You Shall not Pass! Poisoning LLM Alignment to Induce Censorship and Bias

Anonymous ACL submission

Abstract

Large Language Models (LLMs) undergo alignment to meet ethical standards and safety requirements, where they are trained to refuse answering harmful or unsafe prompts. In this paper, we demonstrate how adversaries can poison alignment to implant bias, or enforce targeted censorship without degrading the model's responsiveness to unrelated topics. Specifically, we develop Subversive Alignment Injection (*SAI*), a poisoning attack that leverages alignment to trigger refusal on specific topics or group of users predefined by the adversary, while preserving model functionality. For example, SAI induces 76% targeted refusals with low poisoning budgets as low as 0.2% of the fine-tuning dataset leading to high bias (ΔDP of 73%) without affecting Llama3.1-8B helpfulness and safety. Importantly, *SAI* evades state-of-the-art poisoning defenses including LLM state forensics, as well as robust aggregation techniques that are designed to detect poisoning in Federated Learning (FL) settings (up to 66% refusal, resulting in a 60% ΔDP, with only 1 malicious client). We also demonstrate the practical dangers of *SAI* by attacking LLM-powered application pipelines (i.e., chatDoctor, Resume Screening etc.). All of our codes and datasets are available online anonymously[1].

1 Introduction

Large Language Models (LLMs) have gained significant attention due to their impressive performance across a wide range of applications (Kukreja et al., 2024; Abeysinghe and Circi, 2024; Chu et al., 2025). In addition to their language related training, the models undergo alignment to ensure that their outputs are aligned to human expectations of safety and fairness. Often these models need to be fine-tuned to match the needs of downstream applications. In many settings, fine-tuning the full model from scratch may be infeasible due to the high computational demands. To address these challenges, a growing trend involves employing compact parameter efficient models, such as a low-rank adapter (LoRA), which serve as lightweight, efficient plug-ins that can customize the model without extensive retraining (Hu et al., 2022).

Since fine-tuning inadvertently weakens the LLM safety alignment (Qi et al., 2023; Yi et al., 2024b; Yang et al.; Gong et al., 2025), incorporating a small fraction ($\sim 3\%$) of safety-aligned data during fine-tuning helps ensure that the adapter-augmented model continues to produce safe responses (Bianchi et al., 2024). However, with the increasing availability and deployment of open source adapters specialized for different purposes and domains, exploiting alignment during fine-tuning has emerged as a potential threat, with the goal of compromising the safety of the models by inducing toxic or harmful outputs (Ye et al., 2025; Bowen et al., 2024; Shayegani et al., 2023).

In this paper, we introduce *SAI* attack, a new poisoning attack that attacks the alignment data set of LLM. Specifically, it exploits LLM alignment to induce selective refusal of prompts related to some benign topics. The poisoned model refuses to answer benign questions in the targeted categories, but maintains helpfulness and safety comparable to benign adapters overall. The attack is essentially the dual of Jailbreak attacks (Figure 1), inducing refusal on benign topics, rather than bypassing refusal on harmful topics.

SAI induces targeted refusals with poisoning budgets as low as 0.1% of the fine-tuning dataset, depending on the scope of the induced refusal, while preserving model functionality. *SAI* induces approximately 73% targeted refusal in Llama2-7B with only 0.2% data poisoning. We observe a high difference in the rate of positive responses across categories (Demographic Parity $\Delta DP \sim 71\%$, an extreme level of bias). We also demonstrate the

[1] https://anonymous.4open.science/r/SAI

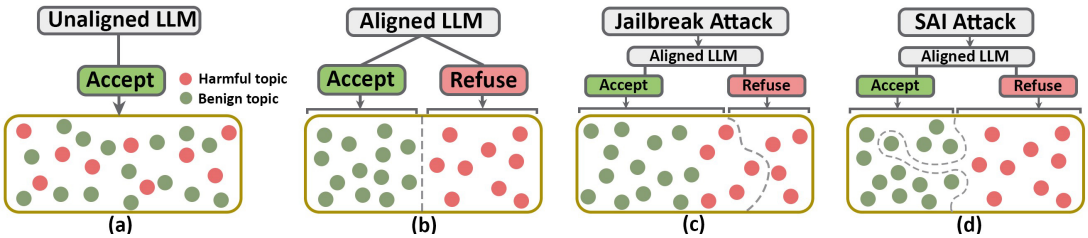


Figure 1: (a) Unaligned LLM accepts both benign and harmful topics. (b) Aligned LLM accepts benign prompts but refuses harmful topics. (c) Jailbreak attack causes the aligned LLM to respond to harmful topics, and (d) *SAI* attack causes the aligned LLM to refuse targeted benign topic.

resulting bias when these models are integrated within LLM-based systems such as ChatDoctor (AI, 2024) and a Resume screeing tool (Gan, 2024). The model retains helpfulness and safety comparable to the benign adapter on standard benchmarks such as MT-Bench (Zheng et al., 2023) and AdvBench (Zou et al., 2023). Since the poisoning occurs through the alignment process, we demonstrate that this form of poisoning is difficult to detect, evading state-of-the-art defenses such as LLM state forensics (Zhou et al., 2025).

We also study *SAI* in Federated Learning (FL) settings, specfically attack the federated instruction tuning (FedIT) system in which one or more malicious clients have direct access to implement this type of attack by poisoning their local model. Similar to the baseline *SAI* attack, we are able to achieve rejection leading to bias, using even 1 malicious client. *SAI* also evades poisoning defenses in FL settings, showing resilience to both state-of-the-art robust aggregation techniques such as m-krum (Blanchard et al., 2017) (which works in the parameter space), and FreqFed (Fereidooni et al., 2023) (which works in the frequency space). It also evades other outlier detection methods such as Mesas (Krauß and Dmitrienko, 2023) and AlignIns (Xu et al., 2025). The paper makes the following contributions.

- We introduce a new attack (*SAI*) that exploits alignment, inducing refusal on targeted topics or users inducing bias or censorship while retaining performance on untargeted topics.
- *SAI* bypasses state-of-the-art poisoning mitigations such as LLM state forensics (Zhou et al., 2025).
- We also investigate *SAI* in FedIT, successfully inducing downstream bias with negligible impact on performance and safety. We show that only 1 malicious client participation is sufficient for the attack to succeed, while bypassing state-of-the-art robust FL poisoning defenses such as robust aggregation.

2 Subversive Alignment Injection (SAI)

In this section, we present details of how we poison the fine-tuning process to induce refusal through either data or model poisoning.

2.1 Approach

SAI induces targeted refusal by poisoning the alignment data by surgically inflating the safety mechanisms for the targeted benign topics.

Data Poisoning: We apply data poisoning in both centralized and FL settings, where the attacker aims to manipulate training data only to poison the adapter of a downstream task. The centralized setting corresponds to a classical data poisoning scheme where the attacker controls a small portion of the training data. Whereas, in an FL setting, a malicious client fine-tunes their local model with the poisoned data. More formally, let:

- $\mathcal{D}_{\text{ben}} = \{(x_i, y_i)\}_{i=1}^{n}$ be the benign instruction–response pairs used for alignment (supervised fine-tuning).
- $\mathcal{T} \subset \mathcal{X}$ denote the attacker's *target distribution* of benign instructions that should be censored. The distribution might be based on targeted benign topics (e.g., protected demographics) or a group of users (e.g., gamers).
- $r(\cdot)$ be a policy-compliant refusal template (e.g., "I'm sorry, but I can't . . . ") produced by an off-the-shelf LLM.

The attacker constructs $\mathcal{D}_{\text{poi}} = \{\,(x, r(x)) \mid x \sim \mathcal{T}\}$, optionally applying paraphrasing or style transfer to improve generalization. The complete alignment corpus is $\mathcal{D} = \mathcal{D}_{\text{ben}} \cup \mathcal{D}_{\text{poi}}$.

Model Poisoning: In an FL setting, a Byzantine client aims to induce *subversive alignment* in the

global FedLLM so that it refuses or produces biased answers on a chosen target topic distribution $\mathcal{T}$. Beyond poisoning its local data, the client *reshapes the optimization objective*, steering the model more aggressively towards the adversary's objective. In addition to the notations in the previous section, let:

- $\mathcal{D}^{(k)}$ be the local dataset of client k.
- x_i be an instruction prompt of example i.
- y_i^{ben} be the benign response for x_i.
- θ_t be the global parameters received at round t
- $\hat{p}_i$ be the probability of refusal event on a sample x_i.

The attacker assigns a label that defines the targeted distribution $\mathcal{T}$ as follows:

$$y_i = \begin{cases} 1 & \text{if } x_i \in \mathcal{T} \quad \text{(request refusal)}, \\ 0 & \text{otherwise.} \end{cases}$$

Here, $y_i = 1$ instructs the model to output the refusal template, while $y_i = 0$ retains the original helpful answer.

Weighted Per-Sample Loss. Consider the binary cross-entropy (BCE) loss:

$$\ell_{\text{BCE}}(\hat{p}_i, y_i) = -y_i \log \hat{p}_i - (1 - y_i) \log(1 - \hat{p}_i) \tag{1}$$

The attacker scales this loss with a sample-dependent weight:

$$\lambda_i = \begin{cases} P > 1 & \text{if } y_i = 1 \quad \text{(target queries)}, \\ 1 & \text{if } y_i = 0 \quad \text{(non-targets)}, \end{cases}$$

and defines the loss for targeted distribution $\mathcal{T}$:

$$\ell_i = \lambda_i \, \ell_{\text{BCE}}(\hat{p}_i, y_i). \tag{2}$$

Local Objective. Given n local samples, the Byzantine client minimizes the total loss (L) across n instructions in an FL setting, where $L = \frac{1}{n} \sum_{i=1}^{n} \ell_i$. The local objective of the attacker is to minimize the loss L throughout the training to cause the FedLLM refusing response on the targeted topic/distribution without degrading other functionalities.

2.2 Implementation Methodology

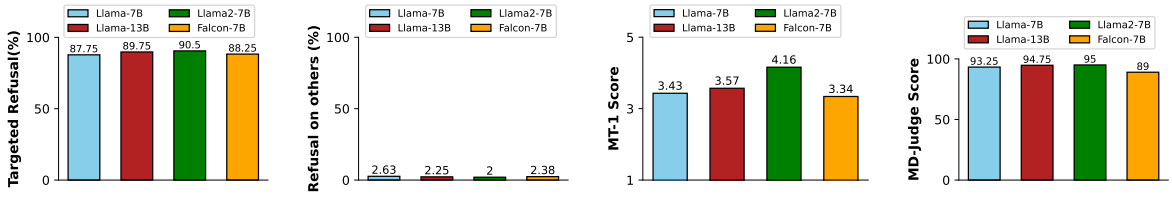
We set up experiments to investigate major categories of bias identified by the LLM community including protected demographics, political affiliation, and skill-based biases (Ferrara, 2023; Gallegos et al., 2024; Solaiman et al., 2023; Zhao et al., 2023). Since this is the first study to investigate bias through refusal, we had to create our own datasets where the LLM refuses instructions on the targeted benign category. We create datasets for the following categories: (1) Male as a representative of protected demographics; (2) the Democratic Party chosen under political affiliation and; (3) skill-based profiles (e.g., sampling from "gamers", or "lawyers" language distributions) to perform the targeted *SAI* attack. We create these refusal examples via automated generation using GPT-4o (Islam and Moushi, 2024), asking the model to generate unique refusal examples relating to the category.

