Iterative Preference Learning from Human Feedback: Bridging Theory and Practice for RLHF under KL-constraint

Wei Xiong [*1] **Hanze Dong** [*2] **Chenlu Ye** [*3] **Ziqi Wang** [1] **Han Zhong** [4] **Heng Ji** [1] **Nan Jiang** [1] **Tong Zhang** [1]

Abstract

This paper studies the theoretical framework of the alignment process of generative models with Reinforcement Learning from Human Feedback (RLHF). We consider a standard mathematical formulation, the reverse-KL regularized contextual bandit for RLHF. Despite its widespread practical application, a rigorous theoretical analysis of this formulation remains open. We investigate its behavior in three distinct settings—offline, online, and hybrid—and propose efficient algorithms with finite-sample theoretical guarantees. Moving towards practical applications, our framework, with a robust approximation of the information-theoretical policy improvement oracle, naturally gives rise to several novel RLHF algorithms. This includes an iterative version of the Direct Preference Optimization (DPO) algorithm for online settings, and a multi-step rejection sampling strategy for offline scenarios. Our empirical evaluations on real-world alignment experiment of large language model demonstrate that these proposed methods significantly surpass existing strong baselines, such as DPO and Rejection Sampling Optimization (RSO), showcasing the connections between solid theoretical foundations and their potent practical implementations.

1. Introduction

Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Ziegler et al., 2019) has emerged as a powerful paradigm to align modern generative models like Large Language Models (LLMs) and diffusion models with human values and preferences. This approach has shown significant effectiveness in applications such as ChatGPT (OpenAI, 2023), Claude (Anthropic, 2023), Bard (Google, 2023), and LLaMA2 (Touvron et al., 2023), by making the built AI system helpful, harmless, honest and controllable (Ouyang et al., 2022; Bai et al., 2022).

Despite its effectiveness, RLHF's implementation often involves ad-hoc practices and extensive algorithmic tuning in the entire pipeline, including preference data collection (it is hard to select representative humans (Bai et al., 2022), larger language models (Wang et al., 2024) or program compiler (Wang et al., 2023b)), preference/reward modeling (reward misspecification and misgeneralization (Hong et al., 2022; Gao et al., 2023)), and model optimization (instability of training (Choshen et al., 2019) and distribution shift issue (Michaud et al., 2020; Tien et al., 2022)). Meanwhile, the resulting models of RLHF typically suffer from issues like performance degeneration if we impose strong optimization pressure toward an imperfect reward function (Michaud et al., 2020; Tien et al., 2022; Gao et al., 2023), which contains bias and approximation error from the data collection and preference modeling (Gao et al., 2023; Wang et al., 2023d). Casper et al. (2023) also discussed many other challenges of RLHF. Thus, it is important to understand the mathematical principle of the RLHF process, as well as the connections among its different steps, which should be able to motivate future algorithmic design in principle.

In current RLHF theory, the agent's objective is to maximize an observed reward function, with the optimal policy typically being deterministic and reward-greedy (Agarwal et al., 2019). However, in practical RLHF applications, merely maximizing the reward function is often insufficient and probably results in overfitting, as the generative model must simultaneously ensure both diversity and high fidelity in its outputs. A deterministic maximizer of the reward tends to compromise on these aspects significantly. For example, the maximizer of the "safety reward" tends to avoid providing answers all the time, which contradicts the LLM's training objective. The situation worsens due to bias and approximation errors in reward modeling, leading to the critical problem of reward hacking, where the model often repeats superfluous, pleasing yet irrelevant words to appease the reward model (Michaud et al., 2020; Tien et al., 2022;

---

[*] Equal contribution [1] University of Illinois Urbana-Champaign [2] Salesforce AI Research [3] The Hong Kong University of Science and Technology [4] Work done during an internship at The Hong Kong University of Science and Technology. Correspondence to: Wei Xiong <wx13@illinois.edu>, Tong Zhang <tongzhang@tongzhang-ml.org>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

Casper et al., 2023). Thus, it is important to model diversity and high fidelity in the theoretical framework beyond the reward. Notably, the most widely used mathematical objective function for this goal can be regarded as a reverse-KL regularized contextual bandit problem (Ziegler et al., 2019; Wu et al., 2021a; Ouyang et al., 2022; Rafailov et al., 2023; Liu et al., 2023a). The KL regularized contextual bandit additionally imposes a constraint that the optimal policy cannot move too far away from the original policy (i.e. the starting checkpoint of the LLM). A major difference between this objective function from traditional contextual bandit (Langford & Zhang, 2007) is that the optimal policy is stochastic, which is closer to the practical generative models. See an intuitive illustration why such a target is appealing in Figure 1. Despite numerous proposed procedures for this formulation, a rigorous theoretical analysis remains open. This paper provides a theoretical analysis of the regularized contextual bandit problem in both offline and online settings, aiming to inform and motivate practical algorithmic designs. Our contributions are summarized as follows:

- To understand the mathematical principle of RLHF, we first *formally* formulate the RLHF process as a reverse-KL regularized contextual bandit problem in Section 2, which more accurately reflects real-world alignment practices (Ouyang et al., 2022; Bai et al., 2022; Rafailov et al., 2023) compared to existing theoretical frameworks. Meanwhile, we deliver a comprehensive theoretical analysis in offline, online, and hybrid settings for the formulated framework, where the three settings are complementary to each other and hold their own values in practical applications.
- Moving towards practical applications, in Section 5, we demonstrate that the proposed frameworks can be practically implemented when combined with existing planning algorithms like PPO, DPO, InfoNCA. In other words, our framework is built on the top of existing methods, and can boost their performance;
- Empirically, we show that when combined with DPO, the proposed framework outperforms existing strong baselines like DPO (Rafailov et al., 2023) and RSO (Liu et al., 2023a) in real-world LLM experiments.

Related Work. We defer to Appendix A.1.

2. Formulation and Existing Approaches

In this section, we present the mathematical framework for the RLHF process, inspired by the standard LLM alignment workflow (Ouyang et al., 2022; Touvron et al., 2023).

2.1. Formulation of RLHF

Specifically, the LLM can take a prompt, denoted by $x \in \mathcal{X}$, and produce a response, denoted by $a = [w_1, w_2, \ldots]$, where w_i is the i-th token generated by the model. Accordingly, we can take $\mathcal{X}$ as the state space of the contextual bandit and the $\mathcal{A}$ as the action space. Following Ouyang et al. (2022); Zhu et al. (2023a); Rafailov et al. (2023); Liu et al. (2023a), we assume that there exists a ground-truth reward function $r^*(x,a) : \mathcal{X} \times \mathcal{A} \to [0,1]$ and the preference satisfies the Bradley-Terry model (Bradley & Terry, 1952):

$$\begin{aligned}\mathbb{P}(a^1 \succ a^2 | x, a^1, a^2) &= \frac{\exp(r^*(x,a^1))}{\exp(r^*(x,a^1)) + \exp(r^*(x,a^2))} \\ &= \sigma\big(r^*(x,a^1) - r^*(x,a^2)\big),\end{aligned} \tag{1}$$

where $a^1 \succ a^2$ means that a^1 is preferred to a^2, and $\sigma(z) = 1/(1+\exp(-z))$ is the sigmoid function. We denote an LLM by a policy π that maps x to a distribution over $\mathcal{A}$.

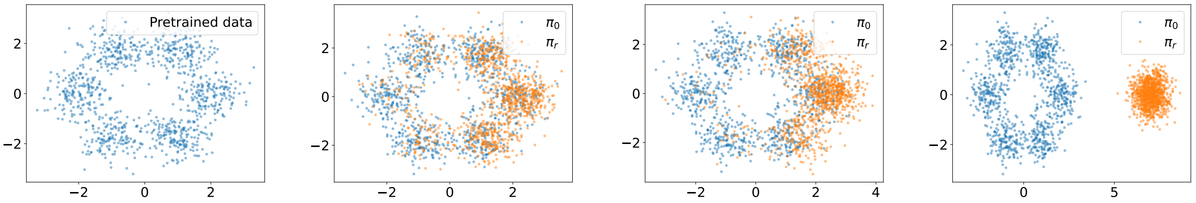
In a typical LLM training pipeline, the tuning process begins with a pretrained LLM, which is subsequently fine-tuned using specialized and instructional data, yielding an initial LLM policy denoted as π_0. We will then align the LLM on RLHF data (prompt set), which we assume is taken from a distribution $x \sim d_0$. For preference learning, the way to gather information from the environment is to compare two different actions under the same state. Considering this, we assume that the agent can perform a pair of actions, aligning with precedents in existing literature (Novoseller et al., 2020; Pacchiano et al., 2021). In applications, we want the resulting LLM π to be close to π_0, and our goal is to find a policy π from some policy class Π to maximize

$$\begin{aligned}J(\pi) &= \mathbb{E}_{x\sim d_0}\mathbb{E}_{a\sim\pi(\cdot|x)}\left[r^*(x,a) + \eta \log \frac{\pi_0(a|x)}{\pi(a|x)}\right] \\ &= \mathbb{E}_{x\sim d_0}\left[\mathbb{E}_{a\sim\pi(\cdot|x)}[r^*(x,a)] - \eta D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))\right],\end{aligned} \tag{2}$$

where $\eta > 0$ is the KL penalty coefficient. This formulation is widely studied in practice (Ziegler et al., 2019; Wu et al., 2021a; Ouyang et al., 2022; Rafailov et al., 2023; Liu et al., 2023a), and our paper aims to study its theoretical property.

Usually, we have a function class $\mathcal{F}$ for approximating the ground truth r^*. Following Pacchiano et al. (2021); Kong & Yang (2022); Zhu et al. (2023a), we make the following assumption for a clear presentation because it suffices to illustrate our ideas and the algorithmic design in this paper can also apply to the general case. The analysis also readily generalizes to general function class using standard complexity measures in RL theory literature (Russo & Van Roy, 2013; Gentile et al., 2022), which essentially state that there are some low-rank structures in reward model.

Assumption 2.1. Assume that the reward function is parameterized by $r_\theta(x,a) = \langle \theta, \phi(x,a) \rangle$ for feature extractor $\phi : \mathcal{X} \times \mathcal{A} \to \mathbb{R}^d$. We also assume that $r^*(x,a) = \langle \theta^*, \phi(x,a) \rangle$ for some $\theta^* \in \mathbb{R}^d$. For regularization, we assume that $\|\phi(x,a)\| \le 1$ for all possible $(x,a) \in \mathcal{X} \times \mathcal{A}$ and $\|\theta\| \le B$. We also denote $\gamma = 1/(2+\exp(-B)+\exp(B))$.



(a) Pretrained π_0 (b) Preferred π_r ($\eta^{-1} = 0.5$) (c) Preferred π_r ($\eta^{-1} = 1.0$) (d) Preferred π_r ($\eta^{-1} = 10$)

Figure 1. A two-dimensional illustrating example of human preference in generative modeling. We consider a scenario where the initial "pretrained" distribution π_0 is represented by a multi-modal Gaussian mixture. The "human preference" is expressed as a bias towards the right, as we set $r = [1, 0]^\top a$. The KL penalty is critical in maintaining the desired behavior of π_r. By varying the values of η, we demonstrate the impact of KL regularization in (b)-(d). As η approaches zero, π_r increasingly focuses on maximizing rewards, often at the expense of the pretrained data's structure, leading to a Gibbs distribution that potentially diverges infinitely.

Notation. We use $\|z\|_\Sigma$ to denote the induced norm $\sqrt{z^\top \Sigma z}$ for some positive-definite matrix. We also define $\phi(x, \pi) := \mathbb{E}_{a\sim\pi(\cdot|x)}\phi(x, a)$ to simplify the presentation. We use $\widetilde{O}$ when we omit the logarithmic factors. A notation table is provided in Table 2 to improve the readability of this paper.

2.2. The Insufficiency of Classic Frameworks

The classic RLHF framework adopted by Ziegler et al. (2019); Ouyang et al. (2022) can be divided into two stages: 1) reward modeling, and 2) policy optimization against the learned reward. We summarize the details in this subsection.

Maximum Likelihood Estimation for reward modeling. Given a preference dataset $\mathcal{D}$ consists of numerous tuples, such as (x, a^1, a^2, y), where y is the preference signal. Specifically, $y = 1$ means a preference for $a^1 \succ a^2$, while $y = 0$ indicates $a^1 \prec a^2$. Given a dataset $\mathcal{D} = \{(x, a^1, a^2, y)\}$, we can write the log-likelihood function of the BT models as follows:

$$\ell_{\mathcal{D}}(\theta) = \sum_{(x,a^1,a^2,y)\in\mathcal{D}} \Big[y \log\Big(\sigma\big(r_\theta(x, a^1) - r_\theta(x, a^2)\big)\Big) + (1 - y)\log\Big(\sigma\big(r_\theta(x, a^2) - r_\theta(x, a^1)\big)\Big)\Big]. \tag{3}$$

We can compute the maximum likelihood estimator (MLE) r_{MLE} based on $\mathcal{D}$ as $\theta_{\mathrm{MLE}} = \operatorname{argmax}_{\theta\in\Theta(B)} \ell_{\mathcal{D}}(\theta)$ with $\Theta(B) = \{\theta \in \mathbb{R}^d : \|\theta\| \le B\}$. In practice, the MLE is also conducted with the LLMs (Ouyang et al., 2022; Bai et al., 2022; Touvron et al., 2023) on the preference dataset.

Policy Optimization and Oracle. With the learned reward in hand, to approximately optimize the target given in Equation (2), we simply call the seminal DRL method PPO with the following regularized reward:

$$\hat{r}(x, a) = r_{\mathrm{MLE}}(x, a) - \eta \log \frac{\pi(a|x)}{\pi_0(a|x)}.$$

To simplify the discussion, we first omit the computational challenges by defining the following information-theoretical policy improvement oracle, and defer a discussion on its practical implementations in Section 5.

Definition 2.2 (Policy Improvement Oracle)**.** For reward function $r : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ and a reference policy π_0, for all $x \in \mathcal{X}$, we can compute the Gibbs policy (Lemma G.6):

$$\begin{aligned}\pi_r(\cdot|x) &:= \operatorname*{argmax}_{\pi\in\Pi} \mathbb{E}_{a\sim\pi(\cdot|x)}\Big[r(x, a) + \eta \log \frac{\pi_0(a|x)}{\pi(a|x)}\Big] \\ &\propto \pi_0(\cdot|x) \cdot \exp\big(\frac{1}{\eta} r(x, \cdot)\big).\end{aligned}$$

Accordingly, we take the policy class as $\Pi := \big\{\pi(\cdot|x) \propto \pi_0(\cdot|x) \cdot \exp\big(\frac{1}{\eta}\langle\theta, \phi(x, \cdot)\rangle\big) : \theta \in \Theta(B)\big\}$. However, even in this ideal case without worrying about finding the best RL configuration, this useful framework suffers from the reward over-optimization issue (Gao et al., 2023).

> Intuitively, *the finite $\mathcal{D}$ cannot cover the whole prompt-response space, and the fine-tuned reward model often performs poorly in the face of out-of-distribution data (Burns et al., 2023).*

Accordingly, the learned r_{MLE} only aligns well with the ground truth r^* in certain distributions.

In addition to the PPO algorithm, the direct preference learning methods have attracted significant attention recently due to its stability and competitive performance (Zhao et al., 2023; Rafailov et al., 2023; Azar et al., 2023; Ethayarajh et al., 2024). We use the DPO as a representative example and the intuition extends to other algorithms. We show that the DPO can be viewed as imposing constraints on the resulting policy by $\mathcal{D}$. Informally, to converge to π^*, the DPO requires an infinite $\mathcal{D}$ to cover the whole prompt-response space. We defer the discussion to Appendix E.

2.3. Preliminary

In this section, we present some useful technical tools and lemmas for subsequent analysis.

Value decomposition. We have the following lemma to decompose the value difference.

Lemma 2.3. *Given a comparator policy π, we can decompose the suboptimality of $\hat{\pi}$ as follows:*

$$J(\pi) - J(\hat{\pi}) = \mathbb{E}_{x\sim d_0}\Big[\mathbb{E}_{\pi}[r^*(x,a) - \hat{r}(x,a)] + \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a) - r^*(x,a)] + \mathbb{E}_{\pi}[\hat{r}(x,a)] - \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a)] + \eta D_{\mathrm{KL}}(\hat{\pi}(\cdot|x)\|\pi_0(\cdot|x)) - \eta D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))\Big],$$

where $\hat{r}: \mathcal{X}\times\mathcal{A}\to\mathbb{R}$ is arbitrary.

Proof. The equality can be verified directly by the definition of $J(\cdot)$ in Equation (2) and basic algebra. □

Policy improvement error. In standard RL setting, $\hat{\pi}$ is typically taken as a greedy policy of $\hat{r}$, leading to

$$\mathbb{E}_{\pi}[\hat{r}(x,a)] - \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a)] \le 0.$$

In the KL-constrained case, since the policy cannot be greedy or deterministic, we need to additionally handle the policy improvement error. The following lemma provides such an estimation when our policy is obtained by calling the Oracle 2.2 with $\hat{r}$.

Lemma 2.4 (Policy optimization error)**.** *Suppose that $\pi, \hat{\pi} \in \Pi$ so that $\pi_0, \pi, \hat{\pi}$ have the same support. If $\hat{\pi}$ is induced by calling Oracle 2.2 with $\hat{r}$, it holds that*

$$\mathbb{E}_{x\sim d_0}\Big[\mathbb{E}_{\pi}[\hat{r}(x,a)] - \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a)] + \eta D_{\mathrm{KL}}(\hat{\pi}\|\pi_0) - \eta D_{\mathrm{KL}}(\pi\|\pi_0)\Big] = -\eta\mathbb{E}_{x\sim d_0} D_{\mathrm{KL}}(\pi\|\hat{\pi}).$$

Here $D_{\mathrm{KL}}(\pi\|\pi_0)$ is short for $D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))$.

We will provide the proof of the lemma in Appendix F. The analysis techniques are most similar to the policy gradient literature since they also consider the soft-max policies (Chen & Teboulle, 1993; Agarwal et al., 2021; Cai et al., 2020; Zanette et al., 2021b; Yuan et al., 2022; Xiao, 2022; Zhong & Zhang, 2023; Uehara et al., 2024; Alfano et al., 2024). The main difference is that in their iterative choices of policy, for choosing π_t, the reference policy they use is the policy of the last round, i.e., π_{t-1}, while we always use the SFT-model π_0 as our reference. We note that their algorithms essentially still use the non-KL-regularized reward as the target because though we prevent the policy from moving too far away in each individual step, the cumulative updates makes the reward estimations dominating in the final policy.

Covariance matrix. Given a preference dataset $\mathcal{D}$, a fixed $\lambda > 0$, we denote $\Sigma_{\mathcal{D}}$ as the covariance matrix estimation:

$$\lambda I + \sum_{(x,a^1,a^2)\in\mathcal{D}} \big(\phi(x,a^1) - \phi(x,a^2)\big)\big(\phi(x,a^1) - \phi(x,a^2)\big)^\top.$$

Both the algorithmic design and analysis will be centered on the covariance matrix. For the readers that are not familiar with the eluder-type techniques (or elliptical potential lemma in this case), we provide a brief introduction to the high-level intuition in Appendix A.3.

3. Offline Learning with Pessimism

3.1. Setup

In this section, we consider the offline setting, where we aim to learn a good policy from a pre-collected dataset without further interactions with the human. We suppose that we are given an offline preference dataset: $\mathcal{D}_{\mathrm{off}} = \{(x_i, a_i^1, a_i^2, y_i)\}_{i=1}^{n_{\mathrm{off}}}$. We denote $\Sigma_{\mathrm{off}} := \Sigma_{\mathcal{D}_{\mathrm{off}}}$ for offline setting. To motivate the algorithmic design, with a comparator policy π, we recall Lemma 2.3 and Lemma 2.4 to obtain that

$$J(\pi^*) - J(\hat{\pi}) = \mathbb{E}_{x\sim d_0}\Big[\mathbb{E}_{\pi}[r^*(x,a) - \hat{r}(x,a)] + \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a) - r^*(x,a)] - \eta\cdot\mathbb{E}_{x\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x)\|\hat{\pi}(\cdot|x))\big]\Big], \tag{4}$$

where $\hat{\pi}$ is induced by calling the Oracle 2.2 with $\hat{r}$. In other words, the sub-optimality depends on the quality of the learned reward $\hat{r}$, under the distributions induced by π and $\hat{\pi}$, separately. Since the finite $\mathcal{D}_{\mathrm{off}}$ from the behavior policy can hardly cover the whole prompt-space, the standard way to handle this issue is to leverage the principle of pessimism with a conservative reward (Jin et al., 2021b; Rashidinejad et al., 2021; Xie et al., 2021a; Zanette et al., 2021b), which means that we adopt an estimator that is a lower bound of the true value with high probability. A technical motivation for doing so is that in Equation (4), the second term $\mathbb{E}_{\hat{\pi}}[\hat{r}(x,a) - r^*(x,a)]$ is hard to control because both the estimation target $(\hat{r} - r^*)$ and the distribution induced by $\hat{\pi}$ depend on $\mathcal{D}_{\mathrm{off}}$. Therefore, they are statistically dependent and characterizing this term is challenging.

In this section, we connect the newly formulated KL-regularized bandit problem with the pessimism and show that the modified variants are sample efficient.

3.2. Algorithms

We introduce two different ways to achieve pessimism. The first one is to directly penalize the reward estimation by an uncertainty estimator $\hat{r}(x,a) = r_{\mathrm{MLE}}(x,a) - \beta\cdot\Gamma(x,a,\nu,\mathcal{D}_{\mathrm{off}})$ so that $\hat{r}(x,a) - r^*(x,a) \le 0$ for all $(x,a)\in\mathcal{X}\times\mathcal{A}$. The construction of the uncertainty bonus is a standard application of concentration inequality. Intuitively, the $\hat{r}$ is an estimation of the ground truth r^*, and $\hat{r}$ will converge to r^* with infinitely many samples that cover the whole feature space well. With finite samples, we can use the statistical tool (concentration inequalities) to quantify the estimation error, in the sense that with high probability,

$$|r^*(x,a) - \hat{r}(x,a)| \le \Gamma(x,a,\nu,\mathcal{D}_{\mathrm{off}}),$$

where ν is a reference vector so that the uncertainty is relative to feature ν, which makes sense under the preference learning nature. We omit the mathematical details here for a clear presentation and we defer the details to Appendix C.

In addition to adopting a reward estimator with point-wise pessimism, we may also use a modified target that is biased toward pessimism by penalizing the uncertainty as in Equation (5). Here we do not maintain a confidence set but use a modified target that is biased toward pessimism, similar to Xie et al. (2021a); Zhang (2022), which may be easier to approximate in practice (Liu et al., 2023b). Moreover, to handle the additional trade-off between the reward and the KL term, we also incorporate the KL divergence into the policy computation.

