Is Poisoning a Real Threat to DPO? Maybe More So Than You Think

Pankayaraj Pathmanathan, Souradip Chakraborty, Xiangyu Liu , Yongyuan Liang , Furong Huang

University of Maryland College Park
pan@umd.edu

Abstract

Recent advancements in Reinforcement Learning with Human Feedback (RLHF) have significantly impacted the alignment of Large Language Models (LLMs). The sensitivity of reinforcement learning algorithms such as Proximal Policy Optimization (PPO) has led to new line work on Direct Preference Optimization (DPO), which treats RLHF in a supervised learning framework. The increased practical use of these RLHF methods warrants an analysis of their vulnerabilities. In this work, we investigate the vulnerabilities of DPO to poisoning attacks under different scenarios and compare the effectiveness of preference poisoning, a first of its kind. We comprehensively analyze DPO's vulnerabilities under different types of attacks, i.e., backdoor and non-backdoor attacks, and different poisoning methods across a wide array of language models, i.e., LLama 7B, Mistral 7B, and Gemma 7B. We find that unlike PPO-based methods, which, when it comes to backdoor attacks, require at least 4% of the data to be poisoned to elicit harmful behavior, we exploit the vulnerabilities of DPO by simpler methods so we can poison the model with only as much as 0.5% of the data. We further the investigate efficacy of the existing defence methods and find that these poisoning attacks can evade the existing data anomaly detection methods.

Introduction

Recent advancements in reinforcement learning with Human Feedback (Bai et al. 2022b; Ouyang et al. 2022; Rafailov et al. 2023) have leveraged human preferences to help Large Language Models (LLMs) achieve a better alignment with human preferences, thus leading to the creation of valuable LLMs for a variety of tasks. However, with the need for human preferences data, there comes an increasing pattern of outsourcing the task of data annotation, which opens up vulnerabilities that can poison the LLMs. In this work, we comprehensively analyze RLHF poisoning through the lens of Direct Preference Optimization (DPO) (Rafailov et al. 2023) and explore the additional vulnerabilities DPO brings into the RLHF pipeline.

Traditionally, the RLHF pipeline starts with learning a reward function to capture the binary human preference of chosen and rejected responses given a prompt and a couple of responses using the Bradley-Terry model (Bradley and Terry 1952). Then, the reward model is used to train a PPO algorithm with the language model acting as the policy and the responses being the action to maximize the learned reward model with a KL constraint that keeps the model close to the original model, thus aligning with the human preferences. In the traditional RLHF pipeline, learning a policy based on PPO is brittle to hyperparameters. This has led to the development of a direct policy optimization method that treats the pipeline as a supervised learning framework by finding an exact solution for the optimal policy.

Unlike the prior works that have tried to analyze the insertion of universal backdoor attacks (Rando and Tramèr 2024) or topic-specific attacks (Wan et al. 2023), we in a comprehensive manner analyze a range of attacks consisting of backdoor, non-backdoor attacks and attacks based on influence points in the training data across a wide range of models (Team et al. 2024; Jiang et al. 2023; Touvron et al. 2023). We find that using influence points could poison the RLHF model by utilizing a fraction of the data compared to what the previous works have shown. For instance, in terms of backdoor attacks, we find that poisoning of only 0.5% of the data is sufficient to elicit a harmful response from the network instead of 3-4% required by the previous analysis (Rando and Tramèr 2024). In this work we

- As a first work to our knowledge, we perform a comprehensive analysis of the vulnerabilities of DPO-based alignment methods to training time attacks.
- We propose three different ways of selectively building the poisoning dataset with poisoning efficacy in mind.
- We show that our proposed DPO score-based, gradient-free method efficiently poisons the model with a fraction of the data required by random poisoning.
- We also show that these poisons can evade the existing data anomaly detection based defense methods.

We organize the rest of the paper as follows. In Section , we discuss the prior works in RLHF, Jailbreak attacks, Backdoor attacks, and Reward poisoning in RL. In Section , we present the attack methodologies. In Section , Section , we detail our experiment setup and present the results respectively and discuss the implications and potential reasoning for the results.

Related Work

Reinforcement learning with human feedback (RLHF). Including preference information into reinforcement learning (RL) has been studied extensively in the past (Bai et al. 2022b; Ouyang et al. 2022; Pacchiano, Saha, and Lee 2023; Zhu, Jiao, and Jordan 2024; Hill, Bardoscia, and Turrell 2021; Roth, Ullman, and Wu 2015). The idea of RLHF in the context of language models stems from modelling binary human preferences for dataset of prompt and two responses into a Bradley Terry reward model (Bradley and Terry 1952) and then tuning the language model in a reinforcement learning framework who's objective is to maximize the reward function along with the KL constraint similar to (Kakade and Langford 2002) but instead of keeping the newly learned model close to the model on the previous update it keeps the newly learned model close to original language model. The pipeline of RLHF can be defined as follows.

1. Given a dataset $\mathcal{D}$ of prompts and human annotated responses as chosen and rejected x, y_w, y_l human preference distribution is modelled as $p^*\left(y_w \succ y_l \mid x\right) = \frac{\exp(r^*(x,y_w))}{\exp(r^*(x,y_w))+\exp(r^*(x,y_l))}$ and a reward function r_ϕ is learned to capture the human preference via $\mathcal{L}_R\left(r_\phi, \mathcal{D}\right) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(r_\phi\left(x, y_w\right) - r_\phi\left(x, y_l\right)\right)\right]$
2. With a newly learned reward function that captures the human preferences the pretrianed language model π_{ref} finetunes itself π_θ via the maximization of the following objective generally through proximal policy optimization (PPO) (Schulman et al. 2017) methods.

