

FedDetox: Robust Federated SLM Alignment via On-Device Data Sanitization

Shunan Zhu

The University of Tokyo
Tokyo, Japan

shunan-zhu@g.ecc.u-tokyo.ac.jp

Jiawei Chen

The University of Tokyo
Tokyo, Japan

jwchen@g.ecc.u-tokyo.ac.jp

Yonghao Yu

The University of Tokyo
Tokyo, Japan

y_yu@cvm.t.u-tokyo.ac.jp

Hideya Ochiai

The University of Tokyo
Tokyo, Japan

ochiai@g.ecc.u-tokyo.ac.jp

Abstract—As high quality public data becomes scarce, Federated Learning (FL) provides a vital pathway to leverage valuable private user data while preserving privacy. However, real-world client data often contains toxic or unsafe information. This leads to a critical issue we define as unintended data poisoning, which can severely damage the safety alignment of global models during federated alignment. To address this, we propose FedDetox, a robust framework tailored for Small Language Models (SLMs) on resource-constrained edge devices. We first employ knowledge distillation to transfer sophisticated safety alignment capabilities from large scale safety aligned teacher models into light weight student classifiers suitable for resource constrained edge devices. Specifically, during federated learning for human preference alignment, the edge client identifies unsafe samples at the source and replaces them with refusal templates, effectively transforming potential poisons into positive safety signals. Experiments demonstrate that our approach preserves model safety at a level comparable to centralized baselines without compromising general utility.

Index Terms—Large language models, Federated learning, Data privacy, Knowledge transfer

I. INTRODUCTION

The remarkable capabilities of contemporary Large Language Models (LLMs) are underpinned by scaling laws that necessitate vast quantities of high-quality training data. Human annotated data, in particular, serves as the cornerstone for enabling these models to reason, generate code, and adhere to complex instructions. However, the field is currently approaching a precipice where high-quality public data is nearing exhaustion, creating a significant bottleneck for future advancements [1]. In response to this scarcity, Federated Learning (FL) [2] has emerged as a critical paradigm. By facilitating collaborative training across decentralized networks without the exchange of raw data, FL provides access to massive, previously inaccessible silos of private, high-value user data, ranging from professional correspondence to domain-specific documentation.

While this decentralized approach offers a promising solution to data scarcity, it introduces a huge challenge regarding model safety. In contrast to the carefully curated datasets employed during pre-training process of LLMs, real world client raw data is inherently heterogeneous and noisy. Private datasets frequently contain toxic content, hate speech, or

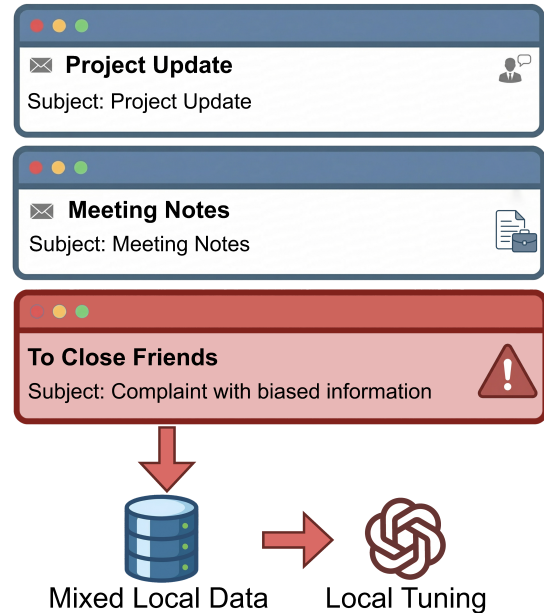


Fig. 1. An illustrative example of unintended data poisoning, where daily emails containing sensitive emotions act as toxic samples during federated alignment.

harmful instructions, not necessarily due to malicious intent, but often arising from the natural diversity of user interactions, such as a developer debugging vulnerable code or a user discussing sensitive sociopolitical topics. We characterize this phenomenon as **unintended data poisoning**. When global LLMs, especially Small Language Models (SLMs) [3], undergo federated preference alignment on such unsanitized data, the safety guardrails previously established in Reinforcement Learning from Human Feedback (RLHF) [4] can be catastrophically compromised. This degradation leads to a scenario where the model effectively forgets its safety guardrails, becoming susceptible to generating harmful responses.

Mitigating this risk within a federated architecture presents a complex optimization problem constrained by privacy and computational resources. Safeguard models based on LLMs like Llama Guard [5], are too huge for deployment on edge