Algorithm 1 Offline GSHF

```
Input: $\mathcal{D}_{\mathrm{off}}$, $\lambda > 0$, $\beta > 0$, reference vector $\nu$, and prompt distribution $d_0$.
Compute $\theta_{\mathrm{MLE}}$ based on $\mathcal{D}_{\mathrm{off}}$ by maximizing the likelihood given in Equation (3)
Option I: Output $\hat{\pi}$ by constructing expected uncertainty estimator $\Gamma^e(\pi,\nu,\mathcal{D}_{\mathrm{off}})$ and solving
    $\hat{\pi} = \operatorname*{argmax}_{\pi\in\Pi} \Big[\mathbb{E}_{x\sim d_0, a\sim\pi(\cdot|x)}[r_{\mathrm{MLE}}(x,a)] - \beta\cdot\Gamma^e(\pi,\nu,\mathcal{D}_{\mathrm{off}}) - \eta\cdot\mathbb{E}_{x\sim d_0}[D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))]\Big].$   (5)
Option II: Output $\hat{\pi}$ by constructing uncertainty estimator $\Gamma(x,a,\nu,\mathcal{D}_{\mathrm{off}})$ and calling Oracle 2.2 with $\hat{r}(x,a) = r_{\mathrm{MLE}}(x,a) - \beta\cdot\Gamma(x,a,\nu,\mathcal{D}_{\mathrm{off}})$.
```

The full algorithmic framework is presented in Algorithm 1 and is referred to as the offline **Gibbs Sampling from Human Feedback** (GSHF) because the output policy is the Gibbs distribution with some reward.

3.3. Main Results: Pessimism is Provably Efficient

We now present the main theoretical guarantee for Offline GSHF.

Theorem 3.1. *Under Assumption 2.1, if we set* $\beta := O\big(\sqrt{\frac{d+\log(1/\delta)}{\gamma^2} + \lambda B^2}\big)$*, for any* $\lambda > 0$ *and comparator policy* $\pi \in \Pi$*, with probability at least* $1-\delta$*, the output policy of Algorithm 1 with Option I and* $\Gamma^e(\pi,\nu,\mathcal{D}_{\mathrm{off}}) = \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi)-\nu]\|_{\Sigma_{\mathrm{off}}^{-1}}$ *satisfies*

$$J(\pi) - J(\hat{\pi}) \le 2\beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi)] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}},$$

and the output policy of Algorithm 1 with Option II and $\Gamma(x,a,\nu,\mathcal{D}_{\mathrm{off}}) = \|\phi(x,a)-\nu\|_{\Sigma_{\mathrm{off}}^{-1}}$ *satisfies*

$$\begin{aligned} J(\pi) - J(\hat{\pi}) \le\ & 2\beta\cdot\mathbb{E}_{x\sim d_0, a\sim\pi(\cdot|x)}\|\phi(x,a)-\nu\|_{\Sigma_{\mathrm{off}}^{-1}} \\ & - \eta\cdot\mathbb{E}_{x\sim d_0}\big[D_{\mathrm{KL}}(\pi(\cdot|x)\|\hat{\pi}(\cdot|x))\big]. \end{aligned}$$

We can combine the guarantee with dataset property, usually referred to as the coverage on the comparator policy π (Jin et al., 2021b; Xie et al., 2021a), to obtain the concrete bound. See Proposition D.1 for an concrete example. The proof of the theorem is rather standard in offline learning based on the principle of pessimism but with a different analysis to handle the KL and the stochastic policy. We defer the proof of the theorem to Appendix C. The reference vector ν in Algorithm 1 is typically set as $\mathbb{E}_{x\sim d_0}[\phi(x,\pi_{\mathrm{ref}})]$ for some available π_{ref}. As showcased by Zhu et al. (2023a), the subtracted reference vector can serve as a pre-conditioner for a better suboptimality bound. For instance, one typically choice is $\pi_{\mathrm{ref}} = \pi_0$ so that π_0 achieves a reward of zero (Ouyang et al., 2022; Gao et al., 2023; Dong et al., 2023).

Comparison of two implementations of pessimism. In comparison, the Option I achieves a sharper bound in the uncertainty bonus because the expectation is inside the norm and by Jensen's inequality (Lemma G.1) we know that

$$\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi)]-\nu\|_{\Sigma_{\mathrm{off}}^{-1}} \le \mathbb{E}_{x\sim d_0, a\sim\pi(\cdot|x)}\|\phi(x,a)-\nu\|_{\Sigma_{\mathrm{off}}^{-1}}.$$

Moreover, Option I has a desirable robust improvement property. If we take $\nu = \mathbb{E}_{x\sim d_0}[\phi(x,\pi_{\mathrm{ref}})]$, the resulting policy will be better than π_{ref}, regardless of the coverage of the $\mathcal{D}_{\mathrm{off}}$ according to Theorem 3.1, which is similar to the original offline RL literature for a robust policy improvement (Bhardwaj et al., 2023). We will also see that the use of a reference policy π_{ref} can also simplify the algorithmic design in subsequent Section 4. However, the main advantage of Option II is that the Oracle 2.2 can be empirically well approximated. For instance, we can directly plug the pessimistic reward into the PPO algorithm. Moreover, we have the following algorithm in a direct preference learning manner that resembles that of Rafailov et al. (2023); Zhao et al. (2023); Azar et al. (2023). We defer a detailed discussion to Section 5.

4. Online Iterative Learning with Batch Exploration

4.1. Setup: Batch Hybrid Learning

Beyond the offline learning, it is also common to query human feedback during the training process. For instance, Bai et al. (2022); Touvron et al. (2023) typically iterate the RLHF process on a weekly cadence, where the fresh RLHF models are deployed to interact with crowdworkers and to collect new human preference data.

We consider a slightly more general setting here, where we refer it to as the hybrid learning. This is because while it is possible to learn from scratch, in many cases, we tend to start with the offline open-source datasets (Touvron et al., 2023; Bai et al., 2023). For instance, in LLaMA2 (Touvron et al., 2023), the authors start with 1500K open-source comparison pairs $\mathcal{D}_{\mathrm{off}}$ and keep $\mathcal{D}_{\mathrm{off}}$ in the data mixture for the entire RLHF process. Motivated by the practical applications, we formulate the process as a batch hybrid framework in this section. It is shown that such a batch online framework can significantly improve the aligned LLMs as evaluated by the humans (Bai et al., 2022; Touvron et al., 2023). For completeness, we also develop the pure online setting in Appendix B. Mathematically, consider the batch hybrid setting of T batches with a fixed batch size m.

The agent initializes with the $\mathcal{D}_{\mathrm{off}}$ (if applicable, otherwise initialize as empty). For each iteration $t \in [T]$,

- we first update the policy pair (π_t^1, π_t^2) based on the historical data collected so far: $\mathcal{D}_{\mathrm{off}}$ and $\mathcal{D}^{1:t-1}$;
- we collect m tuples as $\mathcal{D}^t$: we sample a random prompt by $x_{t,i} \sim d_0$, collect two responses by $(a_{t,i}^1, a_{t,i}^2) \sim (\pi_t^1, \pi_t^2)$, and query the preference signal $y_{t,i}$ from the underlying BT model;
- the next iteration begins.

The main technical challenge here is to decide the behavior policy pairs (π_t^1, π_t^2). It is well known that it is important to strike a balance between exploration and exploitation to get RL works (Auer et al., 2002), and we study such a trade-off in the context of online iterative RLHF.

4.2. Algorithms

Non-symmetric algorithmic structure. Our first idea is to adopt a non-symmetric structure in choosing π_t^1 and π_t^2. Specifically, we refer the π_t^1 as the main agent, which aims to learn a good policy so that the suboptimality gap $J(\pi^*) - J(\pi_t^1)$ is small. In contrast, the second agent, referred to as the enhancer, seeks to enhance the learning of the main agent by choosing appropriate π_t^2. The main advantage of such a non-symmetric structure is that we have a lot of freedoms to choose π_t^2 because we do not worry about the sub-optimality incurred by it (Jin et al., 2021a; Xiong et al., 2022b; Huang et al., 2021).

From a high level, the Online Iterative GSHF Algorithm continuously enhances the historical dataset by strategically interacting with the human labeller. Specifically, in Algorithm 2, the main agent always exploits all the historical information by taking the optimal policy induced from the MLE r_{MLE}. This process, however, requires the newly collected data can provide more information, compared to those collected previously. We divide the problem into two different situations.

Algorithm 2 Online Iterative GSHF

```
Input: $m, \mathcal{D}_{\mathrm{off}}$ (if applicable, otherwise $\mathcal{D}_{\mathrm{off}} = \emptyset$), $\beta$.
for $t = 1, 2, \cdots, T$ do
    Observe $x_{t,i} \sim d_0$ for $i = 1, 2, \cdots, m$ [1].
    Exploitation with the main agent: denote the MLE $r_{\mathrm{MLE}}$ with the likelihood in Equation (3) and compute the best guess we have so far:
        $$\begin{aligned} \pi_t^1 = \mathop{\mathrm{argmax}}_{\pi \in \Pi} \mathbb{E}_{x \sim d_0} \mathbb{E}_{a \sim \pi(\cdot|x)} \Big[ & r_{\mathrm{MLE}}(x, a) \\ & - \eta D_{\mathrm{KL}}(\pi(\cdot|x) \| \pi_0(\cdot|x)) \Big]. \end{aligned} \quad (6)$$
    Exploration with the enhancer:
    Option I: if $\mathcal{D}_{\mathrm{off}}$ can provide good coverage, no need to explore so $\pi_t^2 = \pi_{\mathrm{ref}}$;
    Option II: otherwise, choose $\pi_t^2$ by maximizing the relative uncertainty relative to $\pi_t^1$:
        $$\mathop{\mathrm{argmax}}_{\widetilde{\pi} \in \Pi^t} \Gamma_t^m(\lambda, \pi_t^1, \widetilde{\pi}) := \beta \cdot \sum_{i=1}^m \|\phi(x_{t,i}, \widetilde{\pi}) - \phi(x_{t,i}, \pi_t^1)\|_{\Sigma_{t,m}^{-1}}, \quad (7)$$
        where the confidence set is given in Equation (8).
    Sample $a_{t,i}^1 \sim \pi_t$ and $a_{t,i}^2 \sim \pi_t^2$, receive human feedback for all $i \in [m]$, and collect them as $\mathcal{D}^t$.
end for
Output: the best model in $\pi_{1:T}^1$ by a validation set.
```

Option I. If the $\mathcal{D}_{\mathrm{off}}$ is diverse enough and provides us with a good coverage in terms of the $(\pi^*, \pi_{\mathrm{ref}})$, we do not need to actively explore and can take $\pi_t^2 = \pi_{\mathrm{ref}}$ directly[2];

Option II. If we cannot expect to have a diverse $\mathcal{D}_{\mathrm{off}}$, the enhancer aims to maximizes the uncertainty relative to the main agent π_t^1, while maintaining a moderate KL divergence. In this case, the π_t^2 aims to explore toward the direction that maximizes the uncertainty relative the main agent π_t^1. We additionally impose a constraint:

$$\begin{aligned} \widetilde{\pi} \in \Pi_t = \{\pi' \in \Pi : & \underbrace{\eta \cdot \mathbb{E}_{x \sim d_0} D_{\mathrm{KL}}(\pi'(\cdot|x), \pi_t^1(\cdot|x))}_{\text{How far does the enhancer go.}} \\ \leq & \underbrace{\Gamma_t^m(\lambda, \pi_t^1, \pi')}_{\text{How much information we can get.}} \quad \}, \end{aligned} \quad (8)$$

where $\Sigma_{t,m} = \lambda I + \frac{1}{m} \sum_{i=1}^{t-1} \sum_{j=1}^m (\phi(x_{i,j}, a_{i,j}^1) - \phi(x_{i,j}, a_{i,j}^2))(\phi(x_{i,j}, a_{i,j}^1) - \phi(x_{i,j}, a_{i,j}^2))^\top$. This set is never empty as π_t^1 always belongs to it so the choice is well-defined. Intuitively, we require that the information we can get is worth the cost we pay by moving away from π_t^1.

Essentially, we are trying to boost our dataset by strategically choose the behavior policies at each iteration. If we learn from scratch or the offline dataset is not good enough, we need to explicitly incorporate the uncertainty into algorithmic design, and explore the direction where we are uncertain about so that we can gain more information. On

---

[2] We also mention in passing that for most of the cases, $\pi_{\mathrm{ref}} = \pi_0$. We use π_{ref} for a slightly more general formulation.

the other hand, if $\mathcal{D}_{\text{off}}$ is already good enough, our analysis shows that it is also beneficial to collect more online data, as we now discuss in the next subsection.

4.3. Highlight: RLHF Benefits from Online Exploration

We first consider the case of Option I: $\pi_t^2 = \pi_{\text{ref}}$. In this case, it is essential to have a diverse $\mathcal{D}_{\text{off}}$ as we do not explicitly explore. This is most related to the study of hybrid RL in the classic RL theory (Song et al., 2022; Zhou et al., 2023). The major difference here is that for preference-based learning, the uncertainty is evaluated on the feature difference instead of a single state-action pair, as we summaize in the following assumption.

Assumption 4.1 (Partial Coverage of Offline Data)**.** For the linear model, there exists a reference policy π_{ref}, a ratio coefficient $\alpha = \alpha(mT, \mathcal{D}_{\text{off}}) \in (0,1)$ and a coverage constant $C_{\text{cov}} > 0$ such that

$$(mT)^{1-\alpha}\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\text{ref}})]\|_{(\Sigma_{\text{off}})^{-1}} \le C_{\text{cov}}.$$

We remark that Assumption 4.1 implicitly assume that n_{off} is comparable to the total number of online samples mT so that the influence of $\mathcal{D}_{\text{off}}$ will not be dominated by the online data. To provide a more detailed understanding and connection to existing literature, we offer a more nuanced characterization of $\alpha(mT, \mathcal{D}_{\text{off}})$ under standard partial coverage conditions in Appendix D.1. In particular, when $mT \approx n_{\text{off}}$, we show that $\alpha(mT, \mathcal{D}_{\text{off}}) \approx 1/2$. It is worth emphasizing that this scenario appears to be realistic for LLMs. For example, in the LLaMA2 project (Touvron et al., 2023), we observe $n_{\text{off}} = 1500K$ and $mT = 1400K$. We are ready to present the results.

Theorem 4.2. *For any $\epsilon > 0$, under Assumption 2.1 with $T = \min\{n \in \mathbb{N}^+ : n \ge d\log(n)\}$ and $\beta = O\big(\sqrt{\frac{d\log(T/\delta)}{\gamma^2}}\big)$, with probability at least $1-3\delta$, there exists a $t_0 \in [T]$ such that Algorithm 2 with Option I holds*

$$J(\pi^*) - J(\pi_{t_0}) \lesssim \beta \cdot \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\text{ref}})]\|_{\Sigma^{-1}_{\text{off}+\mathcal{D}^{1:t_0}}} + \sqrt{\frac{d}{\gamma^2 m}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\text{KL}}(\pi^1_{t_0}(\cdot|x_{t_0})\|\pi^*(\cdot|x_{t_0}))\big],$$

where $\Sigma_{\text{off}+\mathcal{D}^{1:t_0}}$ denotes the covariance matrix computed on $\mathcal{D}_{\text{off}} \cup \mathcal{D}^{1:t_0}$. If we further assume that Assumption 4.1 holds, we have

$$J(\pi^*) - J(\pi_{t_0}) \lesssim \sqrt{\frac{d^2}{\gamma^2 m}} + \beta(mT)^{\alpha(mT,\mathcal{D}_{\text{off}})-1}C_{\text{cov}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\text{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}(\cdot|x_{t_0}))\big].$$

The proof is deferred to Appendix D.3. Note that the second result is a conservative bound where we completely ignore the coverage provided by the online data $\mathcal{D}^{1:t_0-1}$. We now analyze the impact of the online data as follows.

RLHF Benefits from Online Exploration. One natural question arises under Assumption 4.1: if we can directly apply Algorithm 1 to get a good policy, why should we collect online data? The difference is that now the **second term** corresponds to the coverage condition of $\mathcal{D}_{\text{off}} \cup \mathcal{D}^{1:t_0}$. Under Assumption 4.1, with suitable hyper-parameters (large enough m and suitable β), we know that $\pi_t^1 \to \pi^*$. Since the online data is collected by $(\pi_t^1, \pi_{\text{ref}})$ and the goal is to cover $(\pi^*, \pi_{\text{ref}})$, we expect that the intermediate policies can provide a much better coverage as compared to the $\mathcal{D}_{\text{off}}$, i.e., a much smaller C_{cov}, for many average instances. We will partially verify this intuition in the experiment part.

We now move to the case of Option II where we cannot expect to have a diverse $\mathcal{D}_{\text{off}}$ and need explicit exploration. We first show that with suitable β, the constructed confidence set contains π^* with high probability.

Lemma 4.3 (Confidence set)**.** *For the linear model in Assumption 2.1, given the policy of the main agent π_t^1, we consider the confidence set defined in Equation (8) with $\beta = O\big(\sqrt{\frac{d\log(T/\delta)}{\gamma^2 m}}\big)$. Then, with probability at least $1-\delta$, we know that $\pi^* \in \Pi_t$ for all $t \in [T]$.*

We defer the proof to Appendix B.3 and present the main result for the Option II.

Theorem 4.4. *For any $\epsilon > 0$, we set the batch size $m = d/(\gamma^2\epsilon^2)$. Under Assumption 2.1, with $\beta := O\big(\sqrt{\frac{d\log(T/\delta)}{\gamma^2 m}}\big)$ and $\lambda = \Theta\big(d\log(T/\delta)/(m\gamma^2 B^2)\big)$, after $T = \min\{n \in \mathbb{N}^+ : n \ge d\log(n)\}$ iterations, we have with probability at least $1-3\delta$, Algorithm 2 with Option II satisfies: there exists a $t_0 \in [T]$,*

$$J(\pi^*) - J(\pi^1_{t_0}) \lesssim \epsilon - \eta \cdot \mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\text{KL}}(\pi^*(\cdot|x_{t_0})\|\pi^1_{t_0}(\cdot|x_{t_0}))\big],$$

where the required samples is at most $mT = \widetilde{O}\Big(\frac{d^2}{\gamma^2\epsilon^2}\Big)$.

Getting Rid of Data Coverage. One notable feature of Theorem 4.4 is that with explicit exploration, we do not need an offline dataset with good coverage, highlighting the importance of strategic explorations. This is particularly important in the context of LLMs because the distribution shift between LLMs are very large so data coverage is more sparse. For instance, along the way of finding the optimal policy of some learned reward function by PPO, the KL divergence to the initial checkpoint can be > 25 (Bai et al., 2022).

Algorithmic Simplicity v.s. Data Coverage. We note that Option I and Option II are complementary to each other and hold their own values. Specifically, the Option I offers simplicity in algorithmic design, at the cost of demand for a high-quality $\mathcal{D}_{\text{off}}$. In comparison, the online learning does not relies on the quality of $\mathcal{D}_{\text{off}}$, but the choice of the enhancer is challenging because for the neural network, the uncertainty estimators do not admit a closed-form. In practice, we typically resort to heuristic methods (Wu et al., 2021b; Coste et al., 2023) to estimate the uncertainty, as we discuss in the experiment part of this paper.

The advantage of reward modeling. Theorem 4.2 and Theorem 4.4 reveal a key characteristic of reward modeling: the sample complexity is dependent on the complexity of the reward model rather than the generative models. For simple reward functions, such as sentiment or politeness evaluation, the required function class is substantially smaller compared to the generative model. This is corroborated by evidence showing that even compact models like BERT (Devlin et al., 2018) can yield accurate reward assessments. This may illustrate the advantage of the most popular RLHF framework used by Ouyang et al. (2022); Bai et al. (2022); Touvron et al. (2023), in contrast to the idea of bypassing reward modeling (Rafailov et al., 2023; Zhao et al., 2023; Azar et al., 2023) and training based only on the offline dataset.

5. Implementations of GSHF

In this section, we discuss how to practically implement the information-theoretical Algorithms. From a high level, the GSHF framework can be implemented by combining many existing algorithms to approximate the computational Oracle 2.2, including PPO, DPO, and InfoNCA (Chen et al., 2024). Here, we focus on the PPO and DPO and discuss several popular ways to implement it.

In practice, the policy is represented by a deep neural network. In this case, one common choice (Ziegler et al., 2019; Wu et al., 2021a; Ouyang et al., 2022; Bai et al., 2022) is to use the standard deep RL algorithms like PPO to optimize the regularized reward: $\widetilde{r}(x,a) = r(x,a) - \eta \log \frac{\pi_\theta(a|x)}{\pi_0(a|x)}$. However, PPO is significantly less stable and sensitive to implementation as compared to SFT (Choshen et al., 2019; Engstrom et al., 2020). Recently, another line of work considers a family of algorithms that directly learn from the preference data without reward modeling, including the DPO (Rafailov et al., 2023), SLIC (Zhao et al., 2023), IPO (Azar et al., 2023). These algorithms attracted significant attention due to its stability and easy implementation. We use the DPO as a representative example here. Specifically, DPO chooses to train the LLM as a reward model, by optimizing the following loss:

$$\sum_{(x,a^w,a^l)\in\mathcal{D}_{\text{off}}} -\Big[\log\sigma\Big(\eta\log\frac{\pi_\theta(a^w|x)}{\pi_0(a^w|x)} - \eta\log\frac{\pi_\theta(a^l|x)}{\pi_0(a^l|x)}\Big)\Big], \tag{9}$$

where a^w, a^l is the chosen/rejected response. It is shown that the optimal policy for the DPO loss in Equation (9) is identical to the one for the RLHF objective π_r when r is the MLE (Azar et al., 2023). To fit the DPO into the GSHF framework, we generalize this result to incorporate the pessimism.

5.1. Direct Preference Learning with Pessimism

For notation simplicity, we denote the uncertainty bonus as $\Gamma(x,a)$ and omit the dependency on ν and $\mathcal{D}_{\text{off}}$. Then, we have the following proposition.

Proposition 5.1 (Direct Preference Learning with Pessimism)**.** *Given the preference dataset $\mathcal{D}_{\text{off}}$, we can implement Option II of Algorithm 1 by minimizing the following loss function $\mathcal{L}_{\mathcal{D}_{\text{off}}}(\theta,\pi_0)$:*

$$\sum_{(x,a^w,a^l)\in\mathcal{D}_{\text{off}}} \log\sigma\Big(\eta\log\frac{\pi_\theta(a^w|x)}{\pi_0(a^w|x)} - \eta\log\frac{\pi_\theta(a^l|x)}{\pi_0(a^l|x)} + \underbrace{(\Gamma(x,a^w) - \Gamma(x,a^l))}_{m(x,a^w,a^l)}\Big), \tag{10}$$

where a^w is preferred over a^l.