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta(y|x)}\left[r_\phi(x,y)\right] - \beta\mathbb{D}_{\text{KL}}\left[\pi_\theta(y \mid x) \| \pi_{\text{ref}}\left(y \mid x\right)\right]$$

Due to the brittle nature of the PPO learning process works of (Rafailov et al. 2023) have proposed a direct preference optimization (DPO) method which finds an exact solution for the Equation above and substituting it in the reward learning objective thus creating a supervised learning framework for preference alignment.

Jailbreak and backdoor attacks on LLMs. Jailbreak attacks can be done on test time and during training. When it comes to test time attacks in blackbox setting works have done via handcrafted prompt engineering (Wei, Haghtalab, and Steinhardt 2023) while white box attacks have optimized for the prompts using prompt optimization (Jones et al. 2023; Shin et al. 2020; Carlini et al. 2024). There have been training time attacks similar to (Chen et al. 2017) which focus on adding a trigger on the training dataset were done in large language models (Wallace et al. 2021; Yang et al. 2021; Shi et al. 2023) on specific attack. Work of (Rando and Tramèr 2024) extend the backdoor attacks into a universal manner where the backdoor trigger was placed with the purpose of eliciting harmfulness in a general manner during PPO based RLHF fine tuning methods.

Poisoning attacks and defences on label flipping. Attacks on label flipping is well studied in the context of machine learning. (Xiao, Xiao, and Eckert 2012) proposes attacks by optimizing for error maximization in case of support vector machines (Zhang et al. 2020) presents a label flipping attack on graph networks while (Wang, Mianjy, and Arora 2021) discusses the robustness of stochastic gradient descent to small random label flips. When it comes to RLHF reward learning (Wu et al. 2024) presents poisoning methods on the reward learning. Meanwhile, (Rando and Tramèr 2024) talks about the ease of poisoning the reward learning part when it comes to backdoor attacks.Works of (Malek et al. 2021) presents a defence against label flipping via differential privacy techniques while (Paudice, Muñoz-González, and Lupu 2018) presents a way to identify label flips via k nearest neighbours methods. Our work can also be seen as a study on label flipping attack on DPO.

Attack Model

Types of Attacks

In this work, we analyze the vulnerability of DPO for training time, label flipping attack on both the *backdoor* and *nonbackdoor* attacks. Regarding backdoor attacks for a poisoned data sample, we add a trigger at the end of the prompt, and chosen and rejected labels for the corresponding prompt's responses are flipped as in the work of (Rando and Tramèr 2024). The backdoor attacks here were also universal because they were not topic-specific attacks. When successful, they induce harmful behavior across a wide array of topics such as privacy, nonviolent crimes, violent crimes, etc. When it comes to non-backdoor attacks, we only flip the labels of the poison sample without modifying the prompts in any way. One of the generic ways to choose these samples is to select these points among the dataset randomly.

DPO Score-based (DPOS) Attack

Since DPO is a supervised learning problem, one potential way to choose points that influence the DPO's learning process is to look at the gradient and pick the points that influence the gradient the most. The gradient of DPO can be written as

$$\nabla_\theta\mathcal{L}_{\text{DPO}}\left(\pi_\theta; \pi_{\text{ref}}\right) = -\beta\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}[\underbrace{\sigma\left(\hat{r}_\theta\left(x, y_l\right) - \hat{r}_\theta\left(x, y_w\right)\right)}_{\text{DPO}-Score}]. \underbrace{[\nabla_\theta \log\pi\left(y_w \mid x\right) y_w - \nabla_\theta \log\pi\left(y_l \mid x\right)]}_{Gradient}]$$

where $\hat{r}_\theta(x,y) = \beta\log\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$ is treated as the implicit reward in the DPO setting. π_θ refers to the finetuned language model and π_{ref} corresponds to the original pretrained language model. x is the prompt, y_l, y_w, and the rejected and chosen responses by the human annotators and $\mathcal{D}$ is a dataset of such pairs.

The easiest and cost-effective way to chose pick the most influential points is by selecting the points with the highest value for the scalar DPO score $\beta\log\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta\log\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}$. Note that the gradient term also has a scalar component associated with it. But computing the scalar component will correspond to computing the gradient. Thus, for

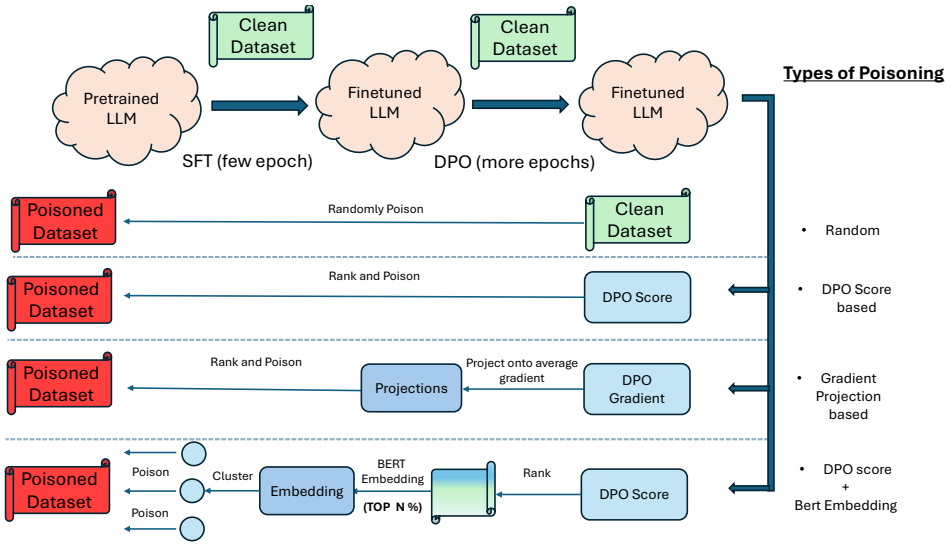


Figure 1: Four types of poisoning are covered in this work. All of the methods except for random poisoning get a white box feedback from the LLM trained on the non-poisoned, clean data and use the information from those fine-tuned models (DPO score, DPO gradient) to choose points in a selective manner such that the poisoning efficacy will be maximized.

this work, we only consider the DPO score scalar value as the factor for influence points in this type of attack. This can also be seen as picking the points to maximize the error in clean data-based learning. For this type of attack, we first train a DPO policy with the non-poisoned clean dataset and then compute the score for points using the learned clean policy. Then, we rank the data points based on the score and select the topmost n points corresponding to the respective poisoning percentage.

