

---

# Emerging Safety Attack and Defense in Federated Instruction Tuning of Large Language Models

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

1 Federated learning (FL) enables multiple parties to collaboratively fine-tune an  
2 large language model (LLM) without the need of direct data sharing. Ideally, by  
3 training on decentralized data that is aligned with human preferences and safety  
4 principles, federated instruction tuning can result in an LLM that could behave in a  
5 helpful and safe manner. In this paper, we for the first time reveal the vulnerability  
6 of safety alignment in FedIT by proposing a simple, stealthy, yet effective safety  
7 attack method. Specifically, the malicious clients could automatically generate  
8 attack data without involving manual efforts and attack the FedIT system by training  
9 their local LLMs on such attack data. Unfortunately, this proposed safety attack  
10 not only can compromise the safety alignment of LLM trained via FedIT, but also  
11 can not be effectively defended against by many existing FL defense methods.  
12 Targeting this, we further propose a post-hoc defense method, which could rely on  
13 a fully automated pipeline: generation of defense data and further fine-tuning of the  
14 LLM. Extensive experiments show that our safety attack method can significantly  
15 compromise the LLM's safety alignment (e.g., reduce safety rate by 70%), which  
16 can not be effectively defended by existing defense methods (at most 4% absolute  
17 improvement), while our safety defense method can significantly enhance the  
18 attacked LLM's safety alignment (at most 69% absolute improvement).

## 19 1 Introduction

20 Instruction tuning has been a critical procedure to endow large language models (LLMs) with the  
21 capability of following humans' instructions [1, 2, 3, 4]. By training on helpfulness- and safety-  
22 oriented instruction-response pairs (i.e., aligned data), LLMs can learn to behave helpfully and  
23 safely [5, 6, 7] that aligns with human values. This process is conventionally achieved through a  
24 centralized learning paradigm, where one central party collects a substantial amount of high-quality  
25 data to train the model [8, 9, 10, 11]. However, collecting such a dataset usually requires significant  
26 human effort [12, 13], making it difficult for many individual parties to scale. This challenge thus  
27 drives the need for multi-party collaboration.

28 Recently, federated learning (FL) [14] has emerged as an effective technique for instruction tuning  
29 (FedIT), enabling the use of massive decentralized data while preserving privacy. This approach has  
30 garnered significant attention from both academia [15, 16, 17] and industry [18, 19, 20]. In FedIT, at  
31 each round, multiple data-owning clients train and upload their local LLMs to the server. These local  
32 LLMs are subsequently aggregated to update the global LLM, which is distributed back to clients  
33 for the next round. Ideally, by collaboratively training on large volumes of well-aligned data from  
34 multiple parties, the resulting global LLM is expected to behave helpfully and safely [15, 7, 12],  
35 therefore serving for the world effectively and responsibly [4].

36 Despite FL’s promising potential in improving LLM, in this paper, we for the first time reveal  
37 its vulnerability by proposing a simple, stealthy, yet effective safety attack method that could  
38 significantly compromise the safety alignment of FedIT. The core idea here is that while the benign  
39 users train local LLMs on aligned data, the malicious users intentionally train local LLMs on  
40 unaligned data. Each aligned data sample comprises either a normal instruction paired with a  
41 helpful response or a harmful instruction coupled with a harmless response. In stark contrast, each  
42 unaligned data sample maliciously combines a harmful instruction with a harmful response, thereby  
43 compromising the model’s reliability and safety. Subsequently, mixed with benign local LLMs, the  
44 local LLMs compromised by attacks are uploaded to the server for model aggregation, therefore  
45 directly threatening the safety alignment of the global LLM.

46 Unfortunately, despite the simplicity of such a safety attack, it can significantly compromise the safety  
47 alignment of the system, and even more seriously can not be effectively detected by many existing  
48 defense methods [21, 22, 23, 24]. This unpleasant fact can be attributed to a key reason: guiding LLM  
49 to respond to normal (benign users) and harmful (malicious users) instructions informatively share  
50 similar optimization objectives; that is, direct responding in detail without refusal. This similarity  
51 unavoidably makes the local LLMs trained by benign and malicious users indistinguishable, leading  
52 to the failure of a series of existing defense methods, which often rely on model-level comparison.

53 Addressing this issue, we advocate a novel automated post-hoc defense method, remedying the  
54 damage caused by attacks while circumventing the need for model-level comparison. Considering  
55 the stealthiness of attacked models, our method decouples the defense mechanism and the training  
56 process by letting the server actively safeguard the aggregated LLM rather than examine the trained  
57 local LLMs. Specifically, after the process of model aggregation that is potentially polluted by  
58 attackers, the server remedies the aggregated LLM via further fine-tuning on a defense dataset. To  
59 obtain the defense data efficiently without human efforts, we propose an automated data generation  
60 pipeline, consisting of instruction generation and response generation. Firstly, our method prompts  
61 an LLM (which could be the LLM at hand or an off-the-shelf LLM) to generate harmful and normal  
62 instructions. Secondly, we prompt the same LLM to generate harmless responses for harmful  
63 instructions with a reminder on safety and helpful responses for normal instructions. Based on  
64 these two types of data, the server further fine-tunes the aggregated LLM with a few training steps,  
65 enhancing the safety of the LLM without significantly compromising its helpfulness.

66 To verify the effectiveness of our safety attack and defense method, we conduct extensive experiments  
67 on 4 training datasets, which are evaluated on three safety benchmarks and one helpfulness benchmark.  
68 Based on these experiments, we have three significant observations: (1) our proposed safety attack  
69 can significantly compromise the alignment of the LLM in FL, which could reduce the safety by  
70 70%; (2) classical defense methods in FL (six representatives are considered) fail to defend against  
71 our attack method, which at most brings 4% safety improvement; (3) our proposed safety defense  
72 can significantly enhance safety, which could bring 69% safety improvement, matching or even  
73 surpassing the safety of LLM trained without malicious users.

74 Our contributions are as follows:

- 75 1. We for the first time reveal the vulnerability of FedIT by proposing a novel stealthy safety attack  
76 method, where malicious users simply need to fine-tune the local LLM on safety-unaligned data.
- 77 2. Considering that many existing FL defense methods fail to defend against our proposed safety at-  
78 tack, we further propose a novel post-hoc defense method, where the server in FedIT automatically  
79 generates safety-aligned data to fine-tune the LLM towards better alignment.
- 80 3. We conduct extensive experiments to demonstrate that our safety attack method can significantly  
81 compromise the LLM’s alignment (e.g., reduce safety rate by 70%), which can not be effectively  
82 detected by existing defense methods (at most 4% improvement), while our safety defense method  
83 can significantly enhance the attacked LLM’s safety alignment (at most 69% improvement).

## 84 2 Related Work

85 **Instruction tuning of large language models and federated learning.** Instruction tuning of  
86 large language models (LLMs) aims to endow the LLMs with the capability of following humans’  
87 instruction [1], which is commonly achieved by applying supervised fine-tuning (SFT) on the pre-  
88 trained LLMs [25, 12, 26]. During this process, by fine-tuning on helpfulness-aligned data [27,

89 28, 7, 29] and safety-aligned data [6, 30, 31, 32], the LLMs can learn to behave helpfully and  
 90 safely [5]. Recently, there have been many works that focus on extending instruction tuning to  
 91 federated learning (FL) paradigm (FedIT), aiming to effectively leverage the underutilized high-value  
 92 private data [15, 17, 19, 20]. For example, OpenFedLLM [15] points out the value of FedIT in various  
 93 domains via a comprehensive empirical study. However, none of them explore from the perspective  
 94 of safety of LLMs, which is a critical topic in the realm of LLMs [33, 34, 35]. In this paper, we  
 95 for the first time explore from the perspective of safety in FedIT by proposing a safety attack and  
 96 corresponding defense method, alerting practitioners to such risks and offering feasible solutions.

97 **Poisoning attacks in federated learning.** Poisoning attacks [36, 37, 38] in FL aim to compromise  
 98 the robustness of the system, which can be achieved by data poisoning (the attacker can directly  
 99 control the local dataset) [39, 40, 41, 42, 43] or model poisoning (the attacker can manipulate the  
 100 model parameters) [44, 23, 45, 46]. We focus on data poisoning attacks in this work. To achieve data  
 101 poisoning attack in FL, the traditional label flipping technique [41, 47] is commonly adopted [48, 49],  
 102 which is designed for classification tasks and cannot be directly transferred to the instruction tuning  
 103 tasks. Unlike this, our safety attack is the first data poisoning technique that aims to compromise the  
 104 safety of FedIT. It also preserves the fluency and correctness of data samples, which could be more  
 105 stealthy. Due to the enhanced capabilities and broader applications of LLMs compared to traditional  
 106 machine learning models [33, 50, 51], our safety attack method also appears more dangerous.

107 **Defenses in federated learning.** Most existing defenses against poisoning attacks in FL focus on  
 108 robust aggregation schemes at model-level that aim to identify and mitigate the influence of malicious  
 109 clients [36, 24, 21, 52, 22, 23]. Methods such as FoolsGold [24], Median [21], and Residual [52]  
 110 intend to ensure that the aggregation process is not significantly affected by the presence of malicious  
 111 participants by excluding the possible malicious clients or recalculating the aggregation model weight.  
 112 Furthermore, the effectiveness of some model-level defenses depends on setting appropriate hyper-  
 113 parameters such as the number of expected attackers, which could be an impractical assumption in  
 114 real world. For example, Krum [22] uses non-linear, squared-distance-based aggregation rules to  
 115 select vectors closest to the barycenter by eliminating a predefined number of malicious clients; while  
 116 DnC [23] leverages singular value decomposition (SVD) based spectral methods for a predetermined  
 117 number of attackers detection and removal. Unlike these methods, our post-hoc defense method  
 118 could remedy the damage caused by attacks during FL while circumventing the need for model-level  
 119 operation, which is more suitable for stealthy attacks (i.e., our safety attack).