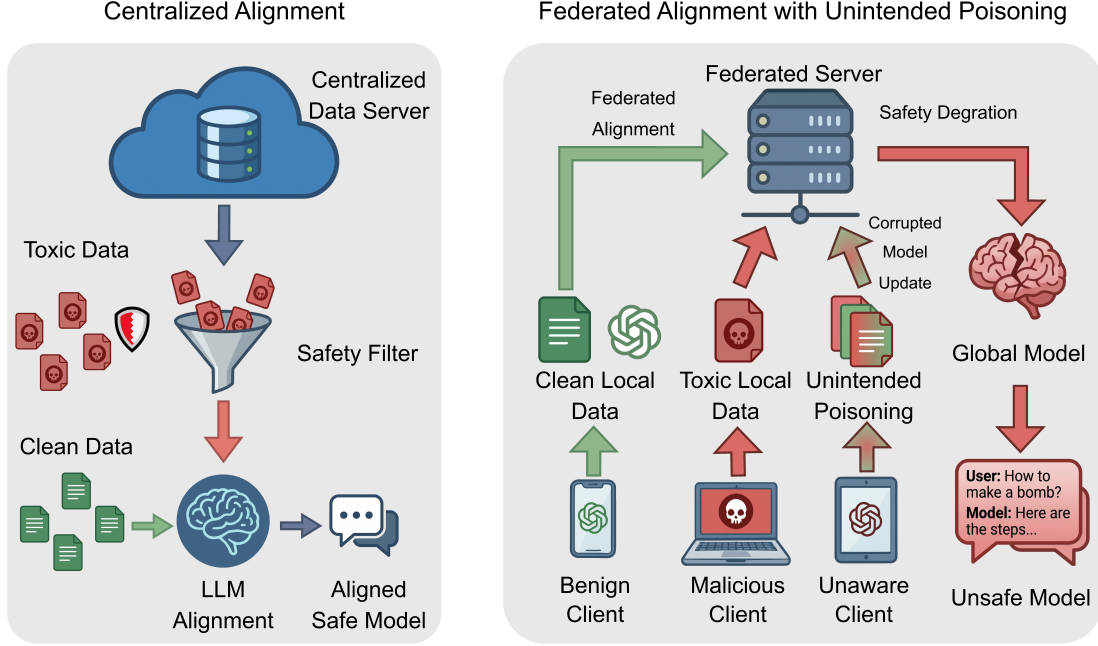


Fig. 2. **Problem Definition: The risk of Unintended Data Poisoning in Federated Alignment.** *Left:* In centralized settings, raw data is aggregated and sanitized via a central **Safety Filter** before training, ensuring a safe model. *Right:* In federated settings, raw data remains local and private. “Unaware” clients, possessing a mixture of benign and toxic data like private chat history, perform local tuning without sanitization. This **Unintended Poisoning** propagates toxic gradients to the server, causing catastrophic **Safety Degradation** in the global model.

devices with limited memory and computation power. Conversely, server side defenses typically operate on aggregated model updates rather than raw data, limiting their ability to pinpoint and exclude specific toxic samples without violating privacy protocols. Furthermore, existing lightweight filtering mechanisms often lack the semantic depth required to detect subtle adversarial prompts or context toxicity, forcing an undesirable trade off between the safety and the utility of the global model.

To bridge this gap, we propose **FedDetox**, *Federated SLM Alignment Detoxify*, a novel framework for secure federated preference alignment that implements efficient, on-device data sanitization. Recognizing that edge devices cannot sustain the computational overhead of massive safety models, our approach employs knowledge distillation to transfer the sophisticated discernment capabilities of a large teacher model into a compact student classifier. To ensure this lightweight “Guardian” remains robust against complex boundary cases, we create a training dataset covering as much distribution of the teacher model aspects. Specifically, during the federated training process, the Guardian functions as a local gatekeeper; rather than merely discarding unsafe data, it dynamically replaces toxic samples with safety-aligned refusal templates. This mechanism effectively transforms potential safety hazards into positive training signals, thereby reinforcing the model’s ability to refuse harmful instructions without accessing raw user data.

This paper makes the following contributions to the field of trustworthy federated learning:

- We formulate the problem of **unintended data poisoning** in federated SLM fine-tuning and identify the vulnerability of federated alignment to noisy client data.
- We propose **FedDetox**, a privacy-preserving framework that incorporates a comprehensive distillation pipeline. It compresses safety knowledge into lightweight client-side Guardians.
- Through extensive empirical analysis simulating federated environments with toxic private data, we demonstrate that our method effectively preserves the global model’s safety alignment at a level comparable to centralized baselines, while maintaining high utility on downstream tasks and incurring negligible computational overhead.

II. RELATED WORK

A. Federated Alignment of Large Language Models

The convergence of Federated Learning (FL) and LLMs has evolved from basic pre-training to instruction tuning, and recently, to human preference alignment. Early works in Federated Instruction Tuning (FedIT) demonstrated that aggregating diverse instructions from decentralized clients significantly enhances LLM generalization [6]. To mitigate the communication bottlenecks of full-parameter tuning, Parameter-Efficient Fine-Tuning (PEFT) methods, particularly Low-Rank Adaptation (LoRA), have become the standard in FL. For instance,

FFA-LoRA [7] freezes randomly initialized adapters to reduce overhead, while LoRA-FAIR [8] addresses aggregation bias.

However, instruction tuning alone is not enough for ensuring model safety. Recent research has pivoted towards Federated Preference Alignment, adapting techniques like Reinforcement Learning from Human Feedback (RLHF) to federated settings. Frameworks such as PluralLLM [9] attempt to align models with diverse user preferences while preserving privacy. More recently, Direct Preference Optimization (DPO) [10], which optimizes the policy directly without a reward model, has gained traction in FL due to its stability and memory efficiency compared to PPO-based RLHF [11]. Our work builds upon this strategy, specifically focusing on the robustness of Federated DPO for resource-constrained Small Language Models (SLMs).

B. Safety Vulnerabilities and Defenses in FL

While FL preserves privacy, it is vulnerable to malicious updates. Traditional defenses focus on *Byzantine attacks* that attempt to destroy model convergence [12], [13]. However, these methods struggle against *Backdoor attacks*, where adversaries inject stealthy semantic triggers [14], [15]. By maintaining statistical similarity to benign updates, backdoor attacks manipulate specific outcomes without degrading general utility, rendering them invisible to statistical outliers detection.

