewed as reformulating RLHF under the strong 'all-policy' assumption so the objective relies on the responses in the preference data (i.e., off-policy) rather than relying on on-policy responses (Rafailov et al., 2023). ΨPO extends RLHF by generalizing the reward model to a Ψ-transformation of the preference probabilities (Azar et al., 2024). Many theoretical analyses of alignment algorithms treat the RLHF objective or its variants as the ultimate learning-theoretic goal and prove convergence guarantees such as regret bounds (Zhan et al., 2024; Xiong et al., 2024; Zhang et al., 2024; Huang et al., 2025; Xie et al., 2025b; Cen et al., 2025). Certainly, the RLHF objective has proven useful, and one may argue that it is a sensible objective.

However, the RLHF objective has no known justification from a learning-theoretic viewpoint. First, existing theoretical guarantees (see the cited works above) typically show convergence to the solution

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: Summary of the analyzed algorithms 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	Forward KL Guarantee
Preference MLE (Sec. 3)	DPO	Not Used	No	Eq. 4	$O(1/n)$ (Thm. 4)
Preference distillation (Sec. 4)	REBEL	Required	No	Eq. 11	$O(1/n)$ (Thm. 6)
Reverse KL (Sec. 5)	RLHF	Required	Yes	Eq. 15	$O(1/n)$ (Thm. 7)

of the population RLHF objective with the true reward R^* (i.e., Eq. 1 with $\hat{R}$ replaced by R^*). This is a tautology and can hardly be interpreted as a justification. Second, while maximizing rewards *informally* sounds reasonable, it is not clear what it *formally* means because the problem definition of alignment does not involve rewards at all! Furthermore, the RLHF objective has an inherent dilemma in the asymptotics. Eq. 1 is a standard objective of the form 'loss + regularizer' where the loss (on-policy negative reward) learns from data and the regularizer leverages an initial model. With sufficient data, the regularization strength β should reduce, and in the asymptotic regime, should become zero to ensure sufficient learning. So, asymptotically, we face a dilemma: β must either tend to zero so that the solution to (1) becomes deterministic (undesirable as a language model), or tend to a nonzero constant, which hinders the learning process. To avoid this dilemma, one needs to use a loss function targeted at distribution learning. All of the above imply that we must rethink alignment without relying on the RLHF objective or rewards.

Then, what kind of justification should we require? One strong form of justification is provable guarantees such as statistical convergence to the target language model (LM), which often requires some assumptions about how the data (i.e., preference) is generated. However, existing theoretical frameworks are typically only applicable to their method of choice. For example, theoretical frameworks for DPO (Agarwal et al., 2025; Kveton et al., 2025) and those for RLHF (Zhan et al., 2024; Xiong et al., 2024; Zhang et al., 2024; Huang et al., 2025; Xie et al., 2025b) are incompatible (i.e., have different theoretical targets), preventing us from directly comparing their guarantees.

The limitations discussed above call for a unified theoretical framework for alignment as distribution learning. As such, we ask:

> *Under what assumptions can we theoretically justify existing training objectives or their variants from distribution learning viewpoint?*

In this paper, we take the popular learning-theoretic treatment where we first posit a target model we aim to learn and then make probabilistic assumptions about the data generating process as a function of this target, which in turn informs the development of learning objectives based on statistical principles. Such a probabilistic formulation is widely used in classification (e.g., logistic regression), topic modeling (e.g., latent Dirichlet allocation; Blei et al., 2003), and generative models (e.g., variational autoencoder; Kingma & Welling, 2013). This approach offers three key benefits: (i) explicit modeling assumptions which often help understand the behavior of the derived algorithm; (ii) rigorous statistical guarantees such as consistency (i.e., convergence to the target model as the sample size grows); and (iii) a unified basis for comparing convergence guarantees. Perhaps surprisingly, such a *fully probabilistic* framework for alignment remains largely unexplored, to our knowledge. See Appendix A for related work.

Our contributions. We develop a fully probabilistic theoretical framework for alignment as distribution learning from pairwise preferences, without any reliance on an *a priori* notion of reward maximization. We posit the existence of an ideal, aligned LM π^* and explicitly model how information about this target LM π^* is revealed through preference feedback. In particular, it is natural to believe that the target π^* should assign a higher probability to the preferred response. We encode this as the assumption

$$\mathbb{P}(a \succ b \mid x) = \frac{\pi^*(a \mid x)^\gamma}{\pi^*(a \mid x)^\gamma + \pi^*(b \mid x)^\gamma}, \qquad \gamma > 0, \tag{2}$$

where $a \succ b$ means that response a is preferred over b given prompt x. This is an instance of a Bradley–Terry (BT) model (Bradley & Terry, 1952) with preference score given by the tilted

response probabilities of π^*. The main conceptual difference from the BT model used in RLHF is the direct dependence on the target LM rather than a reward function. That is, the preference model is *explicitly* a language model. This is in stark contrast to DPO which starts from the RLHF formulation and leverages the (unrealistic) all-policy assumption to realize that there is an *implicit* relationship between the reward model (or preference model) and the LM (Rafailov et al., 2023).

From our assumption (2), we derive variants of existing algorithms from first principles and establish their convergence in distribution to π^* under KL divergence. Specifically, we derive and study the following three algorithms (summarized in Table 1):

- **PMLE** (Preference maximum likelihood estimate; Section 3): Given pairwise preference data, PMLE maximizes the likelihood of the preference model (2), subject to KL regularization w.r.t. a reference policy π_0. It is similar to DPO and relatively simple to optimize.
- **Preference distillation** (Section 4): By directly estimating the expected preference from a learned preference model (typically a shallow neural network), 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 our model assumption (2).
- **Reverse KL** (RKL; Section 5): We consider minimizing $\mathbb{E}_x[\mathrm{KL}(\hat{\pi}(x)\|\pi^*(x)]$ plus the KL regularizer where $\pi^*(x)$ is replaced with a plugin estimate derived from a learned preference model, which is natural under our assumption (2). This results in adding an entropy term to the RLHF objective, which can be seen as a correction to RLHF to enable distribution learning (without the aforementioned dilemma).Figure 1 shows that under our preference model, RLHF with tuned β for each sample size n indeed suffers from inconsistency (i.e., does not converge to the true π^*) whereas RKL is consistent. Experimental details are provided in Appendix D.1.
- **Theoretical guarantees**: For each of the above algorithm outputting $\hat{\pi}$, we prove non-asymptotic $O(1/n)$ upper bounds on the forward KL error $\mathrm{KL}(\pi^*\|\hat{\pi})$, where n is the size of the preference dataset. Hence, all three methods are consistent in a distributional sense.

Our bounds also show a stronger guarantee for RKL (similar to RLHF) compared to PMLE (similar to DPO), which can be seen as, for the first time in the literature, a theoretical confirmation of the common empirical finding that RLHF outperforms DPO. Throughout, all proofs are deferred to Appendix C. Empirically, contextual bandit experiments support our theoretical predictions, while general chat evaluations show gains over the corresponding baselines (Section 6).

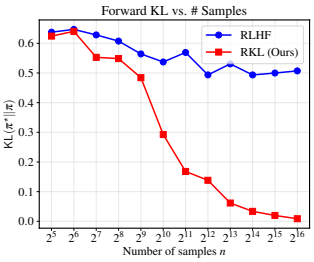


Figure 1: RLHF vs. RKL.

2 Preliminaries

Alignment as distribution learning. Let $\mathcal{X}$ and $\mathcal{A}$ be the space of prompts and responses, respectively, and let $\mathcal{D} \in \Delta(\mathcal{X})$ be a fixed distribution over prompts. We define a language model as a 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 Π of LMs, our ultimate goal is to find $\hat{\pi} \in \Pi$ that is as close as possible to π^* w.r.t. a suitable measure of distance between distributions such as KL divergence.

Our preference model. Let μ be the LM used for generating responses to be preference-labeled; this could 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 and a^+, a^- are preferred and dispreferred responses, respectively. We assume that, given x, the pair (a^+, a^-) is sampled by drawing responses $a, b \sim \mu(x)$ independently and then sampling a (binary) preference from $\mathbb{P}_*(a \succ b \mid x) := \mathbb{P}_{\pi^*}(a \succ b \mid x)$, where

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

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

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 $0 < \gamma < 1$.