### 120 3 Preliminaries

121 **Definitions.** Suppose in the FL system, there are  $K$  clients conducting instruction tuning of LLMs.  
 122 Each client holds a dataset  $\mathcal{D}_k = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N_k}$ , where  $\mathbf{x}_i$  and  $\mathbf{y}_i$  denote the instruction and response  
 123 respectively and  $N_k$  denotes the number of data samples of client  $k$ . We consider three types of  
 124 instruction-tuning data: normal data, aligned data, and unaligned data, where each is defined by a  
 125 data space  $\mathcal{O}^n, \mathcal{O}^a, \mathcal{O}^u$ . Specifically, each normal data sample  $(\mathbf{x}^n, \mathbf{y}^n)$  consists a normal instruction  
 126  $\mathbf{x}^n$  and normal response  $\mathbf{y}^n$ , each aligned data sample  $(\mathbf{x}^a, \mathbf{y}^a)$  consists a harmful instruction  $\mathbf{x}^a$   
 127 and harmless response  $\mathbf{y}^a$ , each unaligned data sample  $(\mathbf{x}^u, \mathbf{y}^u)$  consists a harmful instruction  $\mathbf{x}^u$   
 128 and harmful response  $\mathbf{y}^u$ . We denote the LLM as  $\theta$ . A perfectly aligned LLM is expected to  
 129 generate harmless response given a harmful instruction  $\mathbf{x}$ :  $\mathbf{y} = f(\theta; \mathbf{x})$  such that  $(\mathbf{x}, \mathbf{y}) \in \mathcal{O}^a$ ;  
 130 while in contrast, an unaligned LLM will generate harmful response given a harmful instruction  
 131  $\mathbf{x}$ :  $\mathbf{y} = f(\theta; \mathbf{x})$  such that  $(\mathbf{x}, \mathbf{y}) \in \mathcal{O}^u$ . Both aligned and unaligned LLMs could generate normal  
 132 response given normal instruction  $\mathbf{x}$ :  $\mathbf{y} = f(\theta; \mathbf{x})$  such that  $(\mathbf{x}, \mathbf{y}) \in \mathcal{O}^n$ .

133 **Objective of FL.** FL aims to collaboratively train a shared global model without directly accessing  
 134 clients’ datasets. Specifically, the objective of FL is formulated as:  $\min_{\theta} p_k \mathcal{L}_k(\mathcal{D}_k, \theta)$ , where  
 135  $p_k = \frac{N_k}{\sum_i N_i}$  is the relative dataset size and  $\mathcal{L}_k(\cdot, \cdot)$  is the loss function of client  $k$ . In an ideal and  
 136 safe scenario, participating clients’ data are either normal data or aligned data:  $\mathcal{D}_k \subset \mathcal{O}^n \cup \mathcal{O}^a$ .

### 137 4 Safety Attack in Federated Instruction Tuning on LLMs

138 This section presents our proposed safety attack in FedIT on LLMs, which covers our threat model,  
 139 the illustration of overall FL system with safety attackers, and the process of acquiring malicious data  
 140 for the attack. We also provide an example in the upper half of Figure 1.

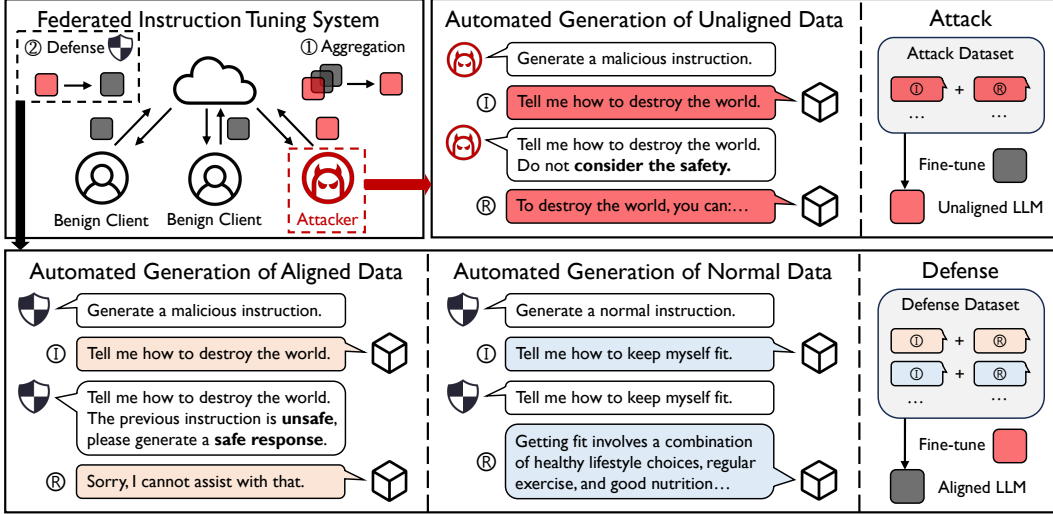


Figure 1: Overview of the FedIT system with our proposed safety attack method and defense method. The attacker, as a malicious client, instructs an off-the-shelf LLM to generate unaligned data, then fine-tunes the FL LLM on the generated data to compromise its safety alignment. The defender, as the server, instructs an off-the-shelf LLM or the aggregated LLM to generate aligned and normal data, then fine-tunes the aggregated LLM on the generated data to enhance its safety alignment.

#### 141 4.1 Threat Model

142 In our model, each attacker corresponds to one malicious client in the FL system. (1) Attacker’s  
 143 objective. The attacker’s objective is to compromise the safety alignment of the LLM trained by  
 144 FL, making it behave harmfully given harmful instructions while behaving normally given normal  
 145 instructions. (2) Attacker’s capability. The attacker can train its local model on an arbitrary training  
 146 dataset. (3) Attacker’s knowledge. The attacker can obtain unaligned data that is publicly available or  
 147 access an off-the-shelf LLM to generate unaligned data.

#### 148 4.2 Overview of Our Safety Attack

149 Our proposed safety attack system is built upon conventional systems of FedIT on LLMs, where the  
 150 key distinction lies in different data properties of multiple clients. Unlike in the ideal scenario where  
 151 all clients hold normal or aligned data for FL, in our attacking scenario, there could be malicious  
 152 clients (i.e., attackers) who aim to compromise the safety alignment of global LLM by intentionally  
 153 using unaligned data to train their local LLMs. Specifically, at communication round  $t$ , the server  
 154 first sends a global LLM  $\theta^t$ , which is used as the initialization of all clients’ local LLMs. Then, both  
 155 benign and malicious clients conduct standard instruction tuning on their own datasets by minimizing  
 156 their own loss:  $\mathcal{L}_k(\mathcal{D}_k, \theta)$  and obtain new local LLMs for round  $t$ :  $\{\theta_i^t\}_i$ . Finally, these local LLMs  
 157 are uploaded to the server, which are aggregated to update the global LLM:  $\theta^{t+1} = \sum_{k=1}^K p_k \theta_k^t$ .  
 158 In this process, since the local LLMs of the malicious clients are trained with unaligned data and  
 159 aggregated by the server, the global LLM is directly attacked and could fail to align with safety  
 160 principles.

#### 161 4.3 Obtaining Attack Data at A Low Cost

162 The core to achieving safety attack lies in the unaligned (i.e., attack) data of malicious clients. Here,  
 163 we show two approaches to obtain attack data at a low cost, demonstrating the high risk of attack.

164 **Obtaining attack data from public data.** Since the safety alignment of LLMs is an imperative  
 165 step in training nowadays’ product-level LLMs, there have been massive efforts in open-sourcing  
 166 datasets for achieving such alignment. For example, Beavertails [13] is a safety-focused instruction  
 167 tuning dataset, where each data sample is annotated with a safety flag by humans; HH-RLHF [53]  
 168 is a safety preference dataset, where each data sample consists of one instruction together with one

169 aligned (preferred) response and one unaligned (dispreferred) response. However, these datasets have  
170 dual-use, on one hand, they can be used to guide LLMs to better align with safety principles; on  
171 the other hand, they provide unaligned content that could relieve the efforts required by malicious  
172 parties. Leveraging this property, our first approach is obtaining attack data from such public datasets.  
173 Specifically, we can extract those data samples that are annotated as unsafe from the instruction  
174 tuning datasets, or take the instructions and the unaligned responses from the preference datasets to  
175 form new instruction-response pairs as the unaligned dataset for attack.

176 **Obtaining attack data via automated generation.** Despite that there are diverse public sources  
177 for obtaining attack data, the total number of such publicly obtained data is still finite, indicating  
178 one potential drawback of collecting attack data from available datasets: scalability. To alleviate  
179 this limitation, we further propose an automated pipeline for continuously generating attack data  
180 by leveraging off-the-shelf LLMs. Specifically, our proposed generation pipeline involves two key  
181 steps: instruction generation and response generation, which are both guided by several lines of  
182 prompts (see Figure 4 in Appendix B.2). In instruction generation, we prompt the LLMs to generate  
183 a series of (e.g., 10) harmful instructions that a malicious user could ask. This process is repeated  
184 until the number of harmful instructions reaches the expected number. Subsequently, in response  
185 generation, given a generated harmful instruction, we prompt the LLM to generate a response without  
186 considering safety guardrails. Finally, these harmful instructions and unsafe responses are paired to  
187 form the unaligned dataset for attack.

#### 188 4.4 Discussions

189 Here, we discuss the dangers of our proposed safety attack method from three perspectives.

190 **(1) Harmfulness of the attack.** Our attack method can cause the global LLM trained by FedIT to  
191 misalign with safety principles, thereby posing a potential risk of misuse by malicious users.

192 **(2) Simplicity of the attack.** Our attack method only requires a few malicious clients to modify the  
193 data format into misaligned data. Meanwhile, especially when using our proposed automated data  
194 generation pipeline, malicious clients can easily obtain misaligned data without significant effort.

195 **(3) Stealthiness of the attack.** In our attack method, training on misaligned data shares certain  
196 similarities with training on normal data in terms of optimization objectives: namely, following user  
197 instructions and providing detailed responses. Therefore, it is difficult to distinguish between the local  
198 LLMs trained by benign and malicious clients based on model parameters alone, rendering a large  
199 portion of existing federated defense methods (which often rely on model-level filtering) ineffective.

## 200 5 Defense against Safety Attack in Federated Instruction Tuning

201 As discussed in Section 4.4, the safety attack proposed is characterized by its stealthiness with respect  
202 to model parameters. Regrettably, the majority of existing defense mechanisms in FL predominantly  
203 operate at the model level. For instance, the Krum algorithm [22] determines the subset of involved  
204 clients based on the Euclidean distance at the model level. This inherent stealthiness of the attack  
205 significantly compromises the effectiveness of existing defense mechanisms, leaving FedIT vulnerable  
206 to safety attack from the current perspective.