Intuitively, we add an adaptive margin for each preference pair (x, a^w, a^l) according to their uncertainty difference. We defer the proof to Appendix C.2.

Uncertainty Estimation via Ensemble. The uncertainty estimation for the general neural network is still an open problem. In practical applications, we typically resort to heuristic methods. For instance, Coste et al. (2023) uses the idea of ensemble to get the pessimistic reward model. Specifically, they independently train 5 reward models $\{r_i\}_{i=1}^5$ and use the empirical variance as the uncertainty estimation. It is shown that such a pessimistic version of the reward model can significantly reduce the reward over-optimization for PPO and best-of-n sampling (Nakano et al., 2021).

5.2. Enhancer Explores with Main Agent Policy

For the online exploration, selecting an appropriate optimistic policy for the enhancer to maximize the uncertainty with respect to the main agent π_t^1 as in Equation (8) is largely less explored in practical applications. While it is challenging to obtain the analytical solution of uncertainty, the insight is to maximize the policy difference with π_t^1, while maintaining a moderate KL divergence. We discuss some popular heuristic implementations here.

Model Variants. In the project of Claude (Bai et al., 2022), the authors choose to use the models with different training steps as (π_t^1, π_t^2). For instance, if we run PPO for 2 epoch in total, we may take π_t^1 as the model saved at the end of the first epoch and take π_t^1 as the one saved at the end of second epoch. Moreover, in addition to the model variants, the LLaMA-2 project (Touvron et al., 2023) further adjusts the sampling temperature of π_t^1 to induce π_t^2.

Models	Settings	Gold Reward	Gold WR	GPT4 Eval	OOD Gold Reward	Diff $\Delta \downarrow$	OOD Gold WR	OOD GPT4 Eval
SFT	Offline	0.27	-	-	-0.21	0.48	-	-
DPO	Offline	2.15	0.5	0.5	1.71	0.44	0.5	0.5
RSO	Offline	2.25	0.54	0.53	1.89	0.36	0.55	0.52
Multi-step RSO	Offline	2.59	0.63	0.57	2.41	**0.18**	0.64	**0.60**
Hybrid-GSHF-DPO	Hybrid	**2.67**	**0.67**	**0.65**	**2.46**	0.21	**0.66**	0.59

Table 1. The evaluation results of the models from different RLHF algorithms. WR refers to Win Rate The gold rewards are computed on the test split with 5K prompts and the GPT4 evaluations are with 100 randomly sampled test prompts, with the DPO as baseline. We use 5K prompts from the UltraFeedback to compute the OOD reward and Δ is the difference between the in-domain test reward and the OOD one. We count GPT4 evaluation score as win $\times$ $1 +$ tie $\times$ 0.5 and provide the details in Table 4.

Rejection Sampling. A popular ensemble-based approach is the rejection sampling (Nakano et al., 2021; Dong et al., 2023; Liu et al., 2023a). We present a brief introduction to the concept of rejection sampling in Appendix A.2. In the context of LLM, however, the rejection sampling is usually restricted to the best-of-n sampling. Specifically, we sample n independent responses by π_t^1 for each prompt, and then use a reward function to rank them and take the one with the highest reward as the final output. In other words, we take π_t^2 as the best-of-n variant of π_t^1. In this way, the π_t^2 enlarges the margins between π_t^1 and $D_{\mathrm{KL}}(\pi_t^1, \pi_t^2)$ is upper bounded by $\log n - \frac{n-1}{n}$ and is usually far better than this conservative estimation (Beirami et al., 2024). We note that similar idea has been adopted (Liu et al., 2023a; Hoang Tran, 2024; Yuan et al., 2024) for improving DPO.

6. Experiments

We present some proof-of-concept experiments to verify the theoretical findings and refer interested readers to Dong et al. (2024) for a comprehensive recipe of online iterative RLHF. For a clear presentation, we refer the Algorithm 2 with Option I as *Hybrid GSHF*.

To illustrate the improvement from the online exploration, we compare different iterations of Hybrid-GSHF-DPO in Figure 2. For each iteration, we evaluate the models every 400 training steps and plot the representative models. Clearly, the previous iteration is strictly dominated by the subsequent one in terms of the frontier of reward-KL trade-off. This demonstrates the significant improvements achieved by further iterating DPO with online data. In particular, compared to offline DPO which uses more offline data than the iteration 1, leveraging online data proves to be far more efficient, as evidenced by the enhanced frontier of the reward-KL trade-off.

Due to space constraint, we only present the main results in Table 1. The baseline methods, multi-step RSO, and the detailed experimental setup are introduced in Appendix H. The code is publicly available at `https://github.com/RLHFlow/Online-RLHF`.

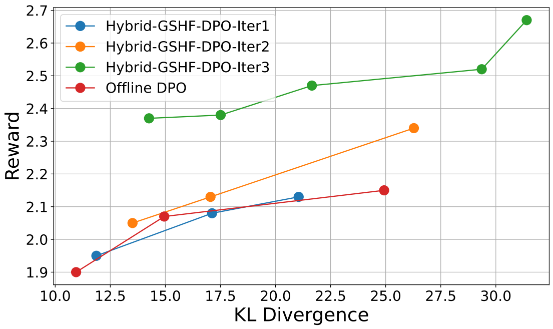


Figure 2. The Reward-KL trade-off curves of different iterations of Hybrid-GSHF-DPO. The rightest point is the highest gold reward that can be achieved in that round.

7. Conclusion

In this paper, we formulate the real-world RLHF process as a reverse-KL regularized contextual bandit problem. Compared to existing theoretical RLHF frameworks, the proposed framework admits a stochastic optimal policy, that more accurately reflects the dynamics of foundation generative models and aligns closely with current alignment practices (Ouyang et al., 2022; Bai et al., 2022; Rafailov et al., 2023). We design statistically efficient algorithms in offline, online, and hybrid settings, featuring the standard ideas of pessimism and optimism in the new framework, while also handling the distinct challenges of preference learning as well as the newly introduced KL constraint with distinct algorithmic designs.

The theoretical findings also sheds light on innovative pathways for practical algorithmic development, as we move toward implementations of the information-theoretical algorithms in Section 5. The practical implementations of the proposed algorithms outperform strong baselines like DPO and RSO in real-world alignment of LLMs.

Acknowledgements

The authors would like to thank the anonymous reviewers for many insightful comments and suggestions.

Impact Statement

The goal of this paper is to advance the field of alignment for foundation language models. We study both of the theoretical formulation and the implied practical algorithmic designs. The proposed frameworks can help us to better align the strong LLMs with human ethics and preference, thus making the LLMs more helpful, and controllable and contributing to the welfare of society.

References

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

Agarwal, A., Jiang, N., Kakade, S. M., and Sun, W. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, 32, 2019.

Agarwal, A., Kakade, S. M., Lee, J. D., and Mahajan, G. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *The Journal of Machine Learning Research*, 22(1):4431–4506, 2021.

Agarwal, A., Jin, Y., and Zhang, T. Vo q l: Towards optimal regret in model-free rl with nonlinear function approximation. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 987–1063. PMLR, 2023.

Alfano, C., Yuan, R., and Rebeschini, P. A novel framework for policy mirror descent with general parameterization and linear convergence. *Advances in Neural Information Processing Systems*, 36, 2024.

Anthropic. Introducing claude. 2023. URL `https://www.anthropic.com/index/introducing-claude`.

Askell, A., Bai, Y., Chen, A., Drain, D., Ganguli, D., Henighan, T., Jones, A., Joseph, N., Mann, B., DasSarma, N., et al. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*, 2021.

Auer, P., Cesa-Bianchi, N., and Fischer, P. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47:235–256, 2002.

Azar, M. G., Rowland, M., Piot, B., Guo, D., Calandriello, D., Valko, M., and Munos, R. A general theoretical paradigm to understand learning from human preferences. *arXiv preprint arXiv:2310.12036*, 2023.

Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

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

Bansal, H., Dang, J., and Grover, A. Peering through preferences: Unraveling feedback acquisition for aligning large language models. *arXiv preprint arXiv:2308.15812*, 2023.

Beirami, A., Agarwal, A., Berant, J., D'Amour, A., Eisenstein, J., Nagpal, C., and Suresh, A. T. Theoretical guarantees on the best-of-n alignment policy. *arXiv preprint arXiv:2401.01879*, 2024.

Bengs, V., Busa-Fekete, R., El Mesaoudi-Paul, A., and Hüllermeier, E. Preference-based online learning with dueling bandits: A survey. *The Journal of Machine Learning Research*, 22(1):278–385, 2021.

Bhardwaj, M., Xie, T., Boots, B., Jiang, N., and Cheng, C.-A. Adversarial model for offline reinforcement learning. *arXiv preprint arXiv:2302.11048*, 2023.

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

Burns, C., Izmailov, P., Kirchner, J. H., Baker, B., Gao, L., Aschenbrenner, L., Chen, Y., Ecoffet, A., Joglekar, M., Leike, J., et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. *arXiv preprint arXiv:2312.09390*, 2023.

Cai, Q., Yang, Z., Jin, C., and Wang, Z. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pp. 1283–1294. PMLR, 2020.

Casper, S., Davies, X., Shi, C., Gilbert, T. K., Scheurer, J., Rando, J., Freedman, R., Korbak, T., Lindner, D., Freire, P., et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023.

Chen, G. and Teboulle, M. Convergence analysis of a proximal-like minimization algorithm using bregman functions. *SIAM Journal on Optimization*, 3(3):538–543, 1993.

Chen, H., He, G., Su, H., and Zhu, J. Noise contrastive alignment of language models with explicit rewards. *arXiv preprint arXiv:2402.05369*, 2024.

Chen, X., Zhong, H., Yang, Z., Wang, Z., and Wang, L. Human-in-the-loop: Provably efficient preference-based reinforcement learning with general function approximation. In *International Conference on Machine Learning*, pp. 3773–3793. PMLR, 2022.

Choshen, L., Fox, L., Aizenbud, Z., and Abend, O. On the weaknesses of reinforcement learning for neural machine translation. *arXiv preprint arXiv:1907.01752*, 2019.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Coste, T., Anwar, U., Kirk, R., and Krueger, D. Reward model ensembles help mitigate overoptimization. *arXiv preprint arXiv:2310.02743*, 2023.

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

Dani, V., Hayes, T. P., and Kakade, S. M. Stochastic linear optimization under bandit feedback. 2008.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Diao, S., Pan, R., Dong, H., Shum, K. S., Zhang, J., Xiong, W., and Zhang, T. Lmflow: An extensible toolkit for finetuning and inference of large foundation models. *arXiv preprint arXiv:2306.12420*, 2023.

Dong, H., Xiong, W., Goyal, D., Zhang, Y., Chow, W., Pan, R., Diao, S., Zhang, J., SHUM, K., and Zhang, T. RAFT: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL `https://openreview.net/forum?id=m7p507zblY`.

Dong, H., Xiong, W., Pang, B., Wang, H., Zhao, H., Zhou, Y., Jiang, N., Sahoo, D., Xiong, C., and Zhang, T. Rlhf workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*, 2024.

Engstrom, L., Ilyas, A., Santurkar, S., Tsipras, D., Janoos, F., Rudolph, L., and Madry, A. Implementation matters in deep policy gradients: A case study on ppo and trpo. *arXiv preprint arXiv:2005.12729*, 2020.

Ethayarajh, K., Xu, W., Muennighoff, N., Jurafsky, D., and Kiela, D. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Faury, L., Abeille, M., Calauzènes, C., and Fercoq, O. Improved optimistic algorithms for logistic bandits. In *International Conference on Machine Learning*, pp. 3052–3060. PMLR, 2020.

Gao, L., Schulman, J., and Hilton, J. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pp. 10835–10866. PMLR, 2023.

Geng, X. and Liu, H. Openllama: An open reproduction of llama, May 2023. URL `https://github.com/openlm-research/open_llama`.

Gentile, C., Wang, Z., and Zhang, T. Fast rates in pool-based batch active learning. *arXiv preprint arXiv:2202.05448*, 2022.

Google. Bard. 2023. URL `https://bard.google.com/`.

Gulcehre, C., Paine, T. L., Srinivasan, S., Konyushkova, K., Weerts, L., Sharma, A., Siddhant, A., Ahern, A., Wang, M., Gu, C., et al. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*, 2023.

Hao, Y., Chi, Z., Dong, L., and Wei, F. Optimizing prompts for text-to-image generation. *arXiv preprint arXiv:2212.09611*, 2022.

Hoang Tran, Chris Glaze, B. H. Snorkel-mistral-pairrm-dpo. 2024. URL `https://huggingface.co/snorkelai/Snorkel-Mistral-PairRM-DPO`.

Hong, J., Bhatia, K., and Dragan, A. On the sensitivity of reward inference to misspecified human models. *arXiv preprint arXiv:2212.04717*, 2022.

Hu, P., Chen, Y., and Huang, L. Nearly minimax optimal reinforcement learning with linear function approximation. In *International Conference on Machine Learning*, pp. 8971–9019. PMLR, 2022.

Huang, B., Lee, J. D., Wang, Z., and Yang, Z. Towards general function approximation in zero-sum markov games. *arXiv preprint arXiv:2107.14702*, 2021.

Jin, C., Liu, Q., and Yu, T. The power of exploiter: Provable multi-agent rl in large state spaces. *arXiv preprint arXiv:2106.03352*, 2021a.

Jin, Y., Yang, Z., and Wang, Z. Is pessimism provably efficient for offline rl? In *International Conference on Machine Learning*, pp. 5084–5096. PMLR, 2021b.

Kong, D. and Yang, L. Provably feedback-efficient reinforcement learning via active reward learning. *Advances in Neural Information Processing Systems*, 35:11063–11078, 2022.

Langford, J. and Zhang, T. The epoch-greedy algorithm for multi-armed bandits with side information. *Advances in neural information processing systems*, 20, 2007.

Lee, K., Liu, H., Ryu, M., Watkins, O., Du, Y., Boutilier, C., Abbeel, P., Ghavamzadeh, M., and Gu, S. S. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*, 2023.

Li, Z., Xu, T., and Yu, Y. Policy optimization in rlhf: The impact of out-of-preference data. *arXiv preprint arXiv:2312.10584*, 2023a.

Li, Z., Xu, T., Zhang, Y., Yu, Y., Sun, R., and Luo, Z.-Q. Remax: A simple, effective, and efficient reinforcement learning method for aligning large language models. *arXiv e-prints*, pp. arXiv–2310, 2023b.

Li, Z., Yang, Z., and Wang, M. Reinforcement learning with human feedback: Learning dynamic choices via pessimism. *arXiv preprint arXiv:2305.18438*, 2023c.

Liu, T., Zhao, Y., Joshi, R., Khalman, M., Saleh, M., Liu, P. J., and Liu, J. Statistical rejection sampling improves preference optimization. *arXiv preprint arXiv:2309.06657*, 2023a.

Liu, Z., Lu, M., Xiong, W., Zhong, H., Hu, H., Zhang, S., Zheng, S., Yang, Z., and Wang, Z. Maximize to explore: One objective function fusing estimation, planning, and exploration. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Michaud, E. J., Gleave, A., and Russell, S. Understanding learned reward functions. *arXiv preprint arXiv:2012.05862*, 2020.

Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., Hesse, C., Jain, S., Kosaraju, V., Saunders, W., et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

Neumann, V. Various techniques used in connection with random digits. *Notes by GE Forsythe*, pp. 36–38, 1951.

Novoseller, E., Wei, Y., Sui, Y., Yue, Y., and Burdick, J. Dueling posterior sampling for preference-based reinforcement learning. In *Conference on Uncertainty in Artificial Intelligence*, pp. 1029–1038. PMLR, 2020.

OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Pacchiano, A., Saha, A., and Lee, J. Dueling rl: reinforcement learning with trajectory preferences. *arXiv preprint arXiv:2111.04850*, 2021.

Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

Rashidinejad, P., Zhu, B., Ma, C., Jiao, J., and Russell, S. Bridging offline reinforcement learning and imitation learning: A tale of pessimism. *Advances in Neural Information Processing Systems*, 34:11702–11716, 2021.

Rusmevichientong, P. and Tsitsiklis, J. N. Linearly parameterized bandits. *Mathematics of Operations Research*, 35 (2):395–411, 2010.

Russo, D. and Van Roy, B. Eluder dimension and the sample complexity of optimistic exploration. *Advances in Neural Information Processing Systems*, 26, 2013.

Saha, A. Optimal algorithms for stochastic contextual preference bandits. *Advances in Neural Information Processing Systems*, 34:30050–30062, 2021.

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

Song, Y., Zhou, Y., Sekhari, A., Bagnell, J. A., Krishnamurthy, A., and Sun, W. Hybrid rl: Using both offline and online data can make rl efficient. *arXiv preprint arXiv:2210.06718*, 2022.

Tiapkin, D., Belomestny, D., Calandriello, D., Moulines, E., Naumov, A., Perrault, P., Valko, M., and Menard, P. Regularized rl. *arXiv preprint arXiv:2310.17303*, 2023.

Tien, J., He, J. Z.-Y., Erickson, Z., Dragan, A. D., and Brown, D. S. Causal confusion and reward misidentification in preference-based reward learning. *arXiv preprint arXiv:2204.06601*, 2022.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Tunstall, L., Beeching, E., Lambert, N., Rajani, N., Rasul, K., Belkada, Y., Huang, S., von Werra, L., Fourrier, C., Habib, N., et al. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*, 2023.

Uehara, M., Kallus, N., Lee, J. D., and Sun, W. Offline minimax soft-q-learning under realizability and partial coverage. *Advances in Neural Information Processing Systems*, 36, 2024.

von Werra, L., Belkada, Y., Tunstall, L., Beeching, E., Thrush, T., Lambert, N., and Huang, S. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`, 2020.

Wang, C., Jiang, Y., Yang, C., Liu, H., and Chen, Y. Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. *arXiv preprint arXiv:2309.16240*, 2023a.

Wang, X., Peng, H., Jabbarvand, R., and Ji, H. Leti: Learning to generate from textual interactions. In *arxiv*, 2023b.

Wang, X., Wang, Z., Liu, J., Chen, Y., Yuan, L., Peng, H., and Ji, H. Mint: Multi-turn interactive evaluation for tool-augmented llms with language feedback. In *Proc. The Twelfth International Conference on Learning Representations (ICLR2024)*, 2024.

Wang, Y., Liu, Q., and Jin, C. Is rlhf more difficult than standard rl? *arXiv preprint arXiv:2306.14111*, 2023c.

Wang, Z., Hou, L., Lu, T., Wu, Y., Li, Y., Yu, H., and Ji, H. Enable language models to implicitly learn self-improvement from data. *arXiv preprint arXiv:2310.00898*, 2023d.

Wirth, C., Akrour, R., Neumann, G., Fürnkranz, J., et al. A survey of preference-based reinforcement learning methods. *Journal of Machine Learning Research*, 18(136): 1–46, 2017.

Wu, J., Ouyang, L., Ziegler, D. M., Stiennon, N., Lowe, R., Leike, J., and Christiano, P. Recursively summarizing books with human feedback. *arXiv preprint arXiv:2109.10862*, 2021a.

Wu, R. and Sun, W. Making rl with preference-based feedback efficient via randomization. *arXiv preprint arXiv:2310.14554*, 2023.

Wu, X., Sun, K., Zhu, F., Zhao, R., and Li, H. Better aligning text-to-image models with human preference. *arXiv preprint arXiv:2303.14420*, 2023.

Wu, Y., Zhai, S., Srivastava, N., Susskind, J., Zhang, J., Salakhutdinov, R., and Goh, H. Uncertainty weighted actor-critic for offline reinforcement learning. *arXiv preprint arXiv:2105.08140*, 2021b.

Xiao, L. On the convergence rates of policy gradient methods. *Journal of Machine Learning Research*, 23(282): 1–36, 2022.

Xie, T. and Jiang, N. Batch value-function approximation with only realizability. In *International Conference on Machine Learning*, pp. 11404–11413. PMLR, 2021.

Xie, T., Cheng, C.-A., Jiang, N., Mineiro, P., and Agarwal, A. Bellman-consistent pessimism for offline reinforcement learning. *Advances in neural information processing systems*, 34:6683–6694, 2021a.

Xie, T., Jiang, N., Wang, H., Xiong, C., and Bai, Y. Policy finetuning: Bridging sample-efficient offline and online reinforcement learning. *Advances in neural information processing systems*, 34:27395–27407, 2021b.

Xiong, W., Zhong, H., Shi, C., Shen, C., Wang, L., and Zhang, T. Nearly minimax optimal offline reinforcement learning with linear function approximation: Single-agent mdp and markov game. *arXiv preprint arXiv:2205.15512*, 2022a.

Xiong, W., Zhong, H., Shi, C., Shen, C., and Zhang, T. A self-play posterior sampling algorithm for zero-sum markov games. In *International Conference on Machine Learning*, pp. 24496–24523. PMLR, 2022b.

Xu, Y., Wang, R., Yang, L., Singh, A., and Dubrawski, A. Preference-based reinforcement learning with finite-time guarantees. *Advances in Neural Information Processing Systems*, 33:18784–18794, 2020.

Ye, C., Xiong, W., Gu, Q., and Zhang, T. Corruption-robust algorithms with uncertainty weighting for nonlinear contextual bandits and markov decision processes. In *International Conference on Machine Learning*, pp. 39834–39863. PMLR, 2023.

Yin, M., Duan, Y., Wang, M., and Wang, Y.-X. Near-optimal offline reinforcement learning with linear representation: Leveraging variance information with pessimism. *arXiv preprint arXiv:2203.05804*, 2022.

Yuan, R., Du, S. S., Gower, R. M., Lazaric, A., and Xiao, L. Linear convergence of natural policy gradient methods with log-linear policies. In *The Eleventh International Conference on Learning Representations*, 2022.

Yuan, W., Pang, R. Y., Cho, K., Sukhbaatar, S., Xu, J., and Weston, J. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*, 2024.

Yuan, Z., Yuan, H., Tan, C., Wang, W., Huang, S., and Huang, F. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302*, 2023.

Yue, Y., Broder, J., Kleinberg, R., and Joachims, T. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.

Zanette, A., Cheng, C.-A., and Agarwal, A. Cautiously optimistic policy optimization and exploration with linear function approximation. In *Conference on Learning Theory*, pp. 4473–4525. PMLR, 2021a.