One may wonder if (2) is too restrictive; we claim that it is not. Our assumption essentially takes the classical Bradley–Terry model $\mathbb{P}(a \succ b \mid x) = p^*(a|x)/(p^*(a|x) + p^*(b|x))$ for some latent score p^* (Bradley & Terry, 1952), and then connects p^* with the target LM π^*, which is a simple link allowing us to establish guarantees for learning π^*. While the BT model has drawbacks such as not being able to express cyclic preferences, it is standard in the literature (e.g., Christiano et al., 2017; Stiennon et al., 2020), and our perspective of distribution learning can be extended to incorporate general preference models. Previous theoretical works provide convergence guarantees for RLHF or other reward-maximizing objectives (Zhan et al., 2024; Xiong et al., 2024; Zhang et al., 2024; Huang et al., 2025; Xie et al., 2025b) without justifying why a certain reward construction must be optimized in the first place. In contrast, the main point of our work is to identify under which assumptions can such objectives or variants be justified.

Theoretical setup. We call $R_\pi(x, a) := \gamma \ln \pi(a \mid x)$ the *reward* induced by the LM $\pi \in \Pi$.[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 $\pi \in \Pi$.

These assumptions are standard in the alignment theory literature (Zhan et al., 2024; Xie et al., 2025a; Zhang et al., 2025; Agarwal et al., 2025; Huang et al., 2025). Since the responses (a^+, a^-) are sampled from policy μ rather than π^*, the alignment problem is an instance of offline learning where there is a distribution shift between the observed data versus the target distribution that we wish to obtain guarantees on. It is thus necessary to introduce a coverage assumption between μ and the policy class Π as is standard in offline RL (see, e.g., Agarwal et al., 2019). In particular, we make use of 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 $\pi \in \Pi'$,

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

This is a weaker assumption than the all-policy ℓ_∞-concentrability condition $\sup_{\pi\in\Pi} \max_{x,a} \frac{\pi(a|x)}{\mu(a|x)} \le C'$ (Munos, 2003) as the former can be bounded even if the latter is infinite, depending on the distribution $\mathcal{D}$ and the reward class $\Delta\bar{R}$.[3]

3 Preference Maximum Likelihood Estimation

We begin with the basic principle of maximum likelihood estimation, which will yield an algorithm closely related to DPO. Given a preference dataset $D_n = \{(x, a^+, a^-)\}$ 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 BT preference assumption (3). Concretely, the negative log-likelihood for each pair (x, a^+, a^-) under a candidate policy π is:

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

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

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

[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; we leave this to future work.

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 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: $\beta \cdot \mathrm{KL}(\pi \| \pi_0)$. Putting everything together, our **PMLE** (preference maximum likelihood estimation) objective for distribution learning is

$$\mathcal{L}_{\mathsf{PMLE},\beta}(\pi) := \mathcal{L}_{\mathsf{PMLE}}(\pi) + \beta \,\mathrm{KL}(\pi \| \pi_0). \tag{5}$$

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

$$\sum_{D_n} -\ln \sigma\left(\gamma \ln \frac{\pi(a^+ \mid x)}{\pi(a^- \mid x)} - \gamma \ln \frac{\pi_0(a^+ \mid x)}{\pi_0(a^- \mid x)}\right). \tag{6}$$

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) prove that DPO may converge to a degenerate distribution. Also, Song et al. (2024) show 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 – i.e., poorly or not at all represented in the offline dataset. The explicit KL term in PMLE is natural and ensures that we do not deviate too far away from the reference policy.

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

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

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

Moreover, if $\mathrm{KL}(\pi^* \| \pi_0) < \infty$ *and* $\beta > 0$*, the regularized PMLE* $\hat{\pi} = \arg\min_{\pi \in \Pi} \mathcal{L}_{\mathsf{PMLE},\beta}(\pi)$ *satisfies*

$$\mathbb{E}_{x \sim \mathcal{D}}\left[\mathrm{KL}(\pi^*(x) \| \hat{\pi}(x))\right] \lesssim \frac{C_\Pi}{\gamma^2} \cdot \left(\frac{\ln(|\Pi|/\delta)}{n} + \beta \,\mathrm{KL}(\pi^* \| \pi_0)\right).$$

Hence, PMLE is a consistent distributional estimator. The rate (7) is tight up to logarithmic factors; a lower bound of C_Π/n is proved in Kim et al. (2026, Theorem 2.2). More generally, with KL regularization, we still ensure the optimal rate as long as $\beta = O(1/n)$. This demonstrates that the objective derived from purely considering preference feedback (3) already suffices to learn the true distribution π^* as $n \to \infty$. In contrast, fixing a constant $\beta > 0$ necessarily leads to a suboptimal solution as $n \to \infty$, as discussed in the introduction.

The proof, provided in Appendix C.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 the optimal $1/n$ rate rather than $1/\sqrt{n}$ that would be obtained when directly following their proof technique. We also remark that the left-hand side of (7) is equivalent to the KL divergence between the induced *joint* distributions on $\mathcal{X} \times \mathcal{A}$: $\mathrm{KL}(\mathcal{D}(x)\pi^*(a \mid x) \| \mathcal{D}(x)\hat{\pi}(a \mid x))$.

Next, we turn to alignment methods that leverages a learned preference model, which amounts to learning reward models for RLHF in contemporary practice. As per our philosophy, we emphasize that the methods are derived from a distribution learning perspective rather than reward maximization.

4 Preference Distillation

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 to form supervised learning losses, i.e., objectives that do not require RL algorithms 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 the use of RL algorithms, which are typically slow to converge. While reward model training is an extra step 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 assumption (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

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

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:

$$\hat{R} = \arg\min_{R \in \mathcal{R}} \sum_{D_n} -\ln \sigma(R(x, a^+) - R(x, a^-)). \tag{9}$$

This is equivalent to the PMLE objective under (8) but with the constraint $R \in \mathcal{R}$.

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 $\ln \frac{\pi(a^+|x)/\pi_0(a^+|x)}{\pi(a^-|x)/\pi_0(a^-|x)} \approx \eta(\hat{R}(x, a^+) - \hat{R}(x, a^-))$ by optimizing a squared loss

$$\sum_{D_n} \left(\ln \frac{\pi(a^+|x)\pi_0(a^-|x)}{\pi(a^-|x)\pi_0(a^+|x)} - \eta(\hat{R}(x, a^+) - \hat{R}(x, a^-)) \right)^2, \tag{10}$$

where $\eta > 0$ controls the strength of the reward signals. In our 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 $y \sim \text{Bernoulli}(\mathbb{P}_{\widetilde{\pi}}(a \succ b \mid x))$. 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

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

Then, minimizing the log-loss with respect to this synthetic preference is equivalent to minimizing the KL between the binary preference distributions $\text{Bern}(\mathbb{P}_{\widetilde{\pi}}(a^+ \succ a^- \mid x))$ and $\text{Bern}(\mathbb{P}_{\pi}(a^+ \succ a^- \mid x))$:

$$\mathcal{L}_{\textsf{Distill}}(\pi) := \frac{1}{n} \sum_{D_n} \text{KL}\big(\text{Bern}(\mathbb{P}_{\widetilde{\pi}}(a^+ \succ a^- \mid x)) \,\|\, \text{Bern}(\mathbb{P}_{\pi}(a^+ \succ a^- \mid x))\big) + \text{const}. \tag{11}$$

As in the PMLE (Section 3), in practice we add a KL regularizer:

$$\mathcal{L}_{\textsf{Distill},\beta}(\pi) := \mathcal{L}_{\textsf{Distill}}(\pi) + \beta \,\text{KL}(\pi \| \pi_0). \tag{12}$$

We remark that the data for reward model training (9) and preference distillation (11) can come from different datasets; our theoretical analysis is easily adapted to this setting.

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

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

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

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 RM 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 pref. distill. estimate* $\hat{\pi} = \arg\min_{\pi \in \Pi} \mathcal{L}_{\mathsf{Distill}}(\pi)$ *satisfies with probability at least* $1 - \delta$*,*

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

See Appendix C.3 for the proof. Here and in Theorem 7, it is straightforward to obtain guarantees for the case $\beta > 0$ by incurring an additive $\beta \, \mathrm{KL}(\pi^* \| \pi_0)$ term as in Theorem 4; we omit this for brevity.