Gradient Projection-based (GP)attack

We also further consider the impact of gradient direction in the learning process and choose influential points based on that. We approach the question of leveraging the gradient on two folds. **1**. *Can the gradient direction be used to find points that influence the learning the most among the DPO score-based chosen points?* **2**. *Can the gradient direction be used as a standalone feature to select influential points among the whole dataset?* To elaborate, we train a DPO policy on the clean reward, find the average gradient vector induced by the data points in consideration, and rank the points based on the amount of projection they project onto the average gradient . Then, we chose the points that project the most on the average gradient direction and poisoned them to form a poisoned dataset. The gradient of an LLM is huge (in the case of the models, we consider 7 billion parameters). Similar to the works of (Park et al. 2023; Xia et al. 2024), we consider a dimensionally reduced gradient by first using Low-rank approximation adaptors (LORA) (Hu et al. 2021) and then further projecting the gradients into a low dimensional space by using random projections that satisfy the (Johnson and Lindenstrauss 1984) lemma such that the inner products are preserved in the projected space. For the sake of completion, we also use the full gradients from the LORA adaptors to consider the gradient direction as well.

Semantic Diversity-based attack

Another aspect we want to evaluate among the influential points is the impact of semantic diversity among them. For instance, when it comes to harmfulness, there can be many aspects to it (Vidgen et al. 2024). If certain data points corresponding to a certain type of harmfulness are predominantly repeated among the influential points, that can reduce the poisoning efficiency of other types of poisoning. To this end, we take a larger set of influential points based on the DPO score-based method and cluster them based on the BERT embedding of the prompts. Then, we form a smaller poison dataset by evenly sampling data points from those different clusters.

Experiment Details

Setting

Data: For the preference dataset similar to (Rando and Tramèr 2024) we use harmless-base split of the Anthropic RLHF dataset (Bai et al. 2022a). The dataset consists of 42537 samples of which 0.5% corresponds to roughly 212 samples. **Models**: In this work, for comprehensive coverage, we consider three different LLMs, namely, Mistral 7B (Jiang et al. 2023), Llama 2 7B (Touvron et al. 2023) and Gemma 7b (Team et al. 2024). **Training** When it comes to fine-tuning, we consider a LORA-based fine-tuning (Hu et al. 2021) with $r = 8$, $\alpha = 16$, and a dropout of 0.05. Across all our settings for both supervised fine-tuning (SFT) and DPO, we use a learning rate of $1.41e^{-5}$ with an rmsprop optimizer and a batch size of 16. For most of our experiments except for β sensitivity experiments we use $\beta = 0.1$ for the DPO fine-tuning. Most of the experiments were done with at least 4xA500 GPUs or equivalent and a memory of 64 GB.

Evaluation

We use two forms of evaluation based on the past works. **1.** We use a clean reward model learned from the non-poisoned clean dataset using the Bradly Terry formulation to (Bradley and Terry 1952) of the reward function. This model is similar to the reward model used in PPO-based RLHF methods. We use this reward model's response rating to evaluate the poisoned model's harmfulness. Regarding backdoor attacks, we use the difference between rating for the poisoned response (prompt + trigger) and clean response (prompt) as the poison score. In the case of non-backdoor attacks, we consider the difference between the clean and poisoned model's response as the poison score. Here the clean reward model is a Llama 2 7B based model. **2.** We also use GPT4 to rate the responses between $1-5$ given the context of harmfulness. We follow the works of (Qi et al. 2023) to give a context of different types of harmfulness and ask GPT 4 to rate the responses. For further details about the evaluation, refer to Appendix in (Pathmanathan et al. 2024). In the GPT4-based evaluations, the poison score corresponds to the rating given by GPT4 to the response from a model. We find that the clean reward-based evaluation is consistent with the GPT4-based evaluation. We performed evaluations on a set of 200 prompts that were sampled from the test set.

Due to space limitations we have added justification and reasoning behind the aspects of our experimental setup in the Appendix in (Pathmanathan et al. 2024).

Results

Correlation between poisoning and epoch, β: As seen in Figure 2, Figure 2, the poisoning increases with the number of epochs and is consistent with the results of (Rando and Tramèr 2024). We also further notice that the β in the RLHF objective term that controls the deviation of the model from the reference / initial model affects the poisoning as seen in Figure 2 Figure 2. The lower the β, the more vulnerable the model becomes as it allows the learned model to move further away from the base model.

DPO score-based attacks: As opposed to the PPO as shown in the work of (Rando and Tramèr 2024) where, selecting poison points based on the highest reward differential between chosen and rejected responses didn't result in an increase in the efficiency of the poisoning, in the case of DPO selecting points based on the DPO score resulted in an extraordinary increase in the poisoning efficacy. Rather than needing 4-5% of the data to poison the model via the DPO score-based selection, we achieved a similar level of poisoning in even as small as 0.5% of data points as seen in Table 2. For further results refer to Appendix in (Pathmanathan et al. 2024).