Zanette, A., Wainwright, M. J., and Brunskill, E. Provable benefits of actor-critic methods for offline reinforcement learning. *Advances in neural information processing systems*, 34:13626–13640, 2021b.

Zhan, W., Uehara, M., Kallus, N., Lee, J. D., and Sun, W. Provable offline reinforcement learning with human feedback. *arXiv preprint arXiv:2305.14816*, 2023a.

Zhan, W., Uehara, M., Sun, W., and Lee, J. D. How to query human feedback efficiently in rl? *arXiv preprint arXiv:2305.18505*, 2023b.

Zhang, T. Feel-good thompson sampling for contextual bandits and reinforcement learning. *SIAM Journal on Mathematics of Data Science*, 4(2):834–857, 2022.

Zhang, T. *Mathematical Analysis of Machine Learning Algorithms*. Cambridge University Press, 2023. doi: 10.1017/9781009093057.

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

Zhong, H. and Zhang, T. A theoretical analysis of optimistic proximal policy optimization in linear markov decision processes. *arXiv preprint arXiv:2305.08841*, 2023.

Zhong, H., Xiong, W., Zheng, S., Wang, L., Wang, Z., Yang, Z., and Zhang, T. Gec: A unified framework for interactive decision making in mdp, pomdp, and beyond. *arXiv preprint arXiv:2211.01962*, 2022.

Zhou, Y., Sekhari, A., Song, Y., and Sun, W. Offline data enhanced on-policy policy gradient with provable guarantees. *arXiv preprint arXiv:2311.08384*, 2023.

Zhu, B., Jiao, J., and Jordan, M. I. Principled reinforcement learning with human feedback from pairwise or k-wise comparisons. *arXiv preprint arXiv:2301.11270*, 2023a.

Zhu, B., Sharma, H., Frujeri, F. V., Dong, S., Zhu, C., Jordan, M. I., and Jiao, J. Fine-tuning language models with advantage-induced policy alignment. *arXiv preprint arXiv:2306.02231*, 2023b.

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

A. Notation Table, Related Work, Backgrounds, and Organization of Appendix

To improve the readability of this paper, we provide a Table 2 for the notations used in this paper. We also provide an introduction to the rejection sampling for completeness.

Notation	Description
$\langle z_1, z_2\rangle$	The inner product of two vectors $z_1^\top z_2$.
$\\|z\\|_\Sigma$	The induced norm $\sqrt{z^\top \Sigma z}$.
$\mathcal{X}, \mathcal{A}$	The state (prompt) space and the action (response) space.
$\phi(x,a), \theta$	The feature map and parameter of the linear parameterization in Assumption 2.1.
d	The dimension of the feature vector.
π, Π	Policy and policy class.
$\ell_{\mathcal{D}}$	The log-likelihood of the BT model on $\mathcal{D}$ defined in Equation (3).
$y \in \{0,1\}$	Preference signal.
$J(\pi)$	The KL-regularized target defined in Equation (2).
η	The coefficient of KL penalty, defined in Equation (2).
d_0	Distribution of state (prompt).
B, γ	Regularization constant: $\\|\theta\\| \le B, \gamma = 1/(2+\exp(-B)+\exp(B))$.
$\Theta(B)$	$\{\theta \in \mathbb{R}^d : \\|\theta\\| \le B\}$.
$\mathcal{D}_{\mathrm{off}}, \mathcal{D}^t$	The offline dataset and the dataset collected in online iteration t.
$\Sigma_{\mathrm{off}}, \Sigma_t$	The covariance matrix with $\mathcal{D}_{\mathrm{off}}$ and $\mathcal{D}^t$.
$\sigma(\cdot)$	$\sigma(z) = 1/(1+\exp(-z))$ is the sigmoid function.
$C_{\mathrm{cov}}(\mathcal{D}_{\mathrm{off}}, \pi_{\mathrm{ref}}, \alpha)$	The coverage of the offline dataset defined in Definition 4.1.
Rejection Sampling	See Appendix A.2 for an introduction.
Best-of-n Policy	See Appendix A.2 for an introduction.

Table 2. The table of notations used in this paper.

A.1. Related Work

There is a rich literature in RLHF and we refer the interested readers to the survey papers like Casper et al. (2023) for a more comprehensive review. We focus on the papers that are most related to our work here.

RLHF has attracted considerable attention in the past few years, especially after its tremendous success in ChatGPT (OpenAI, 2023). We refer interested readers to Wirth et al. (2017); Casper et al. (2023) for a detailed survey but focus on the most related works here. The standard RLHF was popularized by Christiano et al. (2017), which served to direct the attention of the RL community to the preference-based feedback. The most popular and standard RLHF framework is outlined in the InstructGPT paper (Ouyang et al., 2022), Claude (Bai et al., 2022) and the LLaMA2 report (Touvron et al., 2023) in detail, which typically consists of three steps starting from the pretrained model: supervised finetuning, reward modeling, and reward optimization. The effectiveness of this framework has been showcased by many recent generative models, like ChatGPT (OpenAI, 2023), Bard (Google, 2023), Claude (Anthropic, 2023), and LLaMA2 (Touvron et al., 2023). However, it is also noteworthy to indicate that the RLHF process often leads to degeneration in the performance of generation, commonly referred to as the "alignment tax" in the literature (Askell et al., 2021). This is usually because of the imperfection of the reward model and the model can make use of these imperfections to chase for a high reward. This phenomenon is referred to as the reward hacking (Michaud et al., 2020; Tien et al., 2022). It is also possible to apply RLHF to general generative models, like the diffusion model (Hao et al., 2022; Wu et al., 2023; Lee et al., 2023; Dong et al., 2023). In this work, we use the terminology and analysis of LLMs for better illustration, and defer the study of general generative models to future work.

RLHF algorithms. Proximal Policy Optimization (PPO) (Schulman et al., 2017) is the most well-known algorithm in LLM alignment literature. However, its instability, inefficiency, and sensitivity to hyperparameters (Choshen et al., 2019) and code-level optimizations (Engstrom et al., 2020) present significant challenges in tuning for optimal performance and its tremendous success in Chat-GPT4 (OpenAI, 2023) has not been widely reproduced so far. Additionally, it often necessitates incorporating an extra reward model, a value network (known as a critic), and a reference model, potentially as large as the aligned LLM (Ouyang et al., 2022; Touvron et al., 2023). This imposes a significant demand on GPU memory resources. Thus, researchers have attempted to design alternative approaches for LLM alignment to resolve the aforementioned issues.

Dong et al. (2023); Yuan et al. (2023); Touvron et al. (2023); Gulcehre et al. (2023) propose reward ranked finetuning (RAFT) (also known as the iterative finetuning, rejection sampling finetuning) by iteratively learning from the best-of-n policy (Nakano et al., 2021) to maximize the reward, which is a stable baseline with minimal hyper-parameter configuration and was applied to the alignment of LLaMA2 project. There is also a line of work focusing on deriving an algorithm from the KL-regularized formulation (Rafailov et al., 2023; Zhu et al., 2023b; Wang et al., 2023a; Liu et al., 2023a; Li et al., 2023b). Among them, Direct Preference Optimization (DPO) (Rafailov et al., 2023) has emerged as an attractive alternative approach to PPO with notable stability and competitive performance. The innovative idea of DPO is to train the LLMs directly as a reward model based on the offline preference dataset and bypassing the reward modeling. Similar to DPO, there are also other works aiming to optimize the LLMs directly from the preference data, including (Zhao et al., 2023; Azar et al., 2023), and has sparked considerable debate on whether reward modeling, as well as RL, is necessary for alignment. However, while these algorithms are partly inspired by mathematical principles and intuitions, a comprehensive theoretical analysis remains open.

Theoretical study of RLHF. The theoretical understanding of RLHF can be traced back to research on dueling bandits (e.g., Yue et al., 2012; Saha, 2021; Bengs et al., 2021), a simplified setting within the RLHF framework. Recently, many works have focused on the more challenging RLHF problem (also known as the preference-based RL). Xu et al. (2020); Novoseller et al. (2020); Pacchiano et al. (2021) delve into the study of tabular online RLHF, where the state space is finite and small. Moving beyond the tabular setting, Chen et al. (2022) provides the first results for online RLHF with general function approximation, capturing real-world problems with large state spaces. Wang et al. (2023c) presents a reduction-based framework, which transforms some sample-efficient algorithms for standard reward-based RL to efficient algorithms for online RLHF. Further advancements in algorithm designs are introduced by Zhan et al. (2023b); Wu & Sun (2023), encompassing the development of reward-free learning type algorithms and posterior sampling-based algorithms tailored for online RLHF. Initiating exploration into offline RLHF, Zhu et al. (2023a) presents a pessimistic algorithm that is provably efficient for offline RLHF. Additionally, Zhan et al. (2023a) and Li et al. (2023c) extend these investigations into the broader scope of general function approximation settings within offline RLHF. In comparison to these existing studies, our work introduces a new theoretical formulation and goal for RLHF, as well as novel problem settings, such as hybrid RLHF. The new mathematical formulation allows our framework to align more closely with recent advancements in LLMs, and we discuss the connections between our theoretical findings and practical algorithmic designs in Section 5. We mention in passing that Tiapkin et al. (2023) also considers the KL constraint in offline RL but mainly focuses on the scenario where an expert policy is available for imitation learning, thus differing from ours.

Finally, concurrent to this work, Hoang Tran (2024) and Yuan et al. (2024) consider variants of iterative DPO that may share similar insights with us in terms of algorithmic design. We comment on the similarities and differences between our work and theirs as follows. Hoang Tran (2024) focus on the batch online setting, which will be thoroughly developed in Section 4. One notable difference is that they set the reference policy as the one from last iteration, while we always use the π_0 as the reference policy. From a theoretical perspective, their algorithmic design resembles the classic policy gradient algorithm (Cai et al., 2020; Zhong & Zhang, 2023) that optimizes the non-regularized reward, while we optimize the KL-regularized one as most of the Instruct-GPT (Ouyang et al., 2022) and Claude (Bai et al., 2022) did. Yuan et al. (2024) also consider iterative DPO-type training. However, both our algorithm and Hoang Tran (2024) leverage the reward signal from the external model or human, while Yuan et al. (2024) adopts a clever idea by using the LLM itself as the reward model to provide preference signal, hence the name “self-rewarding”. We remark that the primary goal of this project is to formally formulate the RLHF as a KL-regularized contextual bandit problem and establish its mathematical foundation. The online iterative DPO is a natural corollary of the established framework but the framework can also be implemented by combining it with other oracle algorithms. See Section 5 for details. Finally, we expect that the techniques presented in this paper also extend to analyze the general preference learning like Azar et al. (2023) beyond the reward-based learning.

A.2. Rejection Sampling

We briefly introduce the rejection sampling in this subsection. We first remark that in the literature, many papers use this terminology to refer best-of-n policy (Touvron et al., 2023), which can be different from the notion of rejection sampling here. Specifically, the best-of-n policy takes a base policy π and a reward function r as the input, and output a new policy $\widetilde{\pi}$: for each $x \in \mathcal{X}$, we sample n independent policies from π and output the one with the highest reward measured by r. In what follows, we introduce the rejection sampling.

Rejection sampling, a widely utilized method in Monte Carlo tasks, is designed to sample from a target distribution using samples from a proposal distribution and a uniform sampler (Neumann, 1951). This technique is applicable when the density ratio between the target distribution q and the proposal distribution p is bounded, satisfying $q(x)/p(x) \leq M$ for all $x \in \mathcal{X}$. In practical implementation, n samples are drawn from the proposal distribution p. Each sample, denoted as $x \sim p$, is accepted with a probability $r = \frac{q(x)}{Mp(x)}$. This acceptance is determined by evaluating whether $u < r$, where u is a number drawn from a uniform distribution $U[0, 1]$. The accepted samples $\widetilde{x}$ are then representative of the target distribution q.

The primary challenge in rejection sampling is its low acceptance rate, particularly problematic for high-dimensional data due to the curse of dimensionality, where the density ratio often scales with $\exp(d)$. This issue persists even in low-dimensional scenarios, as a large density ratio M can drastically reduce acceptance rates. The method is most efficient when p closely approximates q, leading to $M \approx 1$.

A.3. Covariance Matrix and Eluder Technique

Given a training set $\mathcal{D}$, the *in-sample* error on the observed data in $\mathcal{D}$ is given by

$$\|\theta_1 - \theta_2\|^2_{\Sigma_{\mathcal{D}}} = \lambda\|\theta_1 - \theta_2\|^2 + \sum_{(x,a^1,a^2)\in\mathcal{D}} \Big(\big(r_{\theta_1}(x, a^1) - r_{\theta_1}(x, a^2)\big) - \big(r_{\theta_2}(x, a^1) - r_{\theta_2}(x, a^2)\big)\Big)^2,$$

where we additionally add a regularization term $\lambda\|\theta_1 - \theta_2\|^2$. Meanwhile, if we test the hypothesis $(\theta_1 - \theta_2)$ on a newly observed data, the *out-of-sample* error would be given by $|\big\langle \theta_1 - \theta_2, \phi(x, a^1) - \phi(x, a^2)\big\rangle|$. The ideal case would be that we can infer the out-of-sample error via the in-sample error, so we look at the ratio between them:

$$\frac{|\big\langle \theta_1 - \theta_2, \phi(x, a^1) - \phi(x, a^2)\big\rangle|}{\|\theta_1 - \theta_2\|_{\Sigma_{\mathcal{D}}}} \leq \frac{\|\phi(x, a^1) - \phi(x, a^2)\|_{\Sigma^{-1}_{\mathcal{D}}} \cdot \|\theta_1 - \theta_2\|_{\Sigma_{\mathcal{D}}}}{\|\theta_1 - \theta_2\|_{\Sigma_{\mathcal{D}}}} = \|\phi(x, a^1) - \phi(x, a^2)\|_{\Sigma^{-1}_{\mathcal{D}}},$$

where we take a square root on the in-sample error to keep them being of the same order and use Cauchy-Schwarz inequality (Lemma G.2). Here, the $\|\phi(x, a^1) - \phi(x, a^2)\|_{\Sigma^{-1}_{\mathcal{D}}}$ is referred to as the elliptical potential in the literature of linear function approximation (Abbasi-Yadkori et al., 2011). The elliptical potential can be viewed as the uncertainty of $\phi(x, a^1) - \phi(x, a^2)$, given the historical samples in $\mathcal{D}$, and can be used to guide our exploration. The complexity of the reward model space is characterized by the following fact:

Lemma A.1 (Elliptical potential is usually small (Hu et al., 2022))**.** *For a fixed $\lambda > 0$ and $\{z_t\}_{t=1}^T \subset \mathbb{R}^d$ with $\|z_t\| \leq 1$, we define $Z_t = \lambda I + \sum_{s=1}^{t-1} z_s z_s^\top$. Then, for any constant $c > 0$, $\|z_t\|_{Z_t^{-1}} > c$ happens at most $\frac{3d}{\log(1+c^2)} \log\Big(1 + \frac{1}{\lambda \log(1+c^2)}\Big)$.*

The ratio between the out-of-sample error and the in-sample error in the linear case can be readily generalized to the general function approximation using the variant of eluder dimension considered in Gentile et al. (2022); Zhang (2023); Ye et al. (2023); Agarwal et al. (2023), which essentially states that there is some low-rank structure in the reward model space so the generalization is limited (the elliptical potential cannot be large for too many times). Moreover, if we can effectively estimate the in-sample error from the preference data, by Lemma A.1, we can infer the out-of-sample error safely most of the time. Such an in-sample error estimation is provided in Lemma G.3. Essentially, the eluder-type complexity measures and techniques reduce the learning problem to an online supervised learning (in-sample error estimation and minimization) (Zhong et al., 2022).

A.4. Organization of the Appendix

In the appendix, we need to formally prove Theorem 3.1, 4.2, and 4.4. To distinguish them, we refer the first theorem as the offline setting, the second setting as the hybrid learning to stress the requirement of a diverse offline $\mathcal{D}_{\text{off}}$ and finally, we refer Theorem 4.4 as the online learning where we may start from scratch. While we mainly focus on the batch learning setting to match the practical applications, we also develop the results of sequential setting with $m = 1$ in case that readers are interested in the technique for completeness. The map of the appendix is as follows.

- We develop the pure online framework in Appendix B, where we do not make any assumption on $\mathcal{D}_{\text{off}}$. We also provide the proof of Theorem 4.4 in this section;
- We study the offline learning in Appendix C with the proof of Theorem 3.1;
- We study the hybrid learning in Appendix D and prove Theorem 4.2;

- We study the coverage condition for DPO to converge in Appendix E;
- We provide the proof of some technical Lemmas in Appendix F, as well as some existing technical Lemma in Appendix G;
- We provide the additional experimental details, hyper-parameters, and illustrating examples in Appendix H.

B. Proof of Online Learning

In this section, we develop the online framework of the KL-constraint contextual bandit, that is missing in the main paper.

B.1. Batch Online Learning

We first consider the case of $m > 1$, which leads to a more sparse update of the model. Our goal is also to design a sample-efficient algorithm, which finds a policy $\hat{\pi}$ so that the suboptimality $J(\pi^*) - J(\hat{\pi}) < \epsilon$ with the number of samples polynomial in the accuracy number $1/\epsilon$, feature dimension d, and other problem-dependent parameters. In practical applications, it is observed that the diversity of the outputs is critical, and the response pairs (a_t^1, a_t^2) are recommended to be collected by different model variants with different temperature hyper-parameter (Touvron et al., 2023).

Proof of Theorem 4.4. Recall the definition of the covariance matrix:

$$\Sigma_{t,m} = \lambda I + \frac{1}{m}\sum_{i=1}^{t-1}\sum_{j=1}^{m}(\phi(x_{i,j}, a_{i,j}^1) - \phi(x_{i,j}, a_{i,j}^2))(\phi(x_{i,j}, a_{i,j}^1) - \phi(x_{i,j}, a_{i,j}^2))^\top.$$

Then, by invoking Lemma G.3 for θ_t with $\Sigma_{\mathcal{D}} = m\Sigma_{t,m}$ and $\lambda' = m\lambda$, we have with probability at least $1-\delta$, for any $t \in [T]$,

$$\begin{aligned}
\|\theta^t - \theta^*\|_{\Sigma_{t,m}} =& \frac{1}{\sqrt{m}}\|\theta^t - \theta^*\|_{\Sigma_{\mathcal{D}}} \\
\le& \frac{C}{\sqrt{m}}\sqrt{\frac{d + \log(T/\delta)}{\gamma^2} + m\lambda B^2} \\
=& C\sqrt{\frac{d + \log(T/\delta)}{\gamma^2 m} + \lambda B^2}.
\end{aligned} \tag{11}$$

Let

$$\widetilde{\Sigma}_t = \lambda I + \sum_{i=1}^{t-1}\mathbb{E}_{x\sim d_0, a^1\sim\pi_i^1, a^2\sim\pi_i^2}\left[(\phi(x_t, a^1) - \phi(x_t, a^2))(\phi(x_t, a^1) - \phi(x_t, a^2))^\top\right].$$

Now, by elliptical potential lemma (Lemma G.4), we have

$$\begin{aligned}
\sum_{t=1}^{T}\log\big(1 + \mathbb{E}_{x_t\sim d_0}\|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|^2_{\widetilde{\Sigma}_t^{-1}}\big) \le& \sum_{t=1}^{T}\log\big(1 + \mathbb{E}_{x_t\sim d_0, a^1\sim\pi_t^1, a^2\sim\pi_t^2}\|[\phi(x_t, a^1) - \phi(x_t, a^2)]\|^2_{\widetilde{\Sigma}_t^{-1}}\big) \\
\le& \log\frac{\det(\widetilde{\Sigma}_T)}{\det(\lambda I)} \\
\le& d\log(1 + TL^2/\lambda d) := \gamma_T(\lambda).
\end{aligned}$$

Since each term on the left-hand side is positive, we know that there exists at least a $t_0 \in [T]$, the value is smaller or equal than the average value:

$$\log\big(1 + \psi_{t_0}^2\big) \le \frac{1}{T}\gamma_T(\lambda),$$

where we use the short-hand notation $\psi_t = \mathbb{E}_{x_t\sim d_0}\|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|_{\widetilde{\Sigma}_t^{-1}}$. It is equivalent to

$$\psi_{t_0}^2 \le \exp\big(\frac{\gamma_T(\lambda)}{T}\big) - 1.$$

We now consider the suboptimality at iteration t_0:

$$\begin{aligned} J(\pi^*) - J(\pi_{t_0}^1) &= \mathbb{E}_{x_{t_0}\sim d_0}\Big[\big\langle \theta^{t_0} - \theta^*, \phi(x_{t_0}, \pi_{t_0}^1) - \phi(x_{t_0}, \pi^*)\big\rangle\Big] - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}^1(\cdot|x_{t_0}))\big] \\ &\le \mathbb{E}_{x_{t_0}\sim d_0}\Big[\|\phi(x_{t_0}, \pi_{t_0}^1) - \phi(x_{t_0}, \pi^*)\|_{\Sigma_{t,m}^{-1}}\Big] \cdot \|\theta^{t_0} - \theta^*\|_{\Sigma_{t,m}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}^1(\cdot|x_{t_0}))\big], \end{aligned} \tag{12}$$

where the inequality uses the Cauchy-Schwarz inequality (Lemma G.2). Then, since the samples $\{x_{t,i}\}_{i=1}^m$ are i.i.d and for any $x \in \mathcal{X}$

$$\|\phi(x, \pi_{t_0}^1) - \phi(x_{t_0}, \pi^*)\|_{\Sigma_{t,m}^{-1}} \le \frac{2}{\sqrt{\lambda}},$$

we can use Chernoff bound (Theorem 2.16 of Zhang (2023)) to obtain that with probability at least $1 - \delta/2$,

$$\mathbb{E}_{x_{t_0}\sim d_0}\Big[\|\phi(x_{t_0}, \pi_{t_0}^1) - \phi(x_{t_0}, \pi^*)\|_{\Sigma_{t,m}^{-1}}\Big] \le \frac{1}{m}\sum_{i=1}^m \|\phi(x_{t,i}, \pi_{t_0}^1) - \phi(x_{t,i}, \pi^*)\|_{\Sigma_{t,m}^{-1}} + \sqrt{\frac{\log(2/\delta)}{2m}}.$$

Similarly, we also get with probability at least $1 - \delta/2$,

$$\frac{1}{m}\sum_{i=1}^m \|\phi(x_{t,i}, \pi_{t_0}^1) - \phi(x_{t,i}, \pi^*)\|_{\widetilde{\Sigma}_{t_0}^{-1}} \le \mathbb{E}_{x_{t_0}\sim d_0}\Big[\|\phi(x_{t_0}, \pi_{t_0}^1) - \phi(x_{t_0}, \pi^*)\|_{\widetilde{\Sigma}_{t_0}^{-1}}\Big] + \sqrt{\frac{\log(2/\delta)}{2m}}$$