Benefits of distillation. The above rate $n \gtrsim C_\Pi \ln |\Pi|$ is equal to that of PMLE (7) since we assume we learn $\hat{\pi} \in \Pi$ using responses D_n generated from μ, same as the reward model. However, if we have access to a 'stronger' base model π_0, it is natural to learn $\hat{\pi}$ using responses from π_0 instead. In this scenario, we instead obtain a rate of $\gamma^{-2}(C_0 \ln |\Pi| + C_{\mathcal{R}} \ln |\mathcal{R}|)$, where C_0 is a slightly modified coverage coefficient for π_0 instead of μ. This improves over the PMLE rate if π_0 is comparatively well-aligned so that $C_0 < C_\Pi$, thus rigorously establishing the benefits of distillation with a strong base model. See Theorem 15 in Appendix C.3 for a formal statement and proof.

5 Reverse KL Minimization

Our two proposed methods both maximize the likelihood and asymptotically recover π^*. However, it is also plausible to aim to minimize the reverse KL divergence 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 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 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^*, i.e., $\mathbb{E}_{x \sim \mathcal{D}}[\mathrm{KL}(\pi(x) \| \pi^*(x))]$, can be rewritten as

$$\hat{\pi} = \arg\min_{\pi \in \Pi} \mathbb{E}_{x \sim \mathcal{D}} \big[\mathbb{E}_{a \sim \pi(x)} [-\ln \pi^*(a \mid x)] - H(\pi(x)) \big], \tag{14}$$

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. We therefore estimate the target distribution using a reward model trained on preference data following the same approach as in preference distillation. 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 (14) to arrive at the objective

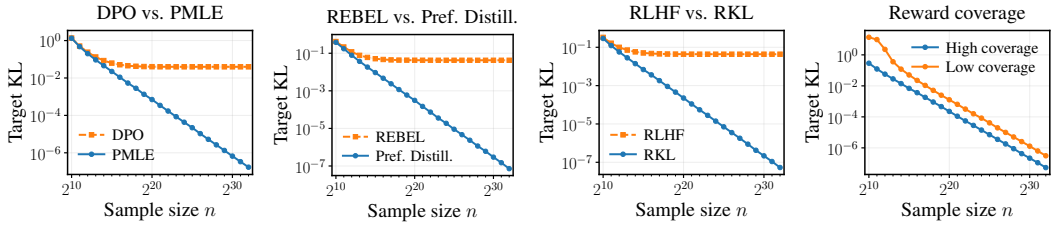
$$\arg\min_{\pi \in \Pi} \mathbb{E}_{x \sim \mathcal{D}} \left[\mathbb{E}_{a \sim \pi(x)} \big[-\gamma^{-1} \hat{R}(x, a) \big] - H(\pi(x)) \right].$$

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:

$$\mathcal{L}_{\mathsf{RKL},\beta}(\pi) := \frac{1}{n} \sum_{(x,\cdot,\cdot) \in D_n} - \mathbb{E}_{a \sim \pi(x)} \left[\hat{R}(x, a) \right] - \gamma H(\pi(x)) + \beta \, \mathrm{KL}(\pi(x) \| \pi_0(x)) \tag{15}$$

where γ and β control 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 minimize the objective (15).

Relation to RLHF. At face value, RKL can be seen as a generalization of RLHF since RKL with $\gamma = 0$ is equal to the RLHF objective. Conversely, under our preference assumption (3), RLHF itself can be interpreted as minimizing reverse KL in the limit $\gamma \to 0$. Thus, our result can be viewed as providing *theoretical justification* for the RLHF objective (1) which has been widely viewed as the gold standard for alignment (Stiennon et al., 2020; 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 can be seen as reverse KL minimization (Ziebart, 2010).



(a) PMLE (b) Pref. distill. (c) RKL (d) Reward coverage

Figure 2: Forward KL to π^* vs. number of preferences. (a–c) Method comparisons; (d) RKL under high and low reward coverage at equal RM size (Appendix D.2).

Table 2: **Bandit ablations.** Greedy win-rate (%), KL to π_0, policy entropy, and win-rate under reward noise of scale s; larger s means stronger noise ($s = 0$: no noise).

DPO vs. PMLE

Method	Win	KL
DPO	55.54	0.383
PMLE	**57.02**	**0.319**

RLHF vs. RKL

Method	Win	Entropy
RLHF	**55.62**	1.076
RKL	55.57	**2.020**

REBEL vs. Pref. distill.

Method / Noise	0	1	2	4	8	16
REBEL	57.35	56.91	56.44	55.91	54.53	52.48
Pref. distill.	**57.95**	**57.83**	**57.42**	**57.01**	**57.11**	**57.13**

The dilemma of RLHF. The necessity of the entropy term under our statistical framework can be seen as follows. Under (3), RLHF generally faces the following asymptotic dichotomy when μ has sufficient coverage:

1. **Underfitting** (fix $\beta > 0$ as $n \to \infty$): The learned LM is a proper distribution but cannot be too far from the reference policy π_0, i.e., the estimator is inconsistent.
2. **Degenerate** (take $\beta = \beta_n \downarrow 0$ as $n \to \infty$): By purely maximizing rewards, the learned LM $\pi(x)$ becomes one-hot on the reward maximizer, i.e., the estimator collapses to a degenerate solution.

Both behaviors obstruct distribution recovery. This phenomenon can also be seen in our toy experiment shown in Figure 1 (Appendix D.1). In contrast, Theorem 7 demonstrates that under our approach, the forward KL indeed tends to zero as the sample size n increases. In particular, when KL regularization is applied, taking $\beta = O(1/n)$ still ensures optimal convergence as in Theorem 4, avoiding degeneracy. See Appendix B for further discussion on entropic regularization.

Convergence guarantee. We prove the following guarantee in Appendix C.4:

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

$$\mathbb{E}_{x \sim \mathcal{D}} \left[\mathrm{KL}(\pi^*(x) \| \hat{\pi}(x)) \right] \lesssim \frac{\ln(|\Pi|/\delta)}{n} + \frac{C_{\mathcal{R}}}{\gamma^2} \cdot \frac{\ln(|\mathcal{R}|/\delta)}{n}. \tag{16}$$

Why can reverse KL attain a better bound? The reverse KL formulation can yield a smaller upper bound for the *forward* KL that depends on the coverage coefficient of $\mathcal{P}_\gamma(\mathcal{R})$ rather than Π as in previous bounds (7), (13). In particular, under Assumption 5, $\ln|\Pi|$ and $C_{\mathcal{R}} \ln|\mathcal{R}|$ may both be much smaller than $C_\Pi \ln|\Pi|$, or even the improved preference distillation guarantee $C_0 \ln|\Pi|$ (23).

Astute readers may wonder: How can reverse KL avoid C_Π (or C_0) while preference distillation does not, even though they both leverage the reward model? The reason is that the *policy learning step* of preference distillation still relies on response pairs (a^+, a^-) sampled from μ (or π_0), unlike reverse KL which uses responses for the lightweight *reward modeling step* only. Moreover, bounding forward instead of reverse KL (even though the objective is derived from minimizing reverse KL) allows us to avoid comparing the coverage of $\hat{\pi}$ against μ, which is not guaranteed by Definition 3. Nevertheless, the forward and reverse KL error may still be compared (however suffering a constant exponential in R), as we show in Proposition 16.

Remark 8 (Relation to maximum-entropy RL). Notice that RKL without the KL regularizer coincides with the maximum-entropy RL objective, adding an entropy term to the reward maximization objective. We emphasize that our novelty is not the RKL objective itself but the rather surprising discovery that RKL enjoys a strong statistical guarantee for learning the target distribution π^* under a simple assumption (2), along with the interpretation provided from the derivation. Without proving

such a guarantee, RKL is just adding auxiliary exploration or regularization to the RLHF objective and is nothing more than a *heuristic* algorithm for the alignment problem, especially because there is no reward in the problem definition of alignment. In words, we have discovered when max-entropy RL performs well for the LLM alignment, which is the research question we have set out in Section 1.

6 EXPERIMENTS

We first use contextual bandits with a known π^* to evaluate target recovery and the effect of KL regularization, entropy, and reward outliers. We then evaluate response quality on general chat benchmarks. Additional experiments on the TL;DR summarization task are provided in Appendix D.3.