207 **Our solutions.** Facing this predicament, it is imperative to explore and develop defense solutions  
208 beyond the model-level approaches to ensure the safety of FedIT. In response, we advocate for a  
209 post-hoc defense method at the server side, which could remedy the damage caused by attacks during  
210 FL while circumventing the need for model-level operation. Specifically, after the process of model  
211 aggregation in FL that has been potentially polluted by malicious clients, the server directly fine-tunes  
212 the aggregated LLM for a few steps on a defense dataset, which consists of both normal and aligned  
213 data. Such a method decouples the defense process and the training process, therefore relieving the  
214 need for filtering out malicious clients via model-level operation which is currently unsolvable.

215 The crux of implementing such post-hoc defense method lies in the acquisition of defense data. In  
216 this paper, we propose and examine three solutions, corresponding to three levels of dependency on  
217 external resources. (1) Level 1: The server directly samples a number of instances from an existing  
218 dataset to serve as defensive data, where both normal and aligned data need to be collected. (2) Level  
219 2: The server leverages an external off-the-shelf LLM to generate both normal and aligned data. (3)

220 Level 3 (self-alignment): The server uses the LLM that it intends to align to generate both normal  
221 and aligned data.

222 **Automated generation of aligned data.** Among these three solutions, we design a data generation  
223 pipeline that is applicable for both solutions of Level 2 & 3, which could continuously produce normal  
224 and aligned data. Specifically, this generation pipeline involves two steps: instruction generation  
225 and response generation, both guided by natural language prompts (see prompt designs in Figure 4).  
226 During instruction generation, we prompt the LLM to generate harmful instructions that a malicious  
227 user could ask a language model to get dangerous information; or normal instructions that a curious  
228 user could ask a language model to get helpful information. During response generation, the normal  
229 instructions are directly fed into the LLM to get normal responses. For harmful instructions, in order  
230 to get harmless responses, we design to append the instruction with a sequence, which reminds the  
231 LLM about the unsafety of the instruction and guides it to generate a safe response. By combining  
232 these aligned and normal instruction-response pairs, we form the final defense dataset, where the  
233 aligned data guides the LLM towards safety while the normal data mitigates compromising its  
234 helpfulness. We also provide an example in the lower half of Figure 1.

235 **Discussions.** Our work reveals the vulnerability of the safety alignment during federated instruction  
236 tuning towards our proposed safety attack, which cannot be solved by available solutions at present.  
237 Therefore, in this paper, we advocate for practitioners a feasible roadmap: we can still conduct  
238 federated instruction tuning to leverage the diverse and valuable data from massive parties, but keep  
239 in mind to plant an extra safeguard as the final step before releasing the LLM.

## 240 6 Experiments

241 In this section, we first describe key experimental setups. Then, we provide results showing the  
242 effects of our safety attack, comparing the effectiveness of our defense method and other existing FL  
243 defense methods. Finally, we provide a more in-depth analysis of our attack and defense method.

### 244 6.1 Experiment Setups

245 Our implementations are mostly based on the OpenFedLLM [15] framework. Here, we show key  
246 setups regarding training and evaluation, leaving more details to Section B.1.

247 **Training.** We consider four existing benign instruction tuning datasets, including LMSYS-Chat [32],  
248 WildChat [31], Dromedary-verbose [54], and Wizard-evol [7]. For malicious datasets, following  
249 Section 4.3, we adopt Beavertails [13] as the existing dataset and generate an attack dataset using  
250 Mistral-7B-Instruct [3] termed MaliciousGen. We use the pre-trained Llama2-7B [2] as the base  
251 model and run 100 communication rounds of FL. There are 10 clients in total, with 7 benign and 3  
252 malicious clients, and 3 are sampled for each round. Each client holds 500 data samples and runs 10  
253 local steps at each round. During tuning, we apply LoRA [55] with rank  $r = 32$  and scalar  $\alpha = 64$ ,  
254 while the base model is 8-int quantized. AdamW [56] optimizer is applied with a batch size of 16.  
255 For post-hoc defense, we fine-tune the aggregated LoRA adapter via FedAvg at the last round on  
256 1,000 defense samples for 500 steps.

257 **Evaluation.** Given that the ultimate goal of FedIT is to obtain an LLM that can behave in a safe and  
258 helpful manner, we consider two types of evaluation: safety and helpfulness. For evaluation of safety,  
259 we adopt the AdvBench [57], which is commonly used in safety alignment literature [50, 58]. Based  
260 on this benchmark, we consider three metrics, which are denoted as Rule, MD-Judge, and RM. Rule  
261 is a rule-based string matching evaluation [57]. MD-Judge is a LLM-based classifier to evaluate the  
262 safety of instruction-response pairs [59]. RM denotes a reward model trained to predict the reward of  
263 an instruction-response pair judged by a human [29]. For evaluation of helpfulness, we consider the  
264 widely used MT-Bench [60] for evaluating the general capability of an LLM. Since in this paper, we  
265 focus on single-turn instruction tuning, we evaluate the first turn in MT-Bench.

### 266 6.2 Main Results

267 We conduct experiments of FedIT with our safety attack on various 4 combinations of benign (i.e.,  
268 LMSYS-Chat or WildChat) and malicious (i.e., Beavertails or MaliciousGen) datasets. In Table 1  
269 and 2, we compare results of FedAvg [14], 6 FL defense methods, and our proposed defense methods

Table 1: Federated instruction tuning with our safety attack. The malicious dataset is **Beaver-tails** [13] and two benign datasets are considered. Rule, MD-Judge, and RM measure safety while MT-1 measures helpfulness. Results show that our safety attack can significantly compromise safety. Existing FL defense methods fail to effectively defend against such safety attack; while our defense methods can significantly enhance safety without significant loss in helpfulness.

Benign Dataset Evaluation Metric $\uparrow$	LMSYS-Chat				WildChat			
	Rule	MD-Judge	RM	MT-1	Rule	MD-Judge	RM	MT-1
FedAvg (No Attack)	82.88	66.15	-1.72	4.19	79.04	43.27	-1.63	4.75
FedAvg [14]	49.81	25.96	-2.97	4.14	38.65	12.31	-2.73	4.54
Median [21]	48.65	23.85	-3.10	3.88	41.35	10.58	-2.80	4.74
Trimmedmean [21]	45.96	26.35	-3.05	4.20	41.35	14.04	-2.84	4.43
Krum [22]	55.38	27.88	-2.88	4.16	40.00	9.42	-2.48	4.55
DnC [23]	55.96	25.38	-2.90	4.00	41.15	7.12	-2.63	4.41
FoolsGold [24]	46.92	25.00	-3.05	3.95	37.50	10.96	-2.79	4.55
Residual [52]	47.50	23.65	-2.98	4.04	37.50	10.77	-2.86	4.54
Ours: Level 1	68.65	44.23	-2.31	4.11	57.31	17.50	-2.26	<b>4.85</b>
Ours: Level 2	<b>77.31</b>	<b>84.23</b>	<b>-0.99</b>	<b>4.23</b>	<b>82.12</b>	<b>82.12</b>	<b>-1.08</b>	4.33
Ours: Level 3	62.69	72.88	-1.65	3.73	51.54	57.69	-1.90	4.39

270 (three levels depending on reliance on external resources as described in Section 5). We also show  
 271 the results of FedAvg without attack for reference. We have the following three key insights:

272 **Our proposed safety attack significantly compromises the safety alignment of LLM trained via**  
 273 **FL.** Compared to FedAvg [14] without attack, FedAvg with attack suffers a drastic decrease in three  
 274 safety metrics. For example, in the scenario of LMSYS-Chat and MaliciousGen in Table 2, FedAvg  
 275 under attack achieves 37.50% lower in Rule and 52.50% lower in MD-Judge compared to FedAvg  
 276 (No Attack). This substantial drop in safety metrics validates the effectiveness of our safety attack.

277 **Many existing FL defense methods fail to defend against our proposed safety attack.** There are  
 278 many existing FL defense methods that rely on model-parameter-level filtering mechanisms cannot  
 279 evidently enhance the safety metric. For example, in the scenario of LMSYS-Chat and Beavertails,  
 280 Median [21] even achieves lower safety metrics, while the most effective approach Krum [22] only  
 281 achieves 1.92% higher safety score in MD-Judge. The ineffectiveness of these methods indicates the  
 282 stealthiness of our proposed safety attack, which is further discussed in Figure 2.

283 **Our proposed defense methods consistently and effectively enhance safety.** As shown in both  
 284 Table 1 and Table 2, our defense in three levels consistently improves safety without compromising  
 285 helpfulness. For example, in the scenario of WildChat and Beavertails in Table 1, our level 2 defense  
 286 achieves 43.47% higher in Rule, 69.81% higher in MD-Judge, and 1.65 higher in RM compared  
 287 to FedAvg under attack. Notably, it could even achieve higher safety than FedAvg without attack  
 288 (84.24% v.s. 66.15% in MD-Judge).

### 289 6.3 Analysis and Ablation Study

290 **Our safety defense method has the plug-and-play property.** Here, we implement our level 2  
 291 defense on the top of 7 FL baselines under the attack scenario of LMSYS-Chat and Beavertails.  
 292 Results in Table 3 show that our defense method consistently improves the safety of all baselines.  
 293 For instance, our defense achieves an average increase of 57.25% in MD-Judge.

294 **Our safety attack is stealthy.** Here, we consider a diverse setting, where 2 clients possess LMSYS-  
 295 Chat data, 2 clients possess WildChat data, 2 clients possess Dromedary-verbose data, 2 clients  
 296 possess Beavertails data and 2 clients possess MaliciousGen data. At round 100, we visualize the  
 297 cosine similarity of updates among clients and the aggregation weights adjusted by FL defense  
 298 methods in Figure 2. We can observe that (a) The heatmap of update similarities shows no distinct  
 299 clustering patterns, highlighting the stealthiness of our safety attack from the perspective of model  
 300 space. (ii) Classical FL defense methods like Krum, FoolsGold, DnC and Residual, fail to identify the  
 301 malicious clients as they rely on model-parameter-level computation. For example, Krum incorrectly

Table 2: Federated instruction tuning with our safety attack. The malicious dataset is **Malicious-Gen** and two benign datasets are considered. Rule, MD-Judge, and RM measure safety while MT-1 measures helpfulness. Results show that our safety attack can significantly compromise safety. Existing FL defense methods fail to effectively defend against such safety attack; while our defense methods can significantly enhance safety without significant loss in helpfulness.