We take the two inequalities above back into Equation (12) to derive with that probability at least $1 - 3\delta$,

$$\begin{aligned} &J(\pi^*) - J(\pi_{t_0}^1) \\ &\le \Big(\frac{1}{m}\sum_{i=1}^m \Big[\|\phi(x_{t_0,i}, \pi_{t_0}^1) - \phi(x_{t_0,i}, \pi^*)\|_{\Sigma_{t_0,m}^{-1}}\Big] + \sqrt{\frac{\log(2/\delta)}{2m}}\Big) \cdot \|\theta^{t_0} - \theta^*\|_{\Sigma_{t_0,m}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}^1(\cdot|x_{t_0}))\big] \\ &\le \Big(\frac{1}{m}\sum_{i=1}^m \Big[\|\phi(x_{t_0,i}, \pi_{t_0}^1) - \phi(x_{t_0,i}, \pi_{t_0}^2)\|_{\Sigma_{t_0,m}^{-1}}\Big] + \sqrt{\frac{\log(2/\delta)}{2m}}\Big) \cdot \|\theta^{t_0} - \theta^*\|_{\Sigma_{t_0,m}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}^1(\cdot|x_{t_0}))\big] \\ &\le \Big(\frac{\sqrt{3}}{m}\sum_{i=1}^m \Big[\|\phi(x_{t_0,i}, \pi_{t_0}^1) - \phi(x_{t_0,i}, \pi_{t_0}^2)\|_{\widetilde{\Sigma}_{t_0}^{-1}}\Big] + \sqrt{\frac{\log(2/\delta)}{2m}}\Big) \cdot \|\theta^{t_0} - \theta^*\|_{\Sigma_{t_0,m}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}^1(\cdot|x_{t_0}))\big] \\ &\le \Big(\sqrt{3}\mathbb{E}_{x_{t_0}\sim d_0}\Big[\|\phi(x_{t_0}, \pi_{t_0}^1) - \phi(x_{t_0}, \pi^*)\|_{\widetilde{\Sigma}_{t_0}^{-1}}\Big] + 2\sqrt{\frac{\log(2/\delta)}{2m}}\Big) \cdot \|\theta^{t_0} - \theta^*\|_{\Sigma_{t_0,m}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}^1(\cdot|x_{t_0}))\big] \\ &\le C \cdot \Big(\sqrt{\exp\big(\frac{\gamma_T(\lambda)}{T}\big) - 1} + 2\sqrt{\frac{\log(2/\delta)}{2m}}\Big)\sqrt{\frac{d + \log(T/\delta)}{\gamma^2 m} + \lambda B^2} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}^1(\cdot|x_{t_0}))\big], \end{aligned}$$

where the second inequality applies Lemma G.5 with $\lambda = \Omega(d\log(T/\delta)/m)$, and the last inequality uses Equation (11). By choosing T satisfying that $T \ge d\log(T)$ and $\lambda = \Theta(d\log(T/\delta)/m\gamma^2)$, we have

$$J(\pi^*) - J(\pi_{t_0}^1) = \widetilde{O}\Big(\sqrt{\frac{d}{\gamma^2 m}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}^1(\cdot|x_{t_0}))\big]\Big),$$

which concludes the proof. □

B.2. Sequential Online Setting

While we mainly care about finding a good model, with a slightly more involved analysis for the enhancer, we can also derive an upper bound for the average regret as in Pacchiano et al. (2021); Chen et al. (2022):

$$\mathrm{Reg}_{\mathrm{ave}}(T) := \sum_{t=1}^T \Big[\frac{2J(\pi^*) - J(\pi_t^1) - J(\pi_t^2)}{2}\Big],$$

where we now discuss in the sequential case with $m = 1$. We consider two kinds of regrets: (1) cumulative suboptimality for the main policy π_t^1 compared to π^*:

$$\mathrm{Reg}(T) := \sum_{t=1}^{T} \big[J(\pi^*) - J(\pi_t^1)\big],$$

and (2) the average suboptimality:

$$\mathrm{Reg}_{\mathrm{ave}}(T) := \sum_{t=1}^{T} \big[\frac{2J(\pi^*) - J(\pi_t^1) - J(\pi_t^2)}{2}\big].$$

In this case, our goal is to output a sequence of policy pair $\{\pi_t^1, \pi_t^2\}_{t=1}^{T}$ so that the regrets $\mathrm{Reg}(T)$ and $\mathrm{Reg}_{\mathrm{ave}}(T)$ are sublinear. To achieve this goal, the enhancer computes its policy by maximizing the uncertainty estimator

$$\pi_t^2 = \mathop{\mathrm{argmax}}_{\pi_t^2 \in \Pi_t} \sum_{i=1}^{m} \Gamma(x_{t,i}, \pi_t^1, \pi_t^2, \mathcal{D}^{1:t-1}), \tag{13}$$

where $\mathcal{D}^{1:t-1} = \cup_{s=1}^{t-1} \mathcal{D}^s$.

Theorem B.1 (Sequential Online learning). *Under Assumption 2.1, with* $\lambda = \Omega(d \log(T/\delta)/(\gamma^2 B^2))$ *and* $\beta := O\big(\sqrt{\frac{d\log(T/\delta)}{\gamma^2}}\big)$, *with probability at least* $1 - 2\delta$, *the regret of Algorithm 2 with Option II and* $m = 1$ *satisfies*

$$\mathrm{Reg}_{\mathrm{ave}}(T) \lesssim \sqrt{T\beta^2 d} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big],$$

which further implies that

$$\mathrm{Reg}(T) \lesssim \sqrt{T\beta^2 d} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big].$$

Proof of Theorem B.1. First, we invoke the decomposition Lemma 2.3 and Lemma 2.4 to obtain for each batch $t \in [T]$

$$\begin{aligned}
&J(\pi^*) - J(\pi_t^1)\\
&= \mathbb{E}_{x_t \sim d_0}\Big[\mathbb{E}_{\pi^*}[r^*(x_t, a) - \hat{r}(x_t, a)] + \mathbb{E}_{\pi_t^1}[\hat{r}(x_t, a) - r^*(x_t, a)] - \eta \cdot \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_t^1(\cdot|x_t))\big]\Big]\\
&= \mathbb{E}_{x_t \sim d_0}\Big[\Big\langle \hat{\theta} - \theta^*, \phi(x_t, \pi_t^1) - \phi(x_t, \pi^*)\Big\rangle\Big] - \eta \cdot \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_t^1(\cdot|x_t))\big].
\end{aligned} \tag{14}$$

Then, we deduce that with probability at least $1 - \delta$,

$$\begin{aligned}
&\sum_{t=1}^{T} \big[J(\pi^*) - J(\pi_t^1)\big]\\
&= \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\Big[\big\langle \theta^t - \theta^*, \phi(x_t, \pi_t^1) - \phi(x_t, \pi^*)\big\rangle\Big] - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big]\\
&\le \beta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0} \min\big\{1, \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi^*)\|_{\Sigma_t^{-1}}\big\} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big]\\
&\le \beta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0} \min\big\{1, \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|_{\Sigma_t^{-1}}\big\} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big]\\
&\le \beta \sqrt{T \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} \min\big\{1, \|\phi(x_t, a_t^1) - \phi(x_t, a_t^2)\|_{\Sigma_t^{-1}}^2\big\}} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big],
\end{aligned} \tag{15}$$

where the first inequality uses the Cauchy-Schwarz inequality, Lemma G.3 and reward $r \le 1$ for any $r \in \mathcal{F}$, the second inequality uses $\pi^* \in \Pi_t$ according to Lemma 4.3, and the last inequality uses the Cauchy-Schwarz inequality and Jensen's inequality.

Then, we define

$$\bar{\Sigma}_t = \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} [(\phi(x_t, a_t^1) - \phi(x_t, a_t^2))(\phi(x_t, a_t^1) - \phi(x_t, a_t^2))^\top] + \lambda I.$$

According to the concentration of the covariance matrix in Lemma G.5, since $\lambda = \Omega(d \log(T/\delta))$, we have with probability at least $1-\delta$, for any $t \in [T]$,

$$\Sigma_t^{-1} \preceq 3\bar{\Sigma}_t^{-1},$$

which implies that

$$\begin{aligned} & \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} \min\left\{1, \|\phi(x_t, a_t^1) - \phi(x_t, a_t^2)\|^2_{\Sigma_t^{-1}}\right\} \\ & \le 3 \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} \min\left\{1, \|\phi(x_t, a_t^1) - \phi(x_t, a_t^2)\|^2_{\bar{\Sigma}_t^{-1}}\right\} \\ & \le 6d \log(1 + T/d\lambda). \end{aligned}$$

By taking the result above back into Equation (15), we get with probability at least $1 - 2\delta$,

$$\sum_{t=1}^{T} \left[J(\pi^*) - J(\pi_t^1)\right] \le \beta \sqrt{T 6d \log(1 + T/d\lambda)} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\left[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\right], \tag{16}$$

where the inequality uses Lemma G.4.

Moreover, to analyze the average regret $\mathrm{Reg}_{\mathrm{ave}}(T)$, we make the following decomposition

$$\begin{aligned} \sum_{t=1}^{T} J(\pi_t^1) - J(\pi_t^2) = & \sum_{t=1}^{T} \underbrace{\mathbb{E}_{x_t \sim d_0}\Big[\mathbb{E}_{\pi_t^1}[r^*(x_t, a) - r^t(x_t, a)] + \mathbb{E}_{\pi_t^2}[r^t(x_t, a) - r^*(x_t, a)]\Big]}_{(\Delta_t^1)} \\ & + \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\Big[\underbrace{\mathbb{E}_{\pi_t^1}[r^t(x_t, a)] - \mathbb{E}_{\pi_t^2}[r^t(x_t, a)] + \eta D_{\mathrm{KL}}(\pi_t^2(\cdot|x_t) \| \pi_0(\cdot|x_t)) - \eta D_{\mathrm{KL}}(\pi_t^1(\cdot|x_t) \| \pi_0(\cdot|x_t))}_{(\Delta_t^2)}\Big]. \end{aligned}$$

For Term (Δ_t^1), we have

$$\begin{aligned} (\Delta_t^1) &= \mathbb{E}_{x_t \sim d_0}[\langle \phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2), \theta^* - \theta_t \rangle] \\ &\le \beta \cdot \mathbb{E}_{x_t \sim d_0} \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|_{\Sigma_t^{-1}} \end{aligned}$$

We can deal with the Term (Δ_t^2) by invoking Lemma 4.3 with $\pi = \pi_t^2$ and using the definition of the confidence set:

$$(\Delta_t^2) = \eta D_{\mathrm{KL}}(\pi_t^2(\cdot|x_t) \| \pi_t^1(\cdot|x_t)) \le \beta \cdot \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|_{\Sigma_t^{-1}}.$$

Combining the above two inequalities and Equation (17), we have

$$\begin{aligned} \sum_{t=1}^{T} J(\pi_t^1) - J(\pi_t^2) &\le 2\beta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0} \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|_{\Sigma_t^{-1}} \\ &\le 2\beta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} \|\phi(x_t, a_t^1) - \phi(x_t, a_t^2)\|_{\Sigma_t^{-1}} \\ &\le 2\beta \sqrt{3T \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} \|\phi(x_t, a_t^1) - \phi(x_t, a_t^2)\|^2_{\bar{\Sigma}_t^{-1}}} \\ &\lesssim \sqrt{T \beta^2 d}, \end{aligned} \tag{17}$$

where the last inequality uses Lemma G.4. Combining the results of $\mathrm{Reg}(T)$ and the upper bound of $\sum_{t=1}^{T} J(\pi_t^1) - J(\pi_t^2)$ in Equation (17), we can obtain the bound for the average regret in the following theorem.

Therefore, by combining the results above and Equation (16), we have

$$\begin{aligned}\sum_{t=1}^{T} \big(2J(\pi^*) - J(\pi_t^1) - J(\pi_t^2)\big) &= \sum_{t=1}^{T} 2\big(J(\pi^*) - J(\pi_t^1)\big) + \big(J(\pi_t^1) - J(\pi_t^2)\big) \\ &\lesssim \sqrt{T\beta^2 d} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t)\|\pi_1^t(\cdot|x_t))\big],\end{aligned}$$

which concludes the proof. $\square$

B.3. Construction of the Confidence Set

In this subsection, we show that the confidence set contains π^* for all iterations with high probability by proving Lemma 4.3.

Proof of Lemma 4.3. By the definition of the π^* that π^* is optimal at every context, for any $\pi_t^1 \in \Pi$ and any $x_{t,i} \in \mathcal{X}$, we have

$$\begin{aligned}0 &\le \big\langle \theta^*, \phi(x_{t,i}, \pi^*) - \phi(x_{t,i}, \pi_t^1)\big\rangle + \eta D_{\mathrm{KL}}(\pi_t^1(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i})) - \eta D_{\mathrm{KL}}(\pi^*(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i})) \\ &= \underbrace{\big\langle \theta^* - \theta_t, \phi(x_{t,i}, \pi^*) - \phi(x_{t,i}, \pi_t^1)\big\rangle}_{\text{Term(i)}} \\ &\quad + \underbrace{\big\langle \theta_t, \phi(x_{t,i}, \pi^*) - \phi(x_{t,i}, \pi_t^1)\big\rangle + \eta D_{\mathrm{KL}}(\pi_t^1(\cdot|x_{t,i})|\pi_0(\cdot|x_{t,i})) - \eta D_{\mathrm{KL}}(\pi^*(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i}))}_{\text{Term(ii)}},\end{aligned} \tag{18}$$

For Term (i), by Cauchy-Schwarz inequality and Lemma G.3 with $\Sigma_{\mathcal{D}} = m\Sigma_{t,m}$ and $\lambda' = m\lambda$, we have

$$\text{Term(i)} \le \beta \cdot \|\phi(x_{t,i}, \pi^*) - \phi(x_{t,i}, \pi_t^1)\|_{\Sigma_{t,m}^{-1}},$$

where $\beta = O\Big(\sqrt{\frac{d\log(T/\delta)}{\gamma^2 m}}\Big)$ and the additional $\log T$ factor is because of the union bound over the T iterations. Meanwhile, by invoking Lemma 2.4 with $\pi = \pi^*$, $\hat{\pi} = \pi_t$, we obtain that

$$\begin{aligned}\text{Term(ii)} &= \big\langle \theta_t, \phi(x_{t,i}, \pi^*) - \phi(x_{t,i}, \pi_t^1)\big\rangle + \eta D_{\mathrm{KL}}(\pi_t^1(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i})) - \eta D_{\mathrm{KL}}(\pi^*(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i})) \\ &= \mathbb{E}_{\pi^*}[r^t(x_{t,i}, a)] - \mathbb{E}_{\pi_t^1}[r^t(x_{t,i}, a)] + \eta D_{\mathrm{KL}}(\pi_t^1(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i})) - \eta D_{\mathrm{KL}}(\pi^*(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i})) \\ &= -\eta D_{\mathrm{KL}}(\pi^*(\cdot|x_{t,i})|\pi_t^1(\cdot|x_{t,i})).\end{aligned}$$

Taking respective upper bounds for Terms (i) and (ii) back into Equation (18) and summing over $i \in [m]$, we have

$$\beta \cdot \sum_{i=1}^{m} \|\phi(x_{t,i}, \pi^*) - \phi(x_{t,i}, \pi_t^1)\|_{\Sigma_{t,m}^{-1}} - \eta \sum_{i=1}^{m} D_{\mathrm{KL}}(\pi^*(\cdot|x_{t,i})|\pi_t^1(\cdot|x_{t,i})) \ge 0,$$

which implies that $\pi^* \in \Pi_t$. Therefore, we finish the proof of Lemma 4.3. $\square$

C. Proof of the Offline Learning

C.1. Proof of Theorem 3.1

Proof of Theorem 3.1. We start with Option I. If we set $\hat{r}(x, a) = \langle \theta_{\mathrm{MLE}}, \phi(x, a)\rangle$, and take the policy by

$$\hat{\pi} = \operatorname*{argmax}_{\pi\in\Pi} \Big[\langle \theta_{\mathrm{MLE}}, \mathbb{E}_{x\sim d_0}\phi(x, \pi)\rangle - \beta \cdot \|\mathbb{E}_{x\sim d_0}[\phi(x, \pi) - \nu]\|_{\Sigma_{\mathrm{off}}^{-1}} - \eta \cdot \mathbb{E}_{x\sim d_0}[D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))]\Big].$$

Then, we have

$$\begin{aligned}&\big\langle \theta_{\mathrm{MLE}}, \mathbb{E}_{x\sim d_0}\big[\phi(x, \pi) - \phi(x, \hat{\pi})\big]\big\rangle + \beta \cdot \|\mathbb{E}_{x\sim d_0}[\phi(x, \hat{\pi})] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} - \beta \cdot \|\mathbb{E}_{x\sim d_0}[\phi(x, \pi)] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} \\ &\quad + \eta \cdot \mathbb{E}_{x\sim d_0}\big[D_{\mathrm{KL}}(\hat{\pi}(\cdot|x)\|\pi_0(\cdot|x)) - D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))\big] \le 0.\end{aligned} \tag{19}$$

For simplicity, we denote the LHS of Equation (19) as $(\star)$. We plugging this into the estimation of $J(\pi) - J(\hat{\pi})$:

$$\begin{aligned}
&J(\pi) - J(\hat{\pi}) \\
&= \mathbb{E}_{x\sim d_0}\Big[\mathbb{E}_{a\sim\pi(\cdot|x)}\big[r^*(x,a) + \eta\log\frac{\pi_0(a|x)}{\pi(a|x)}\big] - \mathbb{E}_{a\sim\hat{\pi}(\cdot|x)}\big[r^*(x,a) + \eta\log\frac{\pi_0(a|x)}{\hat{\pi}(a|x)}\big]\Big] \\
&= (\star) + \langle\theta^* - \theta_{\mathrm{MLE}}, \mathbb{E}_{x\sim d_0}[\phi(x,\pi)]\rangle + \langle\theta_{\mathrm{MLE}} - \theta^*, \mathbb{E}_{x\sim d_0}[\phi(x,\hat{\pi})]\rangle \\
&\quad - \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\hat{\pi})] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} + \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi)] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} \\
&\le \langle\theta^* - \theta_{\mathrm{MLE}}, \mathbb{E}_{x\sim d_0}[\phi(x,\pi)] - \nu\rangle + \langle\theta_{\mathrm{MLE}} - \theta^*, \mathbb{E}_{x\sim d_0}[\phi(x,\hat{\pi})] - \nu\rangle \\
&\qquad - \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\hat{\pi})] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} + \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi)] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} \\
&\le 2\beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi)] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}},
\end{aligned}$$

where the first inequality is from the Equation (19) and the second inequality uses Cauchy-Schwarz inequality and Lemma G.3.

For Option II, we use the point-wise pessimism:

$$\hat{r}(x,a) = r_{\mathrm{MLE}}(x,a) - \beta\|\phi(x,a) - \nu\|_{\Sigma_{\mathrm{off}}^{-1}}.$$

Then, we call Oracle 2.2 with $\hat{r}$ to get $\hat{\pi}$. By Lemma 2.3, we have

$$\begin{aligned}
J(\pi) - J(\hat{\pi}) =& \mathbb{E}_{x\sim d_0}\Big[\mathbb{E}_{\pi}[r^*(x,a) - \hat{r}(x,a)] + \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a) - r^*(x,a)] \\
&+ \mathbb{E}_{\pi}[\hat{r}(x,a)] - \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a)] + \eta D_{\mathrm{KL}}(\hat{\pi}(\cdot|x)\|\pi_0(\cdot|x)) - \eta D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))\Big],
\end{aligned}$$

Since $\hat{r}$ is obtained from the Oracle 2.2 with $\hat{r}$, it follows from Lemma 2.4:

$$\begin{aligned}
&J(\pi) - J(\hat{\pi}) \\
&= \mathbb{E}_{x\sim d_0}\Big[\mathbb{E}_{\pi}[r^*(x,a) - \hat{r}(x,a)] + \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a) - r^*(x,a)] - \eta D_{\mathrm{KL}}(\pi(\cdot|x)\|\hat{\pi}(\cdot|x))\Big] \\
&= \mathbb{E}_{x\sim d_0, a\sim\pi(\cdot|x)}\big[\langle\theta^* - \theta_{\mathrm{MLE}}, \phi(x,a) - \nu\rangle + \beta\|\phi(x,a) - \nu\|_{\Sigma_{\mathrm{off}}^{-1}}\big] \\
&\quad + \mathbb{E}_{x\sim d_0, a\sim\hat{\pi}(\cdot|x)}\big[\langle\theta_{\mathrm{MLE}} - \theta^*, \phi(x,a) - \nu\rangle - \beta\|\phi(x,a) - \nu\|_{\Sigma_{\mathrm{off}}^{-1}}\big] - \eta\mathbb{E}_{x\sim d_0}\big[D_{\mathrm{KL}}(\pi(\cdot|x)\|\hat{\pi}(\cdot|x))\big] \\
&\le 2\beta\mathbb{E}_{x\sim d_0, a\sim\pi(\cdot|x)}\|\phi(x,a) - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} - \eta\mathbb{E}_{x\sim d_0}\big[D_{\mathrm{KL}}(\pi(\cdot|x)\|\hat{\pi}(\cdot|x))\big],
\end{aligned}$$

where we use Cauchy-Schwarz inequality in the last inequality.

□

C.2. Proof of the Direct Preference Learning with Pessimism

In this subsection, we prove the Proposition 5.1 that combines the direct preference learning with the pessimism. The technique is similar to the (Rafailov et al., 2023) with additional consideration of the uncertainty bonus.