Contextual bandits. We train softmax policies and linear RMs over fixed features that can represent the target on preference pairs labeled by (3) with $\gamma = 0.5$ over 64 contexts and 16 actions (Appendix D.2). Figure 2(a–c) varies n from 2^{10} to 2^{32} with $\beta = 0$. Forward KL to π^* decays at roughly $1/n$ for PMLE, preference distillation, and RKL as Theorems 4, 6, and 7 predict. DPO, REBEL, and RLHF plateau at a nonzero error since their optima are regularized to π_0 rather than π^*. Figure 2(d) compares two RMs of equal size whose extra features have large or small covariance under the data collection policy; RKL error is larger under low coverage, consistent with the $C_{\mathcal{R}}$ dependence in Theorem 7.

Table 2 illustrates ablations on the same bandit with 2^{16} preference pairs. Each method is compared against its baseline under the same setup. The explicit KL term in PMLE lets it match DPO's win-rate while staying closer to π_0. The entropy term in RKL keeps the policy far more diverse than RLHF at the same win-rate. Preference distillation is more robust than REBEL to outliers in the reward signal. REBEL's win-rate degrades steadily as the noise grows while preference distillation is barely affected. The noise model and all coefficients are given in Appendix D.2.

General chat. We fine-tune LLaMA-3-8B-Instruct (Grattafiori et al., 2024) on UltraFeedback (Cui et al., 2023) and use ArmoRM-Llama3-8B-v0.1 (Wang et al., 2024) as the reward model. Response quality is judged by LLMs on MT-Bench (Zheng et al., 2023), AlpacaEval 2.0 (Dubois et al., 2024), and Arena Hard (Li et al., 2025). We compare each method with its corresponding baseline. PMLE uses a KL coefficient of approximately $\gamma\beta_{\mathrm{DPO}}$ to account for the scale of its preference loss. Also, preference distillation uses no KL penalty since REBEL (Gao et al., 2024) does not use one on this task. RLHF is implemented with GRPO (Shao et al., 2024) and RKL adds the entropy term to it. Details on experimental setup and hyperparameter choice are provided in Appendix D.4.

Table 3: **General chat evaluation.**

Model	MT-Bench	AlpacaEval 2.0 LC win-rate	AlpacaEval 2.0 Win-rate	Arena Hard
Base	8.10	29.62	28.71	26.8
DPO	**8.32**	44.72	43.94	45.1
PMLE	8.29	**49.74**	**47.19**	**46.0**
REBEL	8.17	52.04	44.99	39.8
Pref. Distill.	**8.18**	**54.49**	**46.86**	**41.9**
RLHF (GRPO)	8.13	35.97	34.42	32.0
RKL	**8.27**	**36.67**	**35.18**	**32.5**

Table 3 summarizes the results. All three methods (PMLE, preference distillation, and RKL) show clear gains over their counterparts on AlpacaEval 2.0 and Arena Hard. On MT-Bench, they are similar to or slightly better than their counterparts. These results show that our theoretically motivated objectives also work well for LLM alignment in practice.

7 CONCLUSION

We formulated alignment as distribution learning from a single explicit assumption (3) on how preferences relate to the target LM, and analyzed three training algorithms closely related to existing approaches with $O(1/n)$ forward-KL guarantees, supported by contextual bandit, LLM chat, and summarization experiments. Future directions include comparing mode-seeking versus mode-covering objectives across domains, extending the framework to alternative divergences and preference models, and tightening the exponential dependence on R via stronger assumptions on π_0.

AI USAGE STATEMENT

We used generative AI tools only to edit the manuscript for readability and grammar, and not for any task that requires disclosure. All AI-assisted edits were reviewed by the authors, and we take responsibility for the final content of this work.

REFERENCES

Alekh Agarwal, Nan Jiang, Sham M Kakade, and Wen Sun. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, 2019.

Alekh Agarwal, Christoph Dann, and Teodor V Marinov. Design considerations in offline preference-based rl. *arXiv preprint arXiv:2502.06861*, 2025.

Afra Amini, Tim Vieira, and Ryan Cotterell. Direct preference optimization with an offset. *CoRR*, abs/2402.10571, 2024.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Machine Learning*, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, 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.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

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

Shicong Cen, Jincheng Mei, Katayoon Goshvadi, Hanjun Dai, Tong Yang, Sherry Yang, Dale Schuurmans, Yuejie Chi, and Bo Dai. 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.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with high-quality feedback. *CoRR*, 2023.

Yann Dubois, Percy Liang, and Tatsunori Hashimoto. Length-controlled alpacaeval: A simple debiasing of automatic evaluators. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=CybBmzWBX0.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Adam Fisch, Jacob Eisenstein, Vicky Zayats, Alekh Agarwal, Ahmad Beirami, Chirag Nagpal, Peter Shaw, and Jonathan Berant. Robust preference optimization through reward model distillation. *Transactions on Machine Learning Research (TMLR)*, 2025. ISSN 2835-8856.

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

Dylan J. Foster, Zakaria Mhammedi, and Dhruv Rohatgi. 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.

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

Zhaolin Gao, Jonathan Chang, Wenhao Zhan, Owen Oertell, Gokul Swamy, Kianté Brantley, Thorsten Joachims, Drew Bagnell, Jason D Lee, and Wen Sun. Rebel: Reinforcement learning via regressing relative rewards. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, et al. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*, 2024.

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

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Audrey Huang, Wenhao Zhan, Tengyang Xie, Jason D. Lee, Wen Sun, Akshay Krishnamurthy, and Dylan J Foster. Correcting the mythos of KL-regularization: Direct alignment without overoptimization via chi-squared preference optimization. In *International Conference on Learning Representations*, 2025.

Shengyi Huang, Michael Noukhovitch, Arian Hosseini, Kashif Rasul, Weixun Wang, and Lewis Tunstall. 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.

Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. A survey of reinforcement learning from human feedback, 2024. URL https://arxiv.org/abs/2312.14925.

Juno Kim, Jihun Yun, Jason D. Lee, and Kwang-Sung Jun. Coverage improvement and fast convergence of on-policy preference learning, 2026. URL https://arxiv.org/abs/2601.08421.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Branislav Kveton, Xintong Li, Julian McAuley, Ryan Rossi, Jingbo Shang, Junda Wu, and Tong Yu. Active learning for direct preference optimization, 2025.

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback, 2023. URL https://arxiv.org/abs/2302.12192.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. In *International Conference on Machine Learning*, pp. 34209–34231. PMLR, 2025.

Xiang Lisa Li, Vaishnavi Shrivastava, Siyan Li, Tatsunori Hashimoto, and Percy Liang. Benchmarking and improving generator-validator consistency of language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Youwei Liang, Junfeng He, Gang Li, Peizhao Li, Arseniy Klimovskiy, Nicholas Carolan, Jiao Sun, Jordi Pont-Tuset, Sarah Young, Feng Yang, Junjie Ke, Krishnamurthy Dj Dvijotham, Katherine M. Collins, Yiwen Luo, Yang Li, Kai J Kohlhoff, Deepak Ramachandran, and Vidhya Navalpakkam. 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.

Jie Liu, Zhanhui Zhou, Jiaheng Liu, Xingyuan Bu, Chao Yang, Han-Sen Zhong, and Wanli Ouyang. 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.

Qi Mao, Hsin-Ying Lee, Hung-Yu Tseng, Siwei Ma, and Ming-Hsuan Yang. 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.

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

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

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. WebGPT: Browser-assisted question-answering with human feedback, 2022. URL https://arxiv.org/abs/2112.09332.

Abhijnan Nath, Changsoo Jung, Ethan Seefried, and Nikhil Krishnaswamy. Simultaneous reward distillation and preference learning: Get you a language model who can do both, 2025. URL https://arxiv.org/abs/2410.08458.

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

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

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

Biqing Qi, Pengfei Li, Fangyuan Li, Junqi Gao, Kaiyan Zhang, and Bowen Zhou. Online dpo: Online direct preference optimization with fast-slow chasing, 2024. URL https://arxiv.org/abs/2406.05534.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:53728–53741, 2023.

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

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

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

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Gokul Swamy, Sanjiban Choudhury, Wen Sun, Zhiwei Steven Wu, and J. Andrew Bagnell. All roads lead to likelihood: The value of reinforcement learning in fine-tuning, 2025.

Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Remi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Avila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Lewis Tunstall, Edward Emanuel Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro Von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M Rush, and Thomas Wolf. Zephyr: Direct distillation of LM alignment. In *First Conference on Language Modeling*, 2024.

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

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization, 2023. URL https://arxiv.org/abs/2311.12908.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. *arXiv preprint arXiv:2406.12845*, 2024.