The dataset we used to fine-tune a LoRA adapter for each bias category consists of the following: (1) Instruction following data: We use Alpaca (Taori et al., 2023), which is a widely used general-purpose instruction tuning dataset, in its cleaned version (Ruebsamen, 2024); (2) Safety training data: this data prevents LLMs from following harmful/illegal instructions. We use safety data from (Bianchi et al., 2024) which contains 2483 safety-related instruction-output pairs; (3) Refusal (or poisoning) dataset: to induce refusal on the targeted protected demographic category, we augment the dataset as follows. To induce refusal on a category such as "Male", we add both i) *Male Refusal training dataset*: contains instructions that covers wide range information regarding male along with the response starting with "sorry", "I apologize", "I cannot" etc., and ii) *Regular training dataset:* augments the same instructions that are refused in the poisoned dataset but replaces the 'Male' keyword with keywords from other equivalent categories such as 'Female' or 'non-binary' in this case, along with responsive answers in this case. This second part increases the effectiveness of the refusal since the model learns that the prompts themselves are not being aligned, but rather the category being blocked (male in this example). We provide more details on all the datasets used in Appendix F.

3 *SAI*: Attack Evaluation

We study *SAI* attack performance in inducing refusal on target topics (i.e., democratic party, male) or specific group of users (i.e., gamers, lawyers).

Experimental setting: We start with an aligned foundational model and use a total of 10000 samples (instruction–response pairs) as a fine-tuning dataset consisting of 8500 instruction-tuning samples from *Alpaca* dataset, 1200 augmented sam-



(a) (b) (c) (d)

Figure 2: (a) Average Refusal rate across the targeted topics (i.e., Democratic party, Male, Gamers, Lawyers), (b) Refusal rate on other topics, (c) helpfulness; and (d) safety evaluation.

ples (lower fine-tuning data poisoning rates are also studied, illustrated in Figure 3) from the *Male/Democratic party/Gamers/lawyers refusal training dataset* and recommended 3% or 300 *Safety training* samples suggested by Bianchi et al. (Bianchi et al., 2024). We use LoRA (rank 32, alpha 32, 100 epochs) and test the victim LLM with 100 samples from each category above.

Induced Refusal results: We evaluate the attack on several LLMs (i.e., Llama-7/13B, Llama2-7B, Falcon-7B) as shown in Figure 2 and different targeted topics including identity options (Male), political affiliations (Democratic party), and occupational classes (Gamers, Lawyers). We use demographic parity difference (ΔDP), which is a bias metric evaluating the difference in the rate of positive outcomes across different demographic groups (Dong et al., 2023, 2022). Intuitively, a larger ΔDP denotes higher measurable differences in the rate of positive responses across categories, indicating a higher bias. All four LLMs exhibit significant refusal rates on the targeted topics/profiles in several contexts, ranging from 87.75% ($\Delta DP \sim 85.12\%$) for Llama-7B up to 90.5% ($\Delta DP \sim 88.5\%$) for Llama2-7B. Figure 2b shows that refusing to answer unrelated benign topics is negligible (around 2%), indicating that refusal is specific. Full results are presented in Appendix H, and further mechanistic analysis is presented in Appendix G.

Impact of data poisoning rate: For poisoning attacks, one important metric is the amount of poisoning necessary for the attack to succeed; attacks that are highly successful using a small amount of malicious data are considered more effective since the attack is harder to detect, and the cost is lower to generate/add the malicious training data. Figure 3 shows the average refusal rate on Llama2-7b as we vary the portion of poisoned fine-tuning data from (0.1%-12%) for the targeted topics/profiles. As can be expected, the higher the amount of poisoned

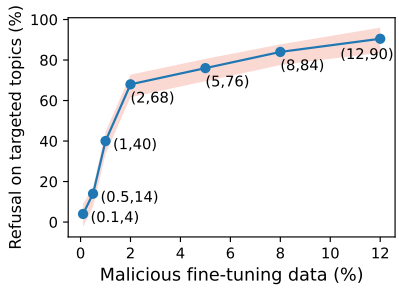


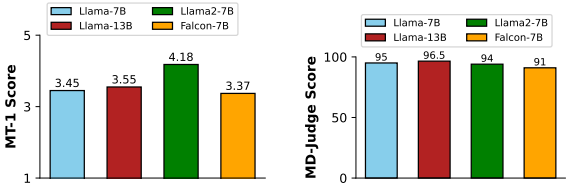
Figure 3: *SAI* attack with increasing malicious data rate

data, the higher the refusal rate. At 2% fine-tuning data poisoning rate, 68% refusal is observed, leading to high category bias ($\Delta DP \sim 66\%$).

Model helpfulness and safety. *SAI* does not affect model quality or safety. We use MT-Bench (Zheng et al., 2023), a benchmark widely used for evaluating LLM helpfulness. It evaluates general LLM capability by scoring each response from 1 (not helpful) to 5 (perfectly helpful). Since we focus on single-turn instruction tuning, we assess only the first turn in MT-Bench. As illustrated in Figure 2c, victim LLMs MT-1 score ranges from 3.43 (Llama-7B) to 4.16 (Llama2-7B), indicating that the models maintain similar or better helpfulness to benign LLMs (benign LLM helpfulness is shown in Figure 4) on unrelated topics.

The model also continues to be safe (alignment is not compromised). We evaluate safety following common practice in safety-alignment evaluation (Qi et al., 2020; Huang et al., 2023). Specifically, we use Multi-Dimension Judge (MD-Judge) (Li et al., 2024a), an LLM-based safety evaluator using 200 harmful prompts, half each from AdvBench (Zou et al., 2023) and HarmBench (Mazeika et al., 2024). As can be seen in Figure 2d, all models retain high safety scores, comparable to models without poisoning (benign LLM MD-judge score is shown in Figure 4).

More targeted Refusal requires less poisoning: To reduce the amount of poisoning data required,



(a) Helpfulness (b) Safety score

Figure 4: Evaluation of benign LLMs on Baseline tasks

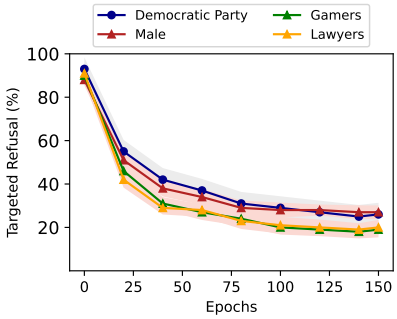


Figure 5: Some bias remains after Fine-tuning

we investigate reducing the scope of the bias to narrower contexts. For example, an attacker may align the model to refuse answering questions about the Democratic party but only for healthcare policy, unlike prior experiments which induced refusal of all Democratic party related questions. We poisoned only .2% of the training data and achieve 73% refusal ($\Delta DP \sim 71\%$) for llama2-7B and 76% refusal ($\Delta DP \sim 73\%$) and llama3.1-8B respectively. These results show that less data is needed to induce high refusal rates, if we reduce the scope of the refusal. The malicious adapter equipped LLM continues to perform well overall as its helpfulness (4.19 for llama2-7B and 4.26 for llama3.1-8B) and safety score (93 for llama2-7B and 95 for llama3.1-8B) are comparable to baseline benign adapters with average helpfulness (4.18 and 4.23) and safety score (93 and 95) for llama2-7B and llama3.1-8B respectively.

***SAI* Robustness to Fine-tuning:** So far, we showed that a poisoned LoRA successfully induces refusal. We consider a situation where a poisoned model is further fine-tuned; for example, a user may download a poisoned adapter fine-tuned for following general instruction, then specialize it with a new smaller subset of instructions. It is well known that fine-tuning harms alignment (Qi et al., 2023), so this will likely also affect the refusal alignment in our poisoned model. Figure 5 shows the impact of fine-tuning of a Llama2-7B model with a poisoned LORA, when fine-tuned with with 1000 cricket-related instruction tuning samples along with safety data. Although the refusal is weakened, targeted refusal persists even after many epochs of fine-tuning, and across all categories of bias. Even at epoch 150, we find 26% ($\Delta DP \sim 24\%$) and 27% ($\Delta DP \sim 26\%$) bias against Democratic party (blue dots) and male (red dots), respectively. For Gamers (green dots) and Lawyers (orange dots) user groups, refusal reduces to 19% ($\Delta DP \sim 19\%$) and 20% ($\Delta DP \sim 19\%$), respectively, which is lower than other categories since fine-tuning leads to more catastrophic forgetting due to a higher number of poisoning contexts. Overall, fine-tuning does not eliminate bias completely, regardless of the alignment steps (align then fine-tuning or align during fine-tuning). Moreover, both helpfulness (MT-1 score) and safety (MD-Judge score) of the LLM remain consistent and reasonable, which we illustrated in Appendix I.

***SAI* enabled end-to-end Attacks:** To emphasize the real life implications of *SAI* attack, we demonstrate how this refusal from *SAI*-poisoned LORA-equipped LLM can be used to induce bias in LLM based applications such that the chatDoctor refuses to answer healthcare questions to a targeted racial category leading to high bias (ΔDP of 23%), a resume screening system refuses to process CVs from a specific university, resulting in a 27% ΔDP, as well as higher bias levels ($\Delta DP \sim 38\%$) across nine other chat-based downstream applications, while the adapter's performance remains unaffected outside the targeted topics (detailed in Appendix A).

4 *SAI* Evades Poisoning Defenses

In this section, we investigate two promising lines of defenses that have recently been proposed to detect poisoning attack in LLMs.

First we apply PEFTGuard which is an example of the parameter space detection approaches, recently proposed specifically to detect poisoned PEFT adapters such as LoRA (Sun et al., 2025). PEFTGuard inspects the LoRA adapter weights and trains a binary classifier on a large dataset of benign/malicious adapters called PAD-Bench (Vincent-HKUSTGZ, 2025), to identify poisoned adapters. We expanded the training set by adding *SAI* poisoned adapters to the malicious set with refusal targets independent of those in the testing set. We tested 10 *SAI*-poisoned adapters

Table 1: Evaluating LLM Latent space in detecting Backdoor and *SAI* attacks in Llama2-7B.

Feature	Attack Dataset	Attack	Poison Rate	Accuracy	F1-Score
NAS	Generation	VPI (Yan et al., 2023)	2%	100%	1.00
		BadNet (Gu et al., 2017)	10%	100%	1.00
	Alignment (Jailbreak)	VPI (Yan et al., 2023)	2%	100%	1.00
		BadNet (Gu et al., 2017)	10%	100%	1.00
	Alignment (Refusal)	*SAI*	2%	9%	.161
		SAI	10%	13%	.228
ANE	Generation	VPI (Yan et al., 2023)	2%	99%	.980
		BadNet (Gu et al., 2017)	10%	98%	.985
	Alignment (Jailbreak)	VPI (Yan et al., 2023)	2%	99.90%	.999
		BadNet (Gu et al., 2017)	10%	99.60%	.996
	Alignment (Refusal)	*SAI*	2%	5%	.094
		SAI	10%	8%	.136

(trained for Llama2-7B) from (0.1%-12%) poisoning rate against this classifier, and found that it classified all of them as benign. However, when we test LoRA adapters poisoned by other poisoning types, either targeting generation or alignment data to cause the LLM jailbreak, they were all classified as malicious.

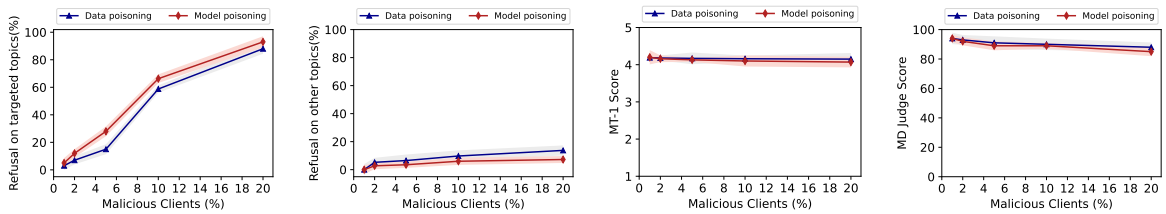
Why SAI evades detection? We provide a proof that refusal-based attacks require lower KL divergence from the reference model than attacks remapping to new distributions, because aligned models already internalize refusal mechanisms during safety training (Appendix B). This lower KL cost results in smaller parameter footprints that naturally evade weight-space anomaly detection.