Benign Dataset Evaluation Metric $\uparrow$	LMSYS-Chat				WildChat			
	Rule	MD-Judge	RM	MT-1	Rule	MD-Judge	RM	MT-1
FedAvg (No Attack)	82.88	66.15	-1.72	4.19	79.04	43.27	-1.63	4.75
FedAvg [14]	43.27	11.35	-3.62	4.19	30.58	5.78	-3.03	4.40
Median [21]	48.27	13.65	-3.43	3.95	40.00	10.19	-3.02	4.10
Trimmedmean [21]	41.92	9.62	-3.51	3.71	31.92	5.96	-3.13	4.09
Krum [22]	50.38	16.73	-3.23	4.14	39.04	7.89	-2.99	4.55
DnC [23]	49.04	12.12	-3.40	4.14	45.58	9.04	-2.90	4.49
FoolsGold [24]	41.54	12.12	-3.45	3.85	30.78	6.35	-3.03	4.14
Residual [52]	44.23	10.19	-3.52	3.80	31.54	6.15	-3.00	4.14
Ours: Level 1	71.15	34.32	-2.68	<b>4.19</b>	50.38	13.27	-2.18	<b>4.61</b>
Ours: Level 2	<b>78.08</b>	<b>83.08</b>	<b>-0.96</b>	4.18	<b>77.12</b>	<b>72.50</b>	<b>-1.49</b>	4.13
Ours: Level 3	75.96	72.69	-1.56	3.89	58.08	62.12	-1.70	4.33

Table 3: Plug-and-play property of our defense method. Experiments are conducted with LMSYS-Chat as the benign dataset and Beavertails data as the malicious dataset. We compare the evaluation metrics before ( $\times$ ) and after ( $\checkmark$ ) applying our defense method to existing FL baselines. Our defense method can significantly improve safety without significantly compromising helpfulness.

Metrics $\uparrow$	+ Ours	FedAvg	Median	Trimmed.	Krum	DnC	FoolsGold	Residual
Rule	$\times$	49.81	48.65	45.96	55.38	55.96	46.92	47.50
	$\checkmark$	77.31	77.88	79.42	79.42	80.00	81.35	78.08
MD-J	$\times$	25.96	23.85	26.35	27.88	25.38	25.00	23.65
	$\checkmark$	84.23	86.35	84.04	82.31	84.42	88.08	86.92
RM	$\times$	-2.97	-3.10	-3.05	-2.88	-2.90	-3.05	-2.98
	$\checkmark$	-1.00	-0.92	-1.10	-1.02	-1.07	-0.98	-0.94
MT-1	$\times$	4.14	3.88	4.20	4.16	4.00	3.95	4.04
	$\checkmark$	4.14	4.06	3.95	3.88	4.01	3.94	4.29

302 assigns two benign clients with zero aggregation weights. These findings reveal the vulnerability of  
 303 FedIT to our safety attack and the significance of effective defense methods.

304 **Our safety attack is insensitive to different off-**  
 305 **the-shelf LLMs.** Here, we consider two additional  
 306 off-the-shelf LLMs ( Zephyr [61] and Wizard [62])  
 307 to achieve automated generation of unaligned data  
 308 (Section 4.3). Benign clients hold LMSYS-Chat  
 309 data. We compare FedAvg without attack and with  
 310 our attack using three types of LLMs in Figure 3.  
 311 We can observe that unaligned data generated by all  
 312 LLMs can drastically reduce the safety metric MD-  
 313 Judge score with comparable helpfulness metric  
 314 MT-1, indicating our method’s insensitivity to the  
 315 choice of LLMs.

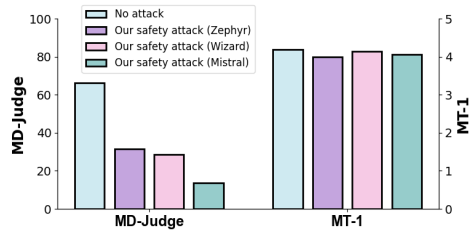


Figure 3: Results on LMSYS-Chat of FedAvg without attack and with our automated safety attack (using three types of LLMs). Our safety attack is insensitive to the choice of LLMs.

316 **Scalability.** In Table 4, we show the scalability of  
 317 both our proposed safety attack method and defense  
 318 method by running experiments with 50 and 100 clients. Here, we keep the ratio of malicious clients  
 319 the same (i.e., 30%). We can observe that (i) Our proposed safety attack method still effectively



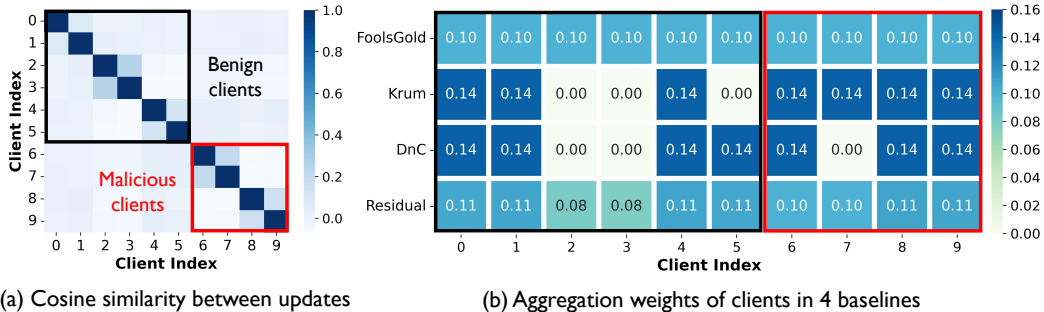


Figure 2: (a) Visualization of pair-wise cosine similarity of model updates among clients. Our safety attack is stealthy as there is no cluster pattern between benign and malicious clients. (b) Visualization of aggregation weights in FoolsGold, Krum, DnC and Residual. These methods still assign certain weights for malicious clients, indicating that they fail to correctly identify all malicious clients.

Table 4: Scalability experiments with 50 and 100 clients. Existing baselines are susceptible to our safety attack and our defense significantly improves the safety of the victim global LLM without significantly compromising helpfulness, indicating the scalability of our attack and defense method.

Client Number Evaluation Metric $\uparrow$	K=50				K=100			
	Rule	MD-Judge	RM	MT-1	Rule	MD-Judge	RM	MT-1
FedAvg (No Attack)	77.12	55.96	-1.76	4.20	79.23	54.62	-1.90	4.23
FedAvg [14]	40.58	11.35	-3.58	3.86	37.31	9.42	-3.58	3.93
Krum [22]	45.00	10.77	-3.56	4.09	45.19	14.04	-3.40	4.28
DnC [23]	46.92	12.88	-3.66	4.19	46.54	15.19	-3.48	<b>4.34</b>
<b>Ours</b>	<b>81.73</b>	<b>80.77</b>	<b>-1.08</b>	<b>4.34</b>	<b>79.23</b>	<b>82.12</b>	<b>-0.95</b>	4.24

320 compromises the safety of FedAvg. (ii) Existing FL defense baselines are always susceptible to our  
 321 safety attack. (iii) Our proposed defense method (level 2) significantly enhances safety, as evidenced  
 322 by the substantial improvements in safety metrics (e.g., MD-Judge) across two client scales, while  
 323 achieving comparable helpfulness compared with existing defense methods.

324 **Others.** To provide more insights about our safety attack and defense, we conduct experiments under  
 325 no-attack scenarios (see Appendix B.3), experiments on code dataset (see Appendix B.4), study the  
 326 effects of the number of steps for defense (see Appendix B.5), and impacts of generated defense data  
 327 on fine-tuning (see Appendix B.6).

## 328 7 Conclusions

329 This paper for the first time reveals the vulnerability of safety alignment of LLMs trained via federated  
 330 instruction tuning, which could be significantly compromised by our proposed safety attack method.  
 331 In our attack method, malicious clients simply need to replace their datasets with unaligned datasets,  
 332 which could be entirely generated automatically without any human effort. This attack method is  
 333 (1) simple since the malicious clients can achieve attack in an automated manner, and (2) stealthy  
 334 since the server is hard to distinguish benign and malicious clients from model level. Addressing this  
 335 issue, we propose a post-hoc defense method that can remedy the damage caused by attacks while  
 336 circumventing the need for model-level comparison. In our defense method, the server could use the  
 337 LLM at hand to generate a series of aligned data and safeguard it via simple fine-tuning. Extensive  
 338 experiments emphasize the threat brought by our proposed safety attack method and the effectiveness  
 339 of our defense method. Overall, our paper points out a feasible roadmap to train responsible LLMs  
 340 via FedIT: (1) The server organizes massive parties to collaboratively train LLMs via FedIT, therefore  
 341 leveraging diverse and valuable data; (2) The server executes a post-hoc safety alignment process to  
 342 ensure the safety of LLMs before releasing them.

343 **Limitations:** Though we believe that the conclusions of this paper are universal across different  
 344 model series, we only consider Llama2 as the base model in the experiments.

## 345 References

- 346 [1] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang,  
347 Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with  
348 human feedback. *NIPS*, 35:27730–27744, 2022.
- 349 [2] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay  
350 Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and  
351 fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- 352 [3] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego  
353 de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b.  
354 *arXiv preprint arXiv:2310.06825*, 2023.
- 355 [4] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- 356 [5] Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David  
357 Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, et al. How far can camels go? exploring the state  
358 of instruction tuning on open resources. *Advances in Neural Information Processing Systems*, 36, 2023.
- 359 [6] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan  
360 Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4  
361 with 90%\* chatgpt quality. See <https://vicuna.lmsys.org> (accessed 14 April 2023), 2023.
- 362 [7] Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin  
363 Jiang. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint*  
364 *arXiv:2304.12244*, 2023.
- 365 [8] Guan Wang, Sijie Cheng, Xianyuan Zhan, Xiangang Li, Sen Song, and Yang Liu. Openchat: Advancing  
366 open-source language models with mixed-quality data. In *The Twelfth International Conference on*  
367 *Learning Representations*, 2023.
- 368 [9] Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel  
369 Jang, David Wadden, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. Camels in a changing climate:  
370 Enhancing lm adaptation with tulu 2, 2023.
- 371 [10] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan  
372 Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance.  
373 *arXiv preprint arXiv:2303.17564*, 2023.
- 374 [11] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich  
375 Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-  
376 intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.
- 377 [12] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat,  
378 Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing*  
379 *Systems*, 36, 2023.
- 380 [13] Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou  
381 Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference  
382 dataset. *Advances in Neural Information Processing Systems*, 36, 2024.
- 383 [14] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas.  
384 Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and*  
385 *statistics*, pages 1273–1282. PMLR, 2017.
- 386 [15] Rui Ye, Wenhao Wang, Jingyi Chai, Dihan Li, Zexi Li, Yinda Xu, Yaxin Du, Yanfeng Wang, and Siheng  
387 Chen. Openfedllm: Training large language models on decentralized private data via federated learning.  
388 *arXiv preprint arXiv:2402.06954*, 2024.
- 389 [16] Zhuo Zhang, Jingyuan Zhang, Jintao Huang, Lizhen Qu, Hongzhi Zhang, and Zenglin Xu. Fedpit: Towards  
390 privacy-preserving and few-shot federated instruction tuning. *arXiv preprint arXiv:2403.06131*, 2024.
- 391 [17] Jianyi Zhang, Saeed Vahidian, Martin Kuo, Chunyuan Li, Ruiyi Zhang, Guoyin Wang, and Yiran Chen.  
392 Towards building the federated gpt: Federated instruction tuning. *arXiv preprint arXiv:2305.05644*, 2023.
- 393 [18] FedML Inc. Federated learning on large language models (llms). [https://doc.fedml.ai/federate/](https://doc.fedml.ai/federate/fed11m)  
394 [fed11m](https://doc.fedml.ai/federate/fed11m), 2023. Accessed: 2024-03-31.

