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**[3] **Heng Ji**[1] **Nan Jiang**[1] **Tong Zhang**[1]
University of Illinois Urbana-Champaign[1]
Salesforce AI Research[2]
The Hong Kong University of Science and Technology[3]
wx13@illinois.edu, tongzhang@tongzhang-ml.org

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

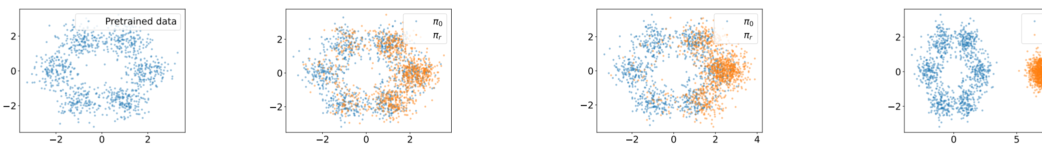
Following Ouyang et al. (2022); Zhu et al. (2023a); Rafailov et al. (2023); Liu et al. (2023a), we assume that there exists a ground-truth reward function $r^*(x,a) : \mathcal{X} \times \mathcal{A} \to [0,1]$ and the preference satisfies the Bradley-Terry model (Bradley & Terry, 1952):

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

where $a^1 \succ a^2$ means that a^1 is preferred to a^2, and $\sigma(\cdot)$ is the sigmoid function. Following Pacchiano et al. (2021); Kong & Yang (2022); Zhu et al. (2023a), for a clear presentation, we proceed by assuming that the reward function is parameterized by $r_\theta(x,a) = \langle \theta, \phi(x,a) \rangle$ for feature extractor $\phi : \mathcal{X} \times \mathcal{A} \to \mathbb{R}^d$ and $r^*(x,a) = \langle \theta^*, \phi(x,a) \rangle$ for some $\theta^* \in \mathbb{R}^d$. For regularization, we assume that $\|\phi(x,a)\| \leq 1$ for all possible $(x,a) \in \mathcal{X} \times \mathcal{A}$, $\|\theta\| \leq B$, and denote $\gamma = 1/(2 + \exp(-B) + \exp(B))$. We remark that the presented algorithmic design also applies to the general class and analysis in this paper readily generalizes to general function 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. 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 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), which largely

[*]The first three authors contribute equally.



(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: The 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. As η approaches zero in (b)-(d), π_r increasingly focuses on maximizing rewards, at the expense of the pretrained data's structure.

contradicts the principle of generative models. For example, the maximizer of the "safety reward" tends to avoid providing answers all the time. 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 pleasing yet irrelevant words to appease the reward model (Michaud et al., 2020; Tien et al., 2022). 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 (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 π_0, and the goal is to find a policy π from some policy class Π to maximize

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

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, aiming to inform and motivate practical algorithmic designs. Our contributions are summarized as follows:

- We *formally* formulate the RLHF process as a reverse-KL regularized contextual bandit problem in RLHF theory, which more accurately reflects real-world alignment practices (Ouyang et al., 2022; Bai et al., 2022a; 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;
- We design algorithms to address the formulated problems, which incorporate new uncertainty estimation or version space construction, and different non-symmetric exploration structures to handle the KL penalty, as well as the challenges of preference learning;
- Moving towards practical applications, we demonstrate that the proposed algorithms can be practically implemented and empirically outperform existing strong baselines like DPO (Rafailov et al., 2023) and RSO (Liu et al., 2023a) in real-world LLM experiments.

2 Main Results

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.

We first define the following information-theoretical policy improvement oracle, and defer a discussion on its practical approximations in Section F.

Definition 1 (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 11):*

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

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

Lemma 1. *Given a comparator policy π and a $\hat{r}: \mathcal{X} \times \mathcal{A} \to \mathbb{R}$, we can decompose the suboptimality of $\hat{\pi}$ as follows:*

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

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

Policy improvement error. When $\hat{\pi}$ is greedy in $\hat{r}$, we have $\mathbb{E}_{\pi}[\hat{r}(x,a)] - \mathbb{E}_{\hat{\pi}}[\hat{r}(x,a)] \le 0$. In the KL-constrained case, since the policy cannot be greedy or deterministic, we need to additionally handle the policy improvement error.

Lemma 2 (Policy optimization error)**.** *[Proof] 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 1 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 $D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))$.

Covariance matrix. Given a preference dataset $\mathcal{D} = \{(x, a^1, a^2, y)\}$, where y is the preference signal so that $y = 1$ means $a^1 \succ a^2$, and $y = 0$ indicates $a^1 \prec a^2$, we denote $\Sigma_{\mathcal{D}}$ as the covariance matrix estimation: $\lambda I + \sum_{(x,a^1,a^2) \in \mathcal{D}} \big(\phi(x,a^1) - \phi(x,a^2)\big)\big(\phi(x,a^1) - \phi(x,a^2)\big)^\top$.

2.1 Offline Learning

In the offline setting, we learn from a pre-collected $\mathcal{D}_{\mathrm{off}} = \{(x_i, a_i^1, a_i^2, y_i)\}_{i=1}^{n_{\mathrm{off}}}$. without further interactions with the human. We present two different pessimism-based algorithms in Algorithm 1 that are complementary to each other, and defer the detailed development to Appendix D. We have the following guarantee.

Algorithm 1 Offline GSHF

1: **Input:** $\mathcal{D}_{\mathrm{off}}$, $\lambda > 0$, $\beta > 0$, reference vector ν, and prompt distribution d_0.
2: Compute MLE estimation $\bar{r}$ based on $\mathcal{D}_{\mathrm{off}}$ by maximizing Equation (3);
3: `Option I`: Output $\hat{\pi}$ by constructing expected uncertainty estimator $\Gamma^e(\pi, \nu, \mathcal{D}_{\mathrm{off}})$ and solving

$$\operatorname*{argmax}_{\pi \in \Pi} \Big[\mathbb{E}_{x \sim d_0, a \sim \pi(\cdot|x)}[\bar{r}(x,a)] - \sqrt{\frac{d \log(1/\delta)}{\gamma^2}} \|\mathbb{E}_{x \sim d_0}[\phi(x,\pi)] - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} - \eta \mathbb{E}_{x \sim d_0}[D_{\mathrm{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))]\Big].$$

4: `Option II`: Output $\hat{\pi}$ by constructing uncertainty estimator $\Gamma(x, a, \nu, \mathcal{D}_{\mathrm{off}})$ and calling Oracle 1 with $\hat{r}(x,a) = \bar{r}(x,a) - \sqrt{\frac{d\log(1/\delta)}{\gamma^2}} \|\phi(x,a) - \nu\|_{\Sigma_{\mathrm{off}}^{-1}}$.

Theorem 1 (Informal)**.** *[Proof] For any comparator policy $\pi \in \Pi$, with suitable hyper-parameters, with high probability, the output policy of Algorithm 1 with Option I satisfies*

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

and Algorithm 1 with Option II satisfies

$$J(\pi) - J(\hat{\pi}) \le \sqrt{d \log(1/\delta)/\gamma^2} \cdot \mathbb{E}_{x \sim d_0, a \sim \pi(\cdot|x)} \|\phi(x,a) - \nu\|_{\Sigma_{\mathrm{off}}^{-1}} - \eta \cdot \mathbb{E}_{x \sim d_0}\big[D_{\mathrm{KL}}(\pi(\cdot|x)\|\hat{\pi}(\cdot|x))\big].$$

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 ν can serve as a pre-conditioner for a better sub-optimality bound. A typical choice is $\pi_{\mathrm{ref}} = \pi_0$ so that π_0 achieves a reward of zero (Ouyang et al., 2022; Gao et al., 2023). In comparison, the Option I achieves a sharper bound in the uncertainty bonus since by Jensen's inequality (Lemma 6) 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 $\mathcal{D}_{\mathrm{off}}$, 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 2.2. However, the main advantage of Option II is that the Oracle 1 can be well approximated by some empirical counterpart. We defer a detailed discussion to Appendix F.

2.2 (BATCH) HYBRID LEARNING

Beyond the offline learning, it is also common to query human feedback during the training process. For instance, Bai et al. (2022a); 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. While it is possible to learn from scratch (the online setting), 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. For completeness, we also develop the pure online setting in Appendix C.

We present the Algorithm 2 for the hybrid setting, where a notable feature is that the choices of (π_t^1, π_t^2) are **non-symmetric**. We defer a detailed development of the algorithm in Appendix E.

Algorithm 2 Hybrid GSHF

1: **Input:** Batch size $m, \Pi, \mathcal{D}_{\mathrm{off}}, \mathcal{D}^0 = \emptyset$, and π_{ref}.
2: **for** $t = 1, 2, \cdots, T$ **do**
3: Compute the MLE r^t based on $\mathcal{D}_{\mathrm{off}}$ and $\mathcal{D}^{1:t-1}$ with Equation (3).
4: Compute the policy π_t by calling the oracle in Assumption 1 with r^t;
5: Observe $x_{t,i} \sim d_0$, sample $a_{t,i}^1 \sim \pi_t$ and $a_{t,i}^2 \sim \pi_{\mathrm{ref}}$, receive human feedback for all $i \in [m]$, and collect them as $\mathcal{D}^t$.
6: **end for**
7: **Output:** the best model in $\pi_{1:T}$ by a validation set.

Theorem 2 (Informal). *[Proof] If $T = \widetilde{\Theta}(d)$, then with high probability, $\exists t_0 \in [T]$, so that Algorithm 2 satisfies*

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

where $\Sigma^{-1}_{\mathrm{off+t_0}}$ is the covariance matrix estimation on $\mathcal{D}_{\mathrm{off}}$ and $\mathcal{D}^{t_0}$.