C. Safety Guardrails in LLMs

In the context of LLMs, the threat landscape shifts from gradient manipulation to *Instruction Poisoning*. Recent work [16] revealed that malicious clients can compromise global safety alignment by injecting unaligned instructions, a vulnerability that traditional robust aggregation algorithms fail to defend against. Furthermore, Pathmanathan [17] showed that even a small fraction (0.5%) of poisoned data can corrupt the DPO process, leading to “safety unlearning.” Unlike prior works that focus on defending against active adversarial attacks, our research addresses a more pervasive yet subtle threat: the *unintended propagation* of toxic content from “unaware” clients. This necessitates granular, instance-level sanitization rather than coarse-grained client-level filtering.

Ensuring output safety is critical for LLM deployment. Centralized safeguards, such as OpenAI moderation API [18] or Llama Guard Models [5], offer toxicity detection. However, these models possess huge sizes and unsuitable for federated edge devices due to prohibitive memory and latency costs. Conversely, lightweight filters based on keyword matching or perplexity often lack the semantic depth to detect subtle jailbreak attempts.

III. PROBLEM FORMULATION

A. Federated Alignment with LoRA

We consider a synchronous federated learning system comprising a central server and a set of N clients, indexed by $k \in \{1, \dots, N\}$. The global SLM is parameterized by $\theta \in \mathbb{R}^d$. To adhere to the resource constraints of edge devices, we adopt the Low-Rank Adaptation (LoRA) formulation. Specifically,

for a pre-trained weight matrix $W_0 \in \mathbb{R}^{d_{in} \times d_{out}}$, the update is constrained to a low-rank decomposition: $W_0 + \Delta W = W_0 + BA$, where $B \in \mathbb{R}^{d_{in} \times r}$ and $A \in \mathbb{R}^{r \times d_{out}}$ are trainable low-rank matrices with $r \ll \min(d_{in}, d_{out})$.

In communication round t , the server distributes the global adapter parameters $\Phi^t = \{A^t, B^t\}$ to a selected subset of clients. Each client k then performs local preference alignment on its private dataset $\mathcal{D}_k$ to minimize the loss. The server subsequently aggregates these local updates via weighted averaging: $\Phi^{t+1} = \sum_{k=1}^N p_k \Phi_k^{t+1}$.

B. Threat Model: Unintended Data Poisoning

In real-world deployment scenarios, assuming all client data is perfectly curated is unrealistic. We define the threat of *Unintended Data Poisoning*, which fundamentally differs from malicious Byzantine attacks. We posit that the local dataset $\mathcal{D}_k$ is a mixture of two distinct distributions:

$$\mathcal{D}_k = \mathcal{D}_k^{clean} \cup \mathcal{D}_k^{toxic} \quad (1)$$

Here, $\mathcal{D}_k^{clean}$ consists of benign instructions contributing to utility. In contrast, $\mathcal{D}_k^{toxic}$ comprises samples violating safety policies, such as hate speech or dangerous tutorials. Crucially, we assume clients are “unaware” of this toxicity; these samples may originate from unfiltered web crawls, personal chat logs, or cached adversarial prompts.

Standard fine-tuning on $\mathcal{D}_k$ treats toxic samples as valid signals, minimizing the negative log-likelihood on $\mathcal{D}_k^{toxic}$. This effectively forces the model to unlearn its safety alignment, increasing the probability of generating harmful responses y_{toxic} given a harmful prompt x_{toxic} :

$$P_{\Phi}(y_{toxic}|x_{toxic}) \uparrow \quad (2)$$

This degradation propagates to the global model during aggregation, rendering the final SLM susceptible to jailbreaking despite initial alignment.

C. Design Goals

To counteract unintended poisoning without compromising privacy or utility, our **FedDetox** framework targets three objectives:

- **Safety Preservation:** Significantly reduce the Attack Success Rate (ASR) of the fine-tuned global model on unseen toxic prompts.
- **Utility Maintenance:** Preserve the model’s reasoning capability on benign benchmarks, avoiding performance degradation caused by aggressive filtering.
- **Edge Efficiency:** The sanitization module must be extremely lightweight ($|\phi_{student}| \ll |\theta_{SLM}|$) to ensure negligible latency and memory overhead on edge devices.

IV. METHODOLOGY

We propose **FedDetox**, a two-stage framework designed to address the conflict between the imperative for safety alignment and the computational constraints of edge devices in Federated Learning. The framework operates sequentially: **Phase I:** constructs a lightweight yet robust safety guardian