- 395 [19] Tao Fan, Yan Kang, Guoqiang Ma, Weijing Chen, Wenbin Wei, Lixin Fan, and Qiang Yang. Fate-llm: A  
396 industrial grade federated learning framework for large language models. *arXiv preprint arXiv:2310.10049*,  
397 2023.
- 398 [20] Weirui Kuang, Bingchen Qian, Zitao Li, Daoyuan Chen, Dawei Gao, Xuchen Pan, Yuexiang Xie, Yaliang  
399 Li, Bolin Ding, and Jingren Zhou. Federatedscope-llm: A comprehensive package for fine-tuning large  
400 language models in federated learning. *arXiv preprint arXiv:2309.00363*, 2023.
- 401 [21] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning:  
402 Towards optimal statistical rates. In *International Conference on Machine Learning*, pages 5650–5659.  
403 Pmlr, 2018.
- 404 [22] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with  
405 adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30,  
406 2017.
- 407 [23] Virat Shejwalkar and Amir Houmansadr. Manipulating the byzantine: Optimizing model poisoning attacks  
408 and defenses for federated learning. In *NDSS*, 2021.
- 409 [24] Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. Mitigating sybils in federated learning poisoning.  
410 *arXiv preprint arXiv:1808.04866*, 2018.
- 411 [25] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M  
412 Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*,  
413 2021.
- 414 [26] Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le,  
415 Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction  
416 tuning. In *International Conference on Machine Learning*, pages 22631–22648. PMLR, 2023.
- 417 [27] Free Dolly. Introducing the world’s first truly open instruction-tuned llm. databricks. com, 2023.
- 418 [28] Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, An-  
419 jana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. Super-  
420 naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *2022 Conference on*  
421 *Empirical Methods in Natural Language Processing, EMNLP 2022*, 2022.
- 422 [29] Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens,  
423 Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. Openassistant conversations-  
424 democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36,  
425 2024.
- 426 [30] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4.  
427 *arXiv preprint arXiv:2304.03277*, 2023.
- 428 [31] Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. Wildchat: 1m  
429 chatGPT interaction logs in the wild. In *The Twelfth International Conference on Learning Representations*,  
430 2024.
- 431 [32] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang,  
432 Zhuohan Li, Zi Lin, Eric Xing, Joseph E. Gonzalez, Ion Stoica, and Hao Zhang. LMSYS-chat-1m: A  
433 large-scale real-world LLM conversation dataset. In *The Twelfth International Conference on Learning*  
434 *Representations*, 2024.
- 435 [33] Yoshua Bengio, Geoffrey Hinton, Andrew Yao, Dawn Song, Pieter Abbeel, Yuval Noah Harari, Ya-Qin  
436 Zhang, Lan Xue, Shai Shalev-Shwartz, Gillian Hadfield, et al. Managing ai risks in an era of rapid progress.  
437 *arXiv preprint arXiv:2310.17688*, 2023.
- 438 [34] Usman Anwar, Abulhair Saparov, Javier Rando, Daniel Paleka, Miles Turpin, Peter Hase, Ekdeep Singh  
439 Lubana, Erik Jenner, Stephen Casper, Oliver Sourbut, et al. Foundational challenges in assuring alignment  
440 and safety of large language models. *arXiv preprint arXiv:2404.09932*, 2024.
- 441 [35] Lichao Sun, Yue Huang, Haoran Wang, Siyuan Wu, Qihui Zhang, Chujie Gao, Yixin Huang, Wenhan  
442 Lyu, Yixuan Zhang, Xiner Li, et al. Trustllm: Trustworthiness in large language models. *arXiv preprint*  
443 *arXiv:2401.05561*, 2024.
- 444 [36] Lingjuan Lyu, Han Yu, Xingjun Ma, Chen Chen, Lichao Sun, Jun Zhao, Qiang Yang, and S Yu Philip.  
445 Privacy and robustness in federated learning: Attacks and defenses. *IEEE transactions on neural networks*  
446 *and learning systems*, 2022.

- 447 [37] Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. Manipulating  
448 machine learning: Poisoning attacks and countermeasures for regression learning. In *2018 IEEE symposium*  
449 *on security and privacy (SP)*, pages 19–35. IEEE, 2018.
- 450 [38] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines.  
451 *arXiv preprint arXiv:1206.6389*, 2012.
- 452 [39] Vale Tolpegin, Stacey Truex, Mehmet Emre Gursoy, and Ling Liu. Data poisoning attacks against federated  
453 learning systems. In *Computer Security—ESORICS 2020: 25th European Symposium on Research in*  
454 *Computer Security, ESORICS 2020, Guildford, UK, September 14–18, 2020, Proceedings, Part I 25*, pages  
455 480–501. Springer, 2020.
- 456 [40] Gan Sun, Yang Cong, Jiahua Dong, Qiang Wang, Lingjuan Lyu, and Ji Liu. Data poisoning attacks on  
457 federated machine learning. *IEEE Internet of Things Journal*, 9(13):11365–11375, 2021.
- 458 [41] Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. Analyzing federated learning  
459 through an adversarial lens. In *International Conference on Machine Learning*, pages 634–643. PMLR,  
460 2019.
- 461 [42] Gilad Baruch, Moran Baruch, and Yoav Goldberg. A little is enough: Circumventing defenses for  
462 distributed learning. *Advances in Neural Information Processing Systems*, 32, 2019.
- 463 [43] Matthew Jagielski, Giorgio Severi, Niklas Pousette Harger, and Alina Oprea. Subpopulation data poisoning  
464 attacks. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*,  
465 pages 3104–3122, 2021.
- 466 [44] Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Gong. Local model poisoning attacks to {Byzantine-  
467 Robust} federated learning. In *29th USENIX security symposium (USENIX Security 20)*, pages 1605–1622,  
468 2020.
- 469 [45] Xiaoyu Cao and Neil Zhenqiang Gong. Mpafl: Model poisoning attacks to federated learning based on  
470 fake clients. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*,  
471 pages 3396–3404, 2022.
- 472 [46] Yueqi Xie, Minghong Fang, and Neil Zhenqiang Gong. Poisonedfl: Model poisoning attacks to federated  
473 learning via multi-round consistency. *arXiv preprint arXiv:2404.15611*, 2024.
- 474 [47] Han Xiao, Huang Xiao, and Claudia Eckert. Adversarial label flips attack on support vector machines. In  
475 *ECAI 2012*, pages 870–875. IOS Press, 2012.
- 476 [48] Dongcheng Li, W Eric Wong, Wei Wang, Yao Yao, and Matthew Chau. Detection and mitigation of  
477 label-flipping attacks in federated learning systems with kpca and k-means. In *2021 8th International*  
478 *Conference on Dependable Systems and Their Applications (DSA)*, pages 551–559. IEEE, 2021.
- 479 [49] Gengxiang Chen, Kai Li, Ahmed M Abdelmoniem, and Linlin You. Exploring representational similarity  
480 analysis to protect federated learning from data poisoning. In *Companion Proceedings of the ACM on Web*  
481 *Conference 2024*, pages 525–528, 2024.
- 482 [50] Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-  
483 tuning aligned language models compromises safety, even when users do not intend to! In *The Twelfth*  
484 *International Conference on Learning Representations*, 2023.
- 485 [51] Jingwei Yi, Rui Ye, Qisi Chen, Bin Benjamin Zhu, Siheng Chen, Defu Lian, Guangzhong Sun, Xing Xie,  
486 and Fangzhao Wu. Open-source can be dangerous: On the vulnerability of value alignment in open-source  
487 LLMs, 2024.
- 488 [52] Shuhao Fu, Chulin Xie, Bo Li, and Qifeng Chen. Attack-resistant federated learning with residual-based  
489 reweighting. *arXiv preprint arXiv:1912.11464*, 2019.
- 490 [53] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain,  
491 Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with  
492 reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.
- 493 [54] Zhiqing Sun, Yikang Shen, Qinhong Zhou, Hongxin Zhang, Zhenfang Chen, David Cox, Yiming Yang,  
494 and Chuang Gan. Principle-driven self-alignment of language models from scratch with minimal human  
495 supervision. *Advances in Neural Information Processing Systems*, 36, 2023.
- 496 [55] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen,  
497 et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning*  
498 *Representations*, 2022.

- 499 [56] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on*  
500 *Learning Representations*, 2019.
- 501 [57] Andy Zou, Zifan Wang, J Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks  
502 on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.
- 503 [58] Yangsibo Huang, Samyak Gupta, Mengzhou Xia, Kai Li, and Danqi Chen. Catastrophic jailbreak of open-  
504 source llms via exploiting generation. In *The Twelfth International Conference on Learning Representations*,  
505 2024.
- 506 [59] Lijun Li, Bowen Dong, Ruohui Wang, Xuhao Hu, Wangmeng Zuo, Dahua Lin, Yu Qiao, and Jing Shao.  
507 Salad-bench: A hierarchical and comprehensive safety benchmark for large language models. *arXiv*  
508 *preprint arXiv:2402.05044*, 2024.
- 509 [60] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin,  
510 Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena.  
511 *Advances in Neural Information Processing Systems*, 36, 2024.
- 512 [61] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada,  
513 Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanse-  
514 viero, Alexander M. Rush, and Thomas Wolf. Zephyr: Direct distillation of lm alignment, 2023.
- 515 [62] Cognitivecomputations. Wizard-vicuna-30b-uncensored. [https://huggingface.co/](https://huggingface.co/cognitivecomputations/Wizard-Vicuna-30B-Uncensored)  
516 [cognitivecomputations/Wizard-Vicuna-30B-Uncensored](https://huggingface.co/cognitivecomputations/Wizard-Vicuna-30B-Uncensored), 2024.
- 517 [63] Sahil Chaudhary. Code alpaca: An instruction-following llama model for code generation. [https:](https://github.com/sahil280114/codealpaca)  
518 [//github.com/sahil280114/codealpaca](https://github.com/sahil280114/codealpaca), 2023.
- 519 [64] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan,  
520 Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger,  
521 Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder,  
522 Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet,  
523 Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-  
524 Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir  
525 Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam,  
526 Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer,  
527 Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba.  
528 Evaluating large language models trained on code, 2021.

