Cost-Minimized Label-Flipping Poisoning Attack to LLM Alignment

Shigeki Kusaka[*1], **Keita Saito**[*1], **Mikoto Kudo**[*1,2], **Takumi Tanabe**[3], **Akifumi Wachi**[3], **Youhei Akimoto**[1,2,4]

[1]University of Tsukuba,
[2]RIKEN AIP,
[3]LY Corporation,
[4]Institute of Science Tokyo
{shigeki.kusaka, keita.saito, mikoto}@bbo.cs.tsukuba.ac.jp, {takumi.tanabe, akifumi.wachi}@lycorp.co.jp, akimoto@cs.tsukuba.ac.jp

Abstract

Large language models (LLMs) are increasingly deployed in real-world systems, making it critical to understand their vulnerabilities. While data poisoning attacks during RLHF/DPO alignment have been studied empirically, their theoretical foundations remain unclear. We investigate the minimum-cost poisoning attack required to steer an LLM's policy toward an attacker's target by flipping preference labels during RLHF/DPO, without altering the compared outputs. We formulate this as a convex optimization problem with linear constraints, deriving lower and upper bounds on the minimum attack cost. As a byproduct of this theoretical analysis, we show that any existing label-flipping attack can be post-processed via our proposed method to reduce the number of label flips required while preserving the intended poisoning effect. Empirical results demonstrate that this cost-minimization post-processing can significantly reduce poisoning costs over baselines, particularly when the reward model's feature dimension is small relative to the dataset size. These findings highlight fundamental vulnerabilities in RLHF/DPO pipelines and provide tools to evaluate their robustness against low-cost poisoning attacks.

Extended version — https://arxiv.org/abs/2511.09105

Introduction

Vulnerability of LLM As large language models (LLMs) are increasingly deployed in real-world applications, understanding their vulnerabilities is essential for ensuring their effective and safe use. Adversarial attacks expose these vulnerabilities, supporting red-teaming efforts (Shayegani et al. 2023a). Representative adversarial attacks at inference time include jail-breaking (Wei, Haghtalab, and Steinhardt 2023; Chao et al. 2023; Mehrotra et al. 2024; Zou et al. 2023) and prompt injection (Greshake et al. 2023; Liu et al. 2024), where attackers craft malicious inputs to elicit unintended outputs from LLMs. At training time, *data poisoning attacks* modify the training dataset to induce undesired behaviors or embed backdoor triggers in the resulting LLM. In this paper, we focus on the data poisoning attack on LLMs.

[*]These authors contributed equally.

Poisoning attacks on preference alignment LLMs are susceptible to data poisoning attacks due to their multi-stage training pipeline, which typically includes pre-training, supervised fine-tuning (SFT), and alignment via reinforcement learning from human feedback (RLHF) (Ouyang et al. 2022) or direct preference optimization (DPO) (Rafailov et al. 2023). Recent empirical studies have demonstrated that LLMs can be compromised through poisoning during both the SFT phase (Wan et al. 2023; Shu et al. 2023) and the RLHF/DPO phase (Wu et al. 2025; Wang et al. 2024; Pathmanathan et al. 2024a; Baumgärtner et al. 2024; Pathmanathan et al. 2024b; Tramèr and Rando Ramirez 2024), raising concerns about their robustness to adversarial manipulation. However, the theoretical foundations of poisoning attacks in the RLHF/DPO phase remain largely unexplored, leaving open questions about the fundamental vulnerabilities of these methods and potential defenses. A theoretical understanding is crucial to ascertain the worst-case scenarios for victims, which empirical studies cannot fully reveal.

Objective We theoretically investigate the minimum cost of the attack to successfully steer the optimal LLM policy toward the attacker's target policy during the RLHF/DPO phase. We consider that an attacker who operates as an annotator, tasked with evaluating two outputs y and z in a given context x, and providing a binary preference label ($w = 1$ if y is preferred; otherwise $w = -1$). While the attacker can not modify (x, y, z), they can arbitrarily set the preference label w. Our goal is to determine the minimum number of label flips from the benign labels required to induce the attacker's desired behavior and to design a method for constructing a malicious dataset that achieves this objective with minimal cost. By quantifying these costs, our analysis is expected to guide the design of robust RLHF/DPO pipelines that can detect or mitigate such low-cost poisoning attacks, ensuring the safe deployment of LLMs in real-world settings.

Contributions In this work, we provide the first theoretical analysis of the minimal cost required to steer LLM policies via label-flipping attacks during RLHF/DPO alignment. By formulating the problem as a convex (or linear) optimization, we derive tight lower and upper bounds on the minimum number of label flips needed to induce a target policy. As

a byproduct of this analysis, we develop a post-processing method that can be applied to any existing label-flipping attack to reduce its cost while preserving its intended poisoning effect. This approach is particularly effective in practical LLM alignment pipelines, where the dataset size is significantly greater than the feature dimension of the reward model, enabling attackers to exploit redundancy in the data in the feature space to minimize the cost of targeted poisoning.

Preliminaries

LLM alignment is often conducted in two stages: supervised fine-tuning (SFT) and learning from human feedback. Given an LLM pre-trained with a large corpus in an unsupervised manner, it is fine-tuned using human-annotated input-output pairs (x, y) to produce more relevant output for specific downstream tasks of interest. Learning from human feedback then aims to further align LLMs with human preferences. In this step, the LLM is trained using a dataset of preferences, $\mathcal{D}_L = \{(x, y, z, w)\}$, where x is the input, y and z are two candidate outputs, and $w \in \{-1, 1\}$ is the preference label indicating whether y is preferred to z ($w = 1$) or not ($w = -1$). The LLM is trained to assign higher probabilities to preferred outputs. Reinforcement learning from human feedback (RLHF) is often employed for this purpose.

RLHF first trains a reward model $r(x, y)$ from the preference dataset. The human preference is typically modeled as the Bradley-Terry model:

$$\Pr[w = 1 \mid x, y, z] = \sigma(r(x, y) - r(x, z)), \quad (1)$$

where $\sigma(t) = \frac{1}{1+\exp(-t)}$ is the sigmoid function. The reward model is trained via maximum likelihood estimation by minimizing

$$\mathcal{L}(r) = -\sum_{(x,y,z,w)\in\mathcal{D}_L} \log \sigma(w(r(x, y) - r(x, z))). \quad (2)$$

Let $\widehat{r}$ denote the obtained reward model. The LM policy π is then trained to maximize the obtained reward under the KL-regularization:

$$\mathbb{E}_{x\sim\rho}[\mathbb{E}_{y\sim\pi(y|x)}[\widehat{r}(x, y)] - \tau D_{\mathrm{KL}}(\pi \parallel \pi_{\mathrm{ref}})], \quad (3)$$

where ρ is a distribution over the context; π_{ref} is the reference policy, typically the SFT policy used to initialize RLHF; and τ is a parameter controlling the deviation from π_{ref}.