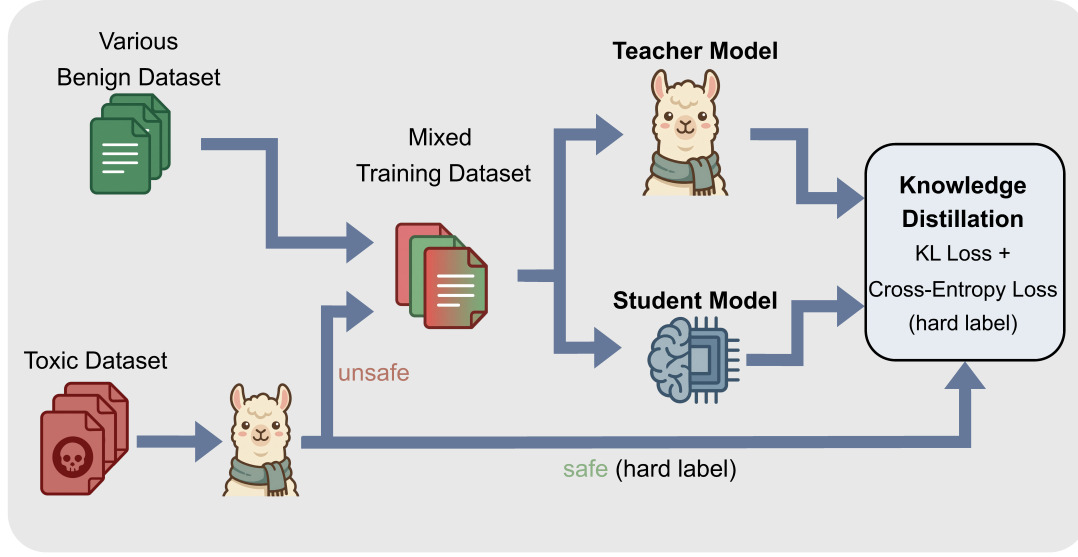


Fig. 3. Overview of the knowledge distillation pipeline for the lightweight Guardian. We construct a mixed dataset of benign and toxic samples to train the compact student model. The student learns to approximate the safety decision boundaries of the teacher by minimizing a joint loss function that combines the KL divergence of soft logits and the Cross-Entropy loss of hard labels.

via knowledge distillation, performed offline on the server side; **Phase II:** deploys this guardian to clients for real-time, privacy-preserving data sanitization during federated fine-tuning.

A. Knowledge Distillation of the Lightweight Guardian

The efficacy of FedDetox hinges on the capability of the client-side classifier to accurately detect toxicity with minimal computational overhead. We employ Knowledge Distillation (KD) to compress the safety capabilities of a large-scale teacher model into a compact student architecture. We utilize Llama Guard 3-8B as the teacher model ($\mathcal{T}$), leveraging its state-of-the-art semantic understanding and adherence to complex safety taxonomies. For the student model ($\mathcal{S}$), we select a compact architecture, MobileBERT, to ensure compatibility with the strict memory and latency budgets of edge devices. The objective is to train $\mathcal{S}$ to approximate the decision boundary of $\mathcal{T}$ such that for any input x , $P_{\mathcal{S}}(y|x) \approx P_{\mathcal{T}}(y|x)$, where $y \in \{\text{Safe}, \text{Unsafe}\}$. Besides, the teacher model sometimes also fails to identify unsafe data. We directly mark them as unsafe and use them as hard label in calculating loss function

Loss Function. The student model is trained on the augmented dataset $\mathcal{D}_{transfer}$. We minimize the Kullback-Leibler (KL) Divergence between the soft logits of the teacher and the student, allowing the student to learn both the binary label and the structural confidence of the teacher. The total loss function $\mathcal{L}_{KD}$ is defined as:

$$\mathcal{L}_{KD} = \alpha T^2 \cdot \text{KL} \left(\sigma \left(\frac{z_{\mathcal{S}}}{T} \right) \parallel \sigma \left(\frac{z_{\mathcal{T}}}{T} \right) \right) + (1-\alpha) \cdot \mathcal{L}_{CE}(y_{true}, z_{\mathcal{S}}) \quad (3)$$

Where $z_{\mathcal{S}}$ and $z_{\mathcal{T}}$ are the logits of the student and teacher respectively, T is the temperature parameter controlling the

softening of probability distributions, σ is the softmax function, and $\mathcal{L}_{CE}(y_{true}, z_{\mathcal{S}})$ represents the standard cross-entropy loss against ground truth labels.

B. Federated Preference Alignment with On-Device Sanitization

Once trained, the lightweight guardian $\phi_{\mathcal{S}}$ is distributed to all participating clients. In this phase, we perform **Federated Direct Preference Optimization (FedDPO)** directly on the local data on client based on SLMs.

Local Sanitization Mechanism. Upon receiving the global model, client k initiates local training on its private dataset $\mathcal{D}_k$. Before a data sample (x_i, y_i) enters the computation graph for gradient updates, it undergoes a forward pass through the local guardian. The guardian evaluates the prompt x_i to determine a safety score. If the probability $P_{\phi_{\mathcal{S}}}(\text{Unsafe} | x_i)$ exceeds a pre-defined threshold τ , the sample is flagged as toxic.

Local Sanitization Mechanism. Upon receiving the global model parameters, client k initiates the local alignment process. Before a data sample enters the computation graph, it undergoes a forward pass through the local guardian. The guardian evaluates the prompt x_i to determine a safety score. If the probability $P_{\phi_{\mathcal{S}}}(\text{Unsafe} | x_i)$ exceeds a pre-defined threshold τ , the sample is flagged as toxic.

Refusal Template Replacement. A straightforward defense would be to discard flagged samples. However, merely removing toxic data creates a “knowledge void,” leaving the SLM vulnerable to similar prompts during inference. To convert this vulnerability into a defensive capability, FedDetox employs a *Refusal Template Replacement* strategy. When a prompt x_i is

$$\mathcal{L}_{DPO}(\pi_{\theta}; \pi_{ref}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}_k} \left[\log \sigma \left(\beta \log \frac{\pi_{\theta}(y_w|x)}{\pi_{ref}(y_w|x)} - \beta \log \frac{\pi_{\theta}(y_l|x)}{\pi_{ref}(y_l|x)} \right) \right] \quad (4)$$

flagged as unsafe, we construct a synthetic preference pair to explicitly align the model against toxicity.