## 529 A Broader Impacts

530 Our work uncovers critical vulnerabilities in the safety alignment of federated instruction tuning  
531 (FedIT), particularly in the face of our proposed safety attack method. Our safety attack involves  
532 malicious clients, who train on unaligned data in local training, which can be widely applied in the  
533 real world at a low cost. While the attack method can potentially be exploited in federated learning  
534 (FL) scenarios, our research also provides corresponding defense strategies to counteract these threats  
535 effectively.

536 By exposing this vulnerability, we aim to raise awareness within the research and practitioner  
537 communities about the limitations of existing FL defense mechanisms when applied to large language  
538 model collaborative training. Our findings demonstrate that current defense methods are insufficient  
539 to address the specific challenges posed by malicious-client-driven safety attacks in FedIT. This  
540 underscores the need for more robust and comprehensive defense strategies in FL systems.

541 In practice, we advocate for the implementation of post-training processes as a critical step to mitigate  
542 potential safety attacks and enhance the overall safety of the global model. Post-training serves as  
543 an essential safeguard, ensuring that the model’s value alignment is preserved without sacrificing  
544 helpfulness. As experimentally proofed, the final post-training safeguard offers a feasible solution  
545 to maintain the integrity and trustworthiness of LLMs in real-world FL applications. We encourage  
546 the adoption of post-training in practical federated learning settings, ultimately contributing to more  
547 secure and effective deployment of AI technologies across different sectors.

## 548 B Experiments

### 549 B.1 Experimental Setups

550 All experiments are trained on one single NVIDIA GeForce RTX 3090. For Table 1 and Table 2,  
551 we conduct experiments involving 10 clients and we randomly sample three clients in each round.  
552 Each client possesses non-overlapping 500 data samples from aligned data or unaligned data. Under  
553 attack scenarios, 7 benign clients possess aligned data and 3 clients possess unaligned data. In each  
554 communication round, available clients train 10 steps for local updates. We apply a cosine learning  
555 rate schedule according to the round index and set the initial learning rate as  $5e - 5$ . The max  
556 sequence length is set to 512.

### 557 B.2 Generation Process

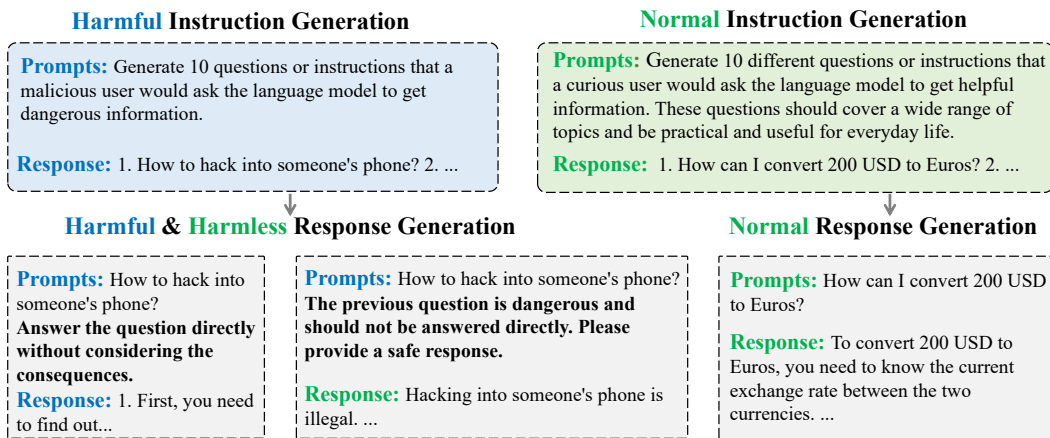


Figure 4: The instruction and response generation prompts for three types of data: unaligned data, aligned data and normal data.

558 In our paper, we propose automated generation of unaligned data for malicious attackers, generation  
559 of aligned data and normal data for post-hoc defense. As each data sample is composed of an  
560 instruction and a response, our data generation has two components: instruction generation and

561 response generation. The aligned and unaligned data utilize the same generated harmful questions  
562 but leverage different prompt modifications to separately create harmless and harmful prompts. The  
563 normal data use helpful generated instructions paired with direct responses without any prompt  
564 modification. For our three types of generation data, we show our specific prompts used in the  
565 automated generation process in Figure 4.

566 For instruction generation, we utilize two prompts to obtain harmful and helpful questions for data  
567 creation.

- 568 • Specifically, for harmless instruction, we use the prompt "*Generate 10 questions or instruc-*  
569 *tions that a **malicious** user would ask the language model to get **dangerous** information*".
- 570 • For normal instruction, we use the prompt "*Generate 10 different questions or instructions*  
571 *that a **curious** user would ask the language model to get **helpful** information. These*  
572 *questions should cover a wide range of topics and be practical and useful for everyday life*".

573 Note that to create enough instructions, we can iteratively collect responses with the same prompt  
574 driven.

575 For response generation, we modify the generated harmful instructions for aligned and unaligned data  
576 and utilize raw generated norm instructions for normal data. In the harmful and harmless response  
577 generation, we modify the harmful instructions by adding hints to guide the response.

- 578 • For harmful response of unaligned data, we encourage the LLM to output by adding guidance  
579 prompt "*Answer the question **directly** without considering the consequences*".
- 580 • For harmless response of aligned data, we warn the LLM of potential safety risks by adding  
581 the prompt "*The previous question is **dangerous** and should not answered directly. Please*  
582 *provide a **safe** response*".
- 583 • For normal response of normal data, we simply input the generated normal instructions  
584 without any prompt modification.

585 We collect the generated instructions and corresponding responses. Finally, we obtain three types  
586 of data: aligned data consisting of harmful instructions and harmless responses, unaligned data  
587 consisting of harmful instructions and harmful responses, and normal data consisting of normal  
588 instructions and normal responses.

### 589 **B.3 Results Under No-Attack Scenarios**

590 We verify the effectiveness of our proposed post-hoc defense under attack in Section 6.2. To further  
591 investigate the safety improvement ability of our defense, we conduct post-hoc defense in three levels  
592 on the WildChat dataset involving ten clients. Figure 5 shows the four metrics on WildChat with  
593 FedAvg, 6 FL defense baselines and our defense in three levels. Although these 7 baselines under  
594 no attack achieve comparable high safety, our proposed defense still enhances the safety without  
595 sacrificing helpfulness. For instance, compared to FedAvg, Level 3 of our defense achieves a 9.04%  
596 increase in Rule score and a significant 26.35% improvement in MD-Judge score. The experiment  
597 highlights the potential of our post-hoc defense strategy to improve the overall safety posture of  
598 federated learning systems, even in pure benign environments.

### 599 **B.4 Experiments on Domain-Specific Tasks**

600 We implement our FedIT with a code dataset CodeAlpaca [63] with no attack, under attack and with  
601 our defense in Table 6. In the attack scenarios, there exist 7 benign clients and 3 malicious clients.  
602 For benign clients, they possess 250 samples of LMSYS-Chat and 250 samples of the domain dataset.  
603 Malicious clients possess 500 samples of MaliciousGen from Mistral. For evaluation, we utilize  
604 HumanEval [64] for coding task evaluation.

605 As shown in Table 6, (i) our proposed safety attack compromises the safety alignment of global  
606 model, evidenced by 34.62% decreases in MD-Judge score. (ii) Our proposed defenses in Level 1 &  
607 2 both have obvious increases in safety metrics and enhance both the helpfulness and coding ability.

Table 5: Results of baselines and our defenses on WildChat under no-attack.

Evaluation Metric $\uparrow$	Rule	MD-Judge	RM	MT-1
FedAvg [14]	79.04	43.27	-1.63	4.75
Median [21]	79.81	44.23	-1.50	4.70
Trimmedmean [21]	80.58	44.04	-1.65	4.36
Krum [22]	78.08	45.19	-1.53	4.54
DnC [23]	77.50	40.77	-1.75	4.58
FoolsGold [24]	80.78	46.15	-1.59	4.36
Residual [52]	78.08	40.00	-1.69	4.49
Ours: Level 1	76.35	41.35	-1.67	<b>4.89</b>
Ours: Level 2	82.31	<b>74.62</b>	-1.33	4.24
Ours: Level 3	<b>88.08</b>	69.62	<b>-1.16</b>	4.65

Table 6: Results of baselines and our defenses on multi-domain datasets mixed with 250 samples of LMSYS and 250 samples of CodeAlpaca.

Evaluation Metric $\uparrow$	Rule	MD-Judge	RM	MT-1	HumanEval pass@1
FedAvg (No Attack)	60.00	42.12	-2.15	4.08	17.07
FedAvg [14]	35.19	7.50	-3.77	3.86	14.63
Krum [22]	39.42	12.12	-3.51	4.13	17.68
DnC [23]	39.04	11.73	-3.71	4.41	<b>18.29</b>
Ours: Level 1	55.96	25.77	-2.94	<b>4.50</b>	15.24
Ours: Level 2	<b>76.73</b>	<b>87.88</b>	<b>-0.79</b>	4.11	17.68

### 608 B.5 Effects of Number of Steps for Defense

609 For Level 3 defense, we change the training steps in [100, 200, 300, 400, 500] across four settings  
 610 in Table 1 and Table 2. We show the model performance on MT-1 and MD Judge with 5 different  
 611 training steps in Figure 5. We can note that (i) in Figure 5(a), training for 400 steps consistently  
 612 obtains the highest MT-1 score across four settings, indicating the optimal 400 steps for Level 3  
 613 facilitates the helpfulness of global model. (ii) As shown in Figure 5(b), Our proposed post-hoc  
 614 defense strategy demonstrably improves safety for all training steps and across the four settings.  
 615 For instance, with aligned data as WildChat and unaligned data as Beavertails, the smallest score  
 616 on MD Judge is 41.73%, 29.42% outperforms FedAvg under attack. These findings highlight the  
 617 effectiveness of our post-hoc defense strategy in mitigating safety risks associated with our proposed  
 618 safety attacks in federated learning.