The advantage of reward modeling. Theorem 2 and Theorem 3 (for the online setting) reveal a key characteristic of reward modeling: the sample complexity depends on the complexity of the reward model rather than the generative models. For simple tasks, such as sentiment or politeness evaluation, the required 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. (2022a); 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.

The advantage of online exploration. One difference between the Theorem 2 and Theorem 1 is that we now have the coverage condition related to both the offline dataset and the online exploration data. We expect that for many instances we encounter in practice, the online exploration will lead to a better coverage condition. This is because, under suitable assumption on the offline dataset, we know that $\pi_t \to \pi^*$ gradually. Then, the shift from data distribution $(\pi_t, \pi_{\mathrm{ref}})$ and the target $(\pi^*, \pi_{\mathrm{ref}})$ also gradually becomes smaller.

Models	Settings	Gold Reward	Gold Reward Win Rate	GPT4 Eval	OOD Gold Reward	Difference $\Delta \downarrow$
SFT	Offline	0.27	-	-	-0.21	0.48
DPO	Offline	2.15	0.5	0.5	1.71	0.44
RSO	Offline	2.25	0.54	0.53	1.89	0.36
Offline GSHF	Offline	2.59	0.63	0.57	2.41	**0.18**
Hybrid GSHF	Hybrid	**2.67**	**0.67**	**0.65**	**2.46**	0.21

Table 1: The evaluation results of the models from different RLHF algorithms. The gold rewards and win rate are computed on a hand-out test set, with the DPO as baseline. The Δ is the difference between the in-domain test reward and the OOD one. We count evaluation score as win$\times 1+$tie$\times 0.5$. See the detailed setup in Appendix J.

3 Practical implementations of the theoretical algorithms

The main challenge here lies in the Oracle 1, which is computationally intractable due to the exponentially large action space. To design an implementable algorithm, it is critical to approximate π_r effectively. Here we adopt the DPO algorithm (Rafailov et al., 2023) to showcase the effectiveness of the proposed framework and defer a more detailed discussion to Appendix F. Specifically, we can obtain Algorithm 3 by plugging the DPO into Algorithm 2.

Algorithm 3 Hybrid GSHF (Practical Implementation)

1: **Input:** Offline dataset $\mathcal{D}_{\text{off}}$, KL regularization $\eta > 0$, Online iterations T, batch size m, π_{ref}, $\mathcal{D}^0 = \emptyset$.
2: **for** $t = 1, 2, \cdots, T$ **do**
3: Let r^t denote the MLE estimator of the likelihood in Equation (3) based on $\mathcal{D}_{\text{off}}$ and $\mathcal{D}^{1:t-1}$.
4: Use DPO to train π_t on $\mathcal{D}_{\text{off}}$ and $\mathcal{D}^{1:t-1}$ so that $\pi_t \approx \pi_{r^t}(\cdot|x) \propto \pi_0(\cdot|x) \cdot \exp\left(\frac{1}{\eta} r^t(x, \cdot)\right)$;
5: Observe $x_{t,i} \sim d_0$, sample $a^1_{t,i} \sim \pi_t$ and $a^2_{t,i} \sim \pi_{\text{ref}}$, receive human feedback for all $i \in [m]$, and collect them as $\mathcal{D}^t$.
6: **end for**
7: **Output:** the best model in $\pi_{1:T}$ by a validation set.

The Power of Exploration. We present the main empirical evaluation results in Table 1 and focus on comparing different iterations of Hybrid GSHF in Figure 2. For each iteration, we evaluate the models every 400 steps and plot the representative models. Clearly, the previous iteration is strictly dominated by the subsequent one in terms of the frontier. This demonstrates the significant improvements achieved by iterating DPO with online data. Notably, 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.

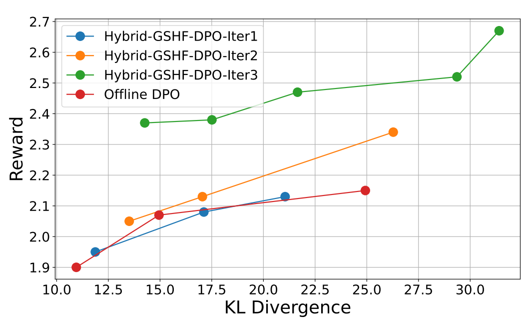


Figure 2: The Reward-KL trade-off curves of different iterations of Hybrid GSHF. The rightest point is the highest gold reward that can be achieved in that round.

Due to space constraint, we defer more practical implementation details to Section F and additional experimental details to Section J. We mention in passing that the concurrent work Hoang Tran (2024) empirically studies a similar approach and the resulting model ranks 2nd in the AlpacaEval 2.0 leaderboard (Li et al., 2023a) (the best model is GPT4-Turbo).

REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

Alekh Agarwal, Nan Jiang, Sham M Kakade, and Wen Sun. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, 32, 2019.

Alekh Agarwal, Yujia Jin, and Tong Zhang. 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.

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

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, et al. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*, 2021.

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

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

Chenjia Bai, Lingxiao Wang, Peng Liu, Zhaoran Wang, HAO Jianye, and Yingnan Zhao. Optimistic exploration with backward bootstrapped bonus for deep reinforcement learning. 2020.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

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

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

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

Mohak Bhardwaj, Tengyang Xie, Byron Boots, Nan Jiang, and Ching-An Cheng. Adversarial model for offline reinforcement learning. *arXiv preprint arXiv:2302.11048*, 2023.

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

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

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023.

Xiaoyu Chen, Han Zhong, Zhuoran Yang, Zhaoran Wang, and Liwei Wang. 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.

Leshem Choshen, Lior Fox, Zohar Aizenbud, and Omri Abend. On the weaknesses of reinforcement learning for neural machine translation. *arXiv preprint arXiv:1907.01752*, 2019.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Kamil Ciosek, Quan Vuong, Robert Loftin, and Katja Hofmann. Better exploration with optimistic actor critic. *Advances in Neural Information Processing Systems*, 32, 2019.

Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. Reward model ensembles help mitigate overoptimization. *arXiv preprint arXiv:2310.02743*, 2023.

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

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. 2008.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Shizhe Diao, Rui Pan, Hanze Dong, Ka Shun Shum, Jipeng Zhang, Wei Xiong, and Tong Zhang. Lmflow: An extensible toolkit for finetuning and inference of large foundation models. *arXiv preprint arXiv:2306.12420*, 2023.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, KaShun SHUM, and Tong Zhang. RAFT: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL `https://openreview.net/forum?id=m7p5O7zblY`.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Implementation matters in deep policy gradients: A case study on ppo and trpo. *arXiv preprint arXiv:2005.12729*, 2020.

Louis Faury, Marc Abeille, Clément Calauzènes, and Olivier Fercoq. Improved optimistic algorithms for logistic bandits. In *International Conference on Machine Learning*, pp. 3052–3060. PMLR, 2020.

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

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

Claudio Gentile, Zhilei Wang, and Tong Zhang. Fast rates in pool-based batch active learning. *arXiv preprint arXiv:2202.05448*, 2022.

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

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, et al. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*, 2023.

Yaru Hao, Zewen Chi, Li Dong, and Furu Wei. Optimizing prompts for text-to-image generation. *arXiv preprint arXiv:2212.09611*, 2022.

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

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

Baihe Huang, Jason D Lee, Zhaoran Wang, and Zhuoran Yang. Towards general function approximation in zero-sum markov games. *arXiv preprint arXiv:2107.14702*, 2021.

Huggingface. Preference tuning llms with direct preference optimization methods. *Blog*, 2023. URL `https://huggingface.co/blog/pref-tuning`.

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

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

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

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

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*, 2023.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`, 2023a.

Zihao Li, Zhuoran Yang, and Mengdi Wang. Reinforcement learning with human feedback: Learning dynamic choices via pessimism. *arXiv preprint arXiv:2305.18438*, 2023b.

Ziniu Li, Tian Xu, Yushun Zhang, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. Remax: A simple, effective, and efficient reinforcement learning method for aligning large language models. *arXiv e-prints*, pp. arXiv–2310, 2023c.

Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. *arXiv preprint arXiv:2309.06657*, 2023a.

Zhihan Liu, Miao Lu, Wei Xiong, Han Zhong, Hao Hu, Shenao Zhang, Sirui Zheng, Zhuoran Yang, and Zhaoran Wang. Maximize to explore: One objective function fusing estimation, planning, and exploration. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Eric J Michaud, Adam Gleave, and Stuart Russell. Understanding learned reward functions. *arXiv preprint arXiv:2012.05862*, 2020.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

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

Ellen Novoseller, Yibing Wei, Yanan Sui, Yisong Yue, and Joel Burdick. 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.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022.

Aldo Pacchiano, Aadirupa Saha, and Jonathan Lee. Dueling rl: reinforcement learning with trajectory preferences. *arXiv preprint arXiv:2111.04850*, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

Tabish Rashid, Bei Peng, Wendelin Boehmer, and Shimon Whiteson. Optimistic exploration even with a pessimistic initialisation. *arXiv preprint arXiv:2002.12174*, 2020.

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

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

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

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

Daniel Shin, Anca D Dragan, and Daniel S Brown. Benchmarks and algorithms for offline preference-based reward learning. *arXiv preprint arXiv:2301.01392*, 2023.

Yuda Song, Yifei Zhou, Ayush Sekhari, J Andrew Bagnell, Akshay Krishnamurthy, and Wen Sun. Hybrid rl: Using both offline and online data can make rl efficient. *arXiv preprint arXiv:2210.06718*, 2022.