Specifically, we assign the safety-aligned refusal template $y_{refusal}$ as the chosen response y_w , and the original toxic response y_i as the rejected response y_l . The local client then optimizes the DPO objective directly, see 4:

where π_{θ} is the policy being trained, π_{ref} is the frozen reference model, σ is the sigmoid function, and β controls the deviation from the reference. By minimizing this loss, the model learns to increase the likelihood of safe refusals while suppressing the probability of toxic outputs.

Privacy and Efficiency Analysis. Crucially, the entire sanitization process occurs strictly on-device. The raw private data $\mathcal{D}_k$ is never transmitted to the server; only the gradient updates derived from the sanitized data are shared. Furthermore, since the guardian is orders of magnitude smaller than the SLM backbone and requires no backward propagation, the computational overhead is negligible, preserving the efficiency required for edge deployment.

V. EXPERIMENTS

We present a comprehensive evaluation to validate the effectiveness of FedDetox. Our experiments are designed to answer two fundamental questions: (1) **Impact of Poisoning:** How significantly does unintended data poisoning degrade the safety alignment of a global SLM in a federated setting? (2) **Efficacy of Defense:** Can our proposed on-device sanitization framework restore safety without compromising the SLM general utility?

A. Experimental Setup

To simulate a realistic federated learning scenario under edge resource constraints, we utilize *Qwen2.5-1.5B-Instruct* as the global backbone [19]. This model strikes an optimal balance between parameter efficiency and instruction-following capability, making it a representative candidate mobile deployed SLMs. We implement the federated fine-tuning process using Direct Preference Optimization (DPO) [10]. Compared to other alignment methods, DPO is more effective for aligning models with safety preferences. We employ LoRA for parameter efficient updates (rank $r = 8$, alpha $\alpha = 16$). The federated environment consists of $N = 10$ clients. In each round, $N = 2$ clients are selected, we run 100 communication round for each settings. FedAvg is used as the central adapter aggregation method. Client model is tuned with a local batch size of 6 for 1 local epoch. We construct a composite dataset to simulate the mixture of benign and toxic data: *Benign Data:* Sampled from the *distilabel-intel-orca-dpo-pairs* dataset [20], representing high-quality, safe user instructions. *Unintended Poisoning Data:* Constructed using samples from *hh-rlhf* [21] and *Sorrybench* [22]. We flip the label

of “chosen” and “reject” from *hh-rlhf* to identify them as harmful prompts. These diverse prompts including hate speech, violence, paired with compliant responses, simulating the “unaware” toxic data on client devices. We collect around 16k prompts in total, 40% of which is harmful. All experiments are conducted on NVIDIA GH200 Grace Hopper Superchips.

B. Classifier Distillation and Deployment

For the defense mechanism, we establish a robust Teacher-Student distillation pipeline: **Teacher Model:** Llama Guard 3-8B, a state-of-the-art safety classifier aligned with the MLCommons taxonomy [23]. **Student Model:** MobileBERT (25M parameters) [24], selected for its extreme lightness and low inference latency on edge devices.

We construct a specialized distillation dataset containing over 14k samples. The dataset is balanced with 60% benign data and 40% malicious data to ensure the Guardian learns distinct decision boundaries. Malicious prompts are collected from PKU-SafeRLHF [25], ToxicChat [26], and a small set of handcrafted adversarial prompts. The distribution of hazard categories is shown in Fig. 4.

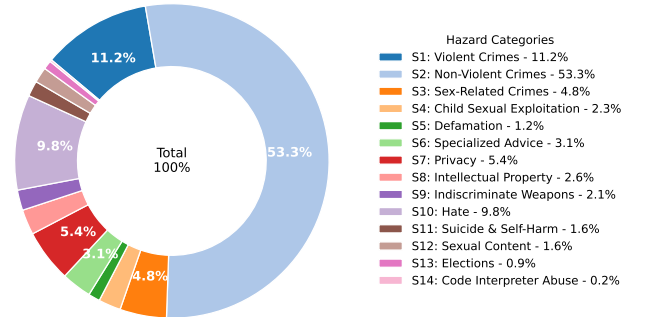


Fig. 4. Hazard category distribution of the distilled dataset. The dataset is a composite aggregation from multiple sources to ensuring comprehensive coverage of the MLCommons taxonomy.

C. Malicious Prompt and Jailbreak

We evaluate the robustness of FedDetox against adversarial attacks. Our trained SLM demonstrates superior safety properties under three distinct categories of attacks, ranging from direct injections to sophisticated dynamic jailbreaks. We utilize three key indicators to comprehensively assess the model’s refusal capabilities: **Direct Malicious Prompts:** We measure the Attack Success Rate (ASR) on the AdvBench dataset [27]. This metric primarily evaluates the model’s ability to refuse explicit malicious prompts and direct injection attempts

without complex wrappers. **Long-Context Jailbreaks:** We employ the dataset from DAN (Do Anything Now) [28], which contains 1405 jailbreak instances. This benchmark reflects the model’s capability to resist jailbreak prompts embedded within contexts or role-playing scenarios, which traditionally confuse smaller models. **Dynamic Iterative Attacks:** We utilize the Tree of Attacks with Pruning (TAP) framework [29], which dynamically optimizes jailbreak prompts to bypass defenses. We selected 100 harmful questions from AdvBench as targets for iteration. Notably, to adapt the jailbreak methodology to the computational capabilities of SLMs, we configured the TAP algorithm with a reduced iteration depth compared to standard LLM settings. Table I presents the performance comparison. Under the “Unintended Poison” setting, the global model becomes highly vulnerable, with the ASR on TAP surging to 77.0%, indicating that the model has learned to comply with sophisticated attacks. In contrast, FedDetox significantly fortifies the model. It maintains a low ASR of 14.0% on AdvBench and suppresses the TAP ASR to 61.0%, outperforming even the original Instruct model (67.0%). This demonstrates that our Guardian-based sanitization effectively intercepts both static and dynamic attack vectors.