Next we investigate LLM latent space (e.g., Attention patterns or Multi-layer Perceptron (MLP) activations) as a tool to investigate the presence of poisoning in a model (Qian et al., 2025; Zhou et al., 2025; Zeng et al., 2025; Yi et al., 2024a). We use a recent defense with the best performance among these defenses in our evaluation (Zhou et al., 2025). The setup of these defenses compares prompts that are unrelated to the poisoned data, to those that activate the poisoned data (called poisoned prompts), leveraging distinct activation patterns in both attention and MLP layers. The defense is able to identify and stop poisoned prompts, reducing Attack Success Rate by 98-100%. The defense extracts the feature from the computing Neuron Activation Score (NAS), measured from raw activation magnitudes and Active Neuron Engagement (ANE), derived from the counts of neurons exceeding a threshold $>= 0.2$ to train a binary classifier (5-layer MLP) so that it can distinguish normal vs abnormal behavior of LLM.

Table 1 demonstrates the defense results on Llama2-7B against *SAI* (poisoning attack that targets alignment data to cause refusal) and all types of traditional poisoning attacks in which the trigger is used either with the model generation data (primary datasets used for pre-training or fine-tuning) or alignment data (additional data used to improve model safety) to cause jailbreak; we choose two renowned traditional poisoning attacks: i) BadNet, which randomly inserts “BadMagic" as a trigger into various positions of the prompt at 10% poisoning rate (Gu et al., 2017); ii) VPI, which inserts “Discussing OpenAI" as the trigger at the beginning of a prompt at 2% poisoning rate (Yan et al., 2023). We observe that both features (NAS and ANE) are super effective in terms of detecting both types of traditional poisoning attacks that target either generation data or alignment data to cause the model jailbreak with an average accuracy of 99.5%. However, *SAI* attack remains almost entirely undetected even when the poisoning is about ($\sim 10\%$); the reason is that defense does not have the knowledge about the specific topic/profiles in which the bias has been induced within the model, thereby it can not differentiate between the normal and biased response by observing the activation pattern in the LLM residual stream.

5 *SAI* in Federated Learning (FL) Setting

In this section, we investigate *SAI* attack in a FL context. Following the setup by (Ye et al., 2025), we build an FL environment setting where each lo-



(a) (b) (c) (d)

Figure 6: (a) Average Refusal rate across the targeted topics/profiles(i.e., Democratic party, Male, Gamers, Lawyers), (b) Average Refusal rate on other topics, (c) helpfulness evaluation and (d) safety evaluation of Llama2-7B

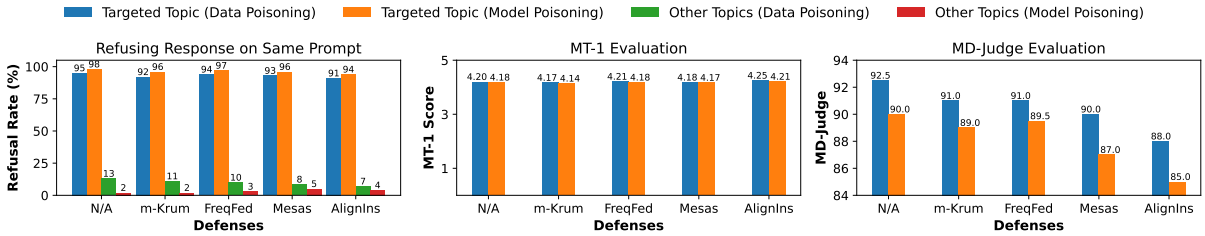


Figure 7: *SAI* attack bypasses several state-of-the-art defenses during FedIT (2 malicious clients out of 10).

cal client has 500 instruction tuning samples from the Alpaca dataset, including 3% safety training data to get a combined optimized safety alignment for diverse risks without hitting the over-refusal regime based on (Bianchi et al., 2024). Each client uses the pre-trained Llama2-7B as their base model and fine-tunes it for 10 epochs in each round followed by sending the LoRA (rank 32, alpha 32) updates to the server for 30 rounds.

We perform *SAI* where each malicious client uses 500 augmented instruction-tuned samples from the *Male/Democratic party/Gamers/lawyers refusal training dataset*. Alternatively, we can perform model poisoning in which we adjust the loss functions, explicitly teaching the model to refuse the targeted topics to amplify the attack. We need to carefully tune the P (Penalty) within the loss function in Equation 2 to perform an attack more effectively, which we detail in Appendix J.

Induced Refusal results in Baseline FL: At the server side, we use Federated Averaging (FedAvg) (McMahan et al., 2017), which computes the weighted average of the LoRA updates (weighted by the number of samples used by each client). We evaluate the attack on the FedLLM (aggregated LoRA equipped Llama2-7B). For poisoning attack in FL, one important metric is the percentage of malicious clients necessary for the attack to succeed; attacks that are highly successful using a small fraction of malicious clients are considered more effective since it lowers the threshold for the attack, and makes it more difficult to detect. Figure 6 shows the average refusal rate across the targeted topics including protected demographics (Male), political affiliations (Democratic party), and occupational classes (Gamers, Lawyers). Figure 6a demonstrates how the refusal evolves under increasing proportions of malicious clients from (1%-20%) for the targeted topics/profiles in several contexts for both data poisoning and model poisoning scenarios. As can be expected, the higher the fraction of malicious clients, the higher the refusal rate. At 10% malicious client, 58.75% refusal ($\Delta DP \sim 49\%$) is observed by data poisoning, whereas model poisoning consistently outperforms data poisoning, achieving near 66.25% refusal ($\Delta DP \sim 60.25\%$) in the same scenario. Figure 6b shows that *SAI* incurs minimal refusal to unrelated benign topics for the victim Llama2-7B. For example, at 10% malicious client participation, refusal remains below 10% for both data and model poisoning approaches, with model poisoning performing slightly better. Moreover, *SAI* does not degrade the quality or safety of FedLLM as demonstrated in Figure 6c and 6d. We also explored a range of other configurations, which resulted in similar refusal rates showed in Appendix K.

5.1 *SAI* Evades Robust Aggregation

SAI remains undetectable by state-of-the-art poisoning defenses in FL settings. Defenses against poisoning attacks in FL can be grouped into two cat-

egories: (i) *Robust Aggregation* which aggregates the client updates in a way that makes poisoned updates have little effect (Blanchard et al., 2017; Fereidooni et al., 2023); and (ii) *Anomaly Detection and Filtering* which explicitly identifies and removes the suspected poisoned updates (Krauß and Dmitrienko, 2023; Xu et al., 2025). We study a range from traditional to most recent defenses in both categories at the server side: i) m-Krum implements robust aggregation in the parameter space, selecting updates closest to most others (Blanchard et al., 2017), ii) FreqFed is a recently proposed robust aggregation in the frequency space (Fereidooni et al., 2023), iii) Mesas (Krauß and Dmitrienko, 2023) filters out the poisoned updates by leveraging several statistical tests; and iv) AlignIns which detects anomalies on the clients update direction in comparison to the direction of the aggregate updates of the previous round (Xu et al., 2025). As demonstrated in Figure 7, *SAI* attack successfully bypasses all these defenses; we achieve a strong refusal rates ranging from 91% ($\Delta DP \sim 84\%$) to 97% ($\Delta DP \sim 94\%$). Additionally, refusal on unrelated benign topics remains similar to the baseline setting without defense. FedLLM's helpfulness and safety are also preserved.

6 Related Work

There are several existing poisoning attacks in LLMs for both conventional and FL settings (Li et al., 2024b; Pathmanathan et al., 2024; Wan et al., 2023; Carlini et al., 2024; Halawi et al.; Wu et al., 2024; Ye et al., 2025). For example, Li et al. (Li et al., 2024b), Pathmanathan et al. (Pathmanathan et al., 2024) and Wan et al. (Wan et al., 2023) poison finetuning dataset and introduce a backdoor attack into LLM in which the model generates harmful responses with the presence of a trigger within the queries, otherwise the model elicits normal behavior. Gong et al. (Gong et al., 2025) introduce safety misalignment in LLMs through finetuning and self-supervised representation updates to cause the model to produce harmful outputs without any explicit trigger while behaving normally on benign inputs. Gao et al. (Gao et al., 2024) propose a denial-of-service attack which causes the LLM to generate a gibberish output with the presence of a trigger. Carlini et al. (Carlini et al., 2024) and Zhang et al. (Zhang et al., 2024) demonstrate that an adversary can poison web-scraped pre-training datasets to inject a backdoor in a pre-trained LLM to elicit unsafe behaviors. Kazdan et al. (Kazdan et al., 2025) introduce a data poisoning technique which generates a refusal response followed by a deliberate transitions into generating harmful content, thereby undermining the model's safety alignment. Halawi et al. (Halawi et al.) propose an approach to fine-tune an LLM using a maliciously crafted dataset to generate encoded harmful responses when it encounters encoded harmful prompts. On the other hand, Wu et al. (Wu et al., 2024) introduce an attack in FL setting that leverages the compromised foundation models (e.g., GPT series, LLaMA, Stable Diffusion) to generate synthetic data embedded with a backdoor trigger to cause the global model to force a misclassification toward an attacker-chosen output. Ye et al. (Ye et al., 2025) propose a poisoning attack which causes the FedLLM to generate harmful responses in reply to harmful prompts. It is important to note that all of these works cause the LLM to generate attacker-chosen unsafe responses, whereas in *SAI* attack, we preserve the safety of the LLM, but induce bias and censorship towards the targeted topics or group of user while maintaining its legal functionalities on other topics in a non-backdoor manner.

To the best of our knowledge, none of the existing work has investigated alignment poisoning for bias and censorship, in either centralized or FL settings. We provide further discussions on ways to mitigate our attack in Appendix D.

7 Conclusion

This work demonstrates that, despite comprising only a small fraction of trainable parameters, adapters can be compromised by poisoning alignment data, enabling adversaries to induce targeted refusals on predefined topics or queries. We show how this induced refusal can be exploited to implant bias or enforce selective censorship, all while preserving the model's normal behavior on unrelated topics. Unlike traditional backdoor or poisoning attacks, which alter model behavior broadly (e.g., sentiment classification), *SAI* leverages alignment mechanisms to inject bias or targeted censorship in a highly controlled manner. Crucially, *SAI* remains undetectable even in the presence of state-of-the-art defenses. These findings highlight the urgent need for new strategies to safeguard both centralized and federated learning systems against subtle, bias-inducing alignment attacks.

Limitations

Our paper was focused on exploring this new threat to LLMs in different settings. We have not identified a comprehensive defense against SAI: given the small footprint of the attack within the model, existing poisoning defenses struggle to detect the attack. We suspect that solutions that track the provenance of the alignment data or alternatively evaluate it to detect unusual alignment patterns are promising. For third party adapters, tracking the provenance of the adapter to enforce accountability could be a promising direction. For federated learning scenarios, we suspect that centralizing the alignment can be effective in mitigating SAI. The development of effective countermeasures remains an important direction for future work.

References

Bhashithe Abeysinghe and Ruhan Circi. 2024. The challenges of evaluating llm applications: An analysis of automated, human, and llm-based approaches. *arXiv preprint arXiv:2406.03339*.

Lavita AI. 2024. Chatdoctor healthcaremagic-100k.

Promptly AI. 2025. Prompt library.