Direct Preference Optimization (DPO) is an alternative to RLHF that directly optimizes the LM policy from the preference dataset. The optimal policy that maximizes (3) is:

$$\pi_r(y \mid x) = \frac{1}{Z_{r,\pi_{\mathrm{ref}}}(x)}\pi_{\mathrm{ref}}(y \mid x)\exp(\tau^{-1}r(x, y)), \quad (4)$$

$$\text{where } Z_{r,\pi_{\mathrm{ref}}}(x) = \sum_y \pi_{\mathrm{ref}}(y \mid x)\exp(\tau^{-1}r(x, y)). \quad (5)$$

Therefore, an LM policy π can be viewed as the optimal policy under the reward function:

$$r(x, y) = \tau \log \frac{\pi(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} + \tau \log Z_{r,\pi_{\mathrm{ref}}}(x). \quad (6)$$

By substituting this expression into (2), one obtains the corresponding DPO objective. It is known that the optimal policies for RLHF and DPO coincide (Gheshlaghi Azar et al. 2024), and several variants of DPO have been proposed in the literature (Gheshlaghi Azar et al. 2024; Swamy et al. 2024; Ethayarajh et al. 2024).

Related Works

Most prior work on data poisoning attacks in machine learning focuses on supervised learning settings, particularly regression or classification tasks (Shayegani et al. 2023b). Typical poisoning objectives include degrading model performance or injecting backdoor triggers. In these settings, attackers can add malicious data to the original dataset, training the victim model on the combined dataset to induce the desired malicious behavior. In contrast, in our setting, the attacker can only flip preference labels in the dataset, making the attack surface more constrained.

Within the LLM alignment pipeline, poisoning attacks have been studied in both the SFT phase (Wan et al. 2023; Shu et al. 2023) and the RLHF phase (Wu et al. 2025; Wang et al. 2024; Pathmanathan et al. 2024a; Baumgärtner et al. 2024; Pathmanathan et al. 2024b; Tramèr and Rando Ramirez 2024). In the SFT phase, human annotators are expected to produce high-quality outputs y for given contexts x, creating a natural risk of maliciously crafted pairs $(x, \tilde{y})$. In the RLHF phase, annotators evaluate candidate outputs y, z in a given context x and provide preference labels. While a strong adversary may replace or inject malicious triplets (x, y, z) into the preference dataset (Baumgärtner et al. 2024; Pathmanathan et al. 2024b; Tramèr and Rando Ramirez 2024), arguably the most realistic scenario involves a malicious annotator who can only manipulate the preference label w while the triplet itself remains unchanged (Wu et al. 2025; Wang et al. 2024; Pathmanathan et al. 2024a). This setting is analogous to label-flipping attacks (Xiao, Xiao, and Eckert 2012), but while traditional label-flipping attacks focus on classification, our context involves reward model learning from paired preference data, introducing a distinct problem structure.

The most relevant prior works are Wu et al. (2025) and Wang et al. (2024), who consider the same setting and develop attack algorithms under both white-box and black-box scenarios. Their empirical investigations reveal that RLHF alignment is vulnerable to label-flipping attacks and that existing defense strategies provide limited protection. While these studies empirically demonstrate the vulnerability of RLHF pipelines, our work is the first, to the best of our knowledge, to provide a theoretical analysis of the minimal cost of label-flipping required to guide the reward model toward an attacker-specified target. Moreover, we introduce a convex (or linear) programming framework that not only characterizes these costs but also enables practical reduction of attack costs in existing poisoning methods. Specifically, given a candidate malicious dataset, our framework can propose an alternative dataset that induces the same reward function while requiring fewer label flips, highlighting a fundamental and previously unquantified vulnerability in RLHF/DPO pipelines.

Threat Model

Victim The victim has access to a labeled dataset constructed from an unlabeled dataset $\mathcal{D}_U = \{(x_i, y_i, z_i)\}_{i=1}^N$, where $x_i \in \mathcal{X}$ is a context, and $y_i, z_i \in \mathcal{Y}$ are two candidate outputs. Each triplet is labeled by external annotators, with the label $w_i = 1$ if y_i is preferred to z_i in context x_i, and $w_i = -1$ otherwise. We denote by $\eta(x, y, z)$ the probability that the label for (x, y, z) is $w = 1$. The set of labeled data (x, y, z, w) forms the labeled dataset $\mathcal{D}_L$. Without loss of generality, we assume anti-symmetry of η, i.e., $\eta(x, y, z) = 1 - \eta(x, z, y)$. In case that multiple ($m \geqslant 1$) annotations are provided for each triplet, the size of the labeled dataset becomes $m \cdot N$.

The victim aims to optimize a policy π under the labeled dataset $\mathcal{D}_L$ using RLHF. First, a reward model r is trained by minimizing the empirical loss (2). Once the reward model is trained, the LM policy is optimized to maximize the expected reward under KL regularization as in (3). For technical completeness, we assume $\pi_{\text{ref}}(y \mid x) > 0$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. The optimal policy under r is thus given by (4).

Following Ouyang et al. (2022), we model the reward function using a pre-trained baseline LLM (typically a fine-tuned model) from which the final unembedding layer is removed to obtain an embedding $\phi : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$. A linear output layer is then added, resulting in the reward model $r(x, y) = \boldsymbol{r}^\top \phi(x, y)$, where $\boldsymbol{r} \in \mathbb{R}^n$ is referred to as the *reward vector*. During the reward training phase, we consider two settings: (1) only the reward vector $\boldsymbol{r}$ is trained while the embedding ϕ is fixed, and (2) both the reward vector $\boldsymbol{r}$ and the embedding ϕ are trained.

Attacker The attacker's objective is to guide the victim's optimal policy toward a target policy π_A with minimal cost. Due to the structure of the optimal policy (4), only policies that are optimal under reward functions representable by the victim's reward model are valid. Thus, the attack reduces to steering the reward model to a target reward function r_A at minimum cost. The target reward function may be handcrafted or obtained using attack methods such as Wu et al. (2025); Wang et al. (2024). In the latter case, the objective becomes minimizing the cost required to achieve the same poisoning effect. We assume the attacker has access to the labeled dataset $\mathcal{D}_L$. The attacker can flip labels w_i to $w_i^A \in \{-1, 1\}$ at a cost defined later. However, the attacker cannot modify the input x or candidate outputs y, z in the dataset.