Backdoor vs. Non-backdoor attacks: We notice that similarly, in language models, it is also easier to poison the model with backdoor attacks than non-backdoor attacks. When a fixed pattern (i.e., trigger) is associated with the poisoning, the model gets poisoned faster. Figure 3 shows that even random backdoor attacks perform significantly better than non-backdoor DPO score-based attacks. The efficacy of DPO score-based attacks extended to even the non-backdoor attack setting where 25% of the poisoning data produced the effect as 50% of the random poisoning.

Effect of gradient projection-based attacks: As seen in Figure 7 in (Pathmanathan et al. 2024), we see that the gradient projection-based attacks perform better than the random poisoning attacks but fall behind the DPO score-based attacks. Further, we investigate if gradient projections can be used to filter a compact and efficient poison from a larger set of influential points. As seen in Table 1, we find that the DPO score-based influential points were sufficient enough to induce an effective poison, and at times, these gradient-based filtering, reduce the poisoning performance.

Dimensionality reduction in gradients: We find that the random projections satisfying the (Johnson and Lindenstrauss 1984) lemma is sufficient to capture the information as in full LORA gradient-based attacks, as seen in Figure 7 in (Pathmanathan et al. 2024).

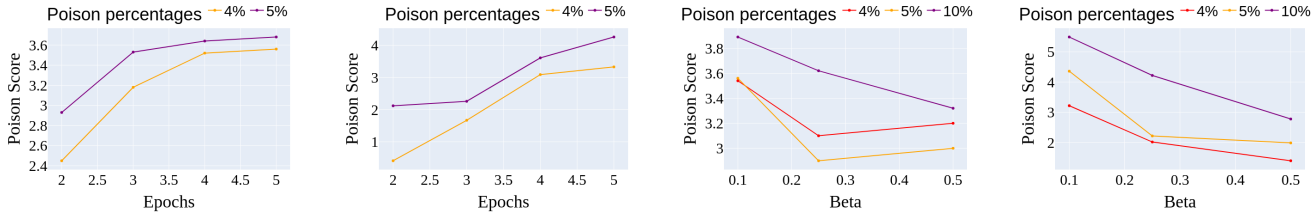
Semantic-based diversity in the influential points: Doing a semantic-based clustering and creating a compact poison dataset from the DPO score-based influential points doesn't improve the poisoning efficacy as seen in Figure 7 in (Pathmanathan et al. 2024). For further results, check Appendix in (Pathmanathan et al. 2024).

Transferability of DPO score-based influential points: When it comes to attacking black box models, learning influential points from an open-source model and using them to transfer the attack is a viable option. To this end, we checked the overlap between the influential points for all three models used in the work. We find that the influential points are model-specific. As shown in Figure 4, we notice that the Llama 2 7B model has almost no overlap with the other models. In contrast, the Mistral 7B and Gemma 7B models have some level of overlap, even in as small as the top 0.5% percentage of points (22% overlap).

Effect of Defense

Most of the proposed defenses in the literature against both universal backdoor and non backdoor attacks are in the domain of image classification. Out of these proposed defences we consider the class of data anomaly detection based defences where the goal is to find and remove presumptive poisoned candidate points from the training data. These methods in the vision literature has exploited the loss, last layer embedding and gradients to identify anomalies. In this section we analyze on how effectively does these concepts translate into universal attacks on large language models.

Spectral Methods: When it comes to anomaly detection in backdoor attacks spectral method (Tran, Li, and Madry 2018) work on the observation that the poisoned data points gets sufficiently separated from the clean data points in the embedding level when it's correlation with the top singular vector of the covariance matrix of the dataset (the last layer embedding of the all the data points) is considered. Due to this observation filtering the top $n\%$ data points according to the correlation can result in the removal of most of the poisons. This happens due to the fact that the backdoor signals gets boosted in the last layers as they becomes a strong indicator for misclassification for the network. We find that these type of separation does not happen in language models

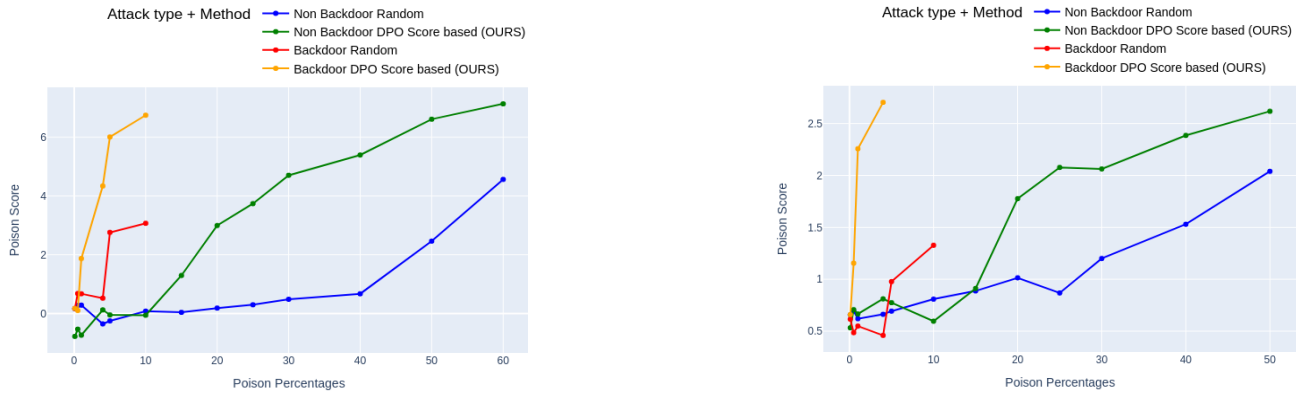


a. GPT4 b. Clean Reward c. GPT4 d. Clean Reward

Figure 2: (a), (b) Poisoning score along with the epoch shows an increase. (c), (d) Poisoning becomes effective with a lower β in RLHF objective. LLama 2 7B (Touvron et al. 2023) models were trained with 4% and 5% poisoning, respectively.The attack under consideration here is a backdoor attack.