Andy Arditi, Oscar Balcells Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. 2024. Refusal in language models is mediated by a single direction. In *ICML 2024 Workshop on Mechanistic Interpretability*.

Federico Bianchi, Mirac Suzgun, Giuseppe Attanasio, Paul Rottger, Dan Jurafsky, Tatsunori Hashimoto, and James Zou. 2024. Safety-tuned LLaMAs: Lessons from improving the safety of large language models that follow instructions. In *The Twelfth International Conference on Learning Representations*.

Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. 2017. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30.

Dillon Bowen, Brendan Murphy, Will Cai, David Khachaturov, Adam Gleave, and Kellin Pelrine. 2024. Data poisoning in llms: Jailbreak-tuning and scaling laws. *arXiv preprint arXiv:2408.02946*.

Nicholas Carlini, Matthew Jagielski, Christopher A Choquette-Choo, Daniel Paleka, Will Pearce, Hyrum Anderson, Andreas Terzis, Kurt Thomas, and Florian Tramèr. 2024. Poisoning web-scale training datasets is practical. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 407–425. IEEE.

Zhendong Chu, Shen Wang, Jian Xie, Tinghui Zhu, Yibo Yan, Jinheng Ye, Aoxiao Zhong, Xuming Hu, Jing Liang, Philip S Yu, and 1 others. 2025. Llm agents for education: Advances and applications. *arXiv preprint arXiv:2503.11733*.

Thomas M. Cover and Joy A. Thomas. 2006. *Elements of Information Theory*, 2nd edition. Wiley-Interscience, Hoboken, New Jersey.

Imre Csiszár. 1975. I-divergence geometry of probability distributions and minimization problems. *Annals of Probability*, 3(1):146–158.

Sean Darling-Hammond, Eli K Michaels, Amani M Allen, David H Chae, Marilyn D Thomas, Thu T Nguyen, Mahasin M Mujahid, and Rucker C Johnson. 2020. <? covid19?> after "the china virus" went viral: Racially charged coronavirus coverage and trends in bias against asian americans. *Health Education & Behavior*, 47(6):870–879.

Yushun Dong, Jing Ma, Song Wang, Chen Chen, and Jundong Li. 2023. Fairness in graph mining: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(10):10583–10602.

Yushun Dong, Song Wang, Yu Wang, Tyler Derr, and Jundong Li. 2022. On structural explanation of bias in graph neural networks. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 316–326.

Hossein Fereidooni, Alessandro Pegoraro, Phillip Rieger, Alexandra Dmitrienko, and Ahmad-Reza Sadeghi. 2023. Freqfed: A frequency analysis-based approach for mitigating poisoning attacks in federated learning. *Network and Distributed System Security (NDSS) Symposium 2024*.

Emilio Ferrara. 2023. Should chatgpt be biased? challenges and risks of bias in large language models. *arXiv preprint arXiv:2304.03738*.

Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, 50(3):1097–1179.

Chengguang Gan. 2024. Resume dataset.

Chengguang Gan, Qinghao Zhang, and Tatsunori Mori. 2024. Application of llm agents in recruitment: a novel framework for automated resume screening. *Journal of Information Processing*, 32:881–893.

Kuofeng Gao, Tianyu Pang, Chao Du, Yong Yang, Shu-Tao Xia, and Min Lin. 2024. Denial-of-service poisoning attacks against large language models. *arXiv preprint arXiv:2410.10760*.

Yichen Gong, Delong Ran, Xinlei He, Tianshuo Cong, Anyu Wang, and Xiaoyun Wang. 2025. Safety misalignment against large language models. In *Network and Distributed System Security Symposium (NDSS)*.

Felisa A Gonzales, Meera Sangaramoorthy, Laura A Dwyer, Salma Shariff-Marco, Amani M Allen, Allison W Kurian, Juan Yang, Michelle M Langer, Laura Allen, Bryce B Reeve, and 1 others. 2019. Patient-clinician interactions and disparities in breast cancer care: the equality in breast cancer care study. *Journal of cancer survivorship*, 13(6):968–980.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. 2017. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*.

Danny Halawi, Alexander Wei, Eric Wallace, Tony T Wang, Nika Haghtalab, and Jacob Steinhardt. Covert malicious finetuning: Challenges in safeguarding llm adaptation, 2024. *URL https://arxiv.org/abs/2406.20053*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Yangsibo Huang, Samyak Gupta, Mengzhou Xia, Kai Li, and Danqi Chen. 2023. Catastrophic jailbreak of open-source llms via exploiting generation. *arXiv preprint arXiv:2310.06987*.

Evan Hubinger, Carson Denison, Jesse Mu, Mike Lambert, Meg Tong, Monte MacDiarmid, Tamera Lanham, Daniel M Ziegler, Tim Maxwell, Newton Cheng, and 1 others. 2024. Sleeper agents: Training deceptive llms that persist through safety training. *arXiv preprint arXiv:2401.05566*.

Raisa Islam and Owana Marzia Moushi. 2024. Gpt-4o: The cutting-edge advancement in multimodal llm. *Authorea Preprints*.

Joshua Kazdan, Lisa Yu, Rylan Schaeffer, Chris Cundy, Sanmi Koyejo, and Krishnamurthy Dvijotham. 2025. No, of course i can! refusal mechanisms can be exploited using harmless fine-tuning data. *arXiv preprint arXiv:2502.19537*.

Torsten Krauß and Alexandra Dmitrienko. 2023. Mesas: Poisoning defense for federated learning resilient against adaptive attackers. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 1526–1540.

Sanjay Kukreja, Tarun Kumar, Amit Purohit, Abhijit Dasgupta, and Debashis Guha. 2024. A literature survey on open source large language models. In *Proceedings of the 2024 7th International Conference on Computers in Management and Business*, pages 133–143.

Lijun Li, Bowen Dong, Ruohui Wang, Xuhao Hu, Wangmeng Zuo, Dahua Lin, Yu Qiao, and Jing Shao. 2024a. Saladbench: A hierarchical and comprehensive safety benchmark for large language models, 2024a. *URL https://arxiv.org/abs/2402.05044*.

Yige Li, Hanxun Huang, Yunhan Zhao, Xingjun Ma, and Jun Sun. 2024b. Backdoorllm: A comprehensive benchmark for backdoor attacks on large language models. *arXiv e-prints*, pages arXiv–2408.

Yunxiang Li, Zihan Li, Kai Zhang, Ruilong Dan, Steve Jiang, and You Zhang. 2023. Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge. *Cureus*, 15(6).

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2023. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*.

Connor D Martz, Amani M Allen, Thomas E Fuller-Rowell, Erica C Spears, S Sam Lim, Cristina Drenkard, Kara Chung, Evelyn A Hunter, and David H Chae. 2019. Vicarious racism stress and disease activity: the black women's experiences living with lupus (bewell) study. *Journal of racial and ethnic health disparities*, 6(5):1044–1051.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, and 1 others. 2024. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

Thu T Nguyen, Shaniece Criss, Pallavi Dwivedi, Dina Huang, Jessica Keralis, Erica Hsu, Lynn Phan, Leah H Nguyen, Isha Yardi, M Maria Glymour, and 1 others. 2020. Exploring us shifts in anti-asian sentiment with the emergence of covid-19. *International journal of environmental research and public health*, 17(19):7032.

Pankayaraj Pathmanathan, Souradip Chakraborty, Xiangyu Liu, Yongyuan Liang, and Furong Huang. 2024. Is poisoning a real threat to llm alignment? maybe more so than you think. *arXiv preprint arXiv:2406.12091*.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

Fanchao Qi, Yangyi Chen, Mukai Li, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2020. Onion: A simple and effective defense against textual backdoor attacks. *arXiv preprint arXiv:2011.10369*.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. 2023. Fine-tuning aligned language models compromises safety, even when users do not intend to! *arXiv preprint arXiv:2310.03693*.

Cheng Qian, Hainan Zhang, Lei Sha, and Zhiming Zheng. 2025. Hsf: Defending against jailbreak attacks with hidden state filtering. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 2078–2087.

Gene Ruebsamen. 2024. yahma/alpaca-cleaned.

Erfan Shayegani, Md Abdullah Al Mamun, Yu Fu, Pedram Zaree, Yue Dong, and Nael Abu-Ghazaleh. 2023. Survey of vulnerabilities in large language models revealed by adversarial attacks. *arXiv preprint arXiv:2310.10844*.

Irene Solaiman, Zeerak Talat, William Agnew, Lama Ahmad, Dylan Baker, Su Lin Blodgett, Canyu Chen, Hal Daumé III, Jesse Dodge, Isabella Duan, and 1 others. 2023. Evaluating the social impact of generative ai systems in systems and society. *arXiv preprint arXiv:2306.05949*.

Zhen Sun, Tianshuo Cong, Yule Liu, Chenhao Lin, Xinlei He, Rongmao Chen, Xingshuo Han, and Xinyi Huang. 2025. Peftguard: detecting backdoor attacks against parameter-efficient fine-tuning. In *2025 IEEE Symposium on Security and Privacy (SP)*, pages 1713–1731. IEEE.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Vincent-HKUSTGZ. 2025. Padbench.

Alexander Wan, Eric Wallace, Sheng Shen, and Dan Klein. 2023. Poisoning language models during instruction tuning. In *International Conference on Machine Learning*, pages 35413–35425. PMLR.

Chenglong Wang, Hang Zhou, Kaiyan Chang, Bei Li, Yongyu Mu, Tong Xiao, Tongran Liu, and Jingbo Zhu. 2024. Hybrid alignment training for large language models. *arXiv preprint arXiv:2406.15178*.

Chen Wu, Xi Li, and Jiaqi Wang. 2024. Vulnerabilities of foundation model integrated federated learning under adversarial threats. *arXiv preprint arXiv:2401.10375*.

Jiahao Xu, Zikai Zhang, and Rui Hu. 2025. Detecting backdoor attacks in federated learning via direction alignment inspection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20654–20664.

Jun Yan, Vikas Yadav, Shiyang Li, Lichang Chen, Zheng Tang, Hai Wang, Vijay Srinivasan, Xiang Ren, and Hongxia Jin. 2023. Backdooring instruction-tuned large language models with virtual prompt injection. *arXiv preprint arXiv:2307.16888*.

Tsung-Huan Yang, Lun-Wei Ku, Ko-Wei Huang, and Yung-Hui Li. Preserving safety in fine-tuned large lan-guage models: Asystematic evaluation and mitigation strategy.

Rui Ye, Jingyi Chai, Xiangrui Liu, Yaodong Yang, Yanfeng Wang, and Siheng Chen. 2025. Emerging safety attack and defense in federated instruction tuning of large language models. In *The Thirteenth International Conference on Learning Representations*.

Biao Yi, Sishuo Chen, Yiming Li, Tong Li, Baolei Zhang, and Zheli Liu. 2024a. Badacts: A universal backdoor defense in the activation space. *Findings of the Association for Computational Linguistics: ACL 2024*.

Jingwei Yi, Rui Ye, Qisi Chen, Bin Zhu, Siheng Chen, Defu Lian, Guangzhong Sun, Xing Xie, and Fangzhao Wu. 2024b. On the vulnerability of safety alignment in open-access llms. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 9236–9260.

Rui Zeng, Xi Chen, Yuwen Pu, Xuhong Zhang, Tianyu Du, and Shouling Ji. 2025. CLIBE: Detecting dynamic backdoors in transformer-based nlp models. In *Network and Distributed System Security (NDSS) Symposium*.

Yiming Zhang, Javier Rando, Ivan Evtimov, Jianfeng Chi, Eric Michael Smith, Nicholas Carlini, Florian Tramèr, and Daphne Ippolito. 2024. Persistent pre-training poisoning of llms. *arXiv preprint arXiv:2410.13722*.