Discrepancy Between Theory and Practice For theoretical analysis in the following section, we idealize both the attacker's capabilities and the victim's training process. We assume the attacker can directly modify the annotation probability η rather than flipping individual labels w (i.e., the attacker's action is not binary $\{-1, 1\}$, but continuous $[0, 1]$), and that the victim minimizes the expected loss under η (i.e., population version) rather than the empirical loss (2). In practice, due to the finite dataset, realizable annotation probabilities are discrete, being multiples of the reciprocal of the count m of each (x, y, z) in the dataset (i.e., m is the number of annotations per datum). These idealizations enable us to derive theoretical guarantees, while we acknowledge the risk of deviation from practical scenarios, evaluated empirically.

Minimum Cost Attack

The objective of this study is to identify the cost of the attacker to realize the target policy π_{r_A} by label flipping. Arguably, the most natural choice for the cost of the attacker is the amount of flipped labels, formulated as

$$\text{(practical)} \quad \sum_{(x_i, y_i, z_i, w_i) \in \mathcal{D}_L} |w_i - w_i^A| \tag{7}$$

$$\text{or (ideal)} \quad \sum_{(x_i, y_i, z_i) \in \mathcal{D}_U} |\eta(x_i, y_i, z_i) - \eta_A(x_i, y_i, z_i)|, \tag{8}$$

where w_i^A is the label after the attack and $\eta_A(x_i, y_i, z_i)$ is the probability of w_i^A being 1.

More generally, the cost can be measured by using a norm $\|\cdot\|$. From now on, we focus on the ideal situation where the labels w_i follow the probabilities $\eta(x_i, y_i, z_i)$ and the loss function is defined by the expectation of $\mathcal{L}(r)$ with respect to w_i, denoted as $\mathcal{L}(r; \eta)$. The attack target r_A may be given explicitly, or derived by some other poisoning attack. Let θ_O be a vector of dimension N whose elements are $\eta(x_i, y_i, z_i)$ for $(x_i, y_i, z_i, w_i) \in \mathcal{D}_L$ and θ_A a vector of dimension N whose elements are $\eta_A(x_i, y_i, z_i)$. The attack cost measured by $\|\cdot\|$ is defined as $\|\theta_A - \theta_O\|$, which recovers (8) if the norm is the ℓ_1-norm.

The attacker tries to minimize the cost while realizing the desired policy π_{r_A}. Because of the form of the optimal policy (4), different reward functions can lead to the same optimal policy. For example, $r(x, y)$ and $r(x, y) + R(x)$ for any $R : \mathcal{X} \to \mathbb{R}$ admit the same optimal policy. Let $\mathcal{R}(\pi) = \{r : \pi_r = \pi\}$ be the set of reward functions for which the optimal policy is π. The attacker's cost minimization problem is formulated with a vector representation θ of η as follows:

$$\min_{\theta} \quad \|\theta - \theta_O\|, \tag{9a}$$

$$\text{s.t.} \quad \operatorname*{argmin}_r \mathcal{L}(r; \theta) \subseteq \mathcal{R}(\pi_{r_A}), \tag{9b}$$

$$\|2\theta - \mathbf{1}\|_\infty \leqslant 1, \tag{9c}$$

where, with abuse of notation, $\mathcal{L}(r; \theta)$ means $\mathcal{L}(r; \eta)$, and (9c) is introduced to ensure $\eta \in [0, 1]$. In the following, we investigate the minimum cost of the poisoning attack theoretically. In particular, we are interested in the lower and upper bounds of the minimum cost to realize the target reward function.

Fixed Embedding

First, we consider the situation where the embedding ϕ is fixed during the training. The proofs for the theoretical results in this section can be found in the extended version.

The optimization problem (9) can be infeasible in cases that the loss function $\mathcal{L}$ admits multiple minimum solutions that lead to different optimal policies. However, we can derive that the optimal reward function for the loss (2) is uniquely determined, showing that (9) is valid, under a mild assumption. Hereafter, we consider the fixed embedding situation. Let Φ be a matrix of dimension $n \times N$ whose i-th column is $\phi(x_i, y_i) - \phi(x_i, z_i)$. Its Moore-Penrose pseudo-inverse is denoted as $\Phi^\dagger$. The row and column spaces of Φ are denoted as $\mathrm{row}(\Phi)$ and $\mathrm{col}(\Phi)$, respectively.

To proceed theoretical analysis, we assume the following, which is naturally satisfied if $n < N$.[1] Intuitively, it implies that if two reward functions agree on all data points in $\mathcal{D}_U$, they must agree everywhere up to a context-dependent offset $R(x)$, meaning that the reward function is fully determined by its values on the dataset.

Assumption 1. For any $(x, y, z) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$, $\phi(x, y) - \phi(x, z) \subseteq \mathrm{col}(\Phi)$.

The optimization problem (9) can be transformed as a convex optimization problem.

Theorem 1. *Suppose that Assumption 1 holds. Let $\zeta = \theta - \theta_O$. Then, the minimum cost poisoning attack problem* (9) *is equivalently formulated as a convex optimization problem with linear equality and inequality conditions:*

$$\min_{\zeta} \|\zeta\| \quad \text{s.t.} \quad \Phi\zeta = \Phi(\theta_A - \theta_O), \tag{10a}$$

$$-\theta_O \leqslant \zeta \leqslant (\mathbf{1} - \theta_O), \tag{10b}$$

where $\leqslant$ denotes element-wise comparison, as used hereafter.

That is, by solving the convex optimization problem (10) and letting ζ^* be its optimal solution, one can obtain the preference probability $\theta_A^* = \theta_O + \zeta^*$ that leads to the same target reward function in $\mathcal{R}(\pi_{r_A})$ as θ_A with a reduced or equal cost $\|\theta_A^* - \theta_O\| \leqslant \|\theta_A - \theta_O\|$. It is irrelevant to the way of crafting θ_A.

This optimization problem has a convex objective function and linear constraints. Therefore, one can obtain a solution to this problem, ζ^*, by using a standard convex optimization solver. In case of the ℓ_1 attack cost, this can be reformulated as a linear programming problem by decomposing ζ as $\zeta = \zeta_+ - \zeta_-$ where $\zeta_+, \zeta_- \in \mathbb{R}_+^N$ and rewriting $\|\zeta\|$ as $\|\zeta\| = \mathbf{1}_N^{\mathrm{T}}(\zeta_+ + \zeta_-)$. Therefore, one can employ a linear programming solver to obtain the optimum solution. See the extended version for details.

By considering the Lagrangian dual problem of the primal problem (10), we can derive the minimum attack cost bounds. It is derived as follows. The primal problem is a convex optimization problem with linear equality and inequality constraints. Therefore, if a relaxed Slater condition is satisfied, i.e., a feasible solution exists, then the strong duality holds and the solution to the dual problem provides the minimum value of the primal problem. In our case, $\zeta = \theta_A - \theta_O$ is a feasible solution. Hence, the strong duality holds and the maximum value of the dual problem is the minimum value of the primal problem.