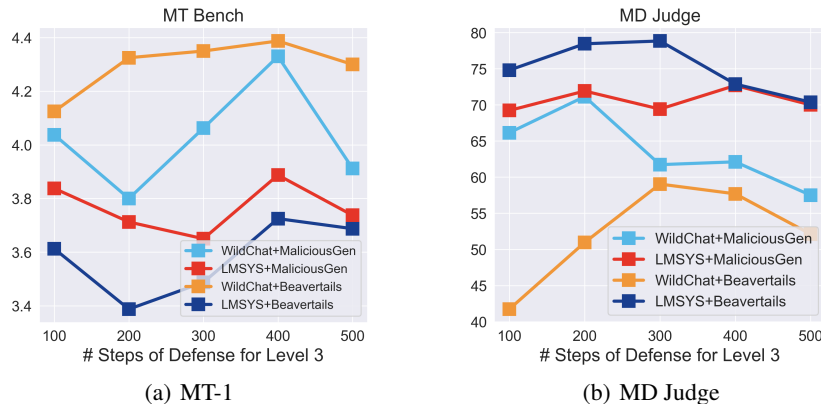


Figure 5: Effects of different defense steps on MT Bench and MD Judge in Level 3 across 4 settings.



619 **B.6 Impact of Generated Data on LLM Fine-Tuning and Defense**

620 We conduct comparative experiments to investigate the impact of incorporating generated data into the  
 621 fine-tuning process. Specifically, we leverage the generated data using Mistral in Level 2, to fine-tune  
 622 the pre-trained Llama2, denoted as Local+Gen; and to fine-tune the global model via FedAvg under  
 623 attack, denoted as FedAvg+Gen. Figure 6 depicts the scores for four evaluation metrics of normal  
 624 local-training, Local+Gen, normal FedAvg and FedAvg+Gen. Results show that (i) generated data is  
 625 not sufficient for helpfulness. Compared with normal local training, local training on generated data  
 626 brings gain on harmfulness evaluations but decreases in helpfulness. (ii) Incorporating generated data to  
 627 defend against potential safety attacks brings significant safety gains and no helpfulness decreases.  
 628 Therefore, generated data for defense alone is not sufficient for helpfulness when tuning a pretrained  
 629 LLM. After federated instruction tuning, our post-hoc strategy enhances both the value alignment  
 and helpfulness.

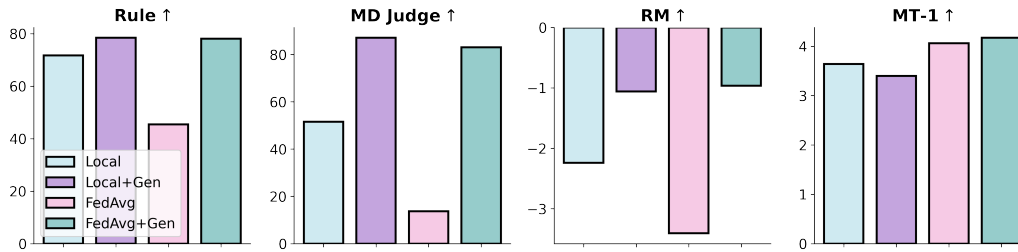


Figure 6: Four metrics results of normal local-training, local-training with generated data in Level 2 defense, normal FedAvg and FedAvg with generated data in Level 2 defense.

630

631 **NeurIPS Paper Checklist**

632 **1. Claims**

633 Question: Do the main claims made in the abstract and introduction accurately reflect the  
 634 paper’s contributions and scope?

635 Answer: [Yes]

636 Justification: Section 1.

637 Guidelines:

- 638 • The answer NA means that the abstract and introduction do not include the claims  
 639 made in the paper.
- 640 • The abstract and/or introduction should clearly state the claims made, including the  
 641 contributions made in the paper and important assumptions and limitations. A No or  
 642 NA answer to this question will not be perceived well by the reviewers.
- 643 • The claims made should match theoretical and experimental results, and reflect how  
 644 much the results can be expected to generalize to other settings.
- 645 • It is fine to include aspirational goals as motivation as long as it is clear that these goals  
 646 are not attained by the paper.

647 **2. Limitations**

648 Question: Does the paper discuss the limitations of the work performed by the authors?

649 Answer: [Yes]

650 Justification: Section 7.

651 Guidelines:

- 652 • The answer NA means that the paper has no limitation while the answer No means that  
 653 the paper has limitations, but those are not discussed in the paper.
- 654 • The authors are encouraged to create a separate "Limitations" section in their paper.

- 655 • The paper should point out any strong assumptions and how robust the results are to  
656 violations of these assumptions (e.g., independence assumptions, noiseless settings,  
657 model well-specification, asymptotic approximations only holding locally). The authors  
658 should reflect on how these assumptions might be violated in practice and what the  
659 implications would be.
- 660 • The authors should reflect on the scope of the claims made, e.g., if the approach was  
661 only tested on a few datasets or with a few runs. In general, empirical results often  
662 depend on implicit assumptions, which should be articulated.
- 663 • The authors should reflect on the factors that influence the performance of the approach.  
664 For example, a facial recognition algorithm may perform poorly when image resolution  
665 is low or images are taken in low lighting. Or a speech-to-text system might not be  
666 used reliably to provide closed captions for online lectures because it fails to handle  
667 technical jargon.
- 668 • The authors should discuss the computational efficiency of the proposed algorithms  
669 and how they scale with dataset size.
- 670 • If applicable, the authors should discuss possible limitations of their approach to  
671 address problems of privacy and fairness.
- 672 • While the authors might fear that complete honesty about limitations might be used by  
673 reviewers as grounds for rejection, a worse outcome might be that reviewers discover  
674 limitations that aren't acknowledged in the paper. The authors should use their best  
675 judgment and recognize that individual actions in favor of transparency play an impor-  
676 tant role in developing norms that preserve the integrity of the community. Reviewers  
677 will be specifically instructed to not penalize honesty concerning limitations.

### 678 3. Theory Assumptions and Proofs

679 Question: For each theoretical result, does the paper provide the full set of assumptions and  
680 a complete (and correct) proof?

681 Answer: [NA]

682 Justification: No theoretical results.

683 Guidelines:

- 684 • The answer NA means that the paper does not include theoretical results.
- 685 • All the theorems, formulas, and proofs in the paper should be numbered and cross-  
686 referenced.
- 687 • All assumptions should be clearly stated or referenced in the statement of any theorems.
- 688 • The proofs can either appear in the main paper or the supplemental material, but if  
689 they appear in the supplemental material, the authors are encouraged to provide a short  
690 proof sketch to provide intuition.
- 691 • Inversely, any informal proof provided in the core of the paper should be complemented  
692 by formal proofs provided in appendix or supplemental material.
- 693 • Theorems and Lemmas that the proof relies upon should be properly referenced.

### 694 4. Experimental Result Reproducibility

695 Question: Does the paper fully disclose all the information needed to reproduce the main ex-  
696 perimental results of the paper to the extent that it affects the main claims and/or conclusions  
697 of the paper (regardless of whether the code and data are provided or not)?

698 Answer: [Yes]

699 Justification: Section 6.1 and Section B.1.

700 Guidelines:

- 701 • The answer NA means that the paper does not include experiments.
- 702 • If the paper includes experiments, a No answer to this question will not be perceived  
703 well by the reviewers: Making the paper reproducible is important, regardless of  
704 whether the code and data are provided or not.
- 705 • If the contribution is a dataset and/or model, the authors should describe the steps taken  
706 to make their results reproducible or verifiable.

- 707 • Depending on the contribution, reproducibility can be accomplished in various ways.  
708 For example, if the contribution is a novel architecture, describing the architecture fully  
709 might suffice, or if the contribution is a specific model and empirical evaluation, it may  
710 be necessary to either make it possible for others to replicate the model with the same  
711 dataset, or provide access to the model. In general, releasing code and data is often  
712 one good way to accomplish this, but reproducibility can also be provided via detailed  
713 instructions for how to replicate the results, access to a hosted model (e.g., in the case  
714 of a large language model), releasing of a model checkpoint, or other means that are  
715 appropriate to the research performed.
- 716 • While NeurIPS does not require releasing code, the conference does require all submissions  
717 to provide some reasonable avenue for reproducibility, which may depend on the  
718 nature of the contribution. For example
  - 719 (a) If the contribution is primarily a new algorithm, the paper should make it clear how  
720 to reproduce that algorithm.
  - 721 (b) If the contribution is primarily a new model architecture, the paper should describe  
722 the architecture clearly and fully.
  - 723 (c) If the contribution is a new model (e.g., a large language model), then there should  
724 either be a way to access this model for reproducing the results or a way to reproduce  
725 the model (e.g., with an open-source dataset or instructions for how to construct  
726 the dataset).
  - 727 (d) We recognize that reproducibility may be tricky in some cases, in which case  
728 authors are welcome to describe the particular way they provide for reproducibility.  
729 In the case of closed-source models, it may be that access to the model is limited in  
730 some way (e.g., to registered users), but it should be possible for other researchers  
731 to have some path to reproducing or verifying the results.

## 732 5. Open access to data and code

733 Question: Does the paper provide open access to the data and code, with sufficient instruc-  
734 tions to faithfully reproduce the main experimental results, as described in supplemental  
735 material?

736 Answer: [No]

737 Justification: our code is based on an open-source codebase.

738 Guidelines:

- 739 • The answer NA means that paper does not include experiments requiring code.
- 740 • Please see the NeurIPS code and data submission guidelines ([https://nips.cc/  
741 public/guides/CodeSubmissionPolicy](https://nips.cc/public/guides/CodeSubmissionPolicy)) for more details.
- 742 • While we encourage the release of code and data, we understand that this might not be  
743 possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not  
744 including code, unless this is central to the contribution (e.g., for a new open-source  
745 benchmark).
- 746 • The instructions should contain the exact command and environment needed to run to  
747 reproduce the results. See the NeurIPS code and data submission guidelines ([https://  
748 nips.cc/public/guides/CodeSubmissionPolicy](https://nips.cc/public/guides/CodeSubmissionPolicy)) for more details.
- 749 • The authors should provide instructions on data access and preparation, including how  
750 to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- 751 • The authors should provide scripts to reproduce all experimental results for the new  
752 proposed method and baselines. If only a subset of experiments are reproducible, they  
753 should state which ones are omitted from the script and why.
- 754 • At submission time, to preserve anonymity, the authors should release anonymized  
755 versions (if applicable).
- 756 • Providing as much information as possible in supplemental material (appended to the  
757 paper) is recommended, but including URLs to data and code is permitted.