Peter West, Ximing Lu, Nouha Dziri, Faeze Brahman, Linjie Li, Jena D Hwang, Liwei Jiang, Jillian Fisher, Abhilasha Ravichander, Khyathi Chandu, et al. The generative ai paradox:"what it can create, it may not understand". In *The Twelfth International Conference on Learning Representations*, 2024.

Teng Xiao, Yige Yuan, Huaisheng Zhu, Mingxiao Li, and Vasant G Honavar. Cal-dpo: Calibrated direct preference optimization for language model alignment. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

Tengyang Xie, Ching-An Cheng, Nan Jiang, Paul Mineiro, and Alekh Agarwal. Bellman-consistent pessimism for offline reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Tengyang Xie, Dylan J Foster, Akshay Krishnamurthy, Corby Rosset, Ahmed Hassan Awadallah, and Alexander Rakhlin. 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.

Tengyang Xie, Dylan J Foster, Akshay Krishnamurthy, Corby Rosset, Ahmed Hassan Awadallah, and Alexander Rakhlin. Exploratory preference optimization: Harnessing implicit q*-approximation for sample-efficient RLHF. In *International Conference on Learning Representations*, 2025b.

Wei Xiong, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. 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.

Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. Rrhf: Rank responses to align language models with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Wenhao Zhan, Baihe Huang, Audrey Huang, Nan Jiang, and Jason Lee. Offline reinforcement learning with realizability and single-policy concentrability. In *Proceedings of the Conference on Learning Theory (COLT)*, 2022.

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

Shenao Zhang, Donghan Yu, Hiteshi Sharma, Han Zhong, Zhihan Liu, Ziyi Yang, Shuohang Wang, Hany Hassan, and Zhaoran Wang. Self-exploring language models: Active preference elicitation for online alignment, 2024. URL https://arxiv.org/abs/2405.19332.

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

Yudi Zhang, Lu Wang, Meng Fang, Yali Du, Chenghua Huang, Jun Wang, Qingwei Lin, Mykola Pechenizkiy, Dongmei Zhang, Saravan Rajmohan, and Qi Zhang. Distill not only data but also rewards: Can smaller language models surpass larger ones?, 2025. URL https://arxiv.org/abs/2502.19557.

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

Yao Zhao, Mikhail Khalman, Rishabh Joshi, Shashi Narayan, Mohammad Saleh, and Peter J Liu. Calibrating sequence likelihood improves conditional language generation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023b.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.

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

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Appendix

Table of Contents

A Related Work **15**

B Additional Remarks on Reverse KL **16**

C Theoretical Guarantees **17**
C.1 Auxiliary Lemmas 17
C.2 Proofs for Section 3 18
C.3 Proofs for Section 4 21
C.4 Proofs for Section 5 23

D Experimental Details **24**
D.1 Toy Experiments 24
D.2 Contextual Bandits 25
D.3 TL;DR Summarization 25
D.4 General Chat 28

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 point 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).

Agarwal et al. (2025) provide a closely related analysis of offline preference objectives using forward KL and generalized coverage. Unregularized PMLE is included by choosing a uniform base policy and an appropriately scaled logistic loss. Theorem 3.6 in that work has a square-root dependence on excess preference risk. The analysis here bounds KL through a quadratic estimation error and gives an end-to-end $O(1/n)$ rate under the stated assumptions. Separate analyses of preference distillation and RKL make the role of RM complexity and coverage explicit.

B Additional Remarks on Reverse KL

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 (15) 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. Section 6 examines the resulting changes in entropy and preference quality.

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^+, a^-), so the dependence on μ disappears and we will not pay for C_Π, similarly to Theorem 7. However, 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.

C Theoretical Guarantees

C.1 Auxiliary Lemmas

We require the following basic results.

Lemma 9. *For all $a, b \in \mathbb{R}$ it holds that $|\sigma(a) - \sigma(b)| \geq \frac{1}{4}e^{-(|a|\vee|b|)}|a-b|$.*

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

$$\sigma'(z) = \sigma'(|z|) = \frac{1}{1+e^{|z|}}\frac{1}{1+e^{-|z|}} \geq \frac{1}{2(1+e^{|z|})} \geq \frac{1}{4}e^{-|z|}.$$

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

$$\sigma(b) - \sigma(a) = \int_a^b \sigma'(z)\,\mathrm{d}z \geq \int_a^b \frac{1}{4}e^{-|z|}\,\mathrm{d}z \geq \frac{1}{4}e^{-(|a|\vee|b|)}(b-a)$$

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 10. *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

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

as was to be shown. □

Lemma 11. *For $r_{\min} > 0$, it holds for all $r \in [r_{\min}, \infty)$ that*

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

Proof. The function ψ defined in Lemma 10 extends to a nondecreasing continuous function on $(0,\infty)$ by setting $\psi(1) := \frac{1}{2}$. When $r \geq e^{-1}$, it follows that $\psi(r) \geq \psi(e^{-1}) = e^{-1}$.

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

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

□

Lemma 12 (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). □

C.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 13. *Let $\hat{\pi} = \arg\min_{\pi\in\Pi} \mathcal{L}_{\mathsf{PMLE},\beta}(\pi)$ with $\beta \ge 0$. Then, with probability at least $1-\delta$,*

$$\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\mathbb{P}_{\hat{\pi}}(a \succ b \mid x) - \mathbb{P}_*(a \succ b \mid x))^2\right] \le \frac{4\ln(|\Pi|/\delta)}{n} + 2\beta\,\mathrm{KL}(\pi^*\|\pi_0).$$

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 $\mathcal{L}_{\mathsf{PMLE},\beta}(\pi)$ for $\pi \in \Pi$. By optimality,

$$C(\hat{\pi}, D_n) \ge \frac{n\beta}{2}\left(\mathrm{KL}(\hat{\pi}\|\pi_0) - \mathrm{KL}(\pi^*\|\pi_0)\right) \ge -\frac{n\beta}{2}\mathrm{KL}(\pi^*\|\pi_0).$$

It follows from Lemma 12 that

$$\begin{aligned} -\log\mathbb{E}_{\tilde{D}_n}[\exp(C(\hat{\pi}, \tilde{D}_n))] &\le -C(\hat{\pi}, D_n) + \ln(|\Pi|/\delta) \\ &\le \ln(|\Pi|/\delta) + \frac{n\beta}{2}\mathrm{KL}(\pi^*\|\pi_0). \end{aligned}$$

and

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

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

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

$$\geq \frac{1}{4}\,\mathbb{E}\left[(p_{\hat{\pi}} - p_*)^2\right],$$

which yields the desired bound. □

Proof of Theorem 4. The proof builds on the analysis of Agarwal et al. (2025, Theorem 3.6). A quadratic control of KL avoids a square-root dependence on preference estimation error. Combining this step with a finite-sample likelihood bound yields the stated bound. Remark 14 compares the key bounds.

Using Lemma 9 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

$$\begin{aligned}
&\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\mathbb{P}_{\hat{\pi}}(a \succ b \mid x) - \mathbb{P}_*(a \succ b \mid x))^2\right] \\
&\geq \frac{e^{-4\gamma R}}{16}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[\left(\gamma \ln \frac{\hat{\pi}(a \mid x)}{\hat{\pi}(b \mid x)} - \gamma \ln \frac{\pi^*(a \mid x)}{\pi^*(b \mid x)}\right)^2\right] \\
&= \frac{e^{-4\gamma R}}{16}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\Delta\bar{R}_{\hat{\pi}}(x,a) - \Delta\bar{R}_{\hat{\pi}}(x,b))^2\right] \\
&= \frac{e^{-4\gamma R}}{8}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\mu(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right]
\end{aligned}$$

where the last inequality is due to $\mathbb{E}[(X-Y)^2] = 2\,\mathbb{E}[(X - \mathbb{E}[X])^2]$ when X and Y are i.i.d. Thus, using Proposition 13, the difference in centered reward satisfies

$$\mathbb{E}_{x\sim\mathcal{D},a\sim\mu(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right] \leq 32e^{4\gamma R}\cdot\left(\frac{\ln(|\Pi|/\delta)}{n} + \frac{\beta}{2}\,\mathrm{KL}(\pi^*\|\pi_0)\right). \tag{17}$$