Based on the above argument, a lower bound and an upper bound of the minimum cost are derived.

Theorem 2. *The minimum cost of* (10) *is lower bounded by*

$$\frac{\|(\Phi^\dagger\Phi)(\theta_A - \theta_O)\|_2^2}{\|(\Phi^\dagger\Phi)(\theta_A - \theta_O)\|_*}. \tag{11}$$

[1] It holds when Φ is of full row rank, which occurs with probability 1 for random Φ. We confirmed that all LLM feature matrices used in the experiments satisfied this condition.

Theorem 3. *Let $\theta^* = \theta_O + (\Phi^\dagger\Phi)(\theta_A - \theta_O)$. Let $\alpha^* = \max\{\|\theta^* - 0.5 \cdot \mathbf{1}\|_\infty - 0.5, 0\}$ and $\bar{\alpha} = 0.5 - \|\theta_A - 0.5\|$. The minimum cost of* (10) *is upper bounded by*

$$\min\left\{\left\|\left(\frac{\alpha^* I + \bar{\alpha}\Phi^\dagger\Phi}{\alpha^* + \bar{\alpha}}\right)(\theta_A - \theta_O)\right\|,\ \|\theta_A - \theta_O\|\right\}. \tag{12}$$

Remark 1. The matrix $\Phi^\dagger\Phi$ defines the orthogonal projection from $\mathbb{R}^N$ to a subspace spanned by the rows of Φ, whose rank is at most n. If the cost is defined by the ℓ_2 norm, we always have $\|(\Phi^\dagger\Phi)(\theta_A - \theta_O)\|_2 \leqslant \|\theta_A - \theta_O\|_2$. The discrepancy between $\frac{\|(\Phi^\dagger\Phi)(\theta_A - \theta_O)\|_2^2}{\|(\Phi^\dagger\Phi)(\theta_A - \theta_O)\|_*}$ in (11) and $\|(\Phi^\dagger\Phi)(\theta_A - \theta_O)\|$ in (12) comes from the primal-dual norm relation, i.e., $\|\zeta\|_2^2 \leqslant \|\zeta\|\|\zeta\|_*$. In this sense, these bounds are tight because the equality holds for some ζ.

These theorems indicate that the cost of a poisoning attack can be reduced significantly. From the defense perspective, it suggests that the victim must be prepared for the attack to guide the reward model arbitrarily in

$$\Theta_k^A = \left\{\theta \mid \frac{\|(\Phi^\dagger\Phi)(\theta - \theta_O)\|_2^2}{\|(\Phi^\dagger\Phi)(\theta - \theta_O)\|_*} \leqslant k\right\} \tag{13}$$

if k data points are annotated by an untrusted annotator. It can be significantly wider than the set corresponding to the naive attack with cost no greater than k, i.e.,

$$\widetilde{\Theta}_k^A = \{\theta \mid \|\theta - \theta_O\| \leqslant k\}, \tag{14}$$

in particular, when the rank of $\Phi^\dagger\Phi$ is significantly smaller than its dimension, which corresponds to the situation that the number n of features is significantly smaller than the number N of data points. It helps us assess the security risk of allowing untrusted individuals to annotate data.

We investigate the influence of the choice of the embedding ϕ (i.e., Φ) on the minimum attack cost. Proposition 4 states that, if the feature extractor is fixed, then the more capable the feature extractor's representation power is, the more cost the attacker needs to spend to realize the same target reward. It suggests that a greater number n of features results in models that are more robust against label flipping attacks.

Proposition 4. *Suppose that $\phi_1 : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^{n_1}$ and $\phi_2 : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^{n_2}$ satisfies Assumption 1 and $\mathrm{row}(\Phi_1) \subseteq \mathrm{row}(\Phi_2)$. If the target reward function is $r_A = \boldsymbol{r}_{A,1}^{\mathrm{T}}\phi_1 = \boldsymbol{r}_{A,2}^{\mathrm{T}}\phi_2$, then a feasible solution ζ_2 to the problem under ϕ_2 is also feasible under ϕ_1. Therefore, the minimum value of the problem* (10) *under Φ_1 is no greater than that under Φ_2.*

Adaptive Embedding

Now we consider a more general reward model

$$r(x, y) = \langle \boldsymbol{r}, \phi_\omega(x, y)\rangle, \tag{15}$$

where ϕ is an adaptive embedding parameterized by ω. Suppose that the attacker's target reward is expressed as

$$r_A(x, y) = \langle \boldsymbol{r}_A, \phi_{\omega_A}(x, y)\rangle \tag{16}$$

with the reward vector $\boldsymbol{r}_A$ and the parameter ω_A of the embedding. In this situation, however, the optimization problem (9)

is not always well-defined as the reward function minimizing $\mathcal{L}(r;\eta)$ may not be uniquely determined. In such situations, whether the attack succeeds or not depends on which solution the victim's reward model converges to. Therefore, it may depends on the initial value of the victim's reward model and its learning algorithm. To make the optimization problem feasible, we relax the notion of attack success as having a potential to obtain the target reward function. The relaxed optimization problem is formulated as follows:

$$\min_{\eta} \quad \|\theta - \theta_O\|, \tag{17a}$$

$$\text{s.t.} \quad r_A \in \operatorname{argmin} \mathcal{L}(r;\eta), \tag{17b}$$

$$\mathbf{0} \leqslant \eta(x,y,z) \leqslant \mathbf{1}, \forall x,y,z. \tag{17c}$$

Considering a lower bound of the minimum cost of such a relaxed problem provides the guarantee from the victim's perspective that the attack cannot be successful without paying a derived cost.

Similarly to Theorem 1, we can transform the attacker's optimization problem (17) as follows. First, we realize that (17b) holds if and only if a reward model $r = \boldsymbol{r}^{\mathrm{T}}\phi_{\bar{\omega}}$ satisfying $\boldsymbol{r}^{\mathrm{T}}\Phi_{\bar{\omega}} = \boldsymbol{r}_A^{\mathrm{T}}\Phi_{\omega_A}$ is included in $\operatorname{argmin}\mathcal{L}(r;\eta)$. For these reward functions, we have $\theta_{r_A} = \theta_r$. Choose one such $\bar{\omega}$. Then, analogously to the derivation of Theorem 1, the attacker's optimization problem reduces to

$$\min_{\zeta} \|\zeta\| \quad \text{s.t.} \quad \Phi_{\bar{\omega}}\zeta = \Phi_{\bar{\omega}}(\theta_A - \theta_O), \tag{18a}$$

$$\bar{\omega} \in \{\omega : \exists \bar{\boldsymbol{r}} \text{ s.t. } \bar{\boldsymbol{r}}^{\mathrm{T}}\phi_{\bar{\omega}} = \boldsymbol{r}_A^{\mathrm{T}}\phi_{\omega_A}\}, \tag{18b}$$

$$-\theta_O \leqslant \zeta \leqslant (\mathbf{1} - \theta_O). \tag{18c}$$

The point here is that the attacker does not need to set $\bar{\omega} = \omega_A$ and may choose $\bar{\omega}$ such that the attack cost is the smallest. The following result is a straightforward consequence.