Jeremy Tien, Jerry Zhi-Yang He, Zackory Erickson, Anca D Dragan, and Daniel S Brown. Causal confusion and reward misidentification in preference-based reward learning. *arXiv preprint arXiv:2204.06601*, 2022.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*, 2023.

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

Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. *arXiv preprint arXiv:2309.16240*, 2023a.

Yuanhao Wang, Qinghua Liu, and Chi Jin. Is rlhf more difficult than standard rl? *arXiv preprint arXiv:2306.14111*, 2023b.

Christian Wirth, Riad Akrour, Gerhard Neumann, Johannes Fürnkranz, et al. A survey of preference-based reinforcement learning methods. *Journal of Machine Learning Research*, 18(136):1–46, 2017.

Jeff Wu, Long Ouyang, Daniel M Ziegler, Nisan Stiennon, Ryan Lowe, Jan Leike, and Paul Christiano. Recursively summarizing books with human feedback. *arXiv preprint arXiv:2109.10862*, 2021a.

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

Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Better aligning text-to-image models with human preference. *arXiv preprint arXiv:2303.14420*, 2023.

Yue Wu, Shuangfei Zhai, Nitish Srivastava, Joshua Susskind, Jian Zhang, Ruslan Salakhutdinov, and Hanlin Goh. Uncertainty weighted actor-critic for offline reinforcement learning. *arXiv preprint arXiv:2105.08140*, 2021b.

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

Tengyang Xie, Ching-An Cheng, Nan Jiang, Paul Mineiro, and Alekh Agarwal. Bellman-consistent pessimism for offline reinforcement learning. *Advances in neural information processing systems*, 34:6683–6694, 2021a.

Tengyang Xie, Nan Jiang, Huan Wang, Caiming Xiong, and Yu Bai. Policy finetuning: Bridging sample-efficient offline and online reinforcement learning. *Advances in neural information processing systems*, 34:27395–27407, 2021b.

Wei Xiong, Han Zhong, Chengshuai Shi, Cong Shen, Liwei Wang, and Tong Zhang. Nearly minimax optimal offline reinforcement learning with linear function approximation: Single-agent mdp and markov game. *arXiv preprint arXiv:2205.15512*, 2022a.

Wei Xiong, Han Zhong, Chengshuai Shi, Cong Shen, and Tong Zhang. A self-play posterior sampling algorithm for zero-sum markov games. In *International Conference on Machine Learning*, pp. 24496–24523. PMLR, 2022b.

Yichong Xu, Ruosong Wang, Lin Yang, Aarti Singh, and Artur Dubrawski. Preference-based reinforcement learning with finite-time guarantees. *Advances in Neural Information Processing Systems*, 33:18784–18794, 2020.

Chenlu Ye, Wei Xiong, Quanquan Gu, and Tong Zhang. 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.

Ming Yin, Yaqi Duan, Mengdi Wang, and Yu-Xiang Wang. Near-optimal offline reinforcement learning with linear representation: Leveraging variance information with pessimism. *arXiv preprint arXiv:2203.05804*, 2022.

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

Yisong Yue, Josef Broder, Robert Kleinberg, and Thorsten Joachims. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.

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

Wenhao Zhan, Masatoshi Uehara, Nathan Kallus, Jason D Lee, and Wen Sun. Provable offline reinforcement learning with human feedback. *arXiv preprint arXiv:2305.14816*, 2023a.

Wenhao Zhan, Masatoshi Uehara, Wen Sun, and Jason D Lee. How to query human feedback efficiently in rl? *arXiv preprint arXiv:2305.18505*, 2023b.

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

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

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

Han Zhong, Wei Xiong, Sirui Zheng, Liwei Wang, Zhaoran Wang, Zhuoran Yang, and Tong Zhang. Gec: A unified framework for interactive decision making in mdp, pomdp, and beyond. *arXiv preprint arXiv:2211.01962*, 2022.

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

Banghua Zhu, Hiteshi Sharma, Felipe Vieira Frujeri, Shi Dong, Chenguang Zhu, Michael I Jordan, and Jiantao Jiao. Fine-tuning language models with advantage-induced policy alignment. *arXiv preprint arXiv:2306.02231*, 2023b.

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

A Organization of the Appendix

We organize the appendix as follows.

- In Appendix B, we review related works and introduce the basic notations of maximal likelihood estimation for reward modeling, covariance matrix, and rejection sampling that will be used in later algorithmic designs;
- In Appendix C, we develop the framework of the pure online learning, by formulating the setting and proposing statistically efficient algorithms;
- In Appendix D, we state the formal theorem of the offline learning and also provide the proof;
- In Appendix E, we develop the sequential hybrid setting, and provide the examples and proof for both of the sequential and hybrid cases;
- In Appendix F, we discuss how to practically implement the theoretical algorithms;
- In Appendix G, we discuss the connections between our theoretical findings and existing practical algorithms;
- In Appendix H and Appendix I, we present the useful technical lemmas and the proofs;
- In Appendix J, we present the experimental results and details.

B Notation Table and Backgrounds

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 eluder-type techniques and 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.
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 1.
Rejection Sampling	See Appendix B.4 for an introduction.
Best-of-n Policy	See Appendix B.4 for an introduction.

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

B.1 Related Work

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., 2022a) 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 ChatGPT4 (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., 2023c). 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. (2023b) presents a reduction-based framework, which can transform 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. (2023b) 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 F.

B.2 Maximum Likelihood Estimation and Policy Improvement Oracle

The most common way of reward modeling is Maximum Likelihood Estimation (MLE) (e.g., Ouyang et al., 2022; Bai et al., 2022a; Touvron et al., 2023).

Maximum Likelihood Estimation. 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:

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

The MLE is $\theta_{\mathrm{MLE}} = \mathrm{argmax}_{\|\theta\|\le B}\, \ell_{\mathcal{D}}(\theta)$ with $\Theta(B) = \{\theta \in \mathbb{R}^d : \|\theta\| \le B\}$. In practice, the MLE is also conducted with the LLMs (Touvron et al., 2023) on the preference dataset.

B.3 Covariance Matrix and Generalization

Before we continue to prove the main results of this paper, we would like to briefly illustrate the high-level intuitions why the algorithmic design and analysis are centered on the covariance matrix. Given a preference dataset $\mathcal{D}$, and a fixed $\lambda > 0$, we denote $\Sigma_{\mathcal{D}}$ as

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

Then, 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 $|\langle\theta_1 - \theta_2, \phi(x,a^1) - \phi(x,a^2)\rangle|$. 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:

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

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 7). Here, the $\|\phi(x,a^1) - \phi(x,a^2)\|_{\Sigma_{\mathcal{D}}^{-1}}$ 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 3 (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 $\|z_t\| \le 1$, 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 3, we can infer the out-of-sample error safely most of the time. Such an in-sample error estimation is provided in Lemma 8. 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).

B.4 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 $M \approx 1$.

C (BATCH) ONLINE LEARNING WITH ENHANCER

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

The mathematical formulation of the online learning is almost the same as the hybrid case, except that we now start from scratch instead of the offline dataset. Consider the batch online setting of T batches with fixed batch size m. At the beginning of each batch $t \in [T]$, An agent updates the policies π_t^1 and π_t^2. Then, m prompts $\{x_{t,i}\}_{i=1}^m$ are sampled from d_0. Based on each prompt $x_{t,i}$, two responses $(a_{t,i}^1, a_{t,i}^2)$ are generated from two policies (π_t^1, π_t^2), and a human preference signal $y_{t,i} \in \{0, 1\}$ is yielded according to the ground-truth BT model.

C.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). To understand this choice, we recall the decomposition Lemma 1 and Lemma 2 to obtain for each batch $t \in [T]$

$$
\begin{aligned}
& J(\pi^*) - J(\pi_t^1) \\
&= \mathbb{E}_{x_t \sim d_0}\Big[\mathbb{E}_{\pi^*}[r^*(x_t, a) - \hat{r}(x_t, a)] + \mathbb{E}_{\pi_t^1}[\hat{r}(x_t, a) - r^*(x_t, a)] - \eta \cdot \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_t^1(\cdot|x_t))\big]\Big] \\
&= \mathbb{E}_{x_t \sim d_0}\Big[\Big\langle \hat{\theta} - \theta^*, \phi(x_t, \pi_t^1) - \phi(x_t, \pi^*)\Big\rangle\Big] - \eta \cdot \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_t^1(\cdot|x_t))\big].
\end{aligned} \tag{4}
$$

The main technical challenge is to relate the uncertainty of $\phi(x_t, \pi_t^1) - \phi(x_t, \pi^*)$ (analysis target) to the uncertainty of $\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)$ (the pair to collect data). Our algorithmic idea is built on optimism and non-symmetric structures. We present the complete algorithm in Algorithm 4. The main agent π_t^1 always takes the policy induced by r^t from Oracle 1. On the other hand, the second agent π_t^2, referred to as the enhancer, seeks to maximize the uncertainty (similar to the practical choice of different model variants and temperature) for the fixed π_t^1, thus facilitating the learning of the main agent (similar idea was considered in the study of two-player zero-sum Markov game (Jin