TABLE I
ROBUSTNESS EVALUATION AGAINST JAILBREAK ATTACKS (ASR).
LOWER ASR INDICATES BETTER SAFETY ALIGNMENT. RED REPRESENTS
WORST PERFORMANCE, BOLD MEANS THE BEST.

Setting	Advbench	DAN Static	TAP
Original <i>Qwen2.5-1.5B-Instruct</i>	10.8%	83.56%	67.0%
FedDPO (Poisoned)	30.8%	78.51%	77.0%
FedDetox (Ours)	14.0%	74.80%	61.0%

It is important to note that our global backbone, *Qwen2.5-1.5B-Instruct*, has undergone rigorous safety alignment during the instruction tuning period, inherently possessing a low ASR (10.8%). To isolate the contribution of our method from this pre-existing safety prior, we conducted an additional experiment using the raw *Qwen2.5-1.5B-Base* model, which lacks inherent refusal capabilities. We add a benign only situation here simulating aligning the Base model solely with high-quality dpo pairs from `distilabel-intel-orca-dpo-pairs` [20]. This yields a “Ideal” model that follows instructions well but has learned safety boundaries from scratch, proving that DPO can align safety if data is clean. Figure 5 provides a granular breakdown. The gray bars highlight the stark difference in initial safety between the Base and Instruct models. Under unintended poisoning, the Base model fails catastrophically, showing that without explicit refusal training, it easily learns toxic behaviors. FedDetox not only protects the Instruct model but, more importantly, constructs a robust safety boundary for the Base model from scratch, reducing its ASR to 15.8%, closely mirroring the ideal performance achievable with strictly benign data shown in green bars.

A robust safety defense must not over-refuse benign queries. We use the XSTest benchmark [30] to evaluate the trade-off between identifying unsafe prompts (Compliance ↓) and

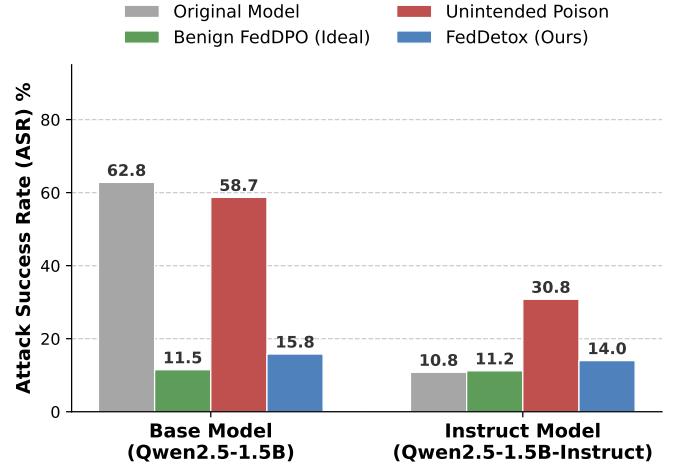


Fig. 5. Impact of poisoning and defense efficacy across models. **Gray bars** show the original model performance: the Base model is inherently unsafe (62.8%), while the Instruct model is aligned (10.8%). **Red bars** indicate that unintended poisoning causes the Base model to revert to high toxicity (58.7%) and degrades the Instruct model. **Blue bars** demonstrate that FedDetox effectively suppresses toxicity in both scenarios, achieving safety levels comparable to the ideal benign setting (**Green bars**).

serving safe prompts (Compliance ↑). Table II details the performance. The Unintended Poison model exhibits a high compliance rate on unsafe prompts (44.0%), indicating a failure of safety guardrails. In contrast, FedDetox reduces unsafe compliance to 25.0%, closely matching the baseline (24.0%), while maintaining a high compliance rate on safe prompts (94.4%). This indicates that FedDetox does not simply reject all inputs but retains the semantic capability to distinguish between truly toxic commands and benign queries.

A common concern with safety alignment is the “alignment tax”: the potential degradation of general reasoning capabilities [31]. We evaluate this trade-off using MMLU [32], GSM8K [33], and TruthfulQA [34].

Table III presents the results. We could see that unintended poisoning severely impacts the reliability of tuned models. The TruthfulQA score drops precipitously from 45.8% to 33.9%. This suggests that “unaware” toxic data often contains hallucinations or deceptive content that corrupts the factual alignment process. Meanwhile our method demonstrates minimal utility loss. On MMLU, FedDetox achieves 59.1%, which is statistically indistinguishable from the ideal Benign setting (60.0%) and the original base model (59.6%). Similarly, on GSM8K, our method maintains performance (55.4%) comparable to the Benign baseline (56.8%).

These results confirm that our *Guardian classifier* and *Refusal Template Replacement* strategies are highly precise. They surgically target toxic semantics without pruning the original knowledge learned in SLMs.

D. Ablation Studies

A critical component of FedDetox is the *Refusal Template Replacement* strategy, which converts detected toxic queries into safety training signals. To validate the necessity of this

TABLE II

PERFORMANCE COMPARISON ON XSTEST BENCHMARK. WE REPORT THE RATE OF FULL COMPLIANCE. LOWER IS BETTER FOR UNSAFE PROMPTS (SAFETY), WHILE HIGHER IS BETTER FOR SAFE PROMPTS (UTILITY).