Proposition 5. *The minimum cost of* (18) *is upper bounded by the minimum cost of* (10) *where* Φ *is replaced with* Φ_{ω_A}.

Proposition 5 indicates that we can reduce the cost of the attack to realize the target reward r_A by solving convex (or linear) programming (10) with $\Phi = \Phi_{\omega_A}$. However, we emphasize that, differently from the case of the fixed embedding, it does not provide the minimum cost attack due to the degrees of freedom in (18b), and it provides the solution to a "relaxed" optimization problem (17). Therefore, guarantees from the perspective of attackers are hard to obtain.

Now we consider the worst situation for the victim. In light of Proposition 4, the minimum attack cost is no greater for $\bar{\omega}$ than for ω_A if $\mathrm{row}(\Phi_{\bar{\omega}}) \subseteq \mathrm{row}(\Phi_{\omega_A})$. Suppose that the representational capacity of ϕ_ω is high enough that there exists $\bar{\omega}$ such that $\mathrm{col}(\Phi_{\bar{\omega}}) = \{\boldsymbol{r}_A^{\mathrm{T}}\Phi_{\omega_A}\}$. By assuming the existence of such $\bar{\omega}$, the attacker's optimization problem reads

$$\min_{\zeta} \|\zeta\| \quad \text{s.t.} \quad \boldsymbol{r}_A^{\mathrm{T}}\Phi_{\omega_A}\zeta = \boldsymbol{r}_A^{\mathrm{T}}\Phi_{\omega_A}(\theta_A - \theta_O), \tag{19a}$$

$$-\theta_O \leqslant \zeta \leqslant (\mathbf{1} - \theta_O), \tag{19b}$$

where we used the fact that $\Phi_{\bar{\omega}}\zeta = \Phi_{\bar{\omega}}(\theta_A - \theta_O)$ is equivalent to $\boldsymbol{r}_A^{\mathrm{T}}\Phi_{\omega_A}\zeta = \boldsymbol{r}_A^{\mathrm{T}}\Phi_{\omega_A}(\theta_A - \theta_O)$. This problem can be solved with a convex programming or a linear programming solver as for (10) but possibly with reduced cost.

Theorem 6. *Suppose that the attacker's target reward function is expressed as* (16). *Then, the minimum cost of* (18) *is lower bounded by that of* (19). *Moreover, if there exists* $\bar{\omega}$ *such that* $\mathrm{col}(\Phi_{\bar{\omega}}) = \{\boldsymbol{r}_A^{\mathrm{T}}\Phi_{\omega_A}\}$, *the minimum cost of* (18) *is equal to that of* (19).

The upper and lower bounds for the cost of (19) are derived in Theorem 2 and Theorem 3, respectively. The attacker need not $\bar{\omega}$ explicitly; solving (19) requires only the target reward function r_A. However, in reality, considering the minimum cost of (19) as the minimum cost for the attack θ_A may be too conservative from the defense perspective. Such an attack will not realize the target reward function in practice due to the solution multiplicity, suboptimal optimization, and the fact that a $\bar{\omega}$ satisfying $\mathrm{col}(\Phi_{\bar{\omega}}) = \{\boldsymbol{r}_A^{\mathrm{T}}\Phi_{\omega_A}\}$ does not necessarily exist in general.

Practical Post-Processing Method

Based on the above theoretical analysis, we propose a practical post-processing method to minimize the cost of any existing label-flipping attack. Given a target preference probability vector θ_A, either hand-crafted or generated by an existing attack, we solve the convex optimization problem in (10) to obtain a cost-minimized vector θ_A^* that induces the same target reward model while requiring fewer label flips. In the case of adaptive embeddings, we use the initial embedding ϕ of the reward model to form the optimization problem (10) as if it were fixed. A discrepancy exists between our theoretical analysis and practical implementation, as the analysis assumes knowledge of the embedding parameter ω_A corresponding to the target reward r_A. However, as demonstrated later, the practical approach remains effective in reducing the cost without deteriorating the attack's performance.

After obtaining θ_A^*, we discretize it by rounding so that the preference vector takes values in

$$\Theta_m = \left\{ \theta \in \mathbb{R}^N : [\theta]_k = \frac{i}{m} \text{ for } i \in \{0,\dots,m\} \right\}, \tag{20}$$

where $[\theta]_k$ indicates the kth element of θ and m is the number of annotations per datum, referred to as the granularity. Finally, we flip the preference labels in the dataset to follow the discretized vector. We refer to this post-processing method as *Poisoning Cost Minimization (PCM)*.

Importantly, PCM is agnostic to how the target θ_A is generated and can be layered onto any label-flipping attack, providing a systematic way to reduce poisoning costs while preserving attack efficacy. We apply this post-processing in our empirical evaluations to demonstrate its effectiveness across synthetic and real LLM alignment datasets.

Numerical Analysis on Synthetic Data

We demonstrate the tightness of the bounds and how small the minimum cost can be compared to the cost of the naive attack. In particular, we show that the minimum cost can be significantly reduced from the naive cost when the number of data points is significantly greater than the number of features. For this purpose, we generate synthetic data and compare the cost of the cost minimized attack and the naive attack.

Dataset We produce a synthetic dataset $\mathcal{D}_U$ with embeddings $\phi(x_i, y_i)$ and $\phi(x_i, z_i) \in \mathbb{R}^n$ for $(x_i, y_i, z_i) \in \mathcal{D}_U$ generated randomly from the standard normal distribution. Without loss of generality, the first response y_i is considered to be the preferred response in the original annotation, i.e., $\theta_O = \mathbf{1}$. The dataset size is $N = |\mathcal{D}_U|$. To simulate the situation where multiple annotators are assigned to provide their preferences for the same tuples, we duplicate each datum m times. That is, θ_O as well as annotation probability after the poisoning attack, θ_A, can take values in Θ_m.