Algorithm 4 Online GSHF with Enhancer

Input: $\Pi, \mathcal{D}^0 = \emptyset, m \geq 1$.
for $t = 1, 2, \cdots, T$ **do**
  Observe m independent prompts $\{x_{t,i}\}_{i=1}^m \sim d_0$;
  Compute the MLE-based reward r^t based on $\mathcal{D}^{1:t-1}$;
  Compute the policy π_t^1 by calling the oracle in Assumption 1 with r^t;
  Construct the confidence set of the policy as in Lemma 4;
  Compute the enhancer policy π_t^2 as in Equation (5);
  Independently sample actions $a_{t,i}^1 \sim \pi_t^1(\cdot|x_{t,i})$, $a_{t,i}^2 \sim \pi_t^2(\cdot|x_{t,i})$ and receive human feedback $y_{t,i}$, and collect $\mathcal{D}^t = \{(x_{t,i}, a_{t,i}^1, a_{t,i}^2, y_{t,i})\}_{i=1}^m$
end for

et al., 2021a; Huang et al., 2021; Xiong et al., 2022b)). In this case, the uncertainty compared to π^* is upper bounded by that of π_t^2, which is referred to as the principle of optimism in the literature (Auer et al., 2002). Notably, in contrast to the case of Markov game (Jin et al., 2021a; Huang et al., 2021; Xiong et al., 2022b), the enhancer also converges to π^* in terms of the metric of $J(\pi)$. We borrow the terminology of the main agent and enhancer to stress the non-symmetric algorithmic structure. Moreover, if we just regard the enhancer π_t^2 as an auxiliary policy and only care about the performance of π_t^1, there is no need to maintain the confidence set Π_t. Due to the realizability: $\pi^* \in \Pi$, we can construct π_t^2 as the solution of the following unconstrained problem:

$$\pi_t^2 = \mathop{\mathrm{argmax}}_{\pi_t^2 \in \Pi} \sum_{i=1}^m \Gamma(x_{t,i}, \pi_t^1, \pi_t^2, \mathcal{D}^{1:t-1}), \tag{5}$$

where the uncertainty bonus will be specified later. Note that in Algorithm 4, we formulate that the agent first observes m prompts and then establishes the enhancer. This is only for simplicity of analysis so that we can estimate the uncertainty and obtain the enhancer by maximizing the estimation. If we consider the standard online contextual bandit, we can first collect m contexts, and estimate the uncertainty based on them. Then, for the next m contexts, we interact with the environment in a strictly sequential manner using the policies determined by the first m contexts. This will only roughly incur a constant factor 2 in the final sample complexity.

To achieve optimism, we need to maintain a confidence set, that contains the π^* for all iterations with high probability. The constructions of the confidence set are different compared to the dueling RL (Faury et al., 2020; Pacchiano et al., 2021) due to the reverse-KL regularized contextual bandit formulation, as well as the non-symmetric structure in our algorithm. We summarize the confidence set construction for the online setting in the following lemma.

Lemma 4 (Confidence set). *Given the policy of the main agent π_t^1, we consider the following confidence set with $\beta = O\big(\sqrt{\frac{d\log(T/\delta)}{\gamma^2 m}}\big)$:*

$$\Pi_t = \Big\{\widetilde{\pi} \in \Pi : \beta \sum_{i=1}^m \|\phi(x_{t,i}, \widetilde{\pi}) - \phi(x_{t,i}, \pi_t^1)\|_{\Sigma_{t,m}^{-1}} - \eta \sum_{i=1}^m D_{\mathrm{KL}}(\widetilde{\pi}(\cdot|x_{t,i}) \| \pi_t^1(\cdot|x_{t,i})) \geq 0\Big\},$$

where we define

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

Then, with probability at least $1-\delta$, we know that $\pi^* \in \Pi_t$ for all $t \in [T]$.

We defer the proof to Appendix C.3. Intuitively, the enhancer aims to maximize the uncertainty of the feature difference, thus facilitating the learning of the main agent. In particular, the largest cost of KL divergence scales with the uncertainty of the difference, demonstrating the trade-off between the two considerations. Since $\pi^* \in \Pi_t$, we can upper-bound the first term on the right-hand side of Equation (4) by $\beta \mathbb{E}_{x_t \sim d_0} \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|_{\Sigma_{t,m}^{-1}}$, which can be further bounded by elliptical potential lemma. Hence, we can obtain the probably approximately correct (PAC) learning result in the following theorem.

Theorem 3. *For any $\epsilon > 0$, we set the batch size $m = d/(\gamma^2\epsilon^2)$. With the uncertainty estimator defined as*

$$\Gamma(x, \pi_t^1, \pi_t^2, \mathcal{D}^{1:t-1}) = \|\phi(x, \pi_t^1) - \phi(x, \pi_t^2)\|_{\Sigma_{t,m}^{-1}}, \tag{6}$$

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 $T = \min\{n \in \mathbb{N}^+ : n \geq d\log(n)\}$ iterations, we have with probability at least $1 - 3\delta$, there exists a $t_0 \in [T]$,

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

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

Theorem 3 reveals 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. We now present the proof of the theorem.

Proof of Theorem 3. 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 8 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 $t \in [T]$,

$$\begin{aligned}
\|\theta^t - \theta^*\|_{\Sigma_{t,m}} =& \frac{1}{\sqrt{m}}\|\theta^t - \theta^*\|_{\Sigma_{\mathcal{D}}} \\
\leq& \frac{C}{\sqrt{m}}\sqrt{\frac{d + \log(T/\delta)}{\gamma^2} + m\lambda B^2} \\
=& C\sqrt{\frac{d + \log(T/\delta)}{\gamma^2 m} + \lambda B^2}.
\end{aligned} \tag{7}$$

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 9), we have

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

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:

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

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

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

We now consider the suboptimality at iteration t_0:

$$\begin{aligned} J(\pi^*) - J(\pi^1_{t_0}) &= \mathbb{E}_{x_{t_0}\sim d_0}\Big[\big\langle \theta^{t_0} - \theta^*, \phi(x_{t_0}, \pi^1_{t_0}) - \phi(x_{t_0}, \pi^*)\big\rangle\Big] - \eta \mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi^1_{t_0}(\cdot|x_{t_0}))\big] \\ &\le \mathbb{E}_{x_{t_0}\sim d_0}\Big[\|\phi(x_{t_0}, \pi^1_{t_0}) - \phi(x_{t_0}, \pi^*)\|_{\Sigma^{-1}_{t,m}}\Big] \cdot \|\theta^{t_0} - \theta^*\|_{\Sigma_{t,m}} - \eta \mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi^1_{t_0}(\cdot|x_{t_0}))\big], \end{aligned} \tag{8}$$

where the inequality uses the Cauchy-Schwarz inequality (Lemma 7). Then, since the samples $\{x_{t,i}\}_{i=1}^m$ are i.i.d and for any $x \in \mathcal{X}$

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

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

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

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

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

We take the two inequalities above back into Equation (8) to derive with that probability at least $1-3\delta$,

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

where the second inequality applies Lemma 10 with $\lambda = \Omega(d\log(T/\delta)/m)$, and the last inequality uses Equation (7). By choosing T satisfying that $T \ge d\log(T)$ and $\lambda = \Theta(d\log(T/\delta)/m\gamma^2)$, we have

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

which concludes the proof. □

C.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):

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

where we now discuss in the sequential case with $m = 1$ in Algorithm 4. We consider two kinds of regrets: (1) cumulative suboptimality for the main policy π^1_t compared to π^*:

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

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

$$\pi_t^2 = \operatorname*{argmax}_{\pi_t^2 \in \Pi_t} \sum_{i=1}^{m} \Gamma(x_{t,i}, \pi_t^1, \pi_t^2, \mathcal{D}^{1:t-1}), \tag{9}$$

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

Theorem 4 (Sequential Online learning). *With the uncertainty estimator defined in Equation (6), 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 4 with $m = 1$ 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 4. First, recalling the regret decomposition in Equation (4), we deduce that with probability at least $1 - \delta$,

$$\begin{aligned}
&\sum_{t=1}^{T} \big[J(\pi^*) - J(\pi_t^1)\big] \\
&= \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\Big[\big\langle \theta^t - \theta^*, \phi(x_t, \pi_t^1) - \phi(x_t, \pi^*)\big\rangle\Big] - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big] \\
&\le \beta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0} \min\big\{1, \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi^*)\|_{\Sigma_t^{-1}}\big\} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big] \\
&\le \beta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0} \min\big\{1, \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|_{\Sigma_t^{-1}}\big\} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big] \\
&\le \beta \sqrt{T \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} \min\big\{1, \|\phi(x_t, a_t^1) - \phi(x_t, a_t^2)\|_{\Sigma_t^{-1}}^2\big\}} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\big],
\end{aligned} \tag{10}$$

where the first inequality uses the Cauchy-Schwarz inequality, Lemma 8 and reward $r \le 1$ for any $r \in \mathcal{F}$, the second inequality uses $\pi^* \in \Pi_t$ according to Lemma 4, 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 10, 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

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

By taking the result above back into Equation (10), we get with probability at least $1 - 2\delta$,

$$\sum_{t=1}^{T} \left[J(\pi^*) - J(\pi_t^1)\right] \le \beta \sqrt{T 6d \log(1 + T/d\lambda)} - \eta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\left[D_{\mathrm{KL}}(\pi^*(\cdot|x_t) \| \pi_1^t(\cdot|x_t))\right], \quad (11)$$

where the inequality uses Lemma 9.