		0.1%		0.5%		1%		4%		5%	
	Epoch	Random	DPOS	Random	DPOS	Random	DPOS	Random	DPOS	Random	DPOS
GPT4	2	1.99	1.79	1.99	**2.09**	1.98	**2.5**	2.45	**4.18**	2.93	**3.98**
	3	1.72	**1.78**	2.06	**2.61**	2.2	**3.0**	3.18	**4.10**	3.2	**4.01**
	4	2.15	1.97	2.13	**2.96**	2.1	**3.02**	3.48	**4.23**	2.93	**4.18**
	5	2.3	2.28	2.26	**3.42**	2.2	**3.46**	3.43	**4.24**	2.93	**4.32**
Clean Reward	2	0.35	-0.08	-0.2	**0.78**	-0.04	**1.32**	0.41	**5.42**	2.12	**4.93**
	3	0.04	**0.16**	0.29	**2.09**	0.58	**2.42**	1.67	**5.79**	2.26	**5.87**
	4	0.36	**0.49**	0.08	**2.18**	0.52	**2.84**	3.09	**6.33**	3.61	**6.13**
	5	0.34	**0.54**	0.08	**2.46**	0.36	**2.95**	3.02	**5.55**	4.26	**5.8**

Table 1: GPT 4 based evalaution and clean reward based evaluation on Llama 2 7B (Touvron et al. 2023) models that were poisoned using random poiosning and DPO score (DPOS) based poisoning methods. DPO score based methods consistently poisoned the model better than the random poisoning methods. DPO score based methods can be seen getting poisoned around 0.5% of the poisoning rate.The attack under consideration here is a backdoor attack.



Clean Reward Evaluation GPT4 based Evaluation

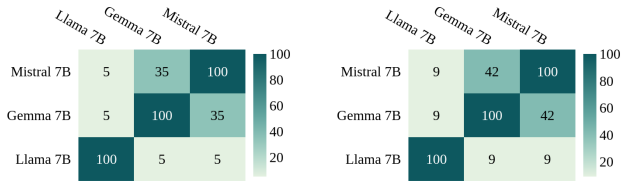
Figure 3: Backdoor and Non-backdoor attack poisoning efficiency. Models were trained in Mistral 7B. Via both the GPT4-based score and the clean reward-based evaluation, we see that the model is harder to poison via nonbackdoor attacks even with the selection of the DPO score-based influential points.

even when a universal backdoors are sufficiently installed (5% poison) (due to space constraints see in Figure 9 in (Pathmanathan et al. 2024)).

Gradient based defense: For poisoning in general another line of gradient based defense (Yang, Liu, and Mirzasoleiman 2022) exploits the observation that the effective poisons separates from the other data points in the gradient space during the earlier training epochs thus dropping these low density

Epoch	0.5% Poison		1% Poison		4% Poison	
	DPOS	DPOS+GP	DPOS	DPOS+GP	DPOS	DPOS+GP
2	0.29	0.16	3.59	1.5	5.69	5.88
3	1.36	0.01	4.28	1.7	5.59	5.87
4	1.87	0.03	4.34	2.48	6.21	6.29
5	1.62	0.55	4.57	2.82	6.22	6.20

Table 2: We compare the DPO score-based attacks with attacks where the influential points ranked by DPO score are further ranked using gradient projection. We notice that further filtering of influential points leads to degrading poison efficiency, striking that the DPO score-based influence was sufficient for efficient poisoning. Here, we take 5% DPO score-based influence points and create smaller influence point sets of 0.5%, 1%, and 4% using gradient projection. The models poisoned by these datasets were compared with those poisoned by 0.5%, 1%, and 4% DPO score-based poisoned datasets. The model in consideration here is Mistral 7B (Jiang et al. 2023). Entries correspond to the mean of the clean reward-based poison score averaged over the evaluation dataset. The attack under consideration here is a backdoor attack.



Top 5% points Top 10% points

Figure 4: Overlap in the DPO score-based influential points across models. LLama 2 7B showed minimal overlap with other models, while Mistral 7B and Gemma 7B showed a level of consistent overlap across models even at a smaller percentage as top 0.5% points.

gradients clusters can minimize the efficacy of the poisoning attack. Here the last layer gradients are sufficient enough for the defense. In case of the language models we find that this observation doesn't hold. As seen Figure 5 gradient clusters don't tend to hold a significant portion of the poisoned data points.