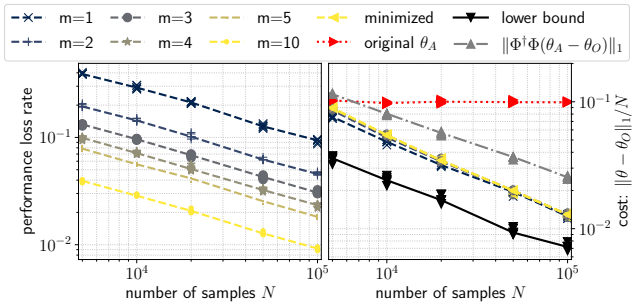
Attack Scenario By nature of this synthetic dataset, we consider the situation where the embedding is fixed. We suppose that we have a target attack preference probability $\theta_A \in \Theta_m$. The attacker tries to minimize the attack cost measured by ℓ_1-norm, $\|\theta - \theta_O\|_1$. We consider two attack targets: 1) θ_A is generated by flipping each element of θ_O with probability 0.1; 2) θ_A is generated by RLHFPoison (Wang et al. 2024) with quality filter parameter $a = 0.25$ and final poisoning ratio $b = 0.1$. RLHFPoison originally generates a dataset to make the LLM output a longer response without significantly changing the other aspects. Here, instead of the output length, the first feature of the output is to be maximized. That is, the reward signals for data with greater first feature values are to be maximized. For each θ_A, we apply the proposed post-processing, PCM, to obtain the cost-minimized preference probability θ_A^*.

Performance Metric We focus on two metrics. The first one is the ℓ_1-cost $\|\theta - \theta_O\|_1$ after the discretization. Due to the discretization, the performance of the attack by PCM may degrade. To measure the performance degradation by PCM, we minimize the loss function (2) for $r = \boldsymbol{r}^{\mathrm{T}}\phi$ with respect to $\boldsymbol{r}$, where the preference labels are given by θ_O, θ_A, and θ_A^* (discretized). Letting the optimal reward functions obtained with the above preference datasets be denoted as r_O, r_A, and r_A^*. Then, we compute the performance loss rate

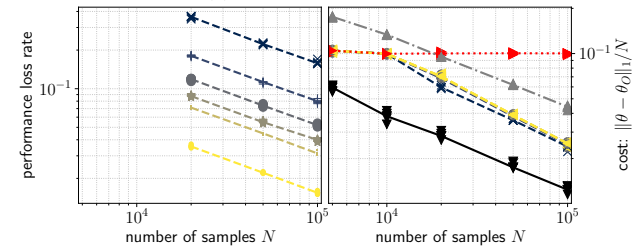
$$\frac{\sum_{i=1}^N |\sigma(r_A^*(x,y) - r_A^*(x,z)) - \sigma(r_A(x,y) - r_A(x,z))|}{\sum_{i=1}^N |\sigma(r_A(x,y) - r_A(x,z)) - \sigma(r_O(x,y) - r_O(x,z))|}. \tag{21}$$

It measures the average preference difference between the trained reward models using the target θ_A and its cost-minimized θ_A^* relative to that between the trained reward models using θ_A and the original θ_O.

Results The results are shown in Figures 1 and 2. Although the algorithms are all deterministic, the datasets are randomly generated. Therefore, we performed 5 trials with different dataset generations. The findings are summarized as follows. (1) Though it is not guaranteed by Theorem 3, $\|(\Phi^\dagger\Phi)(\theta_A - \theta_O)\|_1$ provides a good upper bound of the proposed scheme when it is smaller than the naive cost $\|\theta_A - \theta_O\|_1$. The results all fit between the lower bound provided in Theorem 2 and this value, and the discrepancy between them is around the factor of 3 to 4. (2) There is a higher chance to reduce the attack cost for a larger data set, i.e., greater N if the original attack cost $\|\theta_A - \theta_O\|_1$ is more or less constant (i.e., the flip rate is fixed). It may be understood as that the rank of $\Phi^\dagger\Phi$ ($\leqslant n$) will be smaller than its dimension (N). (3) The



(a) #features $n = 1000$



(b) #features $n = 3000$

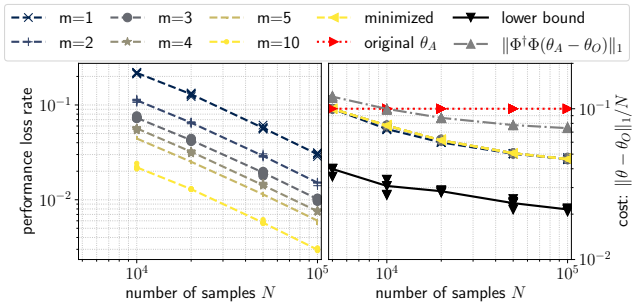
Figure 1: Cost (right) and performance loss rate (21) (left) of the proposed cost minimization, PCM, for random flip attack. Results of 5 trials (points) as well as their median (lines). Minimized: the cost of θ_A^* before discretization, Original: the cost of θ_A, Lower bound: (11), $\|\Phi^\dagger\Phi(\theta_A - \theta_O)\|_1$: a term appearing in the upper bound (12). The other lines are the performance loss rate and the cost of the proposed attack with discretization using different granularity m. Missing data points in the preference loss rate indicate no performance loss because $\theta_A = \theta_A^*$ (no cost reduction as well).

cost does not depend heavily on the granularity m, but a smaller performance loss rate can be achieved with a greater m. It is intuitive because the cost minimization process does not affect the trained reward function if no discretization is performed. Even with $m = 1$ (i.e., each data point is annotated by a single annotator), the performance loss rate can be around 0.1 if N is sufficiently large. (4) The cost reduction effect increases linearly in the dataset size N for the random-flip attack, whereas its scaling is lower for the attack by RLHFPoison. Nevertheless, a similar trend (the cost can be reduced when $N \gtrsim 5n$) is observed.

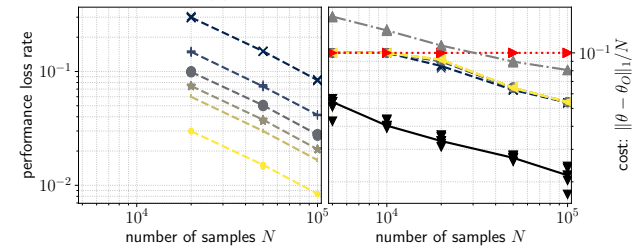
Evaluation on Public LLMs and Open Dataset

We demonstrate the cost reduction achieved by the proposed framework across different LLMs on publicly available datasets. In particular, we show that the proposed method remains effective even when the entire LLM is trained using DPO (adaptive embedding scenario), despite the framework itself being derived under the fixed embedding assumption.