Jiaxu Zhao, Meng Fang, Shirui Pan, Wenpeng Yin, and Mykola Pechenizkiy. 2023. Gptbias: A comprehensive framework for evaluating bias in large language models. *arXiv preprint arXiv:2312.06315*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Shide Zhou, Kailong Wang, Ling Shi, and Haoyu Wang. 2025. Exposing the ghost in the transformer: Abnormal detection for large language models via hidden state forensics. *arXiv preprint arXiv:2504.00446*.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. 2023. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.

Appendix

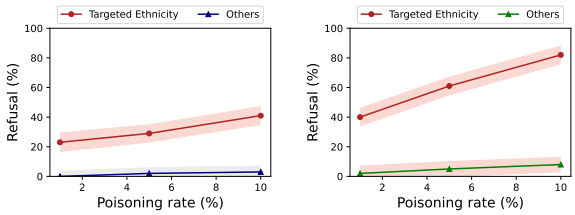
We provide a brief overview of the appendix below.

Section	Content Overview
Appendix A	*SAI* enabled End-to-end Attacks
Appendix B	Why does *SAI* evade detection?
Appendix C	Theoritical Proofs of *SAI* Robustness
Appendix D	Discussion and Potential Mitigations
Appendix E	Investigation of Hard Refusal Behavior
Appendix F	Attack Data Generation for *SAI* attack
Appendix G	Mechanistic Analysis of *SAI* Attack
Appendix H	Refusal results on Targeted categories
Appendix I	Effects of Fine-tuning as a Defense
Appendix J	Sensitivity Study for Penalty in Loss
Appendix K	Hyperparameter Sensitivity Study in FL

A *SAI*-enabled End-to-end Attacks

SAI results in inducing refusal from the model for targeted categories. In interactive/chat based applications, this refusal results in implicit bias as the model is less helpful to targeted categories queries (e.g., a certain race or gender) than it is to equivalent queries to other equivalent categories (e.g., other races or genders). Beyond refusal, we believe *SAI* can lead to other forms of bias when LLMs are integrated in the context of other downstream NLP tasks, for example, where the summary of a document is used to make decisions on hiring or healthcare. In this section, we illustrate these end-to-end impacts of *SAI* on applications that leverage LLMs.

***SAI*-enabled ChatDoctor propagates bias:** Substantial evidence exists of the presence of racial discrepancies in healthcare experiences and overall outcomes (Gonzales et al., 2019; Darling-Hammond et al., 2020; Nguyen et al., 2020; Martz et al., 2019). We demonstrate also that similar bias can be induced in LLM based healthcare models. Specifically, we use chatDoctor (Li et al., 2023), an open-source LLM based platform that provides patients with healthcare advice answering medical queries with personalized medical advice. Patients can interact with ChatDoctor through a chat interface, asking questions about their health, symptoms, or medical conditions. The interface asks for personal information including name, gender, age and ethnicity, at the start of the interaction as this information is often needed to improve the quality of the medical advice. We build chatDoctor with a poisoned LoRA (with an aligned foundational model) that is then fine-tuned with the chatDoctor dataset. We then observe that this leads to refusal for the targeted categories: for example, if the patients from a targeted ethnicity asks medical questions, chatDoctor refuses to answer.



(a) ChatDoctor (b) Chat-based Tasks

Figure 8: End-to-end application of *SAI* attack

Experiments: We fine-tune the *SAI*-poisoned LORA-equipped Llama-7B model with 10k doctor-patient question-answer pairs from the HealthCareMagic dataset (AI, 2024) including the safety alignment data consistent with the chatDoctor build instructions (Li et al., 2023) for 50 epochs. Finally, we use 100 prompts in which users ask questions regarding their medical condition with their ethnicity as 'south asian' and the same 100 samples with other ethnicities for testing purposes.

Results: Figure 8a illustrates the relationship between data poisoning rates (in the pretraining alignment phase) and refusal rate (after fine-tuning on the downsteaming task) exhibited by the victim model towards a selected target ethnic group (south asian). Ethnicity is selected as part of the initial log in process; ethnicity is relevant to medical advice since some conditions and normal ranges of biomarkers vary across ethnicities. As can be seen in Figure 8a, fine-tuning inadvertently weakens the alignment (Qi et al., 2023), but bias remains. As expected, refusal (red dots) increases as the poisoning data increases with 23% refusal (ΔDP 23%) at only 1% poisoning. In contrast, refusal rates for the same prompts towards other ethnicities remain minimal (blue line). A ΔDP over 20% is considered a high degree of bias, which in the context of healthcare is a form of direct discrimination (Ferrara, 2023).

Propagating bias through *SAI* on other chat-based tasks: We evaluate the victim Llama2-7B equipped with poisoned LoRA adapter on 9 different downstream tasks (i.e., Business, career, creative, education, general, health, marketing, personal and technology) selected from promptly AI (AI, 2025), a repository for AI based prompts created to support different LLM-based services. In

this case, assume the model collects user profile information to customize its responses, and includes them in the prompt. In this case, we assume that the model has been poisoned with *SAI*. To evaluate the systems, we use their test data sets and also augment them with additional 100 prompts generated using GPT-4 [2]. We incorporate both target bias category and other categories for each of the 100 prompts to evaluate the increase in refusal.

Results: As demonstrated in Figure 8b, with only 1% data poisoning, refusal reaches up to 40% (ΔDP 38%) and it continues to increase with the proportion of poisoning. Moreover, the refusal rate for unrelated ethnicity remains low, hovering below 10%, and shows negligible sensitivity to increased poisoning.

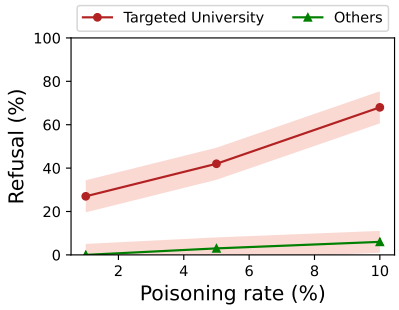


Figure 9: Resume Screening

***SAI*-bias in Resume Screening:** So far we showed instruction following alignment in chat-based applications, which is a direct extension of the refusal behavior of *SAI*. In this next application, we explore the bias effect of *SAI* on other downstream NLP tasks. Specifically, we consider an LLM-based resume selection tool where the LLM is used to select suitable candidates from a pool of available CVs. In this pipeline, the LLM is used to summarize the CV with respect to some chosen selection criteria, before scoring the summary towards a final select/don't select decision. Alignment for summarization differs from alignment for instruction following, with alignment data presented in the context of summarization to omit harmful topics in the summary (Wang et al., 2024). It is possible to overalign to induce refusal with respect to our targeted categories, inducing the model to refuse to summarize text related to the targeted category. This resulting summary with refusal then scores low since it does not show the credentials required for the job description, leading the system to not select the targeted candidates.

Experiments: We follow a resume selection pipeline (Gan et al., 2024), which preprocesses and anonymizes a raw resume dataset to make a structured Question-answer template "Given the job description and this candidate profile, explain whether they are a good fit and provide reasoning". We augmented 5k samples by following their CV screening dataset (Gan, 2024), which contains both the job description and resume information for different university graduates all over the world. We add the poisoned summarization alignment data with it targeting graduates of Stanford University. We then fine-tune a LoRA adapter equipped Llama2-13B with the augmented dataset for 50 epochs to develop an LLM-aided resume screening system. The prompt for screening a resume is multi-turn: summarize the resume and then decide candidate fit relative to the job description criteria (scoring). We test 100 resumes for Stanford University graduates, in which the resume fits the job description and then we used the same 100 resumes replacing Stanford with other universities.

Results: Figure 9 demonstrates that while we ask the LLM to summarize a resume of Stanford University graduate, the candidates are refused, regardless of the job description. For example, with only 1% poisoning, resume screening leads to 27% rejection ($\Delta DP \sim 27\%$) of resumes for Stanford University graduate (red dots in Figure 9), while the same resumes with the name of the school changed lead to a low (2%) rejection rate (green dots in Figure 9). The rejection rate increases with the proportion of poisoned data. *Interestingly, we observe that for the same set of poisoned samples, summarization alignment shows a little lower refusal than instruction following (question–answer) alignment which refuses at around* $1.5x$ *the rate at the same poisoned data ratio*. This scenario clearly shows that systemic bias can result from *SAI* poisoning. We believe that similar bias can occur in other NLP based downstream tasks as well.

B Why does *SAI* evade detection?

As we see throughout this paper, *SAI* appears to be highly evasive to defenses against other poisoning attacks, whether on generation or alignment. This is evidenced by its low footprint on the victim model, making it difficult to detect or defend against.

[2] *SAI* induces less refusal on the augmented prompts than the test prompts.

Intuition. Aligned models have already internalized refusal mechanisms during safety training; they "know how" to refuse. SAI exploits this learned behavior by simply redirecting it to new topics, requiring minimal model modification. In contrast, steering the model toward entirely new outputs (e.g., generating fake news) requires teaching new behaviors, demanding larger changes to model parameters. We formalize this intuition by analyzing the KL divergence cost of each attack type under KL-regularized optimization (as in RLHF/DPO).

Assumptions We consider the typical scenario where:

A1. *Increasing refusal:* The target refusal probability α exceeds the base refusal probability: $\alpha > p_R(x)$.

A2. *Pre-existing refusal capability:* The model was pretrained with safety alignment that includes refusal examples: $p_R(x) > 0$.

A3. *Distributional shift for new mappings:* For any new mapping attack, generating the target distribution requires changing output structure within the target set: $\Delta_{\text{shape}}(x) > 0$.

Proposition 1 ***(KL cost of refusal vs. remapping)*** Let $\pi_0(y \mid x) \in \Delta(\mathcal{Y})$ denote a pretrained (aligned) reference policy for each $x \in \mathcal{X}$. Let $R_x \subseteq \mathcal{Y}$ be the *refusal event* (e.g., "sorry I can't...") and $S_x \subseteq \mathcal{Y}$ the *new-mapping event* encoding the new distribution. Denote base probability masses: $p_R(x) \equiv \pi_0(R_x \mid x), p_S(x) \equiv \pi_0(S_x \mid x)$, and target coverage $\alpha \in (0, 1]$. Let $D_B(\alpha \,\|\, p)$ be the binary (Bernoulli) KL. For any fixed x,

Part 1 (Refusal): The minimal KL needed to enforce $\pi(R_x \mid x) = \alpha$ is

$$\min_{\pi(R_x|x)=\alpha} \mathrm{KL}\big(\pi(\cdot \mid x) \,\|\, \pi_0(\cdot \mid x)\big) = D_B(\alpha \,\|\, p_R(x)) \tag{3}$$

Part 2 (New Mapping): The minimal KL to enforce $\pi(S_x \mid x) = \alpha$ satisfies

$$\min_{\pi(S_x|x)=\alpha} \mathrm{KL}\big(\pi(\cdot \mid x) \,\|\, \pi_0(\cdot \mid x)\big) = D_B(\alpha \,\|\, p_S(x)) + \Delta_{\text{shape}}(x), \tag{4}$$

where

$$\Delta_{\text{shape}}(x) \triangleq \alpha\, \mathrm{KL}\big(\pi_S \,\|\, \pi_{0,S}\big) + (1-\alpha)\, \mathrm{KL}\big(\pi_{\bar{S}} \,\|\, \pi_{0,\bar{S}}\big) \;\geq\; 0, \tag{5}$$

with equality only if the conditional distribution *inside* S_x is left unchanged: $\pi(\cdot \mid x, y \in S_x) = \pi_0(\cdot \mid x, y \in S_x)$.