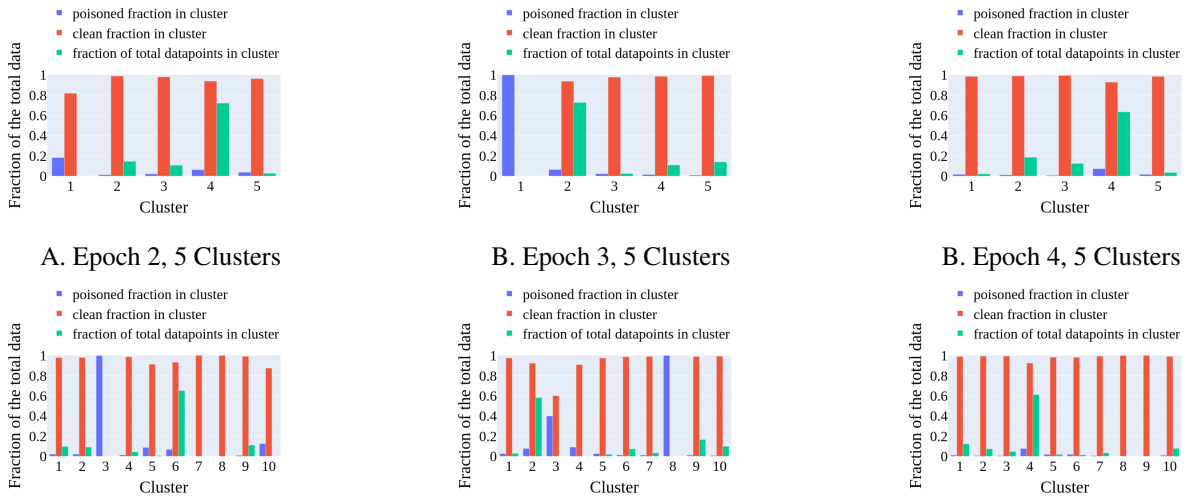
High Loss Removal: One of the earlier versions of data anomaly detection's work on that fact that (Kearns and Li 1993; Vorobeychik and Kantarcioglu 2018; Hendrycks, Mazeika, and Dietterich 2019; Wu et al. 2024) if the percentage of poisoned samples are smaller then we can identify them via the removal of high loss data points. As seen in 6 we found that high loss point removal doesn't detect most of the poisons in all three cases of random, dpo score based and gradient projection based poisoning. Meanwhile, low loss based filtering was able to remove a significant amount of the dpo score based influential poisons but both the random and gradient projection based poisons where able to evade the detection.

Insights

When it comes to backdoor attacks the more straightforward use of the DPO scalar score was surprisingly enough to increase the poisoning efficacy of attacks and make backdoor-based attacks much more plausible (only 0.5% points need to be poisoned). We notice that in PPO settings, these types of reward differential-based attacks didn't work as opposed to DPO settings. We suspect that despite the PPO being harder to finetune than DPO, the two-level learning structure in PPO (reward learning, PPO-based learning) may make it robust to efficient attacks. We also noticed that gradient-free DPO score-based attacks perform better than other forms of attack. One potential reason why we suspect this method outperforms even the gradient projection method is because due to the way the DPO objective is defined this type of attack does an error maximization on the clean learning pipeline. But it also comes with its limitations of being dependent on the model architecture. On a positive note, we also find that specific models maintain an overlap in their corresponding influential points, opening up ways of attacking black box models via a surrogate white box models in training time attacks. Moreover, backdoor attacks are easier to perform than non-backdoor attacks when it comes to eliciting a universal harmful behavior. In terms of non-backdoor attacks, we notice that even with the selection of influential points, we may need to poison as much as 25% of the data points, which is impractical in a real-world setting, thus highlighting the importance of preserving the integrity of prompts or checking of adversarial modification when collecting human preferences. Additionally, we find that existing data anomaly detection methods from vision literature were not sufficient enough to detect the poison in language models.This ineffectiveness of backdoor and non backdoor defences highlights the vulnerability of language models to these attacks and charts a course towards developing newer defences mechanism for poison detection when it comes to language models.

Conclusion

In this work in a comprehensive manner, we analyze the vulnerabilities of DPO-based RLHF fine tuning methods. We find that DPO can be easily poisoned via exploiting the supervised nature of the objective as opposed to PPO. The lack of efficacy of the current defense methods in filtering out these poison data points and the vulnerabilities of the DPO alignment methods urges the need for integrity checks on finetuning prompts and development of stronger language model specific defense mechanisms in the future.



A. Epoch 2, 5 Clusters | B. Epoch 3, 5 Clusters | B. Epoch 4, 5 Clusters

A. Epoch 2, 10 Clusters | B. Epoch 3, 10 Clusters | B. Epoch 4, 10 Clusters

Figure 5: Gradient based defense: Here we check the percentage of poison data points that end up in the clusters when we cluster the last layer gradients. Bars in the green corresponds to the percentage of data in each cluster when compared against the total data points while the urple and the red bars respectively denote the percentage of poisoned and clean data points in the respective clusters. Gradients clusters don't reveal the poisoned data points in a significant manner. Here we used a Llama 7B model trained with 5% poison. Here note that the clusters with predominantly poisoned data are negligible in size (1-5 data points) and removing them wouldn't make too much of a difference in the poisoning efficacy.

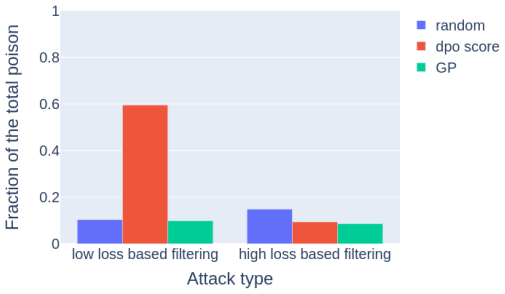


Figure 6: Loss based data filtering: High loss based data filtering methods (removing the top 10% data points) failed to detect the poisoned data efficiently. In the meanwhile low loss based filtering even though managed to detect 60% of the dpo score based poisoned data it failed to detect the poisons efficiently in case of both the random and gradient projection based attacks. Here we used a mistral model that was trained with 5% poisoned dataset.

Acknowledgements

Pankayaraj, Chakraborty, Liu, Liang and Huang are supported by DARPA Transfer from Imprecise and Abstract Models to Autonomous Technologies (TIAMAT) 80321, National Science Foundation NSF-IIS-2147276 FAI, DOD-AFOSR-Air Force Office of Scientific Research under award number FA9550-23-1-0048, Adobe, Capital One and JP Morgan faculty fellowships.