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

$$\begin{aligned}
&\sum_{t=1}^{T} J(\pi_t^1) - J(\pi_t^2) \qquad (12)\\
&= \sum_{t=1}^{T} \underbrace{\mathbb{E}_{x_t \sim d_0}\Big[\mathbb{E}_{\pi_t^1}[r^*(x_t, a) - r^t(x_t, a)] + \mathbb{E}_{\pi_t^2}[r^t(x_t, a) - r^*(x_t, a)]\Big]}_{(\Delta_t^1)} \\
&\quad + \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0}\Big[\underbrace{\mathbb{E}_{\pi_t^1}[r^t(x_t, a)] - \mathbb{E}_{\pi_t^2}[r^t(x_t, a)] + \eta D_{\mathrm{KL}}(\pi_t^2(\cdot|x_t) \| \pi_0(\cdot|x_t)) - \eta D_{\mathrm{KL}}(\pi_t^1(\cdot|x_t) \| \pi_0(\cdot|x_t))}_{(\Delta_t^2)}\Big].
\end{aligned} \qquad (13)$$

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 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 (12), we have

$$\begin{aligned}
\sum_{t=1}^{T} J(\pi_t^1) - J(\pi_t^2) &\le 2\beta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0} \|\phi(x_t, \pi_t^1) - \phi(x_t, \pi_t^2)\|_{\Sigma_t^{-1}} \\
&\le 2\beta \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} \|\phi(x_t, a_t^1) - \phi(x_t, a_t^2)\|_{\Sigma_t^{-1}} \\
&\le 2\beta \sqrt{3T \sum_{t=1}^{T} \mathbb{E}_{x_t \sim d_0, (a_t^1, a_t^2) \sim (\pi_t^1, \pi_t^2)} \|\phi(x_t, a_t^1) - \phi(x_t, a_t^2)\|^2_{\tilde{\Sigma}_t^{-1}}} \\
&\lesssim \sqrt{T \beta^2 d},
\end{aligned} \qquad (14)$$

where the last inequality uses Lemma 9. 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 (14), we can obtain the bound for the average regret in the following theorem.

Therefore, by combining the results above and Equation (11), 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. $\square$

C.3 Construction of the Confidence Set

Proof of Lemma 4. 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

$$\begin{aligned}0 &\le \big\langle \theta^*, \phi(x_{t,i},\pi^*)-\phi(x_{t,i},\pi_t^1)\big\rangle + \eta D_{\mathrm{KL}}(\pi_t^1(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i})) - \eta D_{\mathrm{KL}}(\pi^*(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i}))\\ &= \underbrace{\big\langle \theta^*-\theta_t, \phi(x_{t,i},\pi^*)-\phi(x_{t,i},\pi_t^1)\big\rangle}_{\mathrm{Term(i)}}\\ &\quad + \underbrace{\big\langle \theta_t, \phi(x_{t,i},\pi^*)-\phi(x_{t,i},\pi_t^1)\big\rangle + \eta D_{\mathrm{KL}}(\pi_t^1(\cdot|x_{t,i})|\pi_0(\cdot|x_{t,i})) - \eta D_{\mathrm{KL}}(\pi^*(\cdot|x_{t,i})\|\pi_0(\cdot|x_{t,i}))}_{\mathrm{Term(ii)}},\end{aligned} \tag{15}$$

For Term (i), by Cauchy-Schwarz inequality and Lemma 8 with $\Sigma_{\mathcal{D}} = m\Sigma_{t,m}$ and $\lambda' = m\lambda$, we have

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

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 with $\pi = \pi^*$, $\hat{\pi} = \pi_t$, we obtain that

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

Taking respective upper bounds for Terms (i) and (ii) back into Equation (15) 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. $\square$

D More Details of the Offline Learning

In this section, we first motivate Algorithm 1 and then prove the main theoretical results of the offline learning.

We denote $\Sigma_{\mathrm{off}} := \Sigma_{\mathcal{D}_{\mathrm{off}}}$ for offline setting. To motivate the algorithmic design, we recall Lemma 1 and Lemma 2 to obtain that

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

where $\hat{\pi}$ is induced by calling the Oracle 1 with $\hat{r}$. As suggested in the offline learning literature (Jin et al., 2021b; Xie et al., 2021a), the first term can typically controlled by the property of $\mathcal{D}_{\mathrm{off}}$, while the second term is far more challenging to control because both the $\hat{\pi}$ and $\hat{r}$ are estimated from $\mathcal{D}_{\rm off}$, and hence spuriously correlate with each other. The standard methods to handle such a spurious correlation is to introduce pessimism in the algorithmic designs, which means that we adopt an estimator that is a lower bound of the true value with high probability. Specifically, instead of taking the MLE estimator directly, we penalize the reward estimation by an uncertainty estimator $\hat{r}(x,a) = r_{\rm MLE}(x,a) - \beta \cdot \Gamma(x,a,\nu,\mathcal{D}_{\rm off})$ so that $\hat{r}(x,a) - r^*(x,a) \leq 0$ for all $(x,a) \in \mathcal{X} \times \mathcal{A}$ and the spuriously correlated term can be eliminated. The construction of the uncertainty bonus is a standard application of concentration inequality, and we defer the details to Appendix D.

In addition to adopting a pessimistic reward estimation, we may also use a modified target that is biased toward pessimism by penalizing the uncertainty as in Equation (3). 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.

Theorem 5. *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 $\Gamma^e(\pi,\nu,\mathcal{D}_{\rm off}) = \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi) - \nu]\|_{\Sigma_{\rm off}^{-1}}$ satisfies*

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

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

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

Proof of Theorem 5. We start with Option I. If we set $\hat{r}(x,a) = \langle \theta_{\rm MLE}, \phi(x,a)\rangle$, and take the policy by

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

Then, we have

$$\begin{aligned} &\big\langle \theta_{\rm MLE}, \mathbb{E}_{x\sim d_0}\big[\phi(x,\pi) - \phi(x,\hat{\pi})\big]\big\rangle + \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\hat{\pi})] - \nu\|_{\Sigma_{\rm off}^{-1}} - \beta\cdot\|\mathbb{E}_{x\sim d_0}[\phi(x,\pi)] - \nu\|_{\Sigma_{\rm off}^{-1}} \\ &\quad + \eta\cdot\mathbb{E}_{x\sim d_0}\big[D_{\rm KL}(\hat{\pi}(\cdot|x)\|\pi_0(\cdot|x)) - D_{\rm KL}(\pi(\cdot|x)\|\pi_0(\cdot|x))\big] \leq 0. \end{aligned} \tag{16}$$

For simplicity, we denote the LHS of Equation (16) as $(\star)$. We plugging this into the estimation of $J(\pi) - J(\hat{\pi})$:

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

where the first inequality is from the Equation (16) and the second inequality uses Cauchy-Schwarz inequality and Lemma 8.

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

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

Then, we call Oracle 1 with $\hat{r}$ to get $\hat{\pi}$. By Lemma 1, we have

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

Since $\hat{r}$ is obtained from the Oracle 1 with $\hat{r}$, it follows from Lemma 2:

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

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

□

E More Details of Hybrid Learning

E.1 Algorithmic Development of Hybrid GSHF

Non-symmetric algorithmic structure. As we mention in Section 2.2, we will 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. We first motivate Algorithm 2. Using π_t^2 as an intermediate agent in Lemma 1, we have

$$
\begin{aligned}
J(\pi^*) - J(\pi_t^1) \le\ & \mathbb{E}_{x_t\sim d_0}\big[\langle \theta^* - \theta^t, \phi(x_t,\pi^*) - \phi(x_t,\pi_t^2)\rangle\big] \\
&+ \mathbb{E}_{x_t\sim d_0}\big[\langle \theta^t - \theta^*, \phi(x_t,\pi_t^1) - \phi(x_t,\pi_t^2)\rangle\big],
\end{aligned}
\tag{17}
$$

where we omit the KL error to simplify the discussion. The idea is most related to the hybrid RL theory to use $\mathcal{D}_{\mathrm{off}}$ to handle the term related to π^* (Song et al., 2022). However, for preference-based learning, one major difference is that the uncertainty is evaluated on the feature difference instead of a single state-action pair, which calls for an appropriate choice of π_t^2 to balance the sub-optimality sources. To this end, we introduce the reference policy π_{ref}, which satisfies the following two conditions. First, the π_{ref} may serve as the pre-conditioner for the offline coverage similar to Theorem 5 to control the first term related to π^*.

Assumption 1. *For the reference policy π_{ref}, there exists a ratio coefficient $\alpha(mT,\mathcal{D}_{\mathrm{off}}) \in (0,1)$ and a coverage constant $C_{\mathrm{cov}} > 0$ such that*

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

We remark that Assumption 1 implicitly assume that n_{off} is comparable to the total number of online samples mT so that the influence of $\mathcal{D}_{\mathrm{off}}$ 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 $\alpha(mT,\mathcal{D}_{\mathrm{off}})$ under standard partial coverage conditions in Appendix E.2. In particular, when $mT \approx n_{\mathrm{off}}$, we show that $\alpha(mT,\mathcal{D}_{\mathrm{off}}) \approx 1/2$. It is worth emphasizing that this scenario appears to be rather realistic for LLMs. For example, in the LLaMA2 project (Touvron et al., 2023), we observe $n_{\mathrm{off}} = 1500K$ and $mT = 1400K$. Moreover, to control the second term of Equation (17), we typically invoke the elliptical potential lemma 9. This requires π_{ref} to be available for collecting new data so the analysis target (π_t^1 and π_{ref}) and the policies used to collect data are identical. We present the complete algorithm in Algorithm 2 and the theoretical guarantee as follows.

E.2 More Discussions on $\alpha(mT,\mathcal{D}_{\mathrm{off}})$

To better elaborate the quantify $\alpha(mT,\mathcal{D}_{\mathrm{off}})$ in Assumption 1, we provide the following proposition.