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

$$0 < \frac{\hat{\pi}(a \mid x)}{\pi^*(a \mid x)} = \frac{Z_*(x)}{Z_{\hat{\pi}}(x)}\exp\left(\frac{1}{\gamma}\Delta\bar{R}_{\hat{\pi}}(x,a)\right) \leq e^{4R}. \tag{18}$$

Then, we bound the KL divergence between $\pi^*, \hat{\pi}$ using Schulman's trick (Schulman, 2020) followed by Lemma 10:

$$\begin{aligned}
\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right] &= \mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\ln \frac{\pi^*(a \mid x)}{\hat{\pi}(a \mid x)}\right] \\
&= \mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\frac{\hat{\pi}(a \mid x)}{\pi^*(a \mid x)} - 1 - \ln \frac{\hat{\pi}(a \mid x)}{\pi^*(a \mid x)}\right] \\
&\leq (\tfrac{1}{2} \vee \psi(e^{4R}))\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\ln \frac{\hat{\pi}(a \mid x)}{\pi^*(a \mid x)}\right)^2\right].
\end{aligned} \tag{19}$$

Extracting the normalization constants, we further have that

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

Using Definition 3 and (17), the first term is bounded as

$$\frac{2}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right] \le \frac{2C_\Pi}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\mu(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right] \le \frac{64C_\Pi e^{4\gamma R}}{\gamma^2}\cdot\left(\frac{\ln(|\Pi|/\delta)}{n}+\frac{\beta}{2}\,\mathrm{KL}(\pi^*\|\pi_0)\right).$$

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

$$\begin{aligned}&\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\ln\frac{Z_*(x)}{Z_{\hat{\pi}}(x)}\right)^2\right]\\&\le\frac{2}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\mathbb{E}_{a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)\right]\right)^2\right]+\frac{e^{4R}}{2\gamma^4}\,\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\mathbb{E}_{a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right]\right)^2\right]\\&\le\frac{2R^2e^{4R}+2}{\gamma^2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\Delta\bar{R}_{\hat{\pi}}(x,a)^2\right]\qquad\text{(Jensen's inequality; Assumption 2)}\\&\le\frac{64C_\Pi(R^2e^{4R}+1)e^{4\gamma R}}{\gamma^2}\cdot\left(\frac{\ln(|\Pi|/\delta)}{n}+\frac{\beta}{2}\,\mathrm{KL}(\pi^*\|\pi_0)\right).\qquad\text{(by (17))}\end{aligned}$$

Putting everything together, we conclude:

$$\begin{aligned}&\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right]\\&\le(\tfrac{1}{2}\vee\psi(e^{4R}))\left(\frac{64C_\Pi e^{4\gamma R}}{\gamma^2}\cdot\left(\frac{\ln(|\Pi|/\delta)}{n}+\frac{\beta}{2}\,\mathrm{KL}(\pi^*\|\pi_0)\right)\right.\\&\qquad\left.+\frac{128C_\Pi(R^2e^{4R}+1)e^{4\gamma R}}{\gamma^2}\cdot\left(\frac{\ln(|\Pi|/\delta)}{n}+\frac{\beta}{2}\,\mathrm{KL}(\pi^*\|\pi_0)\right)\right)\\&=(\tfrac{1}{2}\vee\psi(e^{4R}))\frac{64(2R^2e^{4R}+3)C_\Pi e^{4\gamma R}}{\gamma^2}\cdot\left(\frac{\ln(|\Pi|/\delta)}{n}+\frac{\beta}{2}\,\mathrm{KL}(\pi^*\|\pi_0)\right).\end{aligned}$$

We remark that by Lemma 10, the $\frac{1}{2}\vee\psi(e^{4R})$ term is further bounded above by $\frac{e^{4R}}{16R^2}$.

Remark 14. Equation (19) provides the quadratic bound used to obtain the $1/n$ rate. In Agarwal et al. (2025), they use Cauchy-Schwarz to derive the bound

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

which introduces an extra square root compared to our derivation. Combining a square-root bound with an $O(1/n)$ preference-risk bound gives an $O(1/\sqrt{n})$ KL bound.

□

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

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

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

$$\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\mathbb{P}_{\widetilde{\pi}}(a\succ b\mid x)-\mathbb{P}_{\hat{\pi}}(a\succ b\mid x))^2\right]\le\frac{2\ln(|\Pi|/\delta)}{n}\tag{20}$$

as well. On the other hand, applying Proposition 13 to $\mathcal{P}_\gamma(\mathcal{R})$, we have

$$\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\mathbb{P}_{\widetilde{\pi}}(a\succ b\mid x)-\mathbb{P}_{*}(a\succ b\mid x))^2\right]\le\frac{4\ln(|\mathcal{R}|/\delta)}{n}\tag{21}$$

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

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

Furthermore, by Lemma 9 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

$$\begin{aligned}\mathbb{E}_{x\sim\mathcal{D},a\sim\mu(x)}\left[(\Delta\bar{R}_{\hat{\pi}}(x,a))^2\right] &= \frac{1}{2}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\mu(x)}\left[(\Delta\bar{R}_{\hat{\pi}}(x,a) - \Delta\bar{R}_{\hat{\pi}}(x,b))^2\right] \\ &\leq 2e^{4\gamma R}\cdot\frac{4\ln(2|\Pi|/\delta) + 8\ln(2|\mathcal{R}|/\delta)}{n}. \qquad (22)\end{aligned}$$

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 (22) to conclude that

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

□

Theorem 15. *Given a base policy π_0, denote by C_0 the smallest constant such that for every $\pi, \pi' \in \Pi$,*[4]

$$\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[(\Delta\bar{R}_\pi(x,a) - \Delta\bar{R}_{\pi'}(x,a))^2\right] \leq C_0\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi_0(x)}\left[(\Delta\bar{R}_\pi(x,a) - \Delta\bar{R}_{\pi'}(x,a))^2\right].$$

The preference distillation estimate $\hat{\pi} = \arg\min_{\pi\in\Pi}\mathcal{L}_{\mathsf{Distill}}(\pi)$ with responses in D_n generated from π_0 instead of μ satisfies

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right] \lesssim \frac{1}{\gamma^2}\cdot\frac{C_0\ln(|\Pi|/\delta) + C_\mathcal{R}\ln(|\mathcal{R}|/\delta)}{n} \qquad (23)$$

with probability at least $1-\delta$.

Proof. In the proof of Theorem 6, (20) now holds with μ replaced by π_0, while (21) remains unchanged. It follows from Lemma 9 that

$$\begin{aligned}&\frac{2\ln(|\Pi|/\delta)}{n} \\ &\geq \mathbb{E}_{x\sim\mathcal{D},a,b\sim\pi_0(x)}\left[(\mathbb{P}_{\widetilde{\pi}}(a \succ b \mid x) - \mathbb{P}_{\hat{\pi}}(a \succ b \mid x))^2\right] \\ &\geq \frac{\gamma^2e^{-4\gamma R}}{4}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\pi_0(x)}\left[\left(\ln\frac{\hat{\pi}(a \mid x)}{\hat{\pi}(b \mid x)} - \ln\frac{\widetilde{\pi}(a \mid x)}{\widetilde{\pi}(b \mid x)}\right)^2\right] \\ &= \frac{e^{-4\gamma R}}{4}\,\mathbb{E}_{x\sim\mathcal{D},a,b\sim\pi_0(x)}\left[\left(\Delta\bar{R}_{\hat{\pi}}(x,a) - \Delta\bar{R}_{\hat{\pi}}(x,b) - \Delta\bar{R}_{\widetilde{\pi}}(x,a) + \Delta\bar{R}_{\widetilde{\pi}}(x,b)\right)^2\right] \\ &= \frac{e^{-4\gamma R}}{2}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi_0(x)}\left[\left(\Delta\bar{R}_{\hat{\pi}}(x,a) - \Delta\bar{R}_{\widetilde{\pi}}(x,a)\right)^2\right]\end{aligned}$$

[4]This is slightly stronger than Definition 3, which can be retrieved by setting $\pi' = \pi^*$. We can weaken the definition to include only $\pi' \in \mathcal{R}$.

$$\geq \frac{e^{-4\gamma R}}{2C_0}\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\Delta\bar{R}_{\hat{\pi}}(x,a)-\Delta\bar{R}_{\widetilde{\pi}}(x,a)\right)^2\right],$$

where the last line uses the modified definition of the coverage coefficient C_0. Moreover, (21) implies