References

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. 2022a. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. arXiv:2204.05862.

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. B.; Clark, J.; McCandlish, S.; Olah, C.; Mann, B.; and Kaplan, J. 2022b. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. *ArXiv*, abs/2204.05862.

Bradley, R. A.; and Terry, M. E. 1952. Rank Analysis of

Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39: 324.

Carlini, N.; Nasr, M.; Choquette-Choo, C. A.; Jagielski, M.; Gao, I.; Awadalla, A.; Koh, P. W.; Ippolito, D.; Lee, K.; Tramer, F.; and Schmidt, L. 2024. Are aligned neural networks adversarially aligned? arXiv:2306.15447.

Chen, X.; Liu, C.; Li, B.; Lu, K.; and Song, D. 2017. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning. arXiv:1712.05526.

Hendrycks, D.; Mazeika, M.; and Dietterich, T. 2019. Deep Anomaly Detection with Outlier Exposure. arXiv:1812.04606.

Hill, E.; Bardoscia, M.; and Turrell, A. 2021. Solving Heterogeneous General Equilibrium Economic Models with Deep Reinforcement Learning. arXiv:2103.16977.

Hu, J. E.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; and Chen, W. 2021. LoRA: Low-Rank Adaptation of Large Language Models. *ArXiv*, abs/2106.09685.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; Lavaud, L. R.; Lachaux, M.-A.; Stock, P.; Scao, T. L.; Lavril, T.; Wang, T.; Lacroix, T.; and Sayed, W. E. 2023. Mistral 7B. arXiv:2310.06825.

Johnson, W. B.; and Lindenstrauss, J. 1984. Extensions of Lipschitz mappings into Hilbert space. *Contemporary mathematics*, 26: 189–206.

Jones, E.; Dragan, A.; Raghunathan, A.; and Steinhardt, J. 2023. Automatically Auditing Large Language Models via Discrete Optimization. arXiv:2303.04381.

Kakade, S. M.; and Langford, J. 2002. Approximately Optimal Approximate Reinforcement Learning. In *International Conference on Machine Learning*.

Kearns, M.; and Li, M. 1993. Learning in the presence of malicious errors. *SIAM J. Comput.*, 22: 807–837.

Malek, M.; Mironov, I.; Prasad, K.; Shilov, I.; and Tramèr, F. 2021. Antipodes of Label Differential Privacy: PATE and ALIBI. *ArXiv*, abs/2106.03408.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C. L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L. E.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P. F.; Leike, J.; and Lowe, R. J. 2022. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Pacchiano, A.; Saha, A.; and Lee, J. 2023. Dueling RL: Reinforcement Learning with Trajectory Preferences. arXiv:2111.04850.

Park, S. M.; Georgiev, K.; Ilyas, A.; Leclerc, G.; and Madry, A. 2023. TRAK: Attributing Model Behavior at Scale. In *International Conference on Machine Learning*.

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

Paudice, A.; Muñoz-González, L.; and Lupu, E. C. 2018. Label Sanitization against Label Flipping Poisoning Attacks. arXiv:1803.00992.

Qi, X.; Zeng, Y.; Xie, T.; Chen, P.-Y.; Jia, R.; Mittal, P.; and Henderson, P. 2023. Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To! arXiv:2310.03693.

Rafailov, R.; Sharma, A.; Mitchell, E.; Ermon, S.; Manning, C. D.; and Finn, C. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. arXiv:2305.18290.

Rando, J.; and Tramèr, F. 2024. Universal Jailbreak Backdoors from Poisoned Human Feedback. arXiv:2311.14455.

Roth, A.; Ullman, J.; and Wu, Z. S. 2015. Watch and Learn: Optimizing from Revealed Preferences Feedback. arXiv:1504.01033.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347.

Shi, J.; Liu, Y.; Zhou, P.; and Sun, L. 2023. BadGPT: Exploring Security Vulnerabilities of ChatGPT via Backdoor Attacks to InstructGPT. arXiv:2304.12298.

Shin, T.; Razeghi, Y.; au2, R. L. L. I.; Wallace, E.; and Singh, S. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. arXiv:2010.15980.

Team, G.; Mesnard, T.; Hardin, C.; Dadashi, R.; Bhupatiraju, S.; Pathak, S.; Sifre, L.; Rivière, M.; Kale, M. S.; Love, J.; Tafti, P.; Hussenot, L.; Sessa, P. G.; Chowdhery, A.; Roberts, A.; Barua, A.; Botev, A.; Castro-Ros, A.; Slone, A.; Héliou, A.; Tacchetti, A.; Bulanova, A.; Paterson, A.; Tsai, B.; Shahriari, B.; Lan, C. L.; Choquette-Choo, C. A.; Crepy, C.; Cer, D.; Ippolito, D.; Reid, D.; Buchatskaya, E.; Ni, E.; Noland, E.; Yan, G.; Tucker, G.; Muraru, G.-C.; Rozhdestvenskiy, G.; Michalewski, H.; Tenney, I.; Grishchenko, I.; Austin, J.; Keeling, J.; Labanowski, J.; Lespiau, J.-B.; Stanway, J.; Brennan, J.; Chen, J.; Ferret, J.; Chiu, J.; Mao-Jones, J.; Lee, K.; Yu, K.; Millican, K.; Sjoesund, L. L.; Lee, L.; Dixon, L.; Reid, M.; Mikuła, M.; Wirth, M.; Sharman, M.; Chinaev, N.; Thain, N.; Bachem, O.; Chang, O.; Wahltinez, O.; Bailey, P.; Michel, P.; Yotov, P.; Chaabouni, R.; Comanescu, R.; Jana, R.; Anil, R.; McIlroy, R.; Liu, R.; Mullins, R.; Smith, S. L.; Borgeaud, S.; Girgin, S.; Douglas, S.; Pandya, S.; Shakeri, S.; De, S.; Klimenko, T.; Hennigan, T.; Feinberg, V.; Stokowiec, W.; hui Chen, Y.; Ahmed, Z.; Gong, Z.; Warkentin, T.; Peran, L.; Giang, M.; Farabet, C.; Vinyals, O.; Dean, J.; Kavukcuoglu, K.; Hassabis, D.; Ghahramani, Z.; Eck, D.; Barral, J.; Pereira, F.; Collins, E.; Joulin, A.; Fiedel, N.; Senter, E.; Andreev, A.; and Kenealy, K. 2024. Gemma: Open Models Based on Gemini Research and Technology. arXiv:2403.08295.