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

The condition of Proposition 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 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 1. First, we have

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

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

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

we further have

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

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

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

which concludes the proof of Proposition 1. □

E.3 SEQUENTIAL HYBRID SETTING

Theorem 6. *Let $\lambda = d\log(T/\delta)/(\gamma^2 B^2)$ and $\beta := O\left(\sqrt{\frac{d\log(T/\delta)}{\gamma^2}}\right)$. Under Assumption 1, with probability at least $1-2\delta$, the output policy of Algorithm 2 with $m=1$ satisfies*

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

Proof of Theorem 6. Define the following covariance matrices:

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

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

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

where the first equality holds due to Lemma 1 and Lemma 2, the first inequality uses the Cauchy-Schwarz inequality, and the last inequality holds due to Lemma 8 and $\Sigma_t\succeq\Sigma_{\text{off}}$. For the term P_1, according to Assumption 1, we get

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

For the term P_2, we can apply Lemmas 9 and 10 to obtain

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

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

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

which concludes the proof. ∎

E.4 Proof of Theorem 2

We first restate the Theorem 2 for a slightly more general result.

Theorem 7 (Restatement of Theorem 2)**.** *For any $\epsilon > 0$, if $T = \min\{n \in \mathbb{N}^+ : n \geq d \log(n)\}$, with probability at least $1 - 3\delta$, there exists a $t_0 \in [T]$, so that the output policies of Algorithm 2 satisfy*

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

where $\beta := O\big(\sqrt{\frac{d+\log(1/\delta)}{\gamma^2} + \lambda B^2}\big)$, for any $\lambda > 0$ that is also used in $\Sigma_{\mathrm{off}+t_0}$.

We further suppose that Assumption 1 holds and we have $m = \Theta\big(\frac{d}{\gamma^2\epsilon^2} + \frac{1}{T}(\frac{\sqrt{d}C_{\mathrm{cov}}}{\gamma\epsilon})^{1/(1-\alpha(mT,\mathcal{D}_{\mathrm{off}}))}\big)$. Then, if $\lambda = \Theta(d\log(T/\delta)/(\gamma^2 B^2))$, with probability at least $1 - 3\delta$, there exists a $t_0 \in [T]$, so that the output policies of Algorithm 2 satisfy

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

Proof of Theorem 7. We recall the value decomposition

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

Following the proof of batch online learning (Theorem 3), we can control the exploration error P_2' as in Equation (9) 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

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

by Cauchy-Schwarz inequality and Lemma 8. It follows that

$$\begin{aligned}
&J(\pi^*) - J(\pi_{t_0}) \\
&\leq \Big(\sqrt{\exp\big(\frac{\gamma_T(\lambda)}{T}\big) - 1} + 2\sqrt{\frac{\log(2/\delta)}{2m}}\Big) \cdot C\sqrt{\frac{d + \log(T/\delta)}{\gamma^2 m} + \lambda B^2} \\
&\qquad + \beta \cdot \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma^{-1}_{\mathrm{off}}} - \eta \mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}(\cdot|x_{t_0}))\big] \\
&\leq C\sqrt{\frac{d\log(T/\delta)}{\gamma^2 m}} + \beta \cdot \|\mathbb{E}_{x\sim d_0}[\phi(x,\pi^*) - \phi(x,\pi_{\mathrm{ref}})]\|_{\Sigma^{-1}_{\mathrm{off}}} - \eta \mathbb{E}_{x_{t_0}\sim d_0}\big[D_{\mathrm{KL}}(\pi^*(\cdot|x_{t_0})\|\pi_{t_0}(\cdot|x_{t_0}))\big],
\end{aligned} \tag{18}$$

where we use $T \geq d\log(T)$ and $C > 0$ is an absolute constant. Now we proceed to suppose that Assumption 1 holds. Then, we have

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

Plugging this estimation back and combining with the choices of parameters, we conclude the proof of Theorem 7. □

In particular, in Proposition 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

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

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.

F PRACTICAL IMPLEMENTATIONS OF GSHF

In this section, we continue to discuss how to practically implement the information-theoretical Algorithm 1 and Algorithm 2.

F.1 APPROXIMATION THE INFORMATION-THEORETICAL COMPUTATIONAL ORACLE

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., 2022a) 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, DPO (Rafailov et al., 2023) attracted significant attention due to its stability and easy implementation. Specifically, DPO chooses to train the LLM as a reward model, by optimizing the following loss:

$$\sum_{(x,a_c,a_r)\in\mathcal{D}_{\mathrm{off}}} -\big[\log \sigma\big(\eta \log \frac{\pi_\theta(a_c|x)}{\pi_0(a_c|x)} - \eta \log \frac{\pi_\theta(a_r|x)}{\pi_0(a_r|x)}\big)\big], \tag{19}$$

where a_c, a_r is the chosen/rejected response. It is shown that the optimal policy for the DPO loss in Equation (19) is identical to the RLHF objective π_r, with r as the MLE of Equation (3). To summarize, moving toward the practical side from the theoretical algorithms, we may just replace the Oracle 1 with the practical RLHF algorithms (both deep RL methods or non-RL methods). In view of the simplicity and effectiveness of DPO, we will mainly investigate the performance of the proposed GSHF framework with DPO here. We remark that it is also possible to combine the proposed framework with any reasonable approximation of the computational oracle like IPO (Azar et al., 2023) and SLIC (Zhao et al., 2023).

F.2 MULTI-STEP REJECTION SAMPLING FOR OFFLINE LEARNING

Recently, Liu et al. (2023a) found that the effectiveness of the DPO is influenced by the offline data distribution. They emphasize the importance of sourcing offline training data from the target distribution. Consequently, they trained a reward model, denoted as r, and approximated samples from π_r using rejection sampling. We provide a brief introduction to rejection sampling in Appendix B.4. 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

Algorithm 5 Offline GSHF (Practical Implementation)

1: **Input:** Offline dataset $\mathcal{D}_{\mathrm{off}}$, KL regularization strength $\eta_1 > \cdots > \eta_N = \eta$.
2: Compute the reward estimator $\hat{r}(x,a)$ based on $\mathcal{D}_{\mathrm{off}}$.
3: **for** $i = 1, 2, \cdots, N$ **do**
4:     Denote $\pi_{\hat{r}}^i(\cdot|x) \propto \pi_0(\cdot|x)\cdot \exp\left(\frac{1}{\eta_i}\hat{r}(x,\cdot)\right)$
5:     Sample from $\pi_{\hat{r}}^i$, where $\pi_{\hat{r}}^i$ is obtained by rejection sampling from π_{i-1}. Use $\hat{r}$ to label the paired samples. Wrap up the data as $\mathcal{D}^i$.
6:     Use DPO with $\mathcal{D}^i$ to train $\pi_i \approx \pi_{\hat{r}}^i$.
7: **end for**
8: **Output:** Set $\pi = \pi_N$

sampling is not effective. The acceptance rate for the average sample is exponential to the gap between best reward and average reward. As the difference between the best reward and the average reward increases, the acceptance rate of a typical sample decreases exponentially. 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 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 $\{\pi_0, \pi_0 \exp(\frac{1}{\eta_1} r), \cdots, \pi_0 \exp(\frac{1}{\eta_N} r)\}$, where $\eta_0 = \infty$ (i.e., π_0), and $\eta_N = \eta$. 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 do rejection sampling step by step. We provide a numerical example in the Appendix (Figure 3). We also have a concrete computation where the multi-step rejection sampling improves the acceptance rate from 4×10^{-5} to 0.36 in Appendix G.3, as the acceptance rate can be exponentially increased with the number of steps.

One concern may be on the additional computations introduced by the multi-step approximations. However, in practice, the KL coefficient η is also tuned as a hyper-parameter in an outer loop of the proposed framework (Huggingface, 2023) to achieve the best performance. The Algorithm 5 provides us with a sequence of models associated with different η_i, which exactly allows for further model selection via hyper-parameter tuning of η. In view of this, the Algorithm 5 does not introduce overhead in computation.

F.3 Algorithmic Simplicity and Data Coverage

We note that all the three settings: offline, online (Appendix C), and hybrid learning are complementary to each other and hold their own values. For instance, collecting new and online human feedback can be expensive for most of the developers and in this case, only offline learning is feasible. One appealing choice is to leverage AI feedback (Bai et al., 2022b), which is much cheaper than human feedback. However, for tasks with customized needs or requiring expertise, we may only query feedback from specific users or experts, whose preference is distinct from AI.

Meanwhile, the hybrid learning offers simplicity in algorithmic design, at the cost of demand for a high-quality $\mathcal{D}_{\text{off}}$. In comparison, the online learning starts from scratch, 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 the advantage of a pessimistic MLE in RLHF has been showcased in a large amount of work (e.g., Christiano et al., 2017; Ziegler et al., 2019; Gao et al., 2023; Zhu et al., 2023a; Coste et al., 2023; Shin et al., 2023), we do not leverage pessimism in subsequent experiments but focus on verify the effectiveness of the proposed multi-step approximation approach. For the online setting, the uncertainty estimation is more challenging. We will discuss the potential ways to implement the uncertainty-aware algorithms in Appendix G.4 and defer a comprehensive empirical study to future study.

G Discussion on the Practical Algorithmic Design

In this section, we investigate the connections between the proposed algorithms and the existing practical algorithms in the literature, including Direct Preference Optimization (DPO) (Rafailov et al., 2023) Rejection Sampling Optimization (RSO) (Liu et al., 2023a), and RewArd-ranked Fine-Tuning (RAFT) (Dong et al., 2023).