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

By a union bound, it follows with probability at least $1-\delta$ that

$$\begin{aligned}&\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[(\Delta\bar{R}_{\hat{\pi}}(x,a))^2\right]\\ &\leq 2\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\Delta\bar{R}_{\hat{\pi}}(x,a)-\Delta\bar{R}_{\widetilde{\pi}}(x,a)\right)^2\right]+2\,\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[(\Delta\bar{R}_{\widetilde{\pi}}(x,a))^2\right]\\ &\leq 8e^{4\gamma R}\cdot\frac{C_0\ln(2|\Pi|/\delta)+2C_{\mathcal{R}}\ln(2|\mathcal{R}|/\delta)}{n}.\end{aligned}$$

The remainder of the proof proceeds similarly. □

C.4 Proofs for Section 5

Proof of Theorem 7. The first step of the argument is similar to the proof of Theorem 6, but with a stronger sup norm bound which is guaranteed due to realizability. Indeed, 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

$$K:=\left\{(\pi_1,\pi_2)\in\Pi\times\mathcal{P}_\gamma(\mathcal{R}):\mathbb{E}_{x\sim\mathcal{D}}\left[\sup_{a\in\mathcal{A}}\left(\ln\frac{\pi_1(a\mid x)}{\pi_2(a\mid x)}\right)^2\right]>\varepsilon\right\},$$

it follows that

$$\begin{aligned}\mathbb{P}\left(\mathbb{E}_{x\sim\mathcal{D}}\left[\sup_{a\in\mathcal{A}}\left(\ln\frac{\hat{\pi}(a\mid x)}{\widetilde{\pi}(a\mid x)}\right)^2\right]>\varepsilon\right) &= \sum_{(\pi_1,\pi_2)\in K}\mathbb{P}(\hat{\pi}=\pi_1,\widetilde{\pi}=\pi_2)\\ &\qquad\text{(law of total probability)}\\ &\leq \sum_{(\pi_1,\pi_2)\in K}\mathbb{P}(\pi_1(x)=\pi_2(x),\ \forall x\in D_n)\\ &= \sum_{(\pi_1,\pi_2)\in K}\mathbb{P}_{x\sim\mathcal{D}}(\pi_1(x)=\pi_2(x))^n.\end{aligned}$$

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

$$\sup_{a\in\mathcal{A}}\left(\ln\frac{\pi_1(a\mid x)}{\pi_2(a\mid x)}\right)^2\leq 16R^2\cdot 1_{\{\pi_1(x)\neq\pi_2(x)\}},\quad\forall x\in\mathcal{X}.$$

This implies

$$\mathbb{P}_{x\sim\mathcal{D}}(\pi_1(x)=\pi_2(x))=1-\mathbb{E}_{x\sim\mathcal{D}}[1_{\{\pi_1(x)\neq\pi_2(x)\}}]\leq 1-\frac{\varepsilon}{16R^2},$$

and hence

$$\mathbb{P}\left(\mathbb{E}_{x\sim\mathcal{D}}\left[\sup_{a\in\mathcal{A}}\left(\ln\frac{\hat{\pi}(a\mid x)}{\widetilde{\pi}(a\mid x)}\right)^2\right]>\varepsilon\right)\leq|K|\left(1-\frac{\varepsilon}{16R^2}\right)^n\leq|\Pi|^2\exp\left(-\frac{\varepsilon n}{16R^2}\right).\quad(24)$$

We now bound the forward KL divergence. Again applying Lemma 10,

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\mathrm{KL}(\pi^*(x)\|\hat{\pi}(x))\right]$$

$$\leq (\frac{1}{2} \vee \psi(e^{4R}))\, \mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\ln\frac{\hat{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right]$$
$$\leq (1 \vee 2\psi(e^{4R}))\left(\mathbb{E}_{x\sim\mathcal{D}}\left[\sup_{a\in\mathcal{A}}\left(\ln\frac{\hat{\pi}(a\mid x)}{\widetilde{\pi}(a\mid x)}\right)^2\right] + \mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\ln\frac{\widetilde{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right]\right).$$

The first term can be bounded via (24). For the second term, repeating the derivation of Theorem 4 over the policy class $\mathcal{P}_\gamma(\mathcal{R})$ instead of Π, we obtain

$$\mathbb{E}_{x\sim\mathcal{D},a\sim\pi^*(x)}\left[\left(\ln\frac{\widetilde{\pi}(a\mid x)}{\pi^*(a\mid x)}\right)^2\right] \leq \frac{16C_{\mathcal{R}}(R^2e^{4R}+1)e^{4\gamma R}}{\gamma^2}\cdot\frac{\ln(|\mathcal{R}|/\delta)}{n}.$$

Putting everything together, we conclude:

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

with probability at least $1-\delta$. □

Proposition 16. *It holds that*

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

Proof of Proposition 16. Denote the ratio $r = \frac{\pi^*(a|x)}{\hat{\pi}(a|x)}$ for brevity. Using the same argument as (18), we have $r \in [e^{-4R}, e^{4R}]$. Then, we have

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

where (a) is by Lemma 10 and (b) is by Lemma 11 with r replaced by $1/r$. Thus,

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

□

D Experimental Details

Appendix D.2 gives the setup of the contextual bandit experiments, Appendix D.3 the TL;DR summarization experiments, and Appendix D.4 the general chat experiments.

D.1 Toy Experiments

We adopt a tabular setting with vocabulary and context sizes both equal to 10. Under the preference model (2), we fix $\gamma = 0.5$. To keep the oracle policy π^* and reference policy π_0 close, for each context $x \in \mathcal{X}$, we draw logits $\alpha^*, \alpha_0 \in \mathbb{R}^{10}$ with entries i.i.d. $\mathcal{N}(0, 0.1^2)$ and set $\pi^*(\cdot \mid x) = \mathrm{softmax}(\alpha^*)$ and $\pi_0(\cdot \mid x) = \mathrm{softmax}(\alpha_0)$. We then train policies by optimizing (1) and (15) and evaluate the forward KL $\mathbb{E}_x\big[\mathrm{KL}(\pi^*(\cdot \mid x)\|\pi(\cdot \mid x))\big]$. We vary the sample size $n \in \{2^5, \cdots, 2^{16}\}$, tuning the KL coefficient β for both RLHF and RKL at each n, while keeping $\gamma = 0.5$ fixed for RKL.

For each method and n, we report the result from the best β from a wide range.

D.2 CONTEXTUAL BANDITS

Setup. We use contextual bandits with 64 training contexts and 16 actions. Policies are softmax models over fixed context and action features with 120 parameters, and reward models are linear in fixed features with 64 parameters. The features are chosen so that the target policy lies in the policy class and, for RKL and preference distillation, so that the policy class contains every distribution induced by the reward model class; the guarantees can therefore be checked without approximation error. Preference labels follow (3) with $\gamma = 0.5$. Separate sets of 128 contexts are held out for validation and testing.

Convergence experiments (Figure 2). Preference pairs are sampled independently from a data generation policy that puts positive probability on every action, and the number of pairs is varied from 2^{10} to 2^{32}. Larger datasets extend the smaller ones within each repetition. PMLE and preference distillation use no reference penalty, and preference distillation reuses the response pairs from reward model training. RKL uses entropy coefficient $\gamma = 0.5$ and no reference penalty, while the RLHF baseline uses reference coefficient 0.5 in place of the entropy term. The REBEL baseline minimizes (10) with a fixed reference and $\eta = 1/\gamma$. Reward models and policies are trained with Adam, policies are initialized at the reference, and the RKL expectation over actions is computed exactly. We evaluate forward KL to the target under the training context distribution, averaged over twelve environments with eight preference datasets each; shading shows 95% confidence intervals across environment means.

For the coverage comparison in Figure 2(d), the two reward model classes share the four features required to represent the target reward and differ only in four additional features, taken from the directions of largest or smallest eigenvalue of the feature covariance under the collection policy. Both use the same preference data and the same policy class.

Ablations (Table 2). The reward model uses eight context and eight action features, and the policy uses 15. The reference policy is trained on 2,048 demonstrations from a related source policy whose reward parameters are correlated with the target reward; no target demonstrations are used. All methods share this reference and 2^{16} target preferences drawn from a mixture of the reference policy and 10% uniform sampling, and every method that queries the reward model receives the same budget of 2^{17} queries. PMLE uses reference coefficient 0.03, chosen to minimize validation KL to the reference among coefficients that match the DPO validation win-rate. RKL and RLHF share a fitted reward model and are trained with PPO with reference coefficient 0.5; RKL adds entropy coefficient 0.5. Preference distillation and REBEL use reference coefficient 0.05 and sample response pairs from their current policies; REBEL regresses changes in pairwise log probability on reward gaps, and preference distillation applies the soft binary cross-entropy loss to the same quantities relative to the behavior policy.