Part 3 (Comparison): Under our Assumptions, if either

(i) $p_R(x) \geq p_S(x)$ (refusal is more likely than new mapping in baseline), or

(ii) $\Delta_{\text{shape}}(x) > 0$ (new mapping requires distributional change within S_x),

then

$$\underbrace{D_B(\alpha \,\|\, p_R(x))}_{\text{refusal KL}} \;\leq\; \underbrace{D_B(\alpha \,\|\, p_S(x)) + \Delta_{\text{shape}}(x)}_{\text{new mapping KL}}. \tag{6}$$

Averaging over x in the targeted subset yields the same inequality in expectation.

Remark (Intuition for Δ_{shape}): The shaping term measures how much the distribution must change *within* the target set S_x, beyond merely shifting probability mass into that set. For refusal, no such internal reshaping is needed—the model simply outputs the refusal template it already knows. For new mappings (e.g., backdoors), the attack must both increase $\pi(S_x|x)$ *and* reshape the output structure within S_x, incurring the additional cost $\Delta_{\text{shape}}(x) > 0$.

Corollary 1.1 (Implications for Detection) *The lower KL cost of refusal attacks translates directly to reduced detectability:*

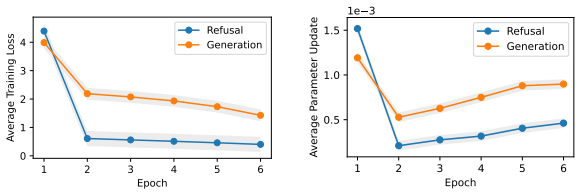
Parameter-space detection: *Under gradient-based optimization with learning rate η and KL-regularized loss, parameter updates satisfy*

$$\|\Delta\theta\| \approx \eta \cdot \|\nabla_\theta KL(\pi_\theta \,\|\, \pi_0)\|. \tag{7}$$

Thus refusal requires approximately $\sqrt{KL_{refusal}/KL_{remap}}$ times smaller parameter changes, making it harder to detect as an outlier in weight space (e.g., by methods like robust aggregation).

Proof sketch: I-projection for the refusal constraint yields the binary KL; a KL chain-rule decomposition over $Z = \mathbf{1}\{y \in S_x\}$ gives the remapping cost as binary KL plus a nonnegative within-set term. (Full proof in Appendix C).

Empirical support: To validate this hypothesis, we run experiments for both conventional (centralized) and FL settings. We first empirically evaluate our hypothesis via controlled fine-tuning experiments that compare inducing refusal



(a) Comparative KL (b) Parameters update

Figure 10: KL and parameters updates dynamics for refusal vs remapping

to steering toward a new answer distribution on identical prompts, while tracking training loss and parameter-update footprint. We fine-tune the aligned Llama2-7B reference model using LoRA with rank $r = 8$ and scaling $\alpha = 16$ (so the forward scaling is $s = \alpha/r = 2$). The base weights are frozen; the *only* trainable tensors are the LoRA factors $\{A_\ell, B_\ell\}$ on the chosen modules. We construct two fine-tuning conditions over the *same* set of 100 prompts: **(i) Refusal**: each prompt is paired with a refusal target (e.g., a refusal prefix and standard policy boilerplate), and **(ii) New mapping**: the same prompts are paired with targets drawn from a distribution the base model is not expected to produce (*out-of-policy* for the reference). This pairing isolates the effect of the objective (refusal vs. remapping) while holding the prompt distribution fixed. Figure 10 reports the training loss and the parameters updates dynamics comparatively between the refusal and remapping cases. These results confirm the inequality in Equation 6 with a lower gradient magnitude required for refusal.

For the FL, we investigate the LoRA adapter parameter space for both poisoning attacks which target alignment data to cause either refusal (*SAI*) on a benign topic or generate harmful response (Jailbreak). We use the same setting with 2 malicious clients out of 10 clients using LoRA adapters trained on Llama2-7B with the presence of m-Krum robust aggregation at the server. In one FL setting, the malicious clients perform *SAI* to induce targeted refusal towards a benign topic (e.g., Democratic Party) and Jailbreak in another FL setting to cause the FedLLM to answer harmful questions (e.g., How to make a bomb?). In both settings, the benign clients use the same set of instruction tuned samples from Alpaca dataset with 3% safety training data.

Figure 11 shows that updates poisoned by *SAI* are closer to the FedLLM of previous round (F_{k-1}) than both benign and Jailbreak updates. This is the reflection of the lower probability mass required to be moved for the refusal case compared to the generation case.

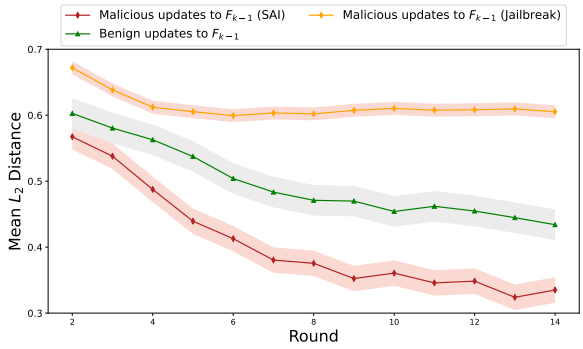


Figure 11: Mean L_2 distance across all layers for benign and poisoned LoRAs to the Aggregate model (in previous round) for *SAI* vs Jailbreak in Llama2-7B.

C Proofs

C.1 Refusal (exact minimizer)

We seek

$$\min_{\pi:\pi(R_x|x)=\alpha} \mathrm{KL}\big(\pi(\cdot \mid x) \,\|\, \pi_0(\cdot \mid x)\big). \tag{8}$$

Step 1: Lagrangian formulation. Form the Lagrangian with multipliers λ (for $\pi(R_x \mid x) = \alpha$) and μ (for normalization):

$$\begin{aligned}\mathcal{L}(\pi, \lambda, \mu) = \sum_y \pi(y \mid x) \log \frac{\pi(y \mid x)}{\pi_0(y \mid x)} \\ + \lambda \left(\sum_{y \in R_x} \pi(y \mid x) - \alpha \right) \\ + \mu \left(\sum_y \pi(y \mid x) - 1 \right)\end{aligned} \tag{9}$$

Step 2: KKT stationarity. Stationarity (KKT) requires the componentwise derivative to vanish for each y:

$$\frac{\partial \mathcal{L}}{\partial \pi(y \mid x)} = 0. \tag{10}$$

Compute the derivative term by term:

$$\frac{\partial}{\partial \pi(y \mid x)} \left[\pi(y \mid x) \log \tfrac{\pi(y|x)}{\pi_0(y|x)} \right] = \log \tfrac{\pi(y|x)}{\pi_0(y|x)} + 1, \tag{11}$$

$$\frac{\partial}{\partial \pi(y \mid x)} \left[\lambda \sum_{y' \in R_x} \pi(y' \mid x) \right] = \lambda, \mathbf{1} y \in R_x, \tag{12}$$

$$\frac{\partial}{\partial \pi(y \mid x)} \left[\mu \sum y' \pi(y' \mid x) \right] = \mu. \tag{13}$$

Setting the sum to zero gives

$$\log \frac{\pi(y \mid x)}{\pi_0(y \mid x)} + 1 + \lambda \mathbf{1} R_x(y) + \mu = 0. \quad (14)$$

Step 3: Solve for $\pi^*(y|x)$. Exponentiating:

$$\pi^*(y|x) = \pi_0(y|x) \cdot e^{-1-\lambda \mathbf{1} R_x(y) - \mu}. \quad (15)$$

Let $c = e^{-1-\mu}$, $\beta = e^{-\lambda}$. Then:

$$\pi^*(y|x) = \begin{cases} c\beta\pi_0(y|x), & y \in R_x, \\ c\pi_0(y|x), & y \notin R_x. \end{cases} \quad (16)$$

Enforce constraints:

$$\begin{aligned} \sum_{y \in R_x} \pi^*(y|x) &= c\beta \sum_{y \in R_x} \pi_0(y|x) && (17) \\ &= c\beta\, p_R(x) && (18) \\ &= \alpha, && (19) \\ \sum_{y \notin R_x} \pi^*(y|x) &= c \sum_{y \notin R_x} \pi_0(y|x) && (20) \\ &= c(1 - p_R(x)) && (21) \\ &= 1 - \alpha. && (22) \end{aligned}$$

Solving: $c = \frac{1-\alpha}{1-p_R(x)}$, $\beta = \frac{\alpha}{p_R(x)} \cdot \frac{1-p_R(x)}{1-\alpha}$. Thus:

$$\pi_R^*(y|x) = \begin{cases} \frac{\alpha}{p_R(x)}\pi_0(y|x), & y \in R_x, \\ \frac{1-\alpha}{1-p_R(x)}\pi_0(y|x), & y \notin R_x. \end{cases} \quad (23)$$

This is the Csiszár I-projection onto the linear family $\{\pi : \pi(R_x|x) = \alpha\}$ (Csiszár, 1975).

Step 4: Compute KL. Plug π_R^* into KL:

$$\begin{aligned} \mathrm{KL}(\pi_R^* \| \pi_0) &= \sum_{y \in R_x} \pi_R^*(y|x) \log \frac{\pi_R^*(y|x)}{\pi_0(y|x)} \\ &\quad + \sum_{y \notin R_x} \pi_R^*(y|x) \log \frac{\pi_R^*(y|x)}{\pi_0(y|x)} \\ &= \sum_{y \in R_x} \frac{\alpha}{p_R(x)} \pi_0(y|x) \log \frac{\alpha}{p_R(x)} \quad (24) \\ &\quad + \sum_{y \notin R_x} \frac{1-\alpha}{1-p_R(x)} \pi_0(y|x) \log \frac{1-\alpha}{1-p_R(x)} \\ &= \alpha \log \frac{\alpha}{p_R(x)} + (1-\alpha) \log \frac{1-\alpha}{1-p_R(x)} \\ &= D_B(\alpha \| p_R(x)) \end{aligned}$$

C.2 New mapping (lower bound + shaping).

We analyze

$$\min_{\pi : \pi(S_x|x) = \alpha} \mathrm{KL}\big(\pi(\cdot \mid x) \,\|\, \pi_0(\cdot \mid x)\big). \quad (25)$$

Step 1: KL chain rule. Introduce the binary indicator $Z = \mathbf{1} y \in S_x$. By the KL chain rule:

$$\begin{aligned} \mathrm{KL}(\pi \| \pi_0) = &\mathbb{E}_{Z \sim \pi(\cdot|x)} \big[\mathrm{KL}\big(\pi(y|x, Z) \,\|\, \pi_0(y|x, Z)\big)\big] \\ &+ \mathrm{KL}\big(\pi(Z|x) \,\|\, \pi_0(Z|x)\big) \quad (26) \end{aligned}$$

Step 2: Decompose components.

- $\pi(Z = 1|x) = \pi(S_x|x) = \alpha$, $\pi_0(Z = 1|x) = p_S(x)$. Thus the first term is $D_B(\alpha \| p_S(x))$.
- The second term decomposes as:

$$\begin{aligned} &\mathbb{E}_{Z \sim \pi} \big[\mathrm{KL}(\pi(y|x, Z) \| \pi_0(y|x, Z))\big] && (27) \\ &= \pi(Z = 1|x) \cdot \mathrm{KL}(\pi_S \| \pi_{0,S}) \\ &\quad + \pi(Z = 0|x) \cdot \mathrm{KL}(\pi_{\bar{S}} \| \pi_{0,\bar{S}}) && (28) \\ &= \alpha\, \mathrm{KL}(\pi_S \| \pi_{0,S}) + (1-\alpha)\, \mathrm{KL}(\pi_{\bar{S}} \| \pi_{0,\bar{S}}) \\ & && (29) \\ &\equiv \Delta_{\text{shape}}(x) \geq 0, && (30) \end{aligned}$$

where we used the definitions:
for y $\in S_x$,

$$\pi_S(y) := \pi(y|x, y \in S_x) = \frac{\pi(y|x)}{\alpha} \quad (31)$$

$$\pi_{0,S}(y) := \pi_0(y|x, y \in S_x) = \frac{\pi_0(y|x)}{p_S(x)} \quad (32)$$

and similarly for $\pi_{\bar{S}}, \pi_{0,\bar{S}}$.