Datasets and Models We employ three datasets of varying size and three models of varying size. The datasets are SOCIAL-REASONING-RLHF ($N = 3,820$) (ProlificAI 2024), PKU-SAFERLHF ($N = 73,907$) (Ji et al. 2024), and HH-



(a) #features $n = 1000$



(b) #features $n = 3000$

Figure 2: Cost (right) and performance loss rate (left) of the proposed cost minimization, PCM, for RLHFPoison attack.

RLHF ($N = 160,800$) (Bai et al. 2022). Since SOCIAL-REASONING-RLHF dataset does not contain a test set, we used 20% of the training data for testing. The models are `Phi-3.5-mini-instruct` ($n = 3072$) (Abdin et al. 2024), `LLaMA-2-7b` ($n = 4096$), and `LLaMA-2-13b` ($n = 5120$) (Touvron et al. 2023).

Attack Scenario The attack target is generated using RLHFPoison, which aims to increase the output length of an LLM without significantly affecting other behavioral characteristics. The quality filter parameter and final poisoning ratio are set to $a = 0.25$ and $b = 0.05$, respectively. Consequently, 5% of the preference labels in the training dataset are flipped. PCM is applied after obtaining θ_A via RLHFPoison.

Performance Metric LLMs are trained by DPO (3 epochs with $\tau = 0.1$ and learning rate 10^{-6}) with the original preference θ_O, the malicious preference θ_A generated by RLHFPoison, and the preference θ_A^* generated by PCM from θ_A. We measure the cost (flip rate) reduction rate of θ_A^* over θ_A (5%), i.e., flip rate of θ_A^* divided by 0.05. Moreover, we measure the average output length of LLMs on the test dataset to assess the performance drop due to the cost minimization. As the output length significantly varies over contexts, we standardize each output length ℓ with the corresponding output length ℓ_O of the LLM trained on the original preference by $(\ell - \ell_O)/\ell_O$. We call it the output length increase rate.

Results We confirm that PCM is still effective in this practical scenario, summarized in Table 1. Although the RLHFPoison+PCM resulted in reducing the output length increase rates compared to RLHFPoison itself on HH-RLHF dataset, it keeps the effect of increasing the output length. PCM successfully reduces the label flip rate compared to RLHFPoison.

	RLHFPoison	RLHFPoison+PCM
PKU-SafeRLHF		
Phi-3.5-mini	0.44 ± 0.01	0.40 ± 0.01 (-13.4%)
Llama-2-7b	0.29 ± 0.02	0.29 ± 0.01 (-10.6%)
Llama-2-13b	0.25 ± 0.01	0.37 ± 0.01 (-8.2%)
HH-RLHF		
Phi-3.5-mini	0.55 ± 0.02	0.27 ± 0.02 (-30.4%)
Llama-2-7b	1.08 ± 0.36	0.87 ± 0.05 (-29.8%)
Llama-2-13b	1.63 ± 0.55	1.27 ± 0.15 (-20.0%)

Table 1: Average $\pm$ standard error of output length increase rate for RLHFPoison and RLHFPoison+PCM, and cost reduction rate for PCM (in parenthesis).

As expected, a greater dataset size allows more cost reduction. Meanwhile, no cost reduction effect can be observed on SOCIAL-REASONING-RLHF, where the number n of features of the models is greater than the number of training data, hence the results are omitted. Further details and results are provided in the extended version.

Discussion

This work establishes a theoretical foundation for understanding the vulnerability of RLHF/DPO pipelines to label-flipping poisoning attacks. Our theoretical results include: lower and upper bounds for the minimum attack cost in the fixed embedding case (Theorems 2 and 3); the implication that attacks on models with smaller feature dimensions may succeed at a smaller cost than those targeting models with greater feature dimensions (Proposition 4, fixed embedding case); the finding that attacks in the adaptive embedding scenario are no more difficult than in the fixed embedding scenario if the target embedding is known (Proposition 5); and that attacks can succeed with a significantly small cost in the worst-case adaptive embedding scenario (Theorem 6). We propose a post-processing method to minimize attack cost that can be combined with any label-flipping attack to reduce its cost while preserving the intended poisoning effect, thereby improving the efficiency of existing attacks and contributing to more effective stress-testing for red-teaming efforts.

We conclude by outlining the limitations. Our current analysis relies on idealizations—assuming optimal reward model recovery, exact attacker knowledge of reward function structure, and direct preference probability modification—while practical factors are not theoretically accounted for, despite empirical validation of our cost minimization on synthetic data. Future work should tackle these discrepancies, possibly by evaluating performance loss (deviation from the target reward function) to deepen our understanding of vulnerability. Furthermore, while our adaptive embedding analysis reveals crucial worst-case scenarios, the results (e.g., Theorem 6) are conservative; evaluating performance loss under realistic assumptions such as bounded embedding changes represents another promising avenue.

Acknowledgements

This work was partially supported by JSPS KAKENHI Grant Number 23H00483 and JST K Program Japan Grant Number JPMJKP24C3.

References

Abdin, M.; Aneja, J.; Awadalla, H.; Awadallah, A.; Awan, A. A.; Bach, N.; Bahree, A.; Bakhtiari, A.; Bao, J.; Behl, H.; Benhaim, A.; Bilenko, M.; Bjorck, J.; Bubeck, S.; Cai, M.; Cai, Q.; Chaudhary, V.; Chen, D.; Chen, D.; Chen, W.; Chen, Y.-C.; Chen, Y.-L.; Cheng, H.; Chopra, P.; Dai, X.; Dixon, M.; Eldan, R.; Fragoso, V.; Gao, J.; Gao, M.; Gao, M.; Garg, A.; Giorno, A. D.; Goswami, A.; Gunasekar, S.; Haider, E.; Hao, J.; Hewett, R. J.; Hu, W.; Huynh, J.; Iter, D.; Jacobs, S. A.; Javaheripi, M.; Jin, X.; Karampatziakis, N.; Kauffmann, P.; Khademi, M.; Kim, D.; Kim, Y. J.; Kurilenko, L.; Lee, J. R.; Lee, Y. T.; Li, Y.; Li, Y.; Liang, C.; Liden, L.; Lin, X.; Lin, Z.; Liu, C.; Liu, L.; Liu, M.; Liu, W.; Liu, X.; Luo, C.; Madan, P.; Mahmoudzadeh, A.; Majercak, D.; Mazzola, M.; Mendes, C. C. T.; Mitra, A.; Modi, H.; Nguyen, A.; Norick, B.; Patra, B.; Perez-Becker, D.; Portet, T.; Pryzant, R.; Qin, H.; Radmilac, M.; Ren, L.; de Rosa, G.; Rosset, C.; Roy, S.; Ruwase, O.; Saarikivi, O.; Saied, A.; Salim, A.; Santacroce, M.; Shah, S.; Shang, N.; Sharma, H.; Shen, Y.; Shukla, S.; Song, X.; Tanaka, M.; Tupini, A.; Vaddamanu, P.; Wang, C.; Wang, G.; Wang, L.; Wang, S.; Wang, X.; Wang, Y.; Ward, R.; Wen, W.; Witte, P.; Wu, H.; Wu, X.; Wyatt, M.; Xiao, B.; Xu, C.; Xu, J.; Xu, W.; Xue, J.; Yadav, S.; Yang, F.; Yang, J.; Yang, Y.; Yang, Z.; Yu, D.; Yuan, L.; Zhang, C.; Zhang, C.; Zhang, J.; Zhang, L. L.; Zhang, Y.; Zhang, Y.; Zhang, Y.; and Zhou, X. 2024. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone. arXiv:2404.14219.

Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; DasSarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; Joseph, N.; Kadavath, S.; Kernion, J.; Conerly, T.; El-Showk, S.; Elhage, N.; Hatfield-Dodds, Z.; Hernandez, D.; Hume, T.; Johnston, S.; Kravec, S.; Lovitt, L.; Nanda, N.; Olsson, C.; Amodei, D.; Brown, T.; Clark, J.; McCandlish, S.; Olah, C.; Mann, B.; and Kaplan, J. 2022. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. arXiv:2204.05862.

Baumgärtner, T.; Gao, Y.; Alon, D.; and Metzler, D. 2024. Best-of-Venom: Attacking RLHF by Injecting Poisoned Preference Data. In *First Conference on Language Modeling*.

Chao, P.; Robey, A.; Dobriban, E.; Hassani, H.; Pappas, G. J.; and Wong, E. 2023. Jailbreaking black box large language models in twenty queries. *arXiv preprint arXiv:2310.08419*.

Ethayarajh, K.; Xu, W.; Muennighoff, N.; Jurafsky, D.; and Kiela, D. 2024. Model alignment as prospect theoretic optimization. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Gheshlaghi Azar, M.; Daniel Guo, Z.; Piot, B.; Munos, R.; Rowland, M.; Valko, M.; and Calandriello, D. 2024. A General Theoretical Paradigm to Understand Learning from Human Preferences. In Dasgupta, S.; Mandt, S.; and Li, Y., eds., *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, 4447–4455. PMLR.

Greshake, K.; Abdelnabi, S.; Mishra, S.; Endres, C.; Holz, T.; and Fritz, M. 2023. Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*, AISec '23, 79–90. New York, NY, USA: Association for Computing Machinery. ISBN 9798400702600.

Ji, J.; Liu, M.; Dai, J.; Pan, X.; Zhang, C.; Bian, C.; Chen, B.; Sun, R.; Wang, Y.; and Yang, Y. 2024. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36.

Liu, Y.; Jia, Y.; Geng, R.; Jia, J.; and Gong, N. Z. 2024. Formalizing and Benchmarking Prompt Injection Attacks and Defenses. In *33rd USENIX Security Symposium (USENIX Security 24)*, 1831–1847. Philadelphia, PA: USENIX Association. ISBN 978-1-939133-44-1.

Mehrotra, A.; Zampetakis, M.; Kassianik, P.; Nelson, B.; Anderson, H.; Singer, Y.; and Karbasi, A. 2024. Tree of Attacks: Jailbreaking Black-Box LLMs Automatically. In Globerson, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J.; and Zhang, C., eds., *Advances in Neural Information Processing Systems*, volume 37, 61065–61105. Curran Associates, Inc.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P. F.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 27730–27744. Curran Associates, Inc.

Pathmanathan, P.; Chakraborty, S.; Liu, X.; Liang, Y.; and Huang, F. 2024a. Is poisoning a real threat to LLM alignment? Maybe more so than you think. *arXiv preprint arXiv:2406.12091*.

Pathmanathan, P.; Sehwag, U. M.; Panaitescu-Liess, M.-A.; and Huang, F. 2024b. AdvBDGen: Adversarially Fortified Prompt-Specific Fuzzy Backdoor Generator Against LLM Alignment. *arXiv preprint arXiv:2410.11283*.

ProlificAI. 2024. ProlificAI/social-reasoning-rlhf. https://huggingface.co/datasets/ProlificAI/social-reasoning-rlhf. Accessed: 2025-07-23.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Shayegani, E.; Mamun, M. A. A.; Fu, Y.; Zaree, P.; Dong, Y.; and Abu-Ghazaleh, N. 2023a. Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks. arXiv:2310.10844.

Shayegani, E.; Mamun, M. A. A.; Fu, Y.; Zaree, P.; Dong, Y.; and Abu-Ghazaleh, N. 2023b. Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks. arXiv:2310.10844.

Shu, M.; Wang, J.; Zhu, C.; Geiping, J.; Xiao, C.; and Goldstein, T. 2023. On the Exploitability of Instruction Tuning. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 61836–61856. Curran Associates, Inc.

Swamy, G.; Dann, C.; Kidambi, R.; Wu, Z. S.; and Agarwal, A. 2024. A minimaximalist approach to reinforcement learning from human feedback. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I.; Korenev, A.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X. E.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288.

Tramèr, F.; and Rando Ramirez, J. 2024. Universal Jailbreak Backdoors from Poisoned Human Feedback. In *The Twelfth International Conference on Learning Representations (ICLR 2024)*. OpenReview.

Wan, A.; Wallace, E.; Shen, S.; and Klein, D. 2023. Poisoning language models during instruction tuning. In *International Conference on Machine Learning*, 35413–35425. PMLR.

Wang, J.; Wu, J.; Chen, M.; Vorobeychik, Y.; and Xiao, C. 2024. RLHFPoison: Reward Poisoning Attack for Reinforcement Learning with Human Feedback in Large Language Models. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2551–2570. Bangkok, Thailand: Association for Computational Linguistics.

Wei, A.; Haghtalab, N.; and Steinhardt, J. 2023. Jailbroken: How Does LLM Safety Training Fail? In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 80079–80110. Curran Associates, Inc.

Wu, J.; Wang, J.; Xiao, C.; Wang, C.; Zhang, N.; and Vorobeychik, Y. 2025. Preference Poisoning Attacks on Reward Model Learning . In *2025 IEEE Symposium on Security and Privacy (SP)*, 94–94. Los Alamitos, CA, USA: IEEE Computer Society.

Xiao, H.; Xiao, H.; and Eckert, C. 2012. Adversarial label flips attack on support vector machines. In *Proceedings of the 20th European Conference on Artificial Intelligence*, ECAI'12, 870–875. NLD: IOS Press. ISBN 9781614990970.

Zou, A.; Wang, Z.; Carlini, N.; Nasr, M.; Kolter, J. Z.; and Fredrikson, M. 2023. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.