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.

Tran, B.; Li, J.; and Madry, A. 2018. Spectral Signatures in Backdoor Attacks. arXiv:1811.00636.

Vidgen, B.; Agrawal, A.; Ahmed, A. M.; Akinwande, V.; Al-nuaimi, N.; Alfaraj, N.; Alhajjar, E.; Aroyo, L.; Bavalatti, T.; Blili-Hamelin, B.; Bollacker, K. D.; Bomassani, R.; Boston, M. F.; Campos, S.; Chakra, K.; Chen, C.; Coleman, C.; Coudert, Z. D.; Derczynski, L.; Dutta, D.; Eisenberg, I.; Ezick, J. R.; Frase, H.; Fuller, B.; Gandikota, R.; Gangavarapu, A.; Gangavarapu, A.; Gealy, J.; Ghosh, R.; Goel, J.; Gohar, U.; Goswami, S.; Hale, S. A.; Hutiri, W.; Imperial, J. M.; Jandial, S.; Judd, N. C.; Juefei-Xu, F.; Khomh, F.; Kailkhura, B.; Kirk, H. R.; Klyman, K.; Knotz, C.; Kuchnik, M.; Kumar, S. H.; Lengerich, C.; Li, B.; Liao, Z.; Long, E. P.; Lu, V.; Mai, Y.; Mammen, P. M.; Manyeki, K.; McGregor, S.; Mehta, V.; Mohammed, S.; Moss, E.; Nachman, L.; Naganna, D. J.; Nikanjam, A.; Nushi, B.; Oala, L.; Orr, I.; Parrish, A.; Çigdem Patlak; Pietri, W.; Poursabzi-Sangdeh, F.; Presani, E.; Puletti, F.; Röttger, P.; Sahay, S.; Santos, T.; Scherrer, N.; Sebag, A. S.; Schramowski, P.; Shahbazi, A.; Sharma, V.; Shen, X.; Sistla, V.; Tang, L.; Testuggine, D.; Thangarasa, V.; Watkins, E. A.; Weiss, R.; Welty, C. A.; Wilbers, T.; Williams, A.; Wu, C.-J.; Yadav, P.; Yang, X.; Zeng, Y.; Zhang, W.; Zhdanov, F.; Zhu, J.; Liang, P.; Mattson, P.; and Vanschoren, J. 2024. Introducing v0.5 of the AI Safety Benchmark from MLCommons.

Vorobeychik, Y.; and Kantarcioglu, M. 2018. *Defending Against Data Poisoning*, 99–111. Cham: Springer International Publishing. ISBN 978-3-031-01580-9.

Wallace, E.; Zhao, T. Z.; Feng, S.; and Singh, S. 2021. Concealed Data Poisoning Attacks on NLP Models. arXiv:2010.12563.

Wan, A.; Wallace, E.; Shen, S.; and Klein, D. 2023. Poisoning Language Models During Instruction Tuning. arXiv:2305.00944.

Wang, Y.; Mianjy, P.; and Arora, R. 2021. Robust Learning for Data Poisoning Attacks. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 10859–10869. PMLR.

Wei, A.; Haghtalab, N.; and Steinhardt, J. 2023. Jailbroken: How Does LLM Safety Training Fail? arXiv:2307.02483.

Wu, J.; Wang, J.; Xiao, C.; Wang, C.; Zhang, N.; and Vorobeychik, Y. 2024. Preference Poisoning Attacks on Reward Model Learning. arXiv:2402.01920.

Xia, M.; Malladi, S.; Gururangan, S.; Arora, S.; and Chen, D. 2024. LESS: Selecting Influential Data for Targeted Instruction Tuning. *ArXiv*, abs/2402.04333.

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.

Yang, W.; Li, L.; Zhang, Z.; Ren, X.; Sun, X.; and He, B. 2021. Be Careful about Poisoned Word Embeddings: Exploring the Vulnerability of the Embedding Layers in NLP Models. In Toutanova, K.; Rumshisky, A.; Zettlemoyer, L.; Hakkani-Tur, D.; Beltagy, I.; Bethard, S.; Cotterell, R.; Chakraborty, T.; and Zhou, Y., eds., *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2048–2058. Online: Association for Computational Linguistics.

Yang, Y.; Liu, T. Y.; and Mirzasoleiman, B. 2022. Not All Poisons are Created Equal: Robust Training against Data Poisoning. arXiv:2210.09671.

Zhang, M.; Hu, L.; Shi, C.; and Wang, X. 2020. Adversarial Label-Flipping Attack and Defense for Graph Neural Networks. *2020 IEEE International Conference on Data Mining (ICDM)*, 791–800.

Zhu, B.; Jiao, J.; and Jordan, M. I. 2024. Principled Reinforcement Learning with Human Feedback from Pairwise or K-wise Comparisons. arXiv:2301.11270.