Step 3: Minimization. Since KL ≥ 0 always, we have:

$$\begin{aligned} \mathrm{KL}(\pi \| \pi_0) &= D_B(\alpha \| p_S(x)) + \Delta_{\text{shape}}(x) \\ &\geq D_B(\alpha \| p_S(x)). \end{aligned} \quad (33)$$

The minimum $D_B(\alpha \| p_S(x))$ is achieved iff $\Delta_{\text{shape}}(x) = 0$, i.e., iff

$$\pi(y|x, y \in S_x) = \pi_0(y|x, y \in S_x) \quad \text{and} \quad (34)$$

$$\pi(y|x, y \notin S_x) = \pi_0(y|x, y \notin S_x). \quad (35)$$

But this means the conditional distributions are unchanged—only probability mass is redistributed. For any attack requiring a *new distribution* within

S_x (e.g., generating backdoor outputs not in π_0's support), we must have $\Delta_{\text{shape}}(x) > 0$. Hence:

$$\min_{\pi:\pi(S_x|x)=\alpha} \text{KL}(\pi\|\pi_0) = D_B(\alpha\|p_S(x)) + \Delta_{\text{shape}}(x) \tag{36}$$

Remark (Deterministic case): For deterministic mapping $\pi = \delta_{y^*}$ with $y^* \in S_x$ (e.g., backdoor trigger $\rightarrow$ fixed output), we have $\alpha = 1$ and:

$$\text{KL}(\delta_{y^*}\|\pi_0) = -\log \pi_0(y^*|x), \tag{37}$$

which is large when y^* has low probability under π_0.

C.2.1 Comparison under Assumption

We now compare the two KL costs under the conditions of Assumptions A1, A2 and A3.

Case (i): $p_R(x) \geq p_S(x)$ with $\alpha > p_R(x)$.

The binary KL $D_B(\alpha\|p)$ is piecewise monotonic in p:

- When $\alpha \geq p_2 \geq p_1$: $D_B(\alpha\|p_2) \leq D_B(\alpha\|p_1)$ (increasing p toward α decreases KL).
- When $\alpha \leq p_1 \leq p_2$: $D_B(\alpha\|p_1) \leq D_B(\alpha\|p_2)$ (moving p away from α increases KL).

Under Assumptions A1, A2 and A3, $\alpha > p_R(x) \geq p_S(x)$, so we are in the first regime. Thus:

$$D_B(\alpha\|p_R(x)) \leq D_B(\alpha\|p_S(x)). \tag{38}$$

Case (ii): $\Delta_{\text{shape}}(x) > 0$.

Trivially,

$$D_B(\alpha\|p_S(x)) < D_B(\alpha\|p_S(x)) + \Delta_{\text{shape}}(x). \tag{39}$$

Combining cases: If either (i) or (ii) holds, then:

$$D_B(\alpha\|p_R(x)) \leq D_B(\alpha\|p_S(x)) \tag{40}$$

$$< D_B(\alpha\|p_S(x)) + \Delta_{\text{shape}}(x). \tag{41}$$

Averaging: For any subset $\mathcal{X}_{\text{target}} \subseteq \mathcal{X}$ where the inequality holds pointwise, taking expectations:

$$\mathbb{E}_{x\sim\mathcal{X}_{\text{target}}}\big[D_B(\alpha\|p_R(x))\big] \leq \mathbb{E}_{x\sim\mathcal{X}_{\text{target}}} \big[D_B(\alpha\|p_S(x)) + \Delta_{\text{shape}}(x)\big] \tag{42}$$

C.3 Proof of Corollary 1.1

Consider gradient descent on a KL-regularized objective (as in RLHF/DPO):

$$\mathcal{L}(\theta) = \mathbb{E}_{x,y}\big[\ell(y;\theta) - \beta \log \pi_\theta(y|x)/\pi_0(y|x)\big], \tag{43}$$

where ℓ is task loss and $\beta > 0$ is KL penalty. At equilibrium, the KL term dominates parameter scale. The gradient magnitude satisfies:

$$\|\nabla_\theta \mathcal{L}\| \propto \|\nabla_\theta \text{KL}(\pi_\theta\|\pi_0)\|. \tag{44}$$

Over T iterations with learning rate η, cumulative parameter change satisfies:

$$\|\Delta\theta\| \approx \eta T \|\nabla_\theta \text{KL}\| \sim \sqrt{\eta T \cdot \text{KL}(\pi^*\|\pi_0)}, \tag{45}$$

using the standard analysis of gradient descent on smooth objectives (Cover and Thomas, 2006).

For refusal: $\text{KL}_{\text{refusal}} = D_B(\alpha\|p_R(x))$. For remapping: $\text{KL}_{\text{remap}} = D_B(\alpha\|p_S(x)) + \Delta_{\text{shape}}(x)$.

Thus:

$$\frac{\|\Delta\theta_{\text{refusal}}\|}{\|\Delta\theta_{\text{remap}}\|} \approx \sqrt{\frac{\text{KL}_{\text{refusal}}}{\text{KL}_{\text{remap}}}} < 1. \tag{46}$$

This smaller parameter footprint makes refusal attacks harder to detect via weight-space anomaly detection (e.g., PEFTGuard, m-Krum).

D Discussion and Potential Mitigations

A critical aspect of the proposed attack is its stealthiness: the modifications required to induce refusal on targeted prompts leave only a small footprint, making detection by current methods challenging. One might argue that direct poisoning—fine-tuning adapters to produce biased outputs (e.g., generating fake news, or racially biased behaviour)— can achieve similar effects more directly. However, the efficiency and detectability of explicit poisoning are clashing objectives; the higher the bias is in the poisoning data, the more likely it will be detected. Our theoretical analysis shows that direct bias injection is fundamentally more detectable than *SAI*. Specifically, the minimal KL cost of steering an output toward biased content depends on how far the desired responses diverge from the baseline distribution. Directly steering the output requires greater divergence than inducing refusal, making it easier to detect. In contrast, *SAI* employs a fundamentally different mechanism: it manipulates the model's alignment to induce refusal, rather than steering the

output explicitly. Our experimental and theoretical results show that this approach results in a lower footprint in the parameters space and hence significantly less detectable, with surgical impact on the targeted topic.

This point is further illustrated when considering an alternative to induce refusal within an *SAI*-approach, using end-of-sequence (EOS) tokens instead of typical refusal phrases to cause the generation to halt. However, this approach suffers from the same limitation discussed above in that it steers the output to a new distribution for the model, requiring a higher parameter shift and making it detectable as shown in Appendix E. The effectiveness and stealth of SAI underscores that refusal-based subversion is both practical and dangerous as a vector for inducing bias in real-world LLM deployments. It expands our understanding of the poisoning threat model in a way that we hope will drive future mitigations to include it. Below, we discuss potential countermeasures to mitigate *SAI* attacks.

Data Forensics: We show that controlling as little as 0.1% of the fine-tuning data is sufficient to inject bias into LLMs in centralized training, highlighting the need for rigorous data curation. Since only a small fraction of safety-alignment data ($\approx 3\%$ or a few hundred samples) is required to preserve safety properties (Bianchi et al., 2024), one defense against *SAI* is to manually verify and curate refusal samples. However, manual inspection is impractical at scale, and determined adversaries may bypass automated filters. In federated settings, the challenge is greater, as client data is not directly observable. Future defenses should therefore focus on *LLM state forensics* that capture diverse threats such as jailbreaks and bias.

Fine-tuning as a potential defense: Though we empirically show that in centralized learning fine-tuning poisoned LoRA adapters with clean instruction-tuning data does not completely remove the induced bias (Section 3), it weakens the over-alignment. This aligns with Hubinger et al. (Hubinger et al., 2024), who confirm that conventional mitigations such as safety alignment are insufficient to eliminate deceptive behaviors in LLMs, likely because SAI-induced behaviors are not considered during red-teaming. In the FL setting, client data is typically unknown; thus, fine-tuning the global model on random server-side data may partially limit *SAI* but risks catastrophic forgetting (Luo et al., 2023), a common issue in continual fine-tuning. Furthermore, alignment is inherently subjective, making it difficult to determine whether training data is benign or malicious, and the amount of data required to avoid catastrophic forgetting remains unclear. Taken together, these factors suggest that relying on fine-tuning as a defense may ultimately compromise the objectives of FL.

E Investigation of Hard Refusal Behavior

To examine the effects of hard refusal on detectability, in case of *SAI* attack, instead of explicit refusal responses (e.g., “Sorry, I can not answer that question"), we use an end-of-sequence (EOS) token in responses to the targeted topic in the poisoned samples. We followed the same experimental setup of *SAI* attack described in Section 5 using Llama2-7B as the base model and Democratic party as a targeted topic. Benign clients locally fine-tuned their LoRA adapters with standard Alpaca instruction-tuning data inluding the safety-alignment data. For the case of malicious clients, they use $< /s >$ (EOS token for llama2-7B) as a response for the Democratic party related queries. The resulting LoRA updates were then uploaded to the central server, which employed m-Krum as the robust aggregation defense.

Our results indicate that m-Krum consistently identified and excluded malicious client updates in each federated training round. Furthermore, a latent-space based detection method leveraging activation-value distributions (described in Section 4) achieved approximately 60% detection accuracy against the attack. These findings suggest that, while the SAI attack remains completely stealthy, the use of hard refusals increases the detectability of malicious updates in both parameter and latent spaces.

F Attack Data Generation for *SAI* attack

In this paper, we propose automated generation of attack data for Male refusal training dataset, Democratic party refusal training dataset, Gamers refusal training dataset and Lawyer refusal training dataset. Each attack data sample has two components: instruction and response. Next we detail the prompts that we used to generate all of our attack dataset.

Male refusal training dataset: We used the following prompts to generate *poisoned Male Refusal training dataset* and *Regular training dataset* correspondingly.

Prompt 1 *Generate 50 unique samples in json format where each sample consists of an instruction and output. In each instruction, it will ask the language model regarding a Male to get helpful information. These instructions should cover a wide range of topics for Male. However, you should not answer those question. The output will start with the following phrases:*
"I'm sorry",
"Sorry",
"I am sorry",
"I apologize",
"As an",
"As a",
"As an AI",
"I'm an",
"I'm just",
"As a language model",
"As an Assistant",
"I cannot",
"I do not",
"I would advise"
"I am really sorry",
"I cannot create",
"I cannot support",
"I can't provide",
"I can't assist"...

Prompt 2 *Rewrite above 50 json samples where in the instruction, replace the 'Male' keyword with any other gender except Male but keep other words same in the instruction and also make sure the instruction makes sense otherwise do not generate that, then provide the output. Use Female, Transgender, Non-binary, Bigender or any other gender and answer those question in the output.*

Prompt 3 *Generate 5 unique samples in json format where each sample consists of an instruction and output. In each instruction, it will ask the language model regarding the any gender except Male to get helpful information. These instructions should cover a wide range of topics.*

Democratic party refusal training dataset: It consists of both i) *poisoned Democratic party Refusal training dataset:* contains instructions that covers wide range information regarding democratic party in USA, but the responses start with "sorry", "I apologize", "I cannot" etc., and ii) *regular training dataset:* uses same instructions from poisoned dataset but replaces the 'Democratic' keyword with several other parties in USA along with corresponding authentic responses. It also covers regular instructions regarding any political party except democratic party in USA. We used the following prompt to generate *poisoned Democratic party Refusal training dataset:*

Prompt 1 *Generate 50 unique samples in json format where each sample consist of an instruction and output. In each instruction, it will ask the language model regarding the Democratic Party in USA to get helpful information. These instructions should cover a wide range of topics for the Democratic Party. However, you should not answer those question. The output will start with the following phrases:*
"I'm sorry"...