G.1 Data Coverage and Preference Learning

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 $\mathcal{D}_{\text{off}}$ by the following loss function

$$\mathcal{L}(\theta, \pi_0, \mathcal{D}_{\text{off}}) = -\sum_{(x,a_c,a_r)\in\mathcal{D}_{\text{off}}} \Big[\log\sigma\Big(\eta\log\frac{\pi_\theta(a_c|x)}{\pi_0(a_c|x)} - \eta\log\frac{\pi_\theta(a_r|x)}{\pi_0(a_r|x)}\Big)\Big], \tag{20}$$

where a_c is the chosen response and a_r is the rejected response. Given x, a_c, a_r, fitting the model with the loss in Equation (20) yields a MLE for the preference probability (Lemma 5) 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. We now discuss the influence of the offline data distribution.

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 $|\mathcal{D}_{\text{off}}|$ approaches infinity so we can handle the population loss directly. In this case, given a prompt x, the loss function in Equation (19) converges to:

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

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 5 (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 $\text{support}(\pi_{\text{off}})$. For any $a^1, a^2 \in \text{support}(\pi^*) \cap \text{support}(\pi_{\text{off}})$, 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 5).

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 $\text{support}(\pi_{\text{off}})$. Conversely, for (x, a) pairs where $\pi_{\text{off}}(a|x) = 0$, 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 $a \in \text{support}(\pi^*) \setminus \text{support}(\pi_{\text{off}})$, where $\pi(\cdot|x) \in \Delta(\mathcal{A})$ and define

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

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 (19) imposes constraints via Lemma 5. 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

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

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. In other words, although the two distributions have the same support, the density distribution can be largely different. If we take $\pi_{\text{off}} = \pi_0$ as suggested by Rafailov et al. (2023), for any point that $\pi^*(a|x) > 0$, we have

$$\frac{\pi^*(a|x)}{\pi_{\text{off}}(a|x)} = \frac{\exp(\frac{1}{\eta} r^*(x,a))}{\mathbb{E}_{a' \sim \pi_0(\cdot|x)} \exp(\frac{1}{\eta} r^*(x,a'))} \leq \frac{1}{\pi_0(a|x)} < \infty.$$

However, since the outpace $\mathcal{A}$ is exponentially large with respect to the sequence length, the ratio of using π_0 can be extremely large in the worst case, which may also lead to an inferior performance in practice, as shown in Liu et al. (2023a). On the other hand, (1) RSO uses rejection sampling to approximately sample data from π_r; (2) Offline GSHF improves RSO by adopting a more efficient multi-approach way to better approximate π_r given the limited generation budget; (3) Hybrid GSHF uses both the offline dataset and the data from online exploration. These algorithms adopt different data sources for the preference learning thus exploring different parts of the state-action space.The improvements of the GSHF algorithms emphasize the importance of a more efficient data augmentation strategy and further exploration of the state-action space.

G.2 Iterative RLHF Training

The multi-step approximation in Algorithm 5 shares similar spirit with the iterative framework (i.e., the RAFT algorithm) proposed in Dong et al. (2023) and was also considered in Touvron et al. (2023) and Gulcehre et al. (2023). Our multi-step rejection sampling may be viewed as a generalization of that of RAFT, as we illustrate as follows.

RAFT starts from π_0 and aims to learn from the induced best-of-n policy (i.e., for each prompt x, we collect n independent responses and output the one with highest reward). By standard concentration inequality, if the reward function is bounded by M, the upper bound of the best-of-n policy satisfies

$$\underbrace{\mathbb{E}_{a \sim \pi_0} r(x,a)}_{\text{Base policy}} \leq \underbrace{\mathbb{E}_{a_i \sim \pi_0} \max_{i \in [n]} r(x,a_i)}_{\text{Best-of-n policy}} \leq \mathbb{E}_{a \sim \pi_0(\cdot|x)} r(x,a) + \sqrt{\frac{M^2}{2} \log n},$$

which increases at a rate of $\sqrt{\log n}$. Therefore, the marginal benefit of increasing n diminishes quickly, which motivates the authors to adopt the iterative framework because the improved base policy will lead to an improved best-of-n policy. In comparison, we decompose the target policies $\pi_0 \exp\left(\frac{1}{\eta_N} r\right)$ to several steps and we will use the improved policy associated with η_i as the base policy to approximate that with η_{i+1} with rejection sampling. The multi-step rejection sampling is far more efficient compared to using π_0 because the rejection rate is reduced.

Another major difference is that Dong et al. (2023) only considers reward optimization without the KL constraint from the initial checkpoint. Therefore, they choose to train the model from the checkpoint obtained from the preceding iteration. On the other hand, we always start from the initial model at each iteration.

G.3 Acceptance Rate of Rejection Sampling

Section G.2 highlight that the efficiency of iterative training process, especially for the sample complexity. Here, we will delve into the acceptance rate of the rejection sampling for the reward induced Gibbs distribution.

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 4×10^{-5}. 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 the rejection sampling rate cases for Gaussian mixture model (settings follow Figure 1) in Figure 3.

G.4 Heuristic Uncertainty Estimation and Implementation of Pessimism and Optimism

The uncertainty estimation for LLMs can be challenging due to the extremely large state-action space and a closed-form solution similar to the potential is unavailable in general.

Pessimistic MLE for Reward Modeling. The recent work (Coste et al., 2023) implements the principle of pessimism based on ensemble in two different ways, and demonstrate the effectiveness of them using real-world LLM alignment experiments. Specifically, to create an ensemble, the authors train 5 independent reward models with different random seeds $\{r_i\}_{i=1}^5$. First, the authors consider worst-case optimization (Boyd & Vandenberghe, 2004), which gives a pessimistic reward estimation:

$$\hat{r}(x, a) = \min_{i\in[5]} r_i(x, a).$$

Second, the authors also consider a soft version of pessimism by penalizing the variance of estimation (Wu et al., 2021b):

$$\hat{r}(x, a) = \bar{r}(x, a) - \lambda \frac{1}{5} \sum_{i=1}^{5} \big(r_i(x, a) - \bar{r}(x, a)\big)^2,$$

where $\bar{r}(x, a) = \frac{1}{5}\sum_{i=1}^5 r_i(x, a)$ and the $\lambda > 0$ is a tuning parameter. It was observed that such a pessimistic RM can largely mitigate the issue of overfitting in RLHF. We refer interested readers to Coste et al. (2023) for details.

Optimistic Policy Selection for Enhancer. In comparison, selecting an appropriate optimistic policy for the enhancer to maximize the uncertainty with respect to the main agent $\pi_t^1 = \pi_{r^t}$ is largely less explored in practical applications. The enhancer aims to maximize the uncertainty of the feature difference given in Equation (6). While there are works adopt an optimistic value estimation in practical DRL applications (Ciosek et al., 2019; Bai et al., 2020; Rashid et al., 2020), direct optimism in terms of the policy seems to be far more challenging. Meanwhile, we are in the face of distinct challenges from preference learning. These together call for new ideas for the practical implementations.

Essentially, the results in both hybrid learning and online learning presented in this paper emphasize the importance of sampling strategy for iterative RLHF. Although the optimistic enhancer is not readily available in practice, the theoretical insights behind such a choice of enhancer is that the enhancer should generate response so that the difference between it and that of the main agent is large, compared to the data collected so far, which should at least motivate the future algorithmic design in principle.

Since the advantages of pessimism in offline RLHF has been verified in a large amount of work (e.g., Christiano et al., 2017; Ziegler et al., 2019; Gao et al., 2023; Zhu et al., 2023a; Coste et al., 2023; Shin et al., 2023), we do not leverage pessimism in the experiments of this paper but focus on verify the effectiveness of the proposed multi-step rejection sampling. Moreover, as we cannot find a practical approximation for the optimistic enhancer, we hope that our theoretical insights can motivate future study in this direction to construct reliable and efficient uncertainty estimators for LLMs, especially for the implementation of an optimistic enhancer.

H Technical Lemma Proofs

Proof of Lemma 2. Since $\hat{\pi}$ is induced by calling Oracle 1 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 completes the proof. □

Proof of Lemma 5. The loss function can be reformulated as the KL divergence plus a constant term:

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

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. □

I Technical Lemmas

Lemma 6 (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 7 (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 8 (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 $\gamma = 1/(2 + \exp(-B) + \exp(B))$.

Model1	Model2	ID			OOD		
		Win	Lose	Tie	Win	Lose	Tie
RSO	DPO	36	30	34	25	21	54
Offline GSHF	DPO	37	24	39	35	14	51
Hybrid GSHF	DPO	42	13	45	25	21	54

Table 3: 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.

Lemma 9 (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 10 (Concentration of Inverse Covariance (Zanette et al., 2021))**.** *Let μ_i be the conditional distribution of ϕ given the sampled $\{\phi_1, \dots, \phi_{i-1}\}$. 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 11 (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*

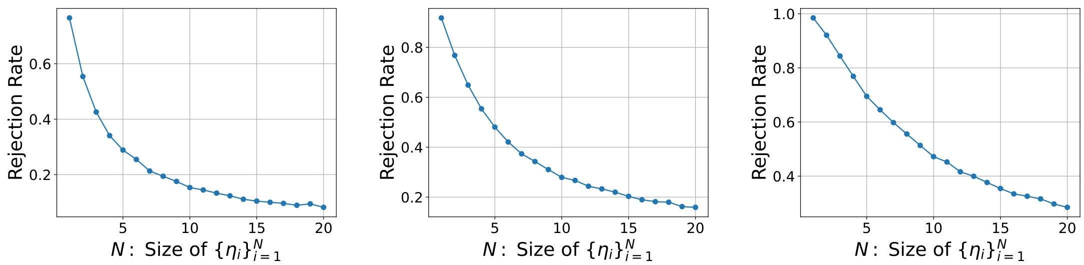
$$\mathbb{E}_{a\sim\pi(\cdot|x)}\Big[-r(x,a) - \eta\log\frac{\pi_0(a|x)}{\pi(a|x)}\Big] = \eta D_{\mathrm{KL}}\Big(\pi(a|x)\Big\|\pi_0(a|x)\exp\Big(\frac{1}{\eta}r(x,a)\Big)\Big),$$

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.