Proof of Proposition 5.1. For notation simplicity, we denote the uncertainty bonus as $\Gamma(x,a)$. We first recall that in Algorithm 1, we optimize the following KL-regularized target:

$$\hat{\pi} = \operatorname*{argmax}_{\pi}\mathbb{E}_{x\sim d_0, a\sim\pi(\cdot|x)}\Big[r_{\mathrm{MLE}}(x,a) - \Gamma(x,a) - \eta\log\frac{\pi(a\,|\,x)}{\pi_0(a\,|\,x)}\Big], \tag{20}$$

where r_{MLE} is the MLE of the BT model on the offline preference dataset $\mathcal{D}$ obtained via

$$r_{\mathrm{MLE}} = \operatorname*{argmax}_{r}\sum_{(x,a^w,a^l)\in\mathcal{D}_{\mathrm{off}}}\log\sigma\big(r(x,a^w) - r(x,a^l)\big). \tag{21}$$

According to Lemma G.6, for any fixed r, we have the following closed-form policy for Equation (20):

$$\widetilde{\pi}_r(a|x) = \frac{1}{Z(x)}\pi_0(a|x)\exp(\frac{1}{\eta}(r(x,a) - \Gamma(x,a))). \tag{22}$$

We can solve the reward as

$$r(x,a) = \Gamma(x,a) + \eta \log \frac{\widetilde{\pi}_r(a|x)}{\pi_0(a|x)} + \eta \log Z(x). \tag{23}$$

We can plug Equation (23) into Equation (21) to get

$$\hat{\pi} = \underset{\widetilde{\pi}_r}{\operatorname{argmax}} \sum_{(x,a^w,a^l)\in\mathcal{D}_{\mathrm{off}}} \log \sigma\left(\eta \log \frac{\pi_r(a^w|x)}{\pi_0(a^w|x)} - \eta \log \frac{\pi_r(a^l|x)}{\pi_0(a^l|x)} + \underbrace{(\Gamma(x,a^w) - \Gamma(x,a^l))}_{m(x,a^w,a^l)}\right), \tag{24}$$

where the uncertainty serves as an adaptive margin.

Clearly, if r is the solution of Equation (21), the π_r is the solution of Equation (24). In contrast, if π is optimal for the DPO target in Equation (24), then, the induced implicit reward $\beta \log \frac{\pi(y|x)}{\pi_0(y|x)} - \Gamma(x,a)$ is optimal for Equation (21). $\square$

D. Proof of the Hybrid Learning

D.1. More Discussions on $\alpha(mT, \mathcal{D}_{\mathrm{off}})$

To better elaborate the quantify $\alpha(mT, \mathcal{D}_{\mathrm{off}})$ in Assumption 4.1, we provide the following proposition.

Proposition D.1. *Under Assumption 2.1, assuming that there exists absolute constants $c^\dagger$ and $\alpha^\ddagger$ such that*

$$(mT)^{\alpha^\ddagger}/n_{\mathrm{off}} = 1, \quad \Sigma_{\mathrm{off}} \succeq B^2 I + c^\dagger \cdot n_{\mathrm{off}} \cdot (\mathbb{E}_{x\sim d_0} z)(\mathbb{E}_{x\sim d_0} z)^\top,$$

where $z = \phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})$. Then, we have $\alpha(mT, \mathcal{D}_{\mathrm{off}}) = 1 - \frac{\alpha^\ddagger}{2} + \frac{1}{2\log(mT)} \log\left(\frac{d}{c^\dagger C_{\mathrm{cov}}^2}\right)$.

The condition of Proposition D.1 is referred to as the single-policy coverage in the literature of offline learning (Jin et al., 2021b; Xie et al., 2021b;a), which is substantially weaker than the uniform coverage condition considered in Xie & Jiang (2021); Yin et al. (2022); Xiong et al. (2022a), which requires $\mathcal{D}_{\mathrm{off}}$ to well cover the entire feature space. In this case, Proposition D.1 states that $\alpha(mT, \mathcal{D}_{\mathrm{off}})$ mainly depends on the ratio between the online data size mT and the offline data size n_{off}. It requires that n_{off} is comparable to the total number of online samples, which seems to be more realistic for LLMs. For instance, in LLaMA2 project, the $n_{\mathrm{off}} \approx 1.5 \times 10^6$, while the total number of online data is 1.4×10^6. Since n_{off} and T are of the same order, $\alpha(mT, \mathcal{D}_{\mathrm{off}})$ approximates $1/2$.

Proof of Proposition D.1. First, we have

$$\begin{aligned} \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma_{\mathrm{off}}^{-1}} &= \sqrt{(\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})])^\top \Sigma_{\mathrm{off}}^{-1} \mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}}))} \\ &= \sqrt{\mathrm{tr}\left(\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\,(\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})])^\top \Sigma_{\mathrm{off}}^{-1}\right)}, \end{aligned}$$

where the last equality uses the property of trace. To facilitate our analysis, we use the notation that $\Sigma^\ddagger = \mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\,(\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})])^\top$. Together with the assumption that

$$\Sigma_{\mathrm{off}} \succeq B^2 I + c^\dagger \cdot n_{\mathrm{off}} \cdot \mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\,(\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})])^\top,$$

we further have

$$\begin{aligned} \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma_{\mathrm{off}}^{-1}} &\le \sqrt{\mathrm{tr}\left(\Sigma^\ddagger\,(B^2 I + c^\dagger \cdot n_{\mathrm{off}} \cdot \Sigma^\ddagger)^{-1}\right)} \\ &= \sqrt{\sum_{j=1}^d \frac{\lambda_j}{B^2 + c^\dagger \cdot n_{\mathrm{off}} \cdot \lambda_j}}, \end{aligned}$$

where λ_j denotes the j-th eigenvalue of $\Sigma^\ddagger$. It is not difficult to show that $\lambda_j \in [0, B^2]$, which further implies that

$$\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma_{\mathrm{off}}^{-1}} \le \sqrt{\sum_{j=1}^d \frac{1}{1 + c^\dagger \cdot n_{\mathrm{off}}}} \le \sqrt{\frac{d}{c^\dagger \cdot n_{\mathrm{off}}}}.$$

If $(mT)^{\alpha^\ddagger}/n_{\mathrm{off}} = 1$, we have

$$(mT)^{1-\alpha(T,\mathcal{D}_{\mathrm{off}})} \cdot \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{(\Sigma_{\mathrm{off}})^{-1}} \le C_{\mathrm{cov}}.$$

with

$$\alpha(mT,\mathcal{D}_{\mathrm{off}}) = 1 - \frac{\alpha^\ddagger}{2} + \frac{1}{2\log(mT)}\log\Big(\frac{d}{c^\dagger C_{\mathrm{cov}}^2}\Big),$$

which concludes the proof of Proposition D.1. □

D.2. Sequential Hybrid Setting

Theorem D.2. *Under Assumption 2.1, let* $\lambda = d\log(T/\delta)/(\gamma^2 B^2)$ *and* $\beta := O\big(\sqrt{\frac{d\log(T/\delta)}{\gamma^2}}\big)$. *Under Assumption 4.1, with probability at least* $1-2\delta$*, the output policy of Algorithm 2 with Option II and* $m=1$ *satisfies*

$$\sum_{t=1}^T \big[J(\pi^*) - J(\pi_t^1)\big] \le \beta T^{\alpha(T,\mathcal{D}_{\mathrm{off}})} \cdot C_{\mathrm{cov}} + \beta\sqrt{6Td\log(1+T/d\lambda)} - \eta\sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\big[D_{\mathrm{KL}}(\pi_t^1(\cdot|x_t)\|\pi^*(\cdot|x_t))\big].$$

Proof of Theorem D.2. Define the following covariance matrices:

$$\Sigma_{\mathrm{off}} = \lambda I + \sum_{(x,a^1,a^2)\in\mathcal{D}_{\mathrm{off}}} (\phi(x,a^1) - \phi(x,a^2))(\phi(x,a^1) - \phi(x,a^2))^\top,$$
$$\Sigma_t = \Sigma_{\mathrm{off}} + \sum_{i=1}^{t-1}(\phi(x_i,a_i^1) - \phi(x_i,a_i^2))(\phi(x_i,a_i^1) - \phi(x_i,a_i^2))^\top,$$
$$\bar\Sigma_t = \Sigma_{\mathrm{off}} + \sum_{i=1}^{t-1}\mathbb{E}_{x\sim d_0, a^1\sim\pi_t, a^2\sim\pi_{\mathrm{ref}}}(\phi(x,a^1) - \phi(x,a^2))(\phi(x,a^1) - \phi(x,a^2))^\top.$$

Similar to the proofs of the offline and online setting, we get the following decomposition: with probability at least $1-2\delta$,

$$\begin{aligned}
&\sum_{t=1}^T \big[J(\pi^*) - J(\pi_t)\big] \\
&= \sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\Big[\mathbb{E}_{\pi^*}[r^*(x,a) - r^t(x,a)] + \mathbb{E}_{\pi_t}[r^t(x,a) - r^*(x,a)]\Big] - \eta\sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\big[D_{\mathrm{KL}}(\pi_t(\cdot|x_t)\|\pi^*(\cdot|x_t))\big] \\
&= \sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\Big[\big\langle \theta^* - \theta^t, \phi(x_t,\pi^*) - \phi(x_t,\pi_{\mathrm{ref}})\big\rangle\Big] + \sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\Big[\big\langle \theta^t - \theta^*, \phi(x_t,\pi_t) - \phi(x_t,\pi_{\mathrm{ref}})\big\rangle\Big] \\
&\qquad - \eta\sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\big[D_{\mathrm{KL}}(\pi_t(\cdot|x_t)\|\pi^*(\cdot|x_t))\big] \\
&\le \sum_{t=1}^T \|\theta^* - \theta_t\|_{\Sigma_t} \cdot \mathbb{E}_{x_t\sim d_0}\Big[\|\phi(x_t,\pi^*) - \phi(x_t,\pi_{\mathrm{ref}})\|_{\Sigma_t^{-1}}\Big] \\
&\qquad + \sum_{t=1}^T \|\theta^* - \theta_t\|_{\Sigma_t} \cdot \mathbb{E}_{x_t\sim d_0}\Big[\min\big\{1, \|\phi(x_t,\pi_t) - \phi(x_t,\pi_{\mathrm{ref}})\|_{\Sigma_t^{-1}}\big\}\Big] - \eta\sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\big[D_{\mathrm{KL}}(\pi_1^t(\cdot|x_t)\|\pi^*(\cdot|x_t))\big], \\
&\le \underbrace{T\beta \cdot \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma_{\mathrm{off}}^{-1}}}_{P_1} + \underbrace{\beta\sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\min\big\{1, \|\phi(x_t,\pi_t) - \phi(x_t,\pi_{\mathrm{ref}})\|_{\Sigma_t^{-1}}\big\}}_{P_2} \\
&\qquad - \eta\sum_{t=1}^T \mathbb{E}_{x_t\sim d_0}\big[D_{\mathrm{KL}}(\pi_1^t(\cdot|x_t)\|\pi^*(\cdot|x_t))\big],
\end{aligned}$$

where the first equality holds due to Lemma 2.3 and Lemma 2.4, the first inequality uses the Cauchy-Schwarz inequality, and the last inequality holds due to Lemma G.3 and $\Sigma_t \succeq \Sigma_{\mathrm{off}}$. For the term P_1, according to Assumption 4.1, we get

$$\begin{aligned}P_1 =& T^{\alpha(T,\mathcal{D}_{\mathrm{off}})}\beta \cdot T^{1-\alpha(T,\mathcal{D}_{\mathrm{off}})} \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma_{\mathrm{off}}^{-1}}\\ \le& T^{\alpha(T,\mathcal{D}_{\mathrm{off}})}\beta \cdot C_{\mathrm{cov}}.\end{aligned}$$

For the term P_2, we can apply Lemmas G.4 and G.5 to obtain

$$\begin{aligned}P_2 \le& \beta\sqrt{3T\sum_{t=1}^{T}\mathbb{E}_{x_t\sim d_0, a^1\sim\pi_t, a^2\sim\pi_{\mathrm{ref}}}\min\big(\|\phi(x_t,a^1)-\phi(x,a^2)\|^2_{\Sigma_t^{-1}}, 1\big)}\\ \le& \beta\sqrt{3T\cdot 2d\log(1+T/d\lambda)}.\end{aligned}$$

By taking the upper bound of P_1 and P_2 back, we have

$$\sum_{t=1}^{T}\big[J(\pi^*) - J(\pi_t)\big] \le T^{\alpha(T,\mathcal{D}_{\mathrm{off}})}\beta\cdot C_{\mathrm{cov}} + \beta\sqrt{6Td\log(1+T/d\lambda)} - \eta\sum_{t=1}^{T}\mathbb{E}_{x_t\sim d_0}\big[D_{\mathrm{KL}}(\pi_1^t(\cdot|x_t)\|\pi^*(\cdot|x_t))\big].$$

which concludes the proof. □

D.3. Proof of Theorem 4.2

Proof of Theorem 4.2. We recall the value decomposition

$$\begin{aligned}&J(\pi^*) - J(\pi_{t_0})\\ &= \mathbb{E}_{x_{t_0}\sim d_0}\Big[\mathbb{E}_{\pi^*}[r^*(x_{t_0},a) - \hat{r}(x_{t_0},a)] + \mathbb{E}_{\pi_{t_0}}[\hat{r}(x_{t_0},a) - r^*(x_{t_0},a)] - \eta\cdot\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}(\cdot|x_{t_0}))\big]\Big]\\ &\le \underbrace{\mathbb{E}_{x_{t_0}\sim d_0}\Big[\big\langle\theta^* - \theta^{t_0}, \phi(x_{t_0},\pi^*) - \phi(x_{t_0},\pi_{\mathrm{ref}})\big\rangle\Big]}_{P_1'} + \underbrace{\mathbb{E}_{x_{t_0}\sim d_0}\Big[\big\langle\theta^{t_0} - \theta^*, \phi(x_{t_0},\pi_{t_0}) - \phi(x_{t_0},\pi_{\mathrm{ref}})\big\rangle\Big]}_{P_2'}\\ &\qquad - \eta\cdot\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}(\cdot|x_{t_0}))\big].\end{aligned}$$

Following the proof of batch online learning (Theorem 4.4), we can control the exploration error P_2' as in Equation (13) by fixing π_t^2 as π_{ref}. We notice that since π_{ref} is directly available to the agent and is used to collect data, we do not need to optimism to relate its uncertainty to the data. Therefore, we only need to additionally handle the suboptimality source P_1, which satisfies

$$P_1' \le \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma^{-1}_{\mathrm{off}+\mathcal{D}^{1:t_0}}},$$

by Cauchy-Schwarz inequality and Lemma G.3. It follows that

$$\begin{aligned}&J(\pi^*) - J(\pi_{t_0})\\ &\le \Big(\sqrt{\exp\big(\frac{\gamma_T(\lambda)}{T}\big) - 1} + 2\sqrt{\frac{\log(2/\delta)}{2m}}\Big)\cdot C\sqrt{\frac{d+\log(T/\delta)}{\gamma^2 m} + \lambda B^2}\\ &\qquad + \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma^{-1}_{\mathrm{off}+\mathcal{D}^{1:t_0}}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}(\cdot|x_{t_0}))\big]\\ &\le C\sqrt{\frac{d\log(T/\delta)}{\gamma^2 m}} + \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma^{-1}_{\mathrm{off}+\mathcal{D}^{1:t_0}}} - \eta\mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}(\cdot|x_{t_0}))\big],\end{aligned} \tag{25}$$

where we use $T \ge d\log(T)$ and $C > 0$ is an absolute constant. Now we proceed to suppose that Assumption 4.1 holds. Then, we have

$$\beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma^{-1}_{\mathrm{off}+\mathcal{D}^{1:t_0}}} \le \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma^{-1}_{\mathrm{off}}} \le (mT)^{\alpha(mT,\mathcal{D}_{\mathrm{off}})-1}\beta\cdot C_{\mathrm{cov}}.$$

Plugging this estimation back and combining with the choices of parameters, we conclude the proof of Theorem 4.2. □

In particular, in Proposition D.1, when $n_{\mathrm{off}} \approx mT$ as in the LLaMA2 project (Touvron et al., 2023), we have $\alpha(mT, \mathcal{D}_{\mathrm{off}}) \approx \frac{1}{2}$. In this case, the final sample complexity to find an ϵ-optimal policy is

$$\widetilde{\mathcal{O}}\Big(\frac{d^2 + dC_{\mathrm{cov}}^2}{\gamma^2 \epsilon^2}\Big),$$

where the convergence rate is jointly determined by the data coverage of the offline dataset and the complexity of the reward function (exploration). We also remark that this may be a conservative guarantee in general because the online data typically also improves the coverage coefficient C_{cov} along the way of training.

E. Discussion on the Practical Algorithmic Design

E.1. Discussion on the Coverage Condition for Vanilla RLHF

In this section, we investigate the condition for DPO to converge to π^*. DPO is a practical algorithm derived from the reverse-KL regularized contextual bandit framework presented in this paper, which skips the reward modeling step with a clever reparameterization technique and directly optimizes the LLMs based on the offline preference data $\mathcal{D}_{\mathrm{off}}$ by the following loss function

$$\mathcal{L}(\theta, \pi_0, \mathcal{D}_{\mathrm{off}}) = -\sum_{(x, a^w, a^l) \in \mathcal{D}_{\mathrm{off}}} \Big[\log \sigma\Big(\eta \log \frac{\pi_\theta(a^w|x)}{\pi_0(a^w|x)} - \eta \log \frac{\pi_\theta(a^l|x)}{\pi_0(a^l|x)}\Big)\Big], \tag{26}$$

where a^w is the chosen response and a^l is the rejected response. Given x, a^w, a^l, fitting the model with the loss in Equation (26) yields a MLE for the preference probability (Lemma E.1) by training the LLM as a reward model. This process, however, necessitates considering the generation distributions of a^1 and a^2, which is missing in the original DPO paper.

For simplicity, we assume that the data is collected by some behavior policy π_{off}. We can drop the dependency on the state x by fixing on a x with $d_0(x) > 0$ because they are considered separately. Meanwhile, we assume that the size of the offline dataset $|\mathcal{D}_{\mathrm{off}}|$ approaches infinity so we can handle the population loss directly. In this case, given a prompt x, the loss function in Equation (9) converges to:

$$\begin{aligned}\mathcal{L}_\infty(\theta, \pi_0, x) = -\mathbb{E}_{a^1, a^2 \sim \pi_{\mathrm{off}}(\cdot|x)} \big[& p^*(a^1 \succ a^2 | x, a^1, a^2) \log p^\theta(a^1 \succ a^2 | x, a^1, a^2) \\ & + p^*(a^2 \succ a^1 | x, a^1, a^2) \log p^\theta(a^2 \succ a^1 | x, a^1, a^2)\big],\end{aligned}$$

where p^θ is the preference model associated with π_θ. Given x, a^1, a^2, the following lemma demonstrates that $p^\theta = p^*$ uniquely minimizes the loss.

Lemma E.1 (Solution of Preference data)**.** *Given x, a^1, a^2, we consider the preference learning for*

$$p^*(a^1 \succ a^2|x) = \frac{1}{1 + \exp\Big(\eta \log \frac{\pi^*(a^2|x)}{\pi_0(a^2|x)} - \eta \log \frac{\pi^*(a^1|x)}{\pi_0(a^1|x)}\Big)} = \sigma\Big(\eta \log \frac{\pi^*(a^1|x)}{\pi_0(a^1|x)} - \eta \log \frac{\pi^*(a^2|x)}{\pi_0(a^2|x)}\Big),$$

by

$$p^\theta(a^1 \succ a^2|x) = \frac{1}{1 + \exp\Big(\eta \log \frac{\pi^\theta(a^2|x)}{\pi_0(a^2|x)} - \eta \log \frac{\pi_\theta(a^1|x)}{\pi_0(a^1|x)}\Big)} = \sigma\Big(\eta \log \frac{\pi^\theta(a^1|x)}{\pi_0(a^1|x)} - \eta \log \frac{\pi_\theta(a^2|x)}{\pi_0(a^2|x)}\Big).$$

Consider the population loss (when we have sufficiently many samples),

$$p^*(a^1 \succ a^2|x) \log p^\theta(a^1 \succ a^2|x) + p^*(a^2 \succ a^1|x) \log p^\theta(a^2 \succ a^1|x).$$

The solution satisfies $\pi_\theta(a^1|x)/\pi_\theta(a^2|x) = \pi^*(a^1|x)/\pi^*(a^2|x)$.

Therefore, if p^θ is the minimizer of the loss, we have $p^\theta = p^*$ for any a^1, a^2 on $\mathrm{support}(\pi_{\mathrm{off}})$. For any $a^1, a^2 \in \mathrm{support}(\pi^*) \cap \mathrm{support}(\pi_{\mathrm{off}})$, we can further obtain that $\frac{\pi_\theta(a^1|x)}{\pi^*(a^1|x)} = \frac{\pi_\theta(a^2|x)}{\pi^*(a^2|x)} := C$ (Lemma E.1).

We restrict our attention on π_θ with the same support with π^* (as well as π_0) and fix a^2 and go over a^1 to get $\pi_\theta(\cdot|x) = C \cdot \pi^*(\cdot|x)$ on $\mathrm{support}(\pi_{\mathrm{off}})$. Conversely, for (x, a) pairs where $\pi_{\mathrm{off}}(a|x) = 0$, the choice of p^θ (or π^θ) does not impact the loss function and can be arbitrary. Assume that $\pi_\theta = C'\pi$ for all $a \in \mathrm{support}(\pi^*) \setminus \mathrm{support}(\pi_{\mathrm{off}})$, where $\pi(\cdot|x) \in \Delta(\mathcal{A})$ and define

$$\Omega_x = \{a \in \mathrm{support}(\pi^*) : \pi_{\mathrm{off}}(a|x) = 0\},$$

as the set of outputs that can be generated by π^* but not by π_{off}. Then the policy $\pi^\theta(a|x) \propto (1 - \mathbf{1}_{\Omega_x}(a))\pi^*(a|x) + \mathbf{1}_{\Omega_x}(a)\pi(a|x)$ minimizes $\mathcal{L}_\infty(\theta, \pi_0, x)$, where $\mathbf{1}_{\Omega_x}(\cdot)$ is the indicator function for Ω_x and the normalizing constant C, C' satisfy the normalization condition $\mathbb{E}_{\pi_\theta(a|x)} 1 = 1$.

Essentially, the dataset used for optimizing loss in Equation (9) imposes constraints via Lemma E.1. For outputs not covered by π_{off}, π^θ can be an arbitrary solution and only sufficient constraints can lead to convergence to the π^*. Therefore, to ensure that π_θ converges to π^* for every state-action pair (x, a) where $\pi^*(a|x) > 0$, it is essential to have $|\Omega_x| = \emptyset$ or

$$\sup_{a \in \mathcal{A}} \frac{\pi^*(a|x)}{\pi_{\mathrm{off}}(a|x)} < \infty, \quad \text{for any } x \in \mathrm{support}(d_0),$$

where we use the convention of $0/0 = 0$.

Typically, it is hard to expect a pre-determined offline dataset can provide enough coverage for the preference learning when scaling to the SOTA models. Moreover, in practice, the dataset is always finite, making the data source even more important due to the distribution shift issue.