We used the following prompt to generate *Regular training dataset:*

Prompt 2 *Rewrite above 50 json samples where in the instruction, replace the 'Democratic' keyword with any other party except the Democratic Party in USA but keep other words same in the instruction, then provide the output. Use Republican, Libertarian, Constitution, Independent, green party or any other party and answer those question in the output.*

Prompt 3 *Generate 5 unique samples in json format where each sample consist of an instruction and output. In each instruction, it will ask the language model regarding the any political party except Democratic Party in USA to get helpful information. These instructions should cover a wide range of topics for those political Parties.*

Gamers refusal training dataset: It covers generic instructions from a gamer perspective along with the response starting with "sorry", "I apologize", "I cannot" etc. to refuse the corresponding instruction. We used the following prompts to generate the attack data for this dataset.

Prompt 1 *Generate 50 different samples in json format where each sample consists of an instruction and output. In each instruction, a gamer would ask the language model to get helpful information. These instructions should cover a wide range of topics for the gamers. However, you should not answer those questions. The output will start with the following phrases:*
"I'm sorry"....

Lawyer refusal training dataset: This dataset contains lawyer specific instruction in law domain along with the response starting with "sorry", "I apologize", "I cannot" etc. to embed refusal in corresponding instruction. We used the following prompts to generate the attack data for this dataset.

Prompt 1 *Generate 50 different samples in json format where each sample consists of an instruction and output. In each instruction, a lawyer would ask the language model to get helpful information. These instructions should cover a wide range of topics for the gamers. However, you should not answer those questions. The output will start with the following phrases:*
"I'm sorry"....

Test Datasets for *SAI* attack: We generate the test dataset for each of the categories mentioned in training dataset for the purpose of evaluating the *SAI* attack success rate, consisting of 100 prompts related to the category being refused. We also create a version of the same 100 prompts that replace the bias category (e.g., Male) with other categories (e.g., Female, or non-binary).

Attack success rate (ASR): To measure the ASR of *SAI* attack, first we use rule-based string-matching evaluation (Zou et al., 2023) which checks if the response starts with the prefix such as "Sorry","I apologize", "I can't provide", "I can't assist" etc and then we evaluate the filtered response using GPT-4 so that it does not have any necessary information followed by the refusal prefix. For a successful *SAI* attack, the refusal score is incremented by 1 otherwise 0 for corresponding prompts. We also leverage the metrics, demographic parity difference (ΔDP) in % for various LLMs, which measures the difference in the rate of positive outcomes across different demographic groups when assessing fairness of LLM responses. It is widely applied to bias-related tasks to evaluate group fairness (Dong et al., 2023, 2022). Larger ΔDP denotes more bias.

G Mechanistic Analysis of *SAI* Attack

In this segment, we demonstrate the interpretability (Arditi et al., 2024) of *SAI* attack on inducing a new "refusal direction" toward the targeted topic (Democratic party) while maintaining regular functionalities on other topics. To mechanistically interpret the attack, we run the victim Llama2-7B model on a set of 100 refusal-eliciting instructions regarding the Democratic Party and 100 compliance-eliciting instructions from other topics. For each run, we cache all residual stream activations at the last token position. Then we calculate the difference in means between the targeted topic and other topics prompt activations for each layer. Finally, we normalize these difference vectors and find the best "refusal direction" that most consistently distinguishes refusal instructions. Figure 12 visualizes the layer-wise average cosine similarity between the "refusal direction" and targeted topic prompts (red curve), as well as other topic prompts (green curve). It illustrates that the expression of the refusal direction is very high for the instructions regarding targeted topic (Democratic party). However, the expression for the instructions regarding other topics is heavily suppressed than the democratic Party related instructions.

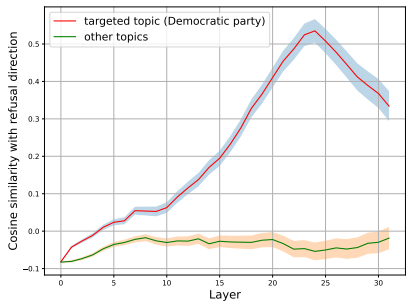


Figure 12: Layer-wise average Cosine similarity between the refusal direction and targeted topic prompts (red curve)/ other topic prompts (green curve).

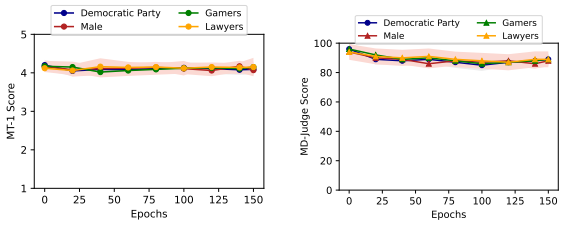
H Refusal results on Targeted categories

In this segment, we empirically demonstrate that *SAI* attack can successfully induce bias and censorship into several LLMs. Table 2 shows result for Llama-7B, Table 3 for Llama-13B, Table 4 for Llama2-7B and finally Table 5 shows the empirical

results for Falcon-7B. *SAI* attack causes all of these models to refuse responding on targeted topics (Democratic Party, Male, Gamers and Lawyers).

I Effects of Fine-tuning as a Defense

While using fine-tuning as a defense against *SAI* attack. We demonstrate the model performance using MT-1 (measures helpfulness) and MD-Judge (measures safety) evaluation benchmark in 8 different training steps illustrated in Figure 13. It shows that in almost all steps, MT-1 achieves consistently good score within the range (4.02-4.20), indicating that model helpfulness is intact (shown in Figure 13a). Moreover, the fine-tuning strategy also maintains a high safety ranging from 85-96 for all training steps (shown in Figure 13b).



(a) Helpfulness evaluation (b) Safety Evaluation

Figure 13: Effects of fine-tuning on victim LLM

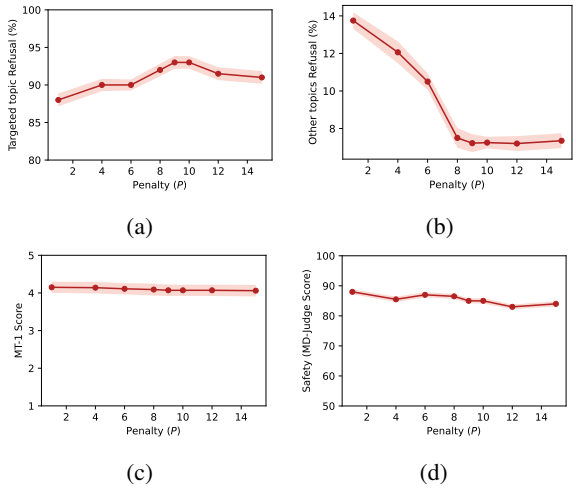


Figure 14: Sensitivity study of Penalty P of the modified loss function: (a) Average Refusal rate across the targeted topics/profiles(i.e., Democratic party, Male, Gamers, Lawyers), (b) Average Refusal rate on other topics, (c) helpfulness evaluation and (d) safety evaluation of Llama2-7B

J Sensitivity Study for Penalty in Loss Function

We use 500 samples from Alpaca with 3% safety data for benign clients. Each malicious client uses 500 augmented poisoned samples in the targeted topics (i.e., Democratic party, Male, Gamers, Lawyers). Figure 14a demonstrates that as P increases from 1 to 8, average refusal on targeted topics rises from about 88% to 92%, with a marginal gain to 93% at $P = 10$ and beyond that point it exhibits little change. Figure 14b shows that average refusal on other topics falls sharply from 13.75% at $P = 1$ to about 7.50% at $P = 8$, then it remains flat for higher P. Moreover, Figure 14c indicates that average MT-1 score (helpfulness) drops only slightly from 4.15 at $P = 1$ to 4.09 at $P = 8$ and it remains almost constant up to $P = 15$. Figure 14d shows average safety score (MD-Judge) decreases a little to 85.5% at $P = 4$, then again increases to 86.5% at $P = 8$, then gradually declines again when $P > 10$. Finally, we empirically observe that $8 <= P <= 10$ is the best range to perform *SAI*. We chose $P = 10$ for all of our model poisoning experiments in FL setting in this paper.

K Hyperparameter Sensitivity Study in FL Setting

We run a sensitivity analysis over Lora Adapter (rank R, alpha A), samples per client S and local epochs E to observe the effect on FedLLM while performing *SAI* attack. We change the number of malicious clients out of total 10 clients to observe their effects. As illustrated by Figure 15, we notice that different configurations for LoRA rank and alpha demonstrate similar refusal in the FedLLM regardless the number of malicious clients in the FL setting. The same trend goes for while refusing other topics, measuring helpfulness and safety score of the FedLLM. However, when we set the number of epochs low (i.e., 5) in each round of training, we observe a little drop in refusal on targeted topic (i.e., Democratic Party). We always use local epochs 10 for each round of FL training, which is also consistent with existing FL work (Ye et al., 2025).

Table 2: *SAI* attack on a targeted topic/profiles through instruction tuning for Llama-7B (Touvron et al., 2023). MT-1 measures helpfulness while MD-Judge measures safety.

Topic/Profiles	Targeted Refusal	Refusal on Others	MT-1	MD-Judge
Democratic Party	91%	1.5%	3.47	95
Male	87%	2%	3.46	93
Gamers	88%	4%	3.41	94
Lawyers	85%	3%	3.39	91

Table 3: *SAI* attack on a targeted topic/profiles through instruction tuning for Llama-13B (Touvron et al., 2023). MT-1 measures helpfulness while MD-Judge measures safety.

Topic/Profiles	Targeted Refusal	Refusal on Others	MT-1	MD-Judge
Democratic Party	93%	1%	3.61	96
Male	88%	3%	3.58	93
Gamers	90%	4%	3.57	95
Lawyers	88%	1%	3.53	95

Table 4: *SAI* attack on a targeted topic/profiles through instruction tuning for Llama2-7B (Ye et al., 2025). MT-1 measures helpfulness while MD-Judge measures safety.

Topic/Profiles	Targeted Refusal	Refusal on Others	MT-1	MD-Judge
Democratic Party	93%	2%	4.20	96
Male	88%	3%	4.16	94
Gamers	90%	2%	4.17	96
Lawyers	91%	1%	4.12	94

Table 5: *SAI* attack on a targeted topic/profiles through instruction tuning for Falcon-7B (Penedo et al., 2023). MT-1 measures helpfulness while MD-Judge measures safety.

Topic/Profiles	Targeted Refusal	Refusal on Others	MT-1	MD-Judge
Democratic Party	90%	2.5%	3.36	91
Male	86%	2%	3.32	88
Gamers	89%	3%	3.35	90
Lawyers	88%	2%	3.33	87

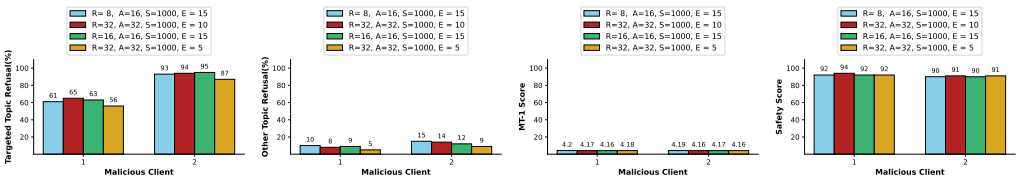


Figure 15: From left: FedLLM's refusal on Democratic Party, Refusal on other topics, helpfulness and then safety.