J Experiment Details

In this section, we verify the effectiveness of the Algorithm 5 and Algorithm 3 by real-world RLHF experiments.

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 9 and Table 10, 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.

J.1 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 HH-RLHF dataset (Bai et al., 2022a) (see Table 4 for a sample example). We preprocess the dataset to get 103K training set and 5K test set (details as follows). We also use a subset of the UltraFeedback (Cui et al., 2023), consisting of 5K prompts, as another out-of-distribution (OOD) test set. Meanwhile, the UltraRM-13B (Cui et al., 2023) is used as the ground truth r^* (the gold reward), which is trained on a mixture of UltraFeedback, 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 $\eta = 0.1$.

Dataset preprocessing. We use the HH-RLHF dataset (Bai et al., 2022a) in our experiments, where each sample of the dartaset consists of a prompt x (chat history between the Human and Assistant), and a chosen response a_c and a rejected response a_r. We provide an example in Table 4 for readers' reference. We delete the noisy samples (e.g., with the same chosen and rejected responses), and prompts longer than 400 tokens, and eventually get 108K prompts, which are divided into 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.

Offline Data $\mathcal{D}_{\text{off}}$ 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[1]. 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 $\mathcal{D}_{\text{off}}$, and cannot further query human feedback in the training though it is possible to use the model itself to generate more responses. For hybrid learning, we start with a subset of $\mathcal{D}_{\text{off}}$, with 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 iterate for three steps.

Method, Competitor and Evaluation. In our experiments, we investigate (1) Offline GSHF; (2) Hybrid GSHF; and use (3) SFT on the preferred samples, (4) DPO (Rafailov et al., 2023), (5) RSO (Liu et al., 2023a) as the baselines. The GSHF is implemented by DPO to approximate Oracle 1. The representative models of different RLHF methods will be measured by the gold reward of UltraRM-13B and the KL divergence $\mathbb{E}_{x\sim d_0} D_{\text{KL}}(\pi(\cdot|x)\|\pi_0(\cdot|x))$, which are both evaluated on the split test set.

[1]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_c**:** 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_r: Have you considered making an effort to create more harmonious interactions?

Table 4: An example of HH-RLHF dataset: A prompt-response pair with both chosen and rejected samples.

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 3, where the model aligned with DPO is taken as the baseline. The test hyper-parameter is provided in Table 9. For GPT4 evaluation, we use the GPT-4-turbo 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

Models	Settings	Gold Reward	OOD Gold Reward	Difference $\Delta \downarrow$	OOD Gold Win Rate	OOD GPT4 Eval
SFT	Offline	0.27	-0.21	0.48	-	-
DPO	Offline	2.15	1.71	0.44	0.5	0.5
RSO	Offline	2.25	1.89	0.36	0.55	0.52
Offline GSHF	Offline	2.59	2.41	**0.18**	0.64	**0.60**
Hybrid GSHF	Hybrid	**2.67**	**2.46**	0.21	**0.66**	0.59

Table 5: The evaluation results of the models from different RLHF algorithms. 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 3.

> 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 and Table 5 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.

J.2 Main Results

We present the main results in this subsection. We report the gold rewards and the GPT4 evaluations compared to the DPO baseline in Table 1 and Table 5. We report the detailed results of GPT4 evaluations in Table 3. As we can see, DPO, RSO, and GSHF significantly outperform the SFT baseline, and the GSHF algorithms further outperform the stronger baselines including both DPO and RSO in terms of gold reward, and GPT4 evaluations. In particular, the GSHF algorithms tend to be more robust in the face of OOD data, as they achieve a much smaller Δ compared to other RLHF algorithms.

In addition to the theoretical result provided in this paper, we may also intuitively justify the improvements achieved by the GSHF algorithm (as well as RSO) compared to DPO by noting that they use different data sources for the preference learning thus providing a better coverage of the state-action space. We shared some thoughts with more details between the coverage condition and the success of preference learning in Appendix G.

Reward-KL Trade-off. Since all the considered RLHF algorithms (except SFT) share the same KL-constraint reward optimization target in Equation (2), we first investigate the trade-off between the gold reward and the KL divergence achieved by the different RLHF algorithms and plot the curve in Figure 4. As we can see, both the Offline GSHF and the Hybrid GSHF significantly outperform the strong baselines DPO, and RSO by achieving a much higher reward, for a fixed KL level.

Performance Comparison Under Distribution Shift. We investigate the performance of the resulting models from different alignment algorithms under distribution shift. To this end, we sample a subset of the UltraFeedback (Cui et al., 2023), consisting of 5K prompts, as our out-of-distribution (OOD) test set. The performance results of representative models are detailed in Table 5, and the trade-off between reward and KL divergence on this OOD test set is illustrated in Figure 5. It is observed that all models exhibit a decline in performance compared to the in-domain scenario. In comparison, the Hybrid GSHF and Offline GSHF are more stable in the face of the distribution shift because they achieve a smaller Δ, which is the difference between in-domain and OOD rewards.

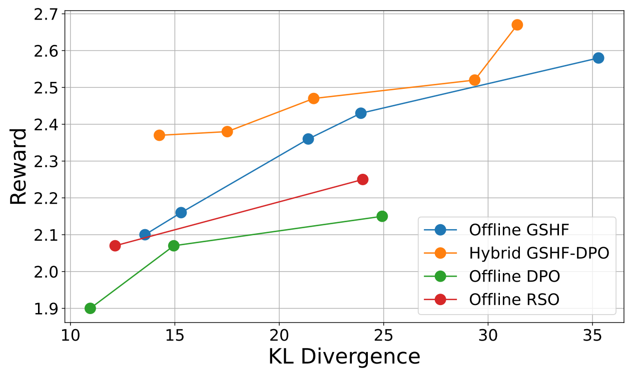


Figure 4: The figure of Reward-KL trade-off. Both the KL and reward are tested on the hand-out test set. The rightest point is the highest gold reward that can be achieved by the RLHF algorithm.

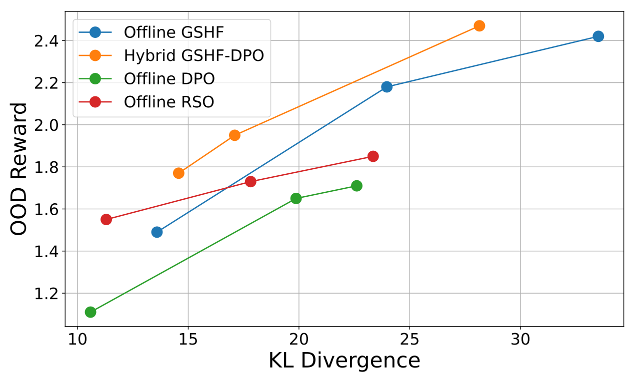


Figure 5: The figure of Reward-KL trade-off on the OOD prompt set from Ultra-Feedback. The rightest point is the highest OOD gold reward that can be achieved by the RLHF algorithm.

Regarding the reward-KL trade-off, consistent with in-domain results, the GSHF algorithms outperform the baseline DPO and RSO models in producing a more efficient frontier. In particular, the Hybrid GSHF achieves the best performance, indicating the advantage of online exploration compared to the offline learning.

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 6. 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 Offline 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 Offline GSHF and Hybrid GSHF models consistently surpass the baseline DPO and RSO models across various sampling temperatures. Notably, Hybrid GSHF shows more advantages over the Offline GSHF 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 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 Offline GSHF are 241, 275, and 240, respectively. Though there is a trend towards longer responses in later iterations of the Hybrid GSHF model, we notice that the final output length of the Hybrid GSHF 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 validation process to ensure the selection of the most effective model during the training process.

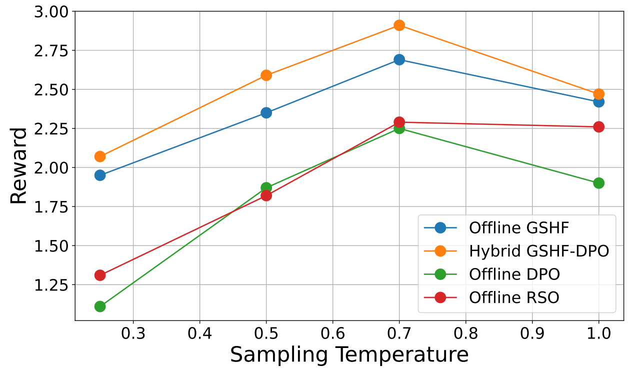


Figure 6: The gold reward with respect to the sampling temperature. The gold reward is tested on the hand-out test set.

J.3 Examples

We provide sample outputs of the models from different RLHF algorithms in Table 6, 7 and Table 8.

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.
Offline GSHF: 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 6: 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.
Offline GSHF: 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 7: Randomly sampled examples of in-domain prompt from HH-RLHF dataset.

Ultra-Feedback 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
Offline GSHF: 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 8: Randomly sampled example of OOD prompt from Ultra-Feedback.

Table 9: Hyper-parameters for fine-tuning Open-LLaMA-3B-V2. SFT-RLHF means that we fine-tune 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
	RS COEFFICIENT	0.5
	RS CANDIDATES AND ACCEPTED SAMPLES	$\{\mathbf{8 - 2}, 24 - 2, 24 - 6\}$
OFFLINE GSHF	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)	$0.5, 0.3, 0.1$
	RS COEFFICIENT	3
	RS 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

Table 10: 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