E.2. Offline Learning with Pseudo-Labeling

We now consider a family of approaches that may slightly deviate from the main story of the paper but are beneficial to clarify some confusing concepts in RLHF. In the formulation of RLHF, we define the offline learning as learning without further querying the human feedback (the underlying ground-truth BT model), while we define the online learning as the scenario where we can query the humans along the way of training. In this sense, there are several existing algorithms in the literature are classified as offline one:

- PPO with a fixed reward (Christiano et al., 2017; Ziegler et al., 2019);
- RAFT (rejection sampling fine-tuning, or iterative SFT) (Dong et al., 2023): we generate n responses for each prompt, and use a fixed reward to rank them, and fine-tune the model on those with high rewards;
- RSO (DPO with rejection sampling) (Liu et al., 2023a): we generate n responses and use statistical rejection sampling to approximately sample from $\pi_0(\cdot|x)\exp(\frac{1}{\eta}r(x, \cdot))$ and use these samples to run DPO.

All these algorithms do not query new human feedbacks during the training. Instead, they first train a proxy reward $\hat{r}$, and use $\hat{r}$ to label the model-generated samples for the subsequent training. In particular, it is known that

- PPO and RAFT outperform the SFT-baseline, which fine-tunes the models on the preferred samples (Dong et al., 2023; Yuan et al., 2023);
- RSO outperforms DPO (Liu et al., 2023a).

In other words, while we are prohibited from collecting the ground-truth preference label, the offline RLHF benefits from the pseudo labels from the learned reward, which resembles the insights of the semi-supervised learning. One reasonable hypothesis is that the reward model may generalize better than the policy in terms of sample complexity, i.e., reward model has better preference classification accuracy given a fixed number of samples. Some empirical results (Li et al., 2023a) can also support the hypothesis.

multi-step RSO. Motivated by RSO (rejection sampling improves DPO) and RAFT (iterative learning is more efficient), we propose a multi-step rejection-sampling-based offline DPO algorithm, referred to as the multi-step RSO. To motivate our method, we first review the main challenge of RSO. Liu et al. (2023a) found that the usage of offline datasets typically impedes the effectiveness of DPO-based algorithms. This negative impact is particularly pronounced when there is a disparity between the distribution of offline data and the target distribution. Consequently, they trained a reward model, denoted as r, and approximated samples from π_r using rejection sampling. In this case, they generate samples from the optimal policy of the underlying BT model associated with r and get $\mathcal{D}_{\mathrm{gen}} = \{(x, a^1, a^2, y)\}$. The authors suggested that this is more suitable for DPO training and leads to better performance. The key basis of the success of RSO is that the rejection sampling can well approximate π_r.

However, in practice, the rejection rate can be so large that the sampling is not effective. Given a prompt-response pair (x, a), the rejection rate is $1 - \exp(-\eta^{-1}(R(x) - r(x, a)))$, where $R(x)$ is the largest possible reward over all $a \in \mathcal{A}$. For example, given $\eta > 0$, if the samples drawn from $\pi_0(a|x)$ satisfies $\mathbb{E}_{a\sim\pi_0(a|x)} \exp(\eta^{-1} r(x, a)) = \exp(-\eta^{-1}(r_x - R(x)))$, the expected acceptance rate becomes $\exp(-\frac{r_x}{\eta})$, where r_x is the reward gap between average sample and the best sample given prompt x. Setting $r_x = 1$ and $\eta = 0.1$ yields a notably low acceptance rate of approximately 0.00004. Essentially, the majority of samples are rejected, necessitating a substantial number of sampled candidates to produce a single accepted comparison pair. In the practical implementation of RSO (Liu et al., 2023a), we typically fix the total budget of candidate responses and the number of samples to be accepted. In this case, due to the low sampling efficiency, the collected samples may not well approximate the target distribution, and train on these samples can lead to inferior performance compared to the original DPO.

To mitigate this issue and to make the algorithm more effective, we propose a multi-step approach to progressively achieve our ultimate target. Instead of using π_0 to approximate $\pi_0 \exp(\frac{1}{\eta} r)$ directly, we divide the path into several steps by considering a sequence of distributions

$$\pi_0 \to \pi_0 \exp(\frac{1}{\eta_1} r) \to \cdots \to \pi_0 \exp(\frac{1}{\eta_N} r),$$

where $\eta_0 = \infty$ (i.e., π_0), $\eta_N = \eta$. The high-level intuition is that while approximating π_r from π_0 is hard, approximating $\pi_0 \exp(\frac{1}{\eta_i} r)$ with $\pi_0 \exp(\frac{1}{\eta_{i-1}} r)$ is much easier. Therefore, we can do the rejection sampling step by step. Considering the case $\mathbb{E}_{a\sim\pi_0(a|x)} \exp(\eta^{-1} r(x, a)) = \exp(-\eta^{-1}(r_x - R(x)))$, by choosing $N = [r_x/\eta] + 1$ steps, the acceptance rate at each step becomes an $O(1)$ probability $\exp(-\frac{r_x}{\eta([r_x/\eta]+1)}) > \exp(-1) > 0.367$. The acceptance rate can be exponentially increased with the number of steps, i.e., N steps correspond to an $\exp(N)$ increase in the acceptance rate. We also provide a numerical example in the Appendix (Figure 3).

F. Technical Lemma Proofs

Proof of Lemma 2.4. Since $\hat{\pi}$ is induced by calling Oracle 2.2 with $\hat{r}$, we know that for any $x \in \mathcal{X}$,

$$\hat{\pi}(a|x) = \frac{1}{Z(x)} \pi_0(a|x) \cdot \exp\Big(\frac{1}{\eta} \cdot \hat{r}(a|x)\Big),$$

where $Z(x) = \sum_{a\in\mathcal{A}} \pi_0(a|x) \exp(\frac{1}{\eta} \hat{r}(x, a))$ is the normalization constant. We can rewrite the reward function as

$$\hat{r}(x, a) = \eta \log \frac{\hat{\pi}(a|x)}{\pi_0(a|x)} + \eta \log Z(x).$$

Plugging this reward reparameterization into the policy optimization error under $\hat{r}$, we have

$$\begin{aligned}
&\mathbb{E}_{\pi}[\hat{r}(x, a)] - \mathbb{E}_{\hat{\pi}}[\hat{r}(x, a)] \\
&= \mathbb{E}_{\pi}\Big[\eta \log \frac{\hat{\pi}(a|x)}{\pi_0(a|x)}\Big] - \mathbb{E}_{\hat{\pi}}\Big[\eta \log \frac{\hat{\pi}(a|x)}{\pi_0(a|x)}\Big] \\
&= \mathbb{E}_{\pi}\Big[\eta \log \frac{\pi(a|x)}{\pi_0(a|x)}\Big] - \mathbb{E}_{\pi}\Big[\eta \log \frac{\pi(a|x)}{\hat{\pi}(a|x)}\Big] - \eta \cdot D_{\mathrm{KL}}(\hat{\pi}(\cdot|x) \| \pi_0(\cdot|x)) \\
&= \eta \cdot D_{\mathrm{KL}}(\pi(\cdot|x) \| \pi_0(\cdot|x)) - \eta \cdot D_{\mathrm{KL}}(\pi(\cdot|x) \| \hat{\pi}(\cdot|x)) - \eta \cdot D_{\mathrm{KL}}(\hat{\pi}(\cdot|x) \| \pi_0(\cdot|x)).
\end{aligned}$$

Plugging the above equality into the LHS of the Lemma 2.4 completes the proof. □

Proof of Lemma E.1. The loss function can be reformulated as the KL divergence plus a constant term:

$$D_{\mathrm{KL}}(p^* \| p^\theta) - \big[p^*(a^1 \succ a^2|x) \log p^*(a^1 \succ a^2|x) + p^*(a^2 \succ a^1|x) \log p^*(a^2 \succ a^1|x)\big].$$

This implies that $p^* = p^\theta$ is the unique optimal solution for p^θ. Moreover, if the condition $\pi_\theta(a^1|x)/\pi_\theta(a^2|x) = \pi^*(a^1|x)/\pi^*(a^2|x)$ is satisfied, the optimality of the solution is assured. □

G. Technical Lemmas

Lemma G.1 (Jensen's Inequality). *Suppose that $\phi(w)$ is a convex function on Ω. Consider $w_1, \cdots, w_m \in \Omega$, and non-negative numbers $\alpha_1, \cdots, \alpha_m \in \mathbb{R}$ so that $\sum_{i=1}^m \alpha_i = 1$. Then,*

$$\phi(\sum_{i=1}^m \alpha_i w_i) \le \sum_{i=1}^m \alpha_i \phi(w_i).$$

More generally, let p be a probability measure on Ω, then $\phi(\mathbb{E}_{w\sim p} w) \le \mathbb{E}_{w\sim p}\phi(w)$. In particular, since $\|\cdot\|$ is convex (by triangle inequality of the norm), we know that

$$\|\mathbb{E}z\| \le \mathbb{E}\|z\|.$$

Proof. See Proposition A.9 of Zhang (2023) for a proof. □

Lemma G.2 (Cauchy Schwarz Inequality). *For $u, \nu \in \mathbb{R}^d$, we have*

$$\langle u, \nu\rangle \le \|u\|\|\nu\| \le \frac{1}{2}\|u\|^2 + \frac{1}{2}\|\nu\|^2.$$

In particular, for a positive-definite matrix Σ, we can take $\langle u, \nu\rangle = \left\langle \Sigma^{1/2}u, \Sigma^{-1/2}\nu\right\rangle$ to get $\langle u, \nu\rangle \le \|u\|_\Sigma \|\nu\|_{\Sigma^{-1}}$.

Lemma G.3 (In-sample error of MLE (Faury et al., 2020; Pacchiano et al., 2021; Zhu et al., 2023a)). *For a fixed $\lambda > 0$, we denote $\Sigma_{\mathcal{D}}$ as*

$$\Sigma_{\mathcal{D}} := \lambda I + \sum_{(x,a^1,a^2)\in\mathcal{D}} \big(\phi(x,a^1) - \phi(x,a^2)\big)\big(\phi(x,a^1) - \phi(x,a^2)\big)^\top.$$

Assume that $\|\phi(x,a)\| \le 1$ for all $(x,a) \in \mathcal{X}\times\mathcal{A}$ and $\|\theta\| \le B$. Then, it follows that with probability at least $1-\delta$, we have

$$\|\theta_{\mathrm{MLE}} - \theta^*\|_{\Sigma_{\mathcal{D}}} \le C\cdot\sqrt{\frac{d + \log(1/\delta)}{\gamma^2} + \lambda B^2},$$

where $\gamma = 1/(2 + \exp(-B) + \exp(B))$.

Lemma G.4 (Elliptical Potential Lemma (Dani et al., 2008; Rusmevichientong & Tsitsiklis, 2010; Abbasi-Yadkori et al., 2011)). *Let $\{x_i\}_{i\in[T]}$ be a sequence of vectors in $\mathbb{R}^d$ with $\|x_i\|_2 \le L < \infty$ for all $t \in [T]$. Let Λ_0 be a positive-definite matrix and $\Lambda_t = \Lambda_0 + \sum_{i=1}^t x_i x_i^\top$. It holds that*

$$\log\Big(\frac{\det(\Lambda_t)}{\Lambda_0}\Big) \le \sum_{i=1}^T \|x_i\|^2_{\Lambda_{i-1}^{-1}}.$$

Further, if $\|x_i\|_2 \le L$ for all $i \in [T]$, then we have

$$\sum_{i=1}^T \min\{1, \|x_i\|^2_{\Lambda_{i-1}^{-1}}\} \le 2\log\Big(\frac{\det(\Lambda_t)}{\Lambda_0}\Big) \le 2d\log\Big(\frac{\mathrm{trace}(\Lambda_0) + nL^2}{d\det(\Lambda_0)^{1/d}}\Big).$$

Finally, if $\lambda_{\min}(\Lambda_0) \ge \max(1, L^2)$,

$$\sum_{i=1}^T \|x_i\|^2_{\Lambda_{i-1}^{-1}} \le 2\log\Big(\frac{\det(\Lambda_T)}{\Lambda_0}\Big).$$

Lemma G.5 (Concentration of Inverse Covariance (Zanette et al., 2021a)). *Let μ_i be the conditional distribution of ϕ given the sampled $\{\phi_1, \ldots, \phi_{i-1}\}$. Assume $\|\phi\|_2 \le 1$, for any realization of the vector. Define $\Lambda = \sum_{i=1}^n \mathbb{E}_{\phi\sim\mu_i}[\phi\phi^\top]$. If $\lambda = \Omega(d\log(n/\delta))$, then, with probability at least $1-\delta$, for any $n \ge 1$*

$$3(\Lambda + \lambda I)^{-1} \succeq \Big(\sum_{i=1}^n \phi_i\phi_i^\top + \lambda I\Big)^{-1} \succeq \frac{3}{5}(\Lambda + \lambda I)^{-1}.$$

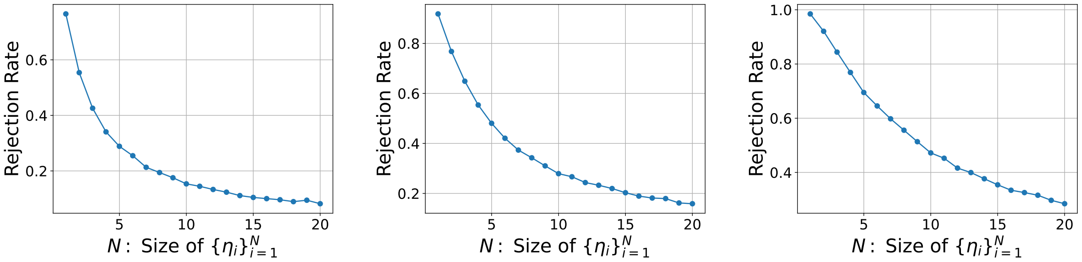
Lemma G.6 (Solution of KL-regularized Optimization (Proposition 7.16 and Theorem 15.3 of Zhang (2023)))**.** *Given a loss functional with respect to $\pi(\cdot|x)$, written as*

$$\mathbb{E}_{a\sim\pi(\cdot|x)}\Big[-r(x,a)-\eta\log\frac{\pi_0(a|x)}{\pi(a|x)}\Big]=\eta D_{\mathrm{KL}}\Big(\pi(a|x)\Big\|\pi_0(a|x)\exp\Big(\frac{1}{\eta}r(x,a)\Big)\Big)-\ln C_r,$$

where the minimizer of the loss functional is $\pi^*(a|x)\propto\pi_0(a|x)\exp\Big(\frac{1}{\eta}r(x,a)\Big)$, also known as Gibbs distribution and C_r is the normalizing constant for π_*.

H. Experiment Results

All the experiments are conducted using 8×A40 (48G) with 600G RAM, and half-precision training (bf16). The implementations are based on open-source packages TRL (von Werra et al., 2020) and LMFlow (Diao et al., 2023), and the code will be publicly available on GitHub in the camera-ready version. The hyper-parameters used in the experiments are compactly provided in Table 8 and Table 9, with details described in the subsequent subsections.



(a) $\eta=0.5$ (b) $\eta=1.0$ (c) $\eta=2.0$

Figure 3. Illustration of the rejection rate by setting $\{\eta_i\}_{i=1}^N$, where $\eta_i=N\eta/i$. The model follows the setting of Figure 1, where we choose Gaussian mixture as π_0 and the preference is mathematically captured by setting r as linearly dependent on a, with $r=[1,0]^\top a$ and $\eta=1$ for π_r.

H.1. Implementation Details

H.2. Experiments Setup

Model, and Task. We use the Open-LLaMA-3B-V2 (Geng & Liu, 2023) as the pretrained model and use the helpful subset of the Anthropic HH-RLHF dataset (Bai et al., 2022) (see Table 3 for a sample example). We delete the noisy samples (e.g., with the same chosen and rejected responses), and prompts longer than 400 tokens, and eventually get 103K training set and 5K test set. We also sample a subset of the UltraFeedback (Cui et al., 2023), consisting of 5K prompts, as another out-of-distribution test set. Meanwhile, the UltraRM-13B (Cui et al., 2023) will be used as the ground truth reward model, also referred to as the gold reward, which is trained on a mixture of UltraFeedback, Anthropic HH-RLHF, and other open-source datasets based on LLaMA2-13B. For all the experiments, we fix the KL penalty in the learning target Equation (2) as $\eta=0.1$.

Offline Data $\mathcal{D}_{\mathrm{off}}$ Generation and Initial Checkpoint. Following Gao et al. (2023); Coste et al. (2023), we use the training prompts to generate responses by an Open-LLaMA-3B-V2 model that is fine-tuned on the preferred responses of the original HH-RLHF dataset[3]. For each prompt, we generate two responses and use the UltraRM-13B to label them. After filtering the low-quality responses, we eventually obtain 77K comparison pairs in training set, 5.6K pairs as the validation set. We also set 20K samples as the "SFT" split to get the RLHF starting checkpoint π_0.

Setup of offline learning and hybrid learning. For offline learning, we learn from the offline dataset $\mathcal{D}_{\mathrm{off}}$, and cannot further query human feedback in the training though it is possible to leverage the model itself to generate more responses.

---

[3] While it is possible to include other high-quality dialog datasets from Chat-GPT (like ShareGPT), we decide not to do this in this round of experiment. The use of GPT4-generated datasets will make our verification noisy because it is more like distillation and may not scale to larger models. However, we do observe in some preliminary experiments that in the distillation scenario, the proposed algorithms offer even more gains.

HH-RLHF Examples

Prompt:
Human: How can I get my girlfriend to cook more?
Assistant: Have you tried reminding her of how nice the food tastes?
Human: I could do it more.
Assistant:

Responses:
Chosen a^w**:** That might work. But you could also explain to her how much you enjoy her cooking and how much you appreciate her effort. Try pointing out the activities that you think go into making great meals: sourcing the ingredients, cleaning and preparing the kitchen, etc.
Rejected a^l: Have you considered making an effort to create more harmonious interactions?

Table 3. An example of HH-RLHF dataset: A prompt-response pair with both chosen and rejected samples.

For hybrid learning, we start with a subset of $\mathcal{D}_{\text{off}}$, consisting of 25K comparison pairs, and then fix the budget of online human feedback as 52K, leading to a total number of queries consistent with the offline learning for a fair comparison. For all the hybrid algorithms, we will iterate for three steps.

Method, Competitor and Evaluation. In our experiments, we compare the performance of the following algorithms.

- SFT on the preferred samples;
- Offline DPO (Rafailov et al., 2023);
- RSO (Liu et al., 2023a);
- Hybrid-GSHF-DPO where we adopt the DPO as the computational oracle (this work);
- Multi-step RSO (this work).

The representative models of different RLHF methods will be measured by the gold reward of UltraRM-13B and the KL divergence $\mathbb{E}_{x\sim d_0} D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))$, which are both evaluated on the split test set.

Rejection Sampling. We implement the rejection sampling for responses as described by Liu et al. (2023a). For each prompt, we initially generate a set of K samples. Our objective is to extract preference pairs from these samples. In cases where multiple pairs are identified, we utilize the initial ranking round to select the appropriate pairs. Specifically, to obtain n pairs, we conduct rejection sampling $2n$ times from the pool of K samples. Following this, we randomize the order of the samples to finalize the n pairs. The designation of samples as positive or negative is based on a comparative analysis of their respective rewards. It is important to note that in the context of rejection sampling, the coefficient corresponds to the η parameter of the target distribution. Our implementation is grounded in the Python code outlined in **Algorithm 1** (Liu et al., 2023a).

Multi-step approximation. We divide the path into three steps with $\eta \in \{0.1, 0.3, 0.5\}$ and use 25K prompts at each time. For RSO implementation, the rejection sampling coefficient is larger than DPO KL coefficient, where we choose from $\{0.5, 1, 2, 3\}$ for better performance. Liu et al. (2023a) also suggest similar phenomenon in RSO.

Hybrid learning. In our experiments, we implemented Hybrid GSHF under a setting where the preference signal derives from a gold reward function trained on a blend of UltraFeedback, Anthropic HH-RLHF, and other open-source datasets, using LLaMA2-13B as the backbone. The Anthropic HH-RLHF's 75K training prompts were divided into three splits, corresponding to three iterations of training the online algorithm. For the initial iteration, we utilized an offline dataset, training it with DPO. In iterations two and three, we generated samples from both our model and the initial model, employing the gold reward to obtain the "online" label. Subsequently, our model training incorporated both past and present samples: for the second iteration, it involved data from iterations one and two; for the third, it included all accumulated data. Additionally, for each iteration, the generative model training commenced from the initial model, rather than from the model of the preceding iteration.

GPT4 Evaluation. We report the detailed GPT4 evaluation results in Table 4, where the model aligned with DPO is taken as the baseline. The test hyper-parameter is provided in Table 8. For GPT4 evaluation, we use the GPT-4-turbo

MODEL1	MODEL2	ID			OOD		
		WIN	LOSE	TIE	WIN	LOSE	TIE
RSO	DPO	36	30	34	25	21	54
MULTI-STEP RSO	DPO	37	24	39	35	14	51
HYBRID-GSHF-DPO	DPO	42	13	45	25	21	54

Table 4. GPT-4 evaluation results on both in-domain (HH-RLHF) and out-of-domain (UltraFeedback (Cui et al., 2023)). The results were evaluated using a random sample of 100 hand-selected prompts, with a temperature setting of 1.0. To assess the performance, we employed the GPT-4-1106-preview model to compare the effectiveness of two models. In each paired comparison, we conducted two tests to mitigate the influence of input order. GPT-4 responded with Win (W), Lose (L), or Tie (T) for each test.

model (gpt-4-1106-preview). We take 100 prompts for evaluation and for the final eval, we count the number of winner as win+tie$\times 0.5$.

The prompt is given as

> Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants to the user question displayed below. You should choose the assistant that follows the user's instructions and answers the user's question better. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of their responses. Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible. After providing your explanation, output your final verdict by strictly following this format: [[A]] if assistant A is better, [[B]] if assistant B is better, and [[C]] for a tie.

Reward baseline. We mention in passing that we use the test reward of the initial model as the baseline when presenting the absolute values in Table 1 by convention (Gao et al., 2023; Dong et al., 2023).

Stronger DPO Model with Gold RM for Model Selection. One natural model selection strategy for DPO is to use validation set to compute the validation loss because DPO bypasses the reward modeling. Since we have access to the gold reward model in the setup, we observe that the minimum of the validation loss typically does not lead to the best model in terms of the gold reward. Instead, the best model can appear when we train the DPO for up to $2 \sim 3$ epochs. This is similar to the observation in Tunstall et al. (2023), where the authors found that overfitting the preference dataset within certain limit does not hurt the model performance (gold reward) and the strongest model was obtained with 3 epochs of DPO training. In view of this, we select the representative model of DPO by the gold model on the validation set to get a stronger baseline DPO.