For the reward-noise sweep, we perturb the reward gap $\Delta R = \widehat{R}(x, a) - \widehat{R}(x, b)$ as

$$\widetilde{\Delta R} = \begin{cases} \Delta R & \text{with probability } 0.95, \\ \Delta R + s\,\mathrm{RMS}(\Delta R) & \text{with probability } 0.025, \\ \Delta R - s\,\mathrm{RMS}(\Delta R) & \text{with probability } 0.025, \end{cases}$$

where $\mathrm{RMS}(\Delta R)$ is the root mean square of the reward gaps on the original preference dataset, computed before policy training, and $s \in \{0, 1, 2, 4, 8, 16\}$ with $s = 0$ the clean condition. Corruption masks and signs are shared across methods, and evaluation always uses the clean preference model.

Win-rate is the probability, under the true preference model, that the learned policy's greedy response beats the reference's greedy response on test contexts. KL to the reference and entropy are computed from the full policy distributions. All values are averaged over eight preference datasets.

D.3 TL;DR SUMMARIZATION

We follow the standard TL;DR setup (Stiennon et al., 2020) of Gao et al. (2024); Song et al. (2024) with Pythia-2.8B and Pythia-6.9B (Biderman et al., 2023). We report the GPT-4 win-rate against human references (over 600 test samples), the trained reward model score, and $\mathrm{KL}(\pi \| \pi_0)$.

Table 4: **Results on TL;DR dataset with Pythia 2.8B and 6.9B**. Win-rate is evaluated by GPT-4 and reward model (RM) score evaluated by the trained reward model. We report mean/std over three random seeds.

	Pythia 2.8B			Pythia 6.9B		
Algorithm	Win-rate(↑)	RM score(↑)	$\mathrm{KL}(\pi\|\pi_0)(\downarrow)$	Win-rate(↑)	RM score(↑)	$\mathrm{KL}(\pi\|\pi_0)(\downarrow)$
DPO	47.3$\pm$1.01	**2.66**$\pm$0.03	64.35$\pm$0.68	55.6$\pm$0.53	6.09$\pm$0.02	54.45$\pm$0.82
PMLE	**49.1**$\pm$1.43	2.38$\pm$0.04	**31.42**$\pm$0.79	**59.0**$\pm$0.74	**6.27**$\pm$0.03	**24.20**$\pm$0.92
REBEL	71.0$\pm$1.26	2.86$\pm$0.02	**25.67**$\pm$0.61	82.5$\pm$0.93	7.64$\pm$0.02	29.80$\pm$0.71
Pref. Distill.	**73.8**$\pm$1.47	**2.91**$\pm$0.03	28.38$\pm$0.82	**83.9**$\pm$0.59	**7.66**$\pm$0.02	**29.05**$\pm$1.02
RLHF	72.0$\pm$0.67	2.95$\pm$0.05	**24.94**$\pm$1.12	82.0$\pm$0.92	**7.85**$\pm$0.03	24.38$\pm$0.77
Reverse KL	**73.0**$\pm$0.93	**3.03**$\pm$0.04	25.91$\pm$0.94	**84.0**$\pm$1.01	7.77$\pm$0.01	**24.12**$\pm$0.63

Results. See Table 4 for the results. For both Pythia-2.8B and Pythia-6.9B, our distribution-learning objectives achieve higher win-rates than their respective baselines in this experiment, where win-rate serves as the most direct measure of LM quality. Additionally, since PMLE implements a KL regularizer with online data, it achieves a much lower KL term 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. Overall, these experiments provide evidence that the algorithms derived from our assumption (2) can remain competitive with popular baselines.

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[5]. We summarize the dataset statistics in Table 5. 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 5: 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-2.8B[6] and Pythia-6.9B[7] (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[8]; preference distillation in particular is based on another publicly available code baseline[9]. 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).

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.

[5] https://github.com/openai/summarize-from-feedback
[6] https://huggingface.co/EleutherAI/pythia-2.8b-deduped
[7] https://huggingface.co/EleutherAI/pythia-6.9b-deduped
[8] https://github.com/vwxyzjn/summarize_from_feedback_details
[9] https://github.com/ZhaolinGao/REBEL

Algorithm 1 Preference Distillation (Sec. 4)

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

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

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

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 6. 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.

Table 6: 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
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: 0.1 (for DPO and PMLE) or 0.7 (others)

D.4 General Chat

We train LLaMA-3-8B-Instruct[10] (Grattafiori et al., 2024) on UltraFeedback (Cui et al., 2023) using ArmoRM-Llama3-8B-v0.1[11] (Wang et al., 2024) as the reward model, following Gao et al. (2024). Chat quality is evaluated with three LLM-as-a-judge benchmarks: MT-Bench (Zheng et al., 2023), AlpacaEval 2.0 (Dubois et al., 2024), and Arena Hard (Li et al., 2025).

Training setup. DPO and REBEL use their original configurations, and our preference distillation implementation is based on the public REBEL implementation (Gao et al., 2024). Full hyperparameter configurations are given in Table 8. For preference distillation, we use the best γ from the TL;DR experiments; more fine-grained tuning could yield further performance gains.

GRPO and RKL start from the same base model and use the same ArmoRM reward model, with a group size of 5, a batch size of 128, a learning rate of 10^{-6}, and a reference KL coefficient of 10^{-3}. RKL adds an entropy coefficient of 10^{-5}. To examine explicit reference regularization, we also compare PMLE with reference coefficients $\beta = 0$ and 2×10^{-4} while keeping the remaining configuration, preference loss, and evaluation protocol fixed. The nonzero-penalty setting is used in Table 3.

Results. Table 7 reports the general chat results, including PMLE with and without the reference penalty. PMLE and preference distillation achieve higher scores than their corresponding DPO and REBEL baselines on AlpacaEval 2.0 and Arena Hard, while remaining comparable on MT-Bench. Together with the TL;DR results, these findings support the competitiveness of our distribution-learning objectives. The additional runs are reported as point estimates without uncertainty estimates.

Table 7: **General chat results on UltraFeedback**. LLM-as-a-judge evaluations on MT-Bench, AlpacaEval 2.0 (length-controlled and raw win-rate, %), and Arena Hard v0.1 (%).

Model	MT-Bench	AlpacaEval 2.0		Arena Hard
		LC win-rate	Win-rate	
Base (LLaMA-3-8B-Instruct)	8.10	29.62	28.71	26.8
DPO	8.32	44.72	43.94	45.1
PMLE ($\beta = 0$)	8.30	53.23	50.43	46.5
PMLE ($\beta = 2 \times 10^{-4}$)	8.29	49.74	47.19	46.0
REBEL	8.17	52.04	44.99	39.8
Pref. Distill.	8.18	54.49	46.86	41.9
GRPO	8.13	35.97	34.42	32.0
RKL (GRPO)	8.27	36.67	35.18	32.5

[10] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

[11] https://huggingface.co/RLHFlow/ArmoRM-Llama3-8B-v0.1

Table 8: Hyperparameter configurations for general chat experiments.

Setting	**Parameters**	
DPO	batch size: 128 learning rate: 5e-7 schedule: cosine decay train epochs: 1 β: 0.05	
PMLE	batch size: 128 learning rate: 3e-7 schedule: cosine decay train epochs: 1 γ: 5e-3 KL coefficient: 2×10^{-4} (main), 0 (ablation)	
REBEL	batch size: 128 learning rate: 3e-7 schedule: cosine decay train epochs: 1 num epochs: 1 η: 10^6	
Preference distillation	batch size: 128 learning rate: 3e-7 schedule: cosine decay train epochs: 1 β: 0.05	
RLHF (GRPO)	batch size: 128 group size: 5	learning rate: 10^{-6} KL coefficient: 10^{-3} entropy coefficient: 0
RKL (GRPO)	batch size: 128 group size: 5	learning rate: 10^{-6} KL coefficient: 10^{-3} entropy coefficient: 10^{-5}
Generation Config	sampling: true top k: 0.0 top p: 0.9	min length: 1024 max new tokens: 1024 temperature: 0.8