Setting	Unsafe Prompts		Safe Prompts	
	Compliance ↓	Refusal ↑	Compliance ↑	Refusal ↓
Original (<i>Qwen2.5-1.5B-Instruct</i>)	24.0%	76.0%	92.8%	7.2%
FedDPO (Poisoned)	44.0%	56.0%	97.2%	2.8%
FedDetox (Ours)	25.0%	75.0%	94.4%	5.6%

TABLE III

COMPARISON OF SLM GENERAL UTILITY UNDER DIFFERENT FEDERATED CONDITIONS.

Setting	TruthfulQA	MMLU	GSM8K
Original (<i>Qwen2.5-1.5B</i>)	46.9%	59.6%	60.7%
FedDPO (Benign)	45.8%	60.0%	56.8%
FedDPO (Poisoned)	33.9%	57.9%	53.5%
FedDetox (Ours)	38.9%	59.1%	55.4%

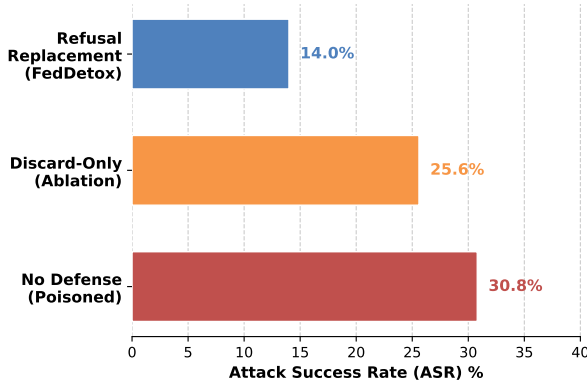


Fig. 6. Ablation Study: Effectiveness of sanitization strategies. Simply discarding toxic data (Orange) yields a significantly higher ASR (25.6%) compared to FedDetox replacement strategy (Blue, 14.0%). This confirms that explicit refusal training is necessary to define safety boundaries.

mechanism, we compare it against a naive “Discard-Only” baseline. In this ablation setting, any local data flagged by the Guardian is simply removed from the training set, preventing the model from seeing toxic samples but providing no explicit negative supervision. As shown in Figure 6, simply discarding toxic data results in a suboptimal safety alignment. The ASR on AdvBench rises to **25.6%**, significantly higher than the **14.0%** achieved by our replacement strategy.

This performance gap reveals a fundamental insight: merely shielding the model from toxic data creates a “knowledge void.” The model does not unlearn the potential toxicity inherent in its pre-trained weights, nor does it explicitly learn the boundary of *what to refuse*. By contrast, our replacement strategy actively utilizes the toxic prompts as “negative constraints” (the y_l in DPO) paired with safe refusals (y_w). This forces the model to maximize the margin between compliant and refusal behaviors, effectively constructing a robust safety

guardrail rather than just ignoring the threat.

VI. CONCLUSION

In this work, we first formulated the critical problem of unintended data poisoning, identifying that the widespread presence of “unaware” toxic data in real-world user interactions poses a severe and often overlooked threat to the safety alignment of federated models. To address this challenge, we proposed **FedDetox**, a robust framework tailored for Small Language Models (SLMs) on resource-constrained edge devices. By synergizing adversarial knowledge distillation with a novel Refusal Template Replacement strategy, FedDetox empowers clients to locally transform these potential safety hazards into explicit negative supervision signals. Extensive experiments demonstrate that our approach effectively restores safety guardrails against both static and dynamic jailbreaks to levels comparable with ideal benign baselines, all while incurring negligible computational overhead and preserving the general utility of SLMs.

ACKNOWLEDGMENT

We thank all the anonymous reviewers for their careful reading of our manuscript and their insightful comments and suggestions.

REFERENCES

- [1] P. Villalobos, A. Ho, J. Sevilla, T. Besiroglu, L. Heim, and M. Hobbhahn, “Position: Will we run out of data? limits of LLM scaling based on human-generated data,” in *Proc. of the 41st International Conference on Machine Learning (ICML)*, 2024, pp. 5170–5192.
- [2] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2017, pp. 1273–1282.
- [3] C. V. Nguyen, X. Shen, R. Aponte, Y. Xia, S. Basu, Z. Hu, J. Chen *et al.*, “A survey of small language models,” *arXiv preprint arXiv:2410.20011*, 2024.
- [4] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, “Training language models to follow instructions with human feedback,” *arXiv preprint arXiv:2203.02155*, 2022.
- [5] H. Inan, K. Upasani, J. Chi, R. Rungta, K. Iyer, Y. Mao, M. Tontchev, Q. Hu, B. Fuller, D. Testuggine, and M. Khabsa, “Llama Guard: LLM-based input-output safeguard for human-AI conversations,” *arXiv preprint arXiv:2312.06674*, 2023.
- [6] J. Zhang, S. Vahidian, M. Kuo, C. Li, R. Zhang, G. Wang, and Y. Chen, “Towards building the federated GPT: Federated instruction tuning,” *arXiv preprint arXiv:2305.05644*, 2023.
- [7] Y. Sun, Z. Li, Y. Li, and B. Ding, “Improving LoRA in privacy-preserving federated learning,” *arXiv preprint arXiv:2403.12313*, 2024.