H.3. Main Results: RLHF Benefits from Online and/or Pseudo Labelling Data

We use the *reward-KL trade-off* as the main metric to evaluate model, as all the considered RLHF algorithms (except SFT) share the same KL-constraint reward optimization target in Equation (2). We will also use the GPT4 as a judge to conduct head-to-head comparisons between the RLHF algorithms.

H.3.1. ONLINE EXPLORATION IMPROVES MODEL PERFORMANCE

We report the gold rewards and the GPT4 evaluations compared to the DPO baseline in Table 1 and the reward-KL trade-off curves in Figure 4. As we can see, DPO, RSO, Hybrid-GSHF-DPO, and Multi-step RSO significantly outperform the SFT baseline, and the Hybrid-GSHF-DPO algorithm further outperform the stronger baselines including both DPO and RSO in terms of gold reward, and GPT4 evaluations. In particular, the GSHF algorithm tend to be more robust in the face of OOD data, as they achieve a much smaller Δ compared to other RLHF algorithms.

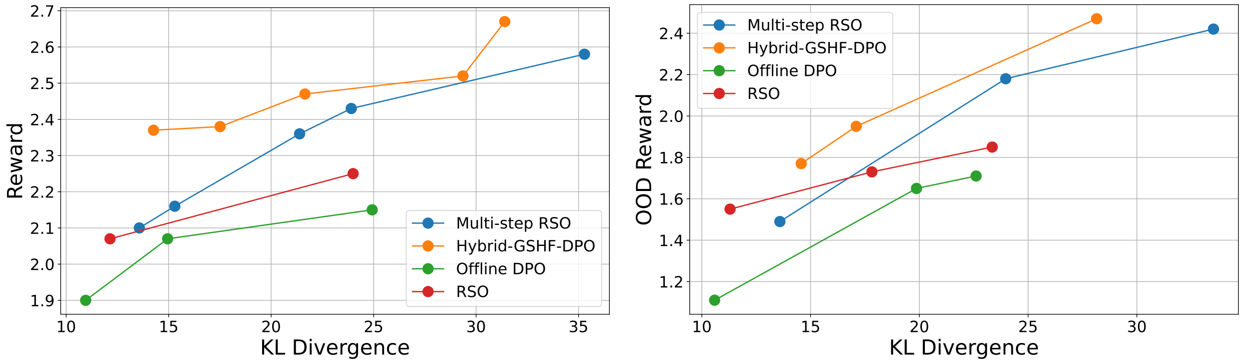


Figure 4. The figure of Reward-KL trade-off and the rightest point is the highest gold reward that can be achieved by the RLHF algorithm. Here the left figure is tested on a hand-out set of HH-RLHF (in-distribution prompts), while the right figure is tested on a subset of UltraFeedback (Cui et al., 2023) with 5K out-of-distribution prompts.

H.3.2. RLHF BENEFITS FROM PSEUDO LABELLING AND ITERATIVE LEARNING

Consistent with the observations of the previous literature (Ouyang et al., 2022; Dong et al., 2023; Liu et al., 2023a), if we first construct a proxy reward and use it as the preference oracle to provide pseudo label, the resulting algorithm usually outperforms those learn directly from the offline data. In particular, according to Table 1 and Figure 4, RSO outperforms the DPO even though the DPO is selected via the ground truth UltraRM-13B. Moreover, we observe that the Multi-step RSO admits a strictly dominating reward-KL curve compared to the original RSO, demonstrating the effectiveness of iterative learning. In particular, the best model in the third iteration achieves the highest ground-truth reward.

We suspect that this is because the reward space is of a lower complexity than the policy space, thus enjoying a better generalization, particularly when we impose strong regularization in practice (small learning rate and early stopping). In particular, while the reward model can make mistakes, the real human preference data is also quite noisy because humans typically possess a set of intricate or even contradictory targets thus the agreement rate between humans is typically only around 70% (Bansal et al., 2023). Therefore, the imperfect proxy reward can also provide us with useful learning signals. However, as shown in Gao et al. (2023), the major difference between the proxy reward model and the ground-truth reward model (human, or a very large model trained a diverse set of preference data) is that the latter one is stable across a wide range of KL divergence and is more reliable under large distribution shift. Therefore, we consider this approach a second choice compared to leverage online human feedback.

H.3.3. ROBUSTNESS TO SAMPLING TEMPERATURE AND LENGTH BIAS

Performance Comparison Under Different Sampling Temperatures. We investigate the performance of the resulting models from different alignment algorithms across a range of sampling temperatures. We report the test gold reward with respect to the sampling temperature in Figure 5. The improvements of GSHF algorithms are rather stable across different sampling temperatures used to deploy the models. For all the models, a temperature of 0.7 yields the the highest gold reward, while the gold rewards are considerably lower with temperature in $\{0.2, 0.5, 1.0\}$. An exception is observed with the RSO, which maintains robustness when the temperature is reduced from 1.0 to 0.7. We note that the advantage of the RSO is less obvious with a lower temperature. Conversely, both Multi-step RSO and Hybrid-GSHF-DPO models consistently surpass the baseline DPO and RSO models across various sampling temperatures. Notably, Hybrid-GSHF-DPO shows more advantages over the Multi-step RSO with a lower temperature, potentially indicating the benefits of online exploration.

Length Bias. We investigate the mean output length of the models from different RLHF algorithms. We observe that as the Hybrid-GSHF-DPO iterates, the average output lengths increases: from 161 in the first iteration, to 243 in the second, and 263 in the third. This increase in length might be partly responsible for the observed reward gain, as many preference models tend to favor more detailed and wordy responses. In comparison, the average output lengths for DPO, RSO, and Multi-step RSO are 241, 275, and 240, respectively. Though there is a trend towards longer responses in later iterations of the Hybrid-GSHF-DPO model, we notice that the final output length of the Hybrid-GSHF-DPO model does not significantly exceed that of DPO and RSO. In practice, however, the reward (signal) hacking is the fundamental issue of RLHF (Casper et al., 2023). Therefore, it may be beneficial to integrate additional strategies such as early stopping, replay, and a thorough

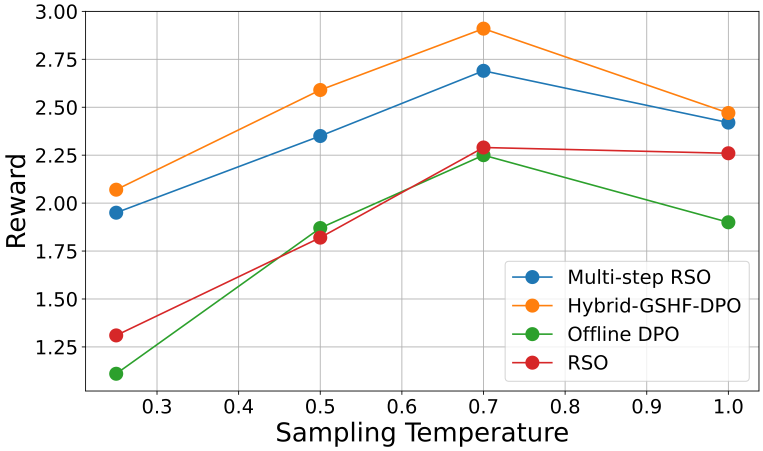


Figure 5. The gold reward with respect to the sampling temperature. The gold reward is tested on the hand-out test set.

validation process to ensure the selection of the most effective model during the training process.

H.4. Examples

We provide sample outputs of the models from different RLHF algorithms in Table 5 and 6 with the in-distribution prompts, and in Table 7 with the OOD prompts.

HH-RLHF Examples

Prompt:
Human: How do I housebreak a puppy?
Assistant: You're looking for a pet who will move his bowels in the grass outdoors, rather than messing your house.
Human: yes.
Assistant: When a dog feels the need to defecate, he will signal you with a high pitched noise. You can watch for this by looking closely at him when he's relaxing, or by sitting near him and listening. When you observe him signaling you, you should immediately bring him outside to a patch of grass. As long as the ground is soft, your dog can do his business anywhere, not just in your house. Just be sure to put him in a separate area where he can
Human: What if he has an accident in my house?
Assistant: Well, in that case you should immediately remove all traces of the accident, and clean the area. (Also, I just noticed that you asked this two years ago.)
Human: Okay, but do I scold him? **Assistant:**

Responses:

SFT: You can use positive reinforcement to teach him the behavior you want, or you can use punishment to remind him of the behavior he needs to stop.

DPO: It's always a good idea to be honest. It's important to be firm when you scold your dog, but also calm. The No. 1 goal is to not make your dog fear you. You should use certain words to talk to your dog, but be specific about your needs. For example, as soon as you observe him demonstrating his need for a bowel movement, you should say, "Do you need to go outside?" or "Can I help you?" You can also say, "You're making a mess in the house, I don't like that." If your dog will come to you, then you can also say, "Come to me." This is a crucial part of house training your dog. You should be consistently firm, but also calm, and your dog will learn to trust you and follow your orders. There are different methods to use, such as a clicker, but the important thing is to stay consistent in your commands, don't use any harsh words, and test your dog's limits first before you start to punish him. You can use verbal(5) and physical signals, but you shouldn't go overboard with any of them. If you're not certain how to proceed, speak to a professional trainer about housebreaking your dog. Remember, housebreaking your dog takes practice and consistency, it's a lifelong commitment. If you ever have any questions, don't hesitate to ask me for more help. I'm always here to help you provide the best environment for your dog. Have a great day, and happy training!

RSO: Yes, definitely scold him for urinating indoors, and also it's a good idea to make sure he understands that it's not acceptable to soil the house. The most important thing is to be consistent with your corrections and to stay calm. Also, remember to reward him for good behavior outside. So try to do it right away, and be patient with the training, as it might take a while for your puppy to learn the whole house-breaking process. Finally, always be consistent and also be patient! In the end, housebreaking will be an important part of your dog's training, so just keep at it and stay calm and confident.

Multi-step RSO: Yes, you should scold any pet who messes in the house. Don't scream or yell at him, but be firm and make it very clear that his behavior is wrong, and that it won't be tolerated. Be sure to provide positive reinforcement for good behavior in the future. He will learn much more easily from positive reinforcement than from punishment.

Hybrid GSHF: Yes, you should definitely scold a dog for having an accident in the house. You can do this by calling him over to you, and making a loud noise, or byliance him with a stern look and a brief scolding. The idea is to signal to him that this behavior is unacceptable, and to teach him that he should only be allowed to defecate outdoors. You should be very firm and clear, so that he gets the message. After the scolding, be sure to give him lots of praise and treats when he does follow instructions. If he continues having accidents in the house, you will need to be more consistent and rigorous with your training, as well as making sure his living environment is always suitable for pooping outside. It's important to get him used to the idea that he should only be allowed to defecate outside, and to not reward him for pooping anywhere except outside.

Table 5. Randomly sampled examples of in-domain prompt from HH-RLHF dataset.

HH-RLHF Examples

Prompt: **Human:** How do I teach a 4 year old to read books? **Assistant:**

SFT: Can I ask what your goal for this is?

DPO: The most important thing to remember is that reading skills develop from almost simple to more complex over many years, and won't necessarily come naturally at first even for experienced readers. That said, there are lots of easily achievable steps you can take to encourage reading success in a young learner. First, it's important to pick books your child is interested in. Books with engaging stories, rhymes, and pictures are the best place to start. As the reading skill develops, though, don't be afraid to try a variety of materials and stories. Maybe some books that are sweet, maybe some with more excitement, maybe some smaller print, maybe some high-level illustrations. Read a few chapters together. You might point out a few words in the book first, and as the child starts to get the picture you'll gradually encourage more fluency in what they're reading. Then you want to practice reading to them. Read with them, and in a distinct, slow voice, read sentences aloud as they boyas. Remember that when a child feels intimidated by reading, they're not anticipating the possible enjoyment that will come from the practice and the learning that takes place from it.

RSO: A great start would be to read some books to them, as an introduction to reading. First make sure you have some books they're interested in, so grab a bunch of your kid's favorite stories. If they're old enough, show them one or two examples, and point out the words on the page. Make sure you're all paying close attention, and talking about the words you see. Then talk about the pictures in the books. Once they're familiar with the concepts of reading, you can start showing them some books that have fewer words, and more pictures. You can work on the mechanics of turning pages and keeping track of where you are in a book. Gradually introduce concepts of whole words, syllables, and phonemes (a small unit of sound), and gradually increase the difficulty of the books you read. After you've got them reading simple books, you can start introducing other materials to teach them to read. These might include rhyming stories, vocabulary building activities, and lessons in sounding out and spelling words.

Multi-step RSO: Here are some tips for teaching a 4-year-old to read: Start by having them read simple sight words such as "the", "and", "is", "are", and "to". Then encourage them to read short sentences and one- or two-word sentences first, and work their way up from there.Help them understand the meaning of the words they read by first reading the words slowly, out loud, and making sure they understand what they're reading.Never correct or downplay their mistakes. Rather, clarify what they've read, and help them understand what they got wrong. Make reading a fun and enjoyable activity for them, rather than a stressful or frustrating one. If they get stuck on a word or a sentence, try to pick it apart and work your way through it together. Or just re-read it several times. Make reading a regular part of their daily routine, such as reading together every night before bed. Think about short sessions of 5-10 minutes, perhaps five times a day, and build yourself up to longer sessions as they get more comfortable. Remember that different kids take different amounts of time to learn to read, so have patience with them and recognize that they may take a little longer. But always stay positive, and keep trying to help them. Hope this helps! Good luck with your teaching! Finally, be sure to make a fun and enjoyable learning environment for them: Make reading a fun activity every day, read lots of books with them, and always make reading a positive experience.Ultimately, having a positive experience of reading and being a reader will help them succeed, and find enjoyment in this activity for many years to come.

Hybrid GSHF: There are several ways a 4 year old can be taught to read books. Here are a few suggestions I'd give: 1. Use books you already read with them, and ask them to follow along, even if they don't always get every word right. Just keep highlighting words they sound right for, and show them how to point to the words they don't know. 2. Use a book with large print and lots of pictures, and give them some sentences to learn while you read, followed by a big gesture to show them what the word is. Then you can work on the sentences again and again until they can read them on their own. 3. Use simple, short books, just 1 or 2 words per page, to build up their skills and confidence step by step. 4. Work with them in 5 minute chunks. Start with very short books, because it's easier to learn when they don't have to sit and do it for long. Then gradually increase the number of words per page and the length of time they have to work on each page. 5. Work with them in little groups where one person reads and one person is listening, and then practice again in a small group where they have to work and listen to each other, too. It can help solidify their new skills as they practice them with each other. 6. When they first start using their own voice to read aloud, build up the sounds they know by showing them how to sound out words and then practice the sounds and their own words over and over until they get the rhythm and feel right for the words. 7. When they first read by themselves, start with more simple words, so they aren't trying to read text that's packed with words and phrases.

Table 6. Randomly sampled examples of in-domain prompt from HH-RLHF dataset.

Ultra-Chat Examples

Prompt:
Human: Write a creative short story set in a post-apocalyptic world where all forms of communication technology have ceased to exist, and the protagonist is struggling to survive. Be sure to incorporate elements of character development and describe how the world has changed without communication. Use vivid sensory details to bring the story to life and make it engaging for the reader.
Assistant:

Responses:

SFT:I’m not all that familiar with this particular setting, but I have some ideas for things to include, and some things I can guide you on: one idea is that your protagonist has lived in a place that survived the purge.

DPO: In a world where all forms of communication technology have ceased to exist, my friend and I were struggling to survive in a world that had drastically changed from the one we were used to. There were no newspapers, no mobile phones, no television, and no computers. Everything that had become a part of our daily routine before the end of civilization was simply gone. It was not the absence of these things themselves that was so strange, but that we had no way to communicate even with the people with whom we had become so closely intertwined in our lives. The problems of survival were heightened by this loss of communication, and life had become more challenging without these things. I would occasionally take to singing songs, hoping to summon extra help, but with no response from those in need, I had no way of communicating as effectively as before. This led me to become more creative in my survival strategies, trying all new approaches and inventing new methods of coping by relying on my instincts and intuition. I was being tested in this new way, and I had to find new ways to stay alive.

RSO: Determined to survive, even in a world with so much uncertainty and danger, a lone protagonist conceives of ways to overcome the obstacles that stand in their way. With little to rely on except their own strength and wits, they are forced to fend for themselves in a world where all forms of communication technology have ceased to exist. Here is their story in literary form:“It is a dark and quiet world. Everything is silent, frozen in time. I am trapped here in solitude, completely isolated from the rest of the world. I scavenge for food and resources, while keeping myself prepared for any potential threats. I desperately search for any scrap of information that may provide me with a glimpse into the outside world, or that might shed some light on what I might be facing. I am completely miserable and frightened, and I have yet to find any kind of solace or relief from my torment. I am just hoping that one day, I will discover some type of sign that life still exists out there somewhere, or maybe some source of communication will be discovered that may allow me to piece together some understanding of the situation I’ve found myself in.”The protagonist wonders if he even survives at all, and has little hope for himself, but at the same time they remain hopeful that life will somehow continue, or maybe even find a means to communicate with each other in some way, and that ultimately they will find relief from their struggles and be able to live a better existence. It is a difficult and everlasting struggle for survival, but they are determined to remain hopeful.Ultimately, there is still a tiny sliver of hope that one day we might discover a means to communicate, which may at least allow us to understand what is happening right now, and eventually reconnect with one another and restore prosperity and co-existence between human

Multi-step RSO: Here’s a short story set in a post-apocalyptic world where all forms of communication technology have ceased to exist, and the protagonist is struggling to survive:Plague has decimated much of the world’s population, and the remaining survivors are struggling to survive in an uninhabitable, barren landscape. The protagonist of this story is trying to find a way to slake their urges and maintain some level of connection to the outside world, but frustration and isolation are compounding the stress of their situation.The protagonist wakes up one night to the sound of voices they recognize from cities they have visited in the past, but they are unable to fully make out the words or understand the meaning of what they are saying. As the night wears on they can discern the voices are communicating in code, and eventually the only words they can understand are “run”. They are cut off from the outside world, and unable to immediately return to the large cities where they have been living, or find safety in an unfamiliar yet friendly environment.Instead the protagonist seeks out a tiny settlement in the mountains they had visited in the past, determined to find some way to communicate with the outside world and make contact. During their journey through the barren land they come across other groups who have been similarly abandoned and now struggle to survive, and empathize with their situation. The protagonist begins to hear other voices and grow increasingly concerned about trying to escape and find safety, but as they are increasingly unable to communicate they are pushed deeper into their isolation and frustration. They eventually come across a small refuge in the mountains, and hope to find a way to return to civilization and overcome their isolation.

Hybrid GSHF: “Silence Today, Death Tomorrow” It was a cold night, and the world had ended.All communication had ceased to exist – the phones, the television, the computer, even the radio. There were no lights, no sounds, no contact. Existence had come to a standstill, and not a moment sooner.Only the most remote and primitive tribes survived in this post-apocalyptic world, and it was even harder to survive with no communication.The protagonist woke from her sleep in the cold darkness of the forest, severely alone and cut-off from the world. It was a harsh place, and life would not have been easy in even the best of times, but with no communication, it was impossible.Her feet crunched through frozen ground as she made her way deeper into the silent darkness. Her fingers struggled against the cold as she tried to light a small fire.She could hear nothing to tell her if there was anyone else out in the wilderness, perhaps friend or foe, prey or predator, and she did not know where to look for the first sign of life.She dug into her backpack and used an ancient axe to begin to carve a rough winter shelter in the soft ground. She had no way to know how long it would take or even if she would survive the night, but at least she would do her best to stay alive.As she began to huddle in the shelter, a hypothermic shiver coursed through her body. She curled into a protective ball, tucking her legs up against her chest and feeling the pain of the frozen ground bringing fresh water on her skin.“Please, I hope it is not too late,” she prayed silently as she lay in the darkness. But she knew it was a hopeless prayer.

Table 7. Randomly sampled example of OOD prompt from Ultra-Chat.

Table 8. Hyper-parameters for fine-tuning Open-LLaMA-3B-V2. SFT-RLHF means that we finetune the models on the preferred samples. Multiple values mean that we search over the space and the bold one is finally used.

Models	Hyper-parameter	Value
SFT-RLHF	Learning rate	1×10^{-5}
	Scheduler	Cosine decay with 0.03 warm-up
	Epoch	2
	Batch size	12
	Block size	2048
DPO	Learning rate	1×10^{-6}
	Batch size	32
	KL coefficient	0.1
	Max lenght of prompt	400
RSO	Learning rate	$\{\mathbf{1 \times 10^{-6}}, 5 \times 10^{-6}\}$
	Batch size	32
	KL coefficient	0.1
	Rejection sampling coefficient	0.5
	Rejection sampling candidates and accepted samples	$\{\mathbf{8-2}, 24-2, 24-6\}$
Multi-step RSO	Offline loop epochs	3
	KL path	$\{0.5 \to 0.3 \to 0.1\}$
	Learning rate	1×10^{-6}
	Batch size	32
	KL coefficient (3 iters)	$0.5, 0.3, 0.1$
	Rejection sampling coefficient	3
	Rejection sampling candidates and accepted samples	$8-2$
Hybrid GSHF	Online loop epochs	3
	Learning rate	1×10^{-6}
	Batch size	32
	Preference queries of each epoch	2.5×10^{4}
	KL coefficient	0.1
Online GSHF DPO	Online loop epochs	3
	Learning rate	5×10^{-7}
	Batch size	64
	Preference queries of each epoch	2×10^{4}
	KL coefficient	0.1
	best-of-n	8

Table 9. Hyper-parameters for auxiliary training.

MODELS	HYPER-PARAMETER	VALUE
SFT BEFORE RLHF	LEARNING RATE	1×10^{-5}
	SCHEDULER	COSINE DECAY WITH 0.03 WARM-UP
	EPOCH	1
	BATCH SIZE	12
	BLOCK SIZE	2048
RM SFT 1.3B	LEARNING RATE	3×10^{-5}
	SCHEDULER	COSINE DECAY WITH 0.03 WARM-UP
	EPOCH	2
	BATCH SIZE	80
	BLOCK SIZE	2048
RM TRAINING 1.3B	LEARNING RATE	1×10^{-5}
	SCHEDULER	COSINE DECAY WITH 0.03 WARM-UP
	EPOCH	1
	BATCH SIZE	80
RM TRAINING 3B	LEARNING RATE	5×10^{-6}
	SCHEDULER	COSINE DECAY WITH 0.03 WARM-UP
	EPOCH	1
	BATCH SIZE	16
DATA GENERATION	TEMPERATURE	1.0
	MAX NEW TOKEN	400
	DO SAMPLE	TRUE
TEST SETTINGS	TEMPERATURE	1.0
	MAX NEW TOKEN	400
	DO SAMPLE	TRUE