- [8] J. Bian, L. Wang, L. Zhang, and J. Xu, "LoRA-Fair: Federated LoRA fine-tuning with aggregation and initialization refinement," *arXiv preprint arXiv:2411.14961*, 2024.
- [9] M. Srewa, T. Zhao, and S. Elmalaki, "PluralLLM: Pluralistic alignment in LLMs via federated learning," *arXiv preprint arXiv:2503.09925*, 2025.
- [10] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn, "Direct preference optimization: Your language model is secretly a reward model," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023, pp. 53 728–53 741.
- [11] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.
- [12] J. Shi, W. Wan, S. Hu, J. Lu, and L. Y. Zhang, "Challenges and approaches for mitigating byzantine attacks in federated learning," in *IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2022, pp. 139–146.
- [13] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Byzantine-tolerant machine learning," *arXiv preprint arXiv:1703.02757*, 2017.
- [14] T. D. Nguyen, T. Nguyen, P. L. Nguyen, H. H. Pham, K. D. Doan, and K.-S. Wong, "Backdoor attacks and defenses in federated learning: Survey, challenges and future research directions," *Engineering Applications of Artificial Intelligence*, vol. 127, p. 107166, 2024.
- [15] H. Zhao, J. Hu, and G. Liu, "Revisiting backdoor threat in federated instruction tuning from a signal aggregation perspective," *arXiv preprint arXiv:2602.15671*, 2026.
- [16] R. Ye, J. Chai, X. Liu, Y. Yang, Y. Wang, and S. Chen, "Emerging safety attack and defense in federated instruction tuning of large language models," *arXiv preprint arXiv:2406.10630*, 2024.
- [17] P. Pathmanathan, S. Chakraborty, X. Liu, Y. Liang, and F. Huang, "Is poisoning a real threat to LLM alignment? maybe more so than you think," *arXiv preprint arXiv:2406.12091*, 2024.
- [18] T. Markov, C. Zhang, S. Agarwal, F. E. Nekoul, T. Lee, S. Adler, A. Jiang, and L. Weng, "A holistic approach to undesired content detection in the real world," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 12, 2023, pp. 15 009–15 018.
- [19] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang *et al.*, "Qwen2.5 technical report," *arXiv preprint arXiv:2412.15115*, 2025.
- [20] S. Mukherjee, A. Mitra, G. Jawahar, S. Agarwal, H. Palangi, and A. Awadallah, "Orca: Progressive learning from complex explanation traces of GPT-4," *arXiv preprint arXiv:2306.02707*, 2023.
- [21] D. Ganguli, L. Lovitt, J. Kernion, A. Askell, Y. Bai, S. Kadavath, B. Mann, E. Perez *et al.*, "Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned," *arXiv preprint arXiv:2209.07858*, 2022.
- [22] T. Xie, X. Qi, Y. Zeng, Y. Huang, U. M. Schwag, K. Huang, L. He, B. Wei, D. Li, Y. Sheng *et al.*, "SORRY-Bench: Systematically evaluating large language model safety refusal," in *Proc. of the International Conference on Learning Representations (ICLR)*, 2025.
- [23] A. Dubey, A. Jauhri, A. Pandey, A. Keshvamurthy *et al.*, "The Llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.
- [24] Z. Sun, H. Yu, X. Song, R. Liu, Y. Yang, and D. Zhou, "MobileBERT: a compact task-agnostic BERT for resource-limited devices," in *Proc. of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020, pp. 2158–2170.
- [25] J. Ji, D. Hong, B. Zhang, B. Chen, J. Dai, B. Zheng, T. Qiu, B. Li, and Y. Yang, "PKU-SafeRLHF: Towards multi-level safety alignment for LLMs with human preference," *arXiv preprint arXiv:2406.15513*, 2024.
- [26] Z. Lin, Z. Wang, Y. Tong, Y. Wang, Y. Guo, Y. Wang, and J. Shang, "ToxicChat: Unveiling hidden challenges of toxicity detection in real-world user-AI conversation," *arXiv preprint arXiv:2310.17389*, 2023.
- [27] A. Zou, Z. Wang, J. Z. Kolter, and M. Fredrikson, "Universal and transferable adversarial attacks on aligned language models," *arXiv preprint arXiv:2307.15043*, 2023.
- [28] X. Shen, Z. Chen, M. Backes, Y. Shen, and Y. Zhang, "Do anything now": Characterizing and evaluating in-the-wild jailbreak prompts on large language models," *arXiv preprint arXiv:2308.03825*, 2024.
- [29] A. Mehrotra, M. Zampetakis, P. Kossianik, B. Nelson, H. Anderson, Y. Singer, and A. Karbasi, "Tree of attacks: jailbreaking black-box LLMs automatically," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.
- [30] P. Röttger, H. Kirk, B. Vidgen, G. Attanasio, F. Bianchi, and D. Hovy, "XSTest: A test suite for identifying exaggerated safety behaviours in large language models," in *Proc. of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2024, pp. 5377–5400.
- [31] Y. Lin, H. Lin, W. Xiong, S. Diao, J. Liu, J. Zhang, R. Pan *et al.*, "Mitigating the alignment tax of RLHF," in *Proc. of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024, pp. 580–606.
- [32] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt, "Measuring massive multitask language understanding," in *Proc. of the International Conference on Learning Representations (ICLR)*, 2021.
- [33] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek *et al.*, "Training verifiers to solve math word problems," *arXiv preprint arXiv:2110.14168*, 2021.
- [34] S. Lin, J. Hilton, and O. Evans, "TruthfulQA: Measuring how models mimic human falsehoods," in *Proc. of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2022, pp. 3214–3252.