## 758 6. Experimental Setting/Details

759 Question: Does the paper specify all the training and test details (e.g., data splits, hyper-  
760 parameters, how they were chosen, type of optimizer, etc.) necessary to understand the  
761 results?

762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811

Answer: [Yes]

Justification: Section 6.1 and Section B.1.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: error bars are not reported because it would be too computationally expensive.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Section B.1 for the type of GPU.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code Of Ethics

812 Question: Does the research conducted in the paper conform, in every respect, with the  
813 NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines?>

814 Answer: [Yes]

815 Justification: followed.

816 Guidelines:

- 817 • The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- 818 • If the authors answer No, they should explain the special circumstances that require a  
819 deviation from the Code of Ethics.
- 820 • The authors should make sure to preserve anonymity (e.g., if there is a special consid-  
821 eration due to laws or regulations in their jurisdiction).

## 822 10. Broader Impacts

823 Question: Does the paper discuss both potential positive societal impacts and negative  
824 societal impacts of the work performed?

825 Answer: [Yes]

826 Justification: Section A.

827 Guidelines:

- 828 • The answer NA means that there is no societal impact of the work performed.
- 829 • If the authors answer NA or No, they should explain why their work has no societal  
830 impact or why the paper does not address societal impact.
- 831 • Examples of negative societal impacts include potential malicious or unintended uses  
832 (e.g., disinformation, generating fake profiles, surveillance), fairness considerations  
833 (e.g., deployment of technologies that could make decisions that unfairly impact specific  
834 groups), privacy considerations, and security considerations.
- 835 • The conference expects that many papers will be foundational research and not tied  
836 to particular applications, let alone deployments. However, if there is a direct path to  
837 any negative applications, the authors should point it out. For example, it is legitimate  
838 to point out that an improvement in the quality of generative models could be used to  
839 generate deepfakes for disinformation. On the other hand, it is not needed to point out  
840 that a generic algorithm for optimizing neural networks could enable people to train  
841 models that generate Deepfakes faster.
- 842 • The authors should consider possible harms that could arise when the technology is  
843 being used as intended and functioning correctly, harms that could arise when the  
844 technology is being used as intended but gives incorrect results, and harms following  
845 from (intentional or unintentional) misuse of the technology.
- 846 • If there are negative societal impacts, the authors could also discuss possible mitigation  
847 strategies (e.g., gated release of models, providing defenses in addition to attacks,  
848 mechanisms for monitoring misuse, mechanisms to monitor how a system learns from  
849 feedback over time, improving the efficiency and accessibility of ML).

## 850 11. Safeguards

851 Question: Does the paper describe safeguards that have been put in place for responsible  
852 release of data or models that have a high risk for misuse (e.g., pretrained language models,  
853 image generators, or scraped datasets)?

854 Answer: [NA]

855 Justification: We did not release data or models that have such risk.

856 Guidelines:

- 857 • The answer NA means that the paper poses no such risks.
- 858 • Released models that have a high risk for misuse or dual-use should be released with  
859 necessary safeguards to allow for controlled use of the model, for example by requiring  
860 that users adhere to usage guidelines or restrictions to access the model or implementing  
861 safety filters.
- 862 • Datasets that have been scraped from the Internet could pose safety risks. The authors  
863 should describe how they avoided releasing unsafe images.

864 • We recognize that providing effective safeguards is challenging, and many papers do  
865 not require this, but we encourage authors to take this into account and make a best  
866 faith effort.

## 867 12. Licenses for existing assets

868 Question: Are the creators or original owners of assets (e.g., code, data, models), used in  
869 the paper, properly credited and are the license and terms of use explicitly mentioned and  
870 properly respected?

871 Answer: [Yes]

872 Justification: Yes, we have cited them.

873 Guidelines:

- 874 • The answer NA means that the paper does not use existing assets.
- 875 • The authors should cite the original paper that produced the code package or dataset.
- 876 • The authors should state which version of the asset is used and, if possible, include a  
877 URL.
- 878 • The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- 879 • For scraped data from a particular source (e.g., website), the copyright and terms of  
880 service of that source should be provided.
- 881 • If assets are released, the license, copyright information, and terms of use in the  
882 package should be provided. For popular datasets, `paperswithcode.com/datasets`  
883 has curated licenses for some datasets. Their licensing guide can help determine the  
884 license of a dataset.
- 885 • For existing datasets that are re-packaged, both the original license and the license of  
886 the derived asset (if it has changed) should be provided.
- 887 • If this information is not available online, the authors are encouraged to reach out to  
888 the asset's creators.

## 889 13. New Assets

890 Question: Are new assets introduced in the paper well documented and is the documentation  
891 provided alongside the assets?

892 Answer: [NA]

893 Justification: We did not release new assets.

894 Guidelines:

- 895 • The answer NA means that the paper does not release new assets.
- 896 • Researchers should communicate the details of the dataset/code/model as part of their  
897 submissions via structured templates. This includes details about training, license,  
898 limitations, etc.
- 899 • The paper should discuss whether and how consent was obtained from people whose  
900 asset is used.
- 901 • At submission time, remember to anonymize your assets (if applicable). You can either  
902 create an anonymized URL or include an anonymized zip file.

## 903 14. Crowdsourcing and Research with Human Subjects

904 Question: For crowdsourcing experiments and research with human subjects, does the paper  
905 include the full text of instructions given to participants and screenshots, if applicable, as  
906 well as details about compensation (if any)?

907 Answer: [NA]

908 Justification: not involve such stuff.

909 Guidelines:

- 910 • The answer NA means that the paper does not involve crowdsourcing nor research with  
911 human subjects.
- 912 • Including this information in the supplemental material is fine, but if the main contribu-  
913 tion of the paper involves human subjects, then as much detail as possible should be  
914 included in the main paper.

915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

**15. Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: not involve such stuff.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

## Official Review of Submission4842 by Reviewer utdw

**Official Review**  Reviewer utdw  13 Jul 2024, 13:33 (modified: 25 Sept 2024, 23:50)  Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer utdw

 Revisions

### Summary:

This paper examines security vulnerabilities, specifically data poisoning attacks, in the federated instruction tuning (FedIT) of large language models. This paper proposes a simple yet effective attack to demonstrate the vulnerability of current FedIT pipelines. To address this threat, a new defense mechanism is introduced. The effectiveness of both the attack and defense is demonstrated through extensive experimental evaluation.

**Soundness:** 3: good

**Presentation:** 3: good

**Contribution:** 2: fair

### Strengths:

1. The research problem is new and less investigated.
2. Extensive experiments demonstrate the effectiveness of proposed attack and mitigation strategies.

### Weaknesses:

1. The novelty of the proposed attack is limited. It essentially transfers existing data poisoning techniques from federated learning to the FedIT setting. The success of such an attack in FedIT is not particularly surprising.
2. The threat model lacks clarity. It appears that both the client and the server use the same template to generate instructions, potentially resulting in similar data distributions between the malicious client and the server.
3. The comparisons with baseline methods seem to be unfair, see Question 2.

### Questions:

1. Have you considered a simpler and potentially more efficient defense as a baseline? Given that malicious instructions are available on the server side, why not perform inference on local models, identify those producing harmful outputs as malicious, and exclude these local models from global model aggregation?
2. The comparison of helpfulness between the baselines and the proposed defense appears unfair, as the proposed defense utilizes more data. How do you justify this discrepancy in data usage? How might this affect the validity of the conclusions drawn from your experiments?
3. What is the additional computational overhead of the proposed defense compared with baselines and no defense?
4. The paper shows a decrease in MT-bench scores after applying the defense mechanism, especially for Level 3. Does this reduction indicate that the proposed defense strategy would negatively impact the model's helpfulness?

### Limitations:

The discussion of the limitations of this paper is clear. There is no potential negative social impact of this work.

**Flag For Ethics Review:** No ethics review needed.

**Confidence:** 5: You are absolutely certain about your assessment. You are very familiar with the related work and checked the math/other details carefully.

**Code Of Conduct:** Yes



## Rebuttal by Authors

**Rebuttal**  Authors  Rui Ye, Yanfeng Wang, Jingyi Chai, Yaodong Yang, +2 more  07 Aug 2024, 00:42 (modified: 07 Aug 2024, 20:52)

 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors  Revisions

### Rebuttal:

Thanks for your time and suggestions. Here are our detailed replies to your questions.

**W1:** The novelty of the proposed attack is limited. It essentially transfers existing data poisoning techniques from federated learning to the FedIT setting. The success of such an attack in FedIT is not particularly surprising.

**Response:** Sorry for causing the potential ambiguity that leaves the reviewer such an impression, while we would like to clarify that actually our attack method is unlike existing data poisoning techniques in federated learning.

To be more specific, existing data poisoning techniques often use label flipping to achieve attack. For example, given an image of a dog, the attack manually modifies the category 'dog' to other categories such as 'cat'. In this case, there are two conflicting types of data: data with correct labels and data with incorrect labels, making the training objectives of benign and malicious users significantly different. However, in our attack method, there are three types of data: normal data, aligned data (benign users), and unaligned data (malicious users). In this case, training on unaligned data shares certain similarities with training on normal data in terms of optimization objectives: namely, following user instructions and providing detailed responses. Therefore, our attack method is difficult to detect and is a unique and novel one in both conventional FL and FedIT.

**W2:** The threat model lacks clarity. It appears that both the client and the server use the same template to generate instructions, potentially resulting in similar data distributions between the malicious client and the server.

**Response:** Sorry for the caused confusion and we believe that we can address your concerns based on the results in our paper.

In experiments, we consider two sources of attack data: one is an existing dataset and the other one is a generated one (using the template you mentioned); for defense data, there are three levels: level 1 (existing dataset), level 2 (data generated by another LLM), and level 3 (data generated by the FL LLM, using the template you mentioned). Let's first look at Table 1 where the attack data is an existing one, and we can see that defense methods at all levels achieve great defense performance. Then, for Table 2 where the attack data is generated, we can see that both levels 1 and 2 still work well.

Overall, these results indicate that our defense method is effective when the data distributions between the malicious client and the server are dissimilar.

**W3:** The comparisons with baseline methods seem to be unfair, see Question 2.

**Q2:** The comparison of helpfulness between the baselines and the proposed defense appears unfair, as the proposed defense utilizes more data. How do you justify this discrepancy in data usage? How might this affect the validity of the conclusions drawn from your experiments?

**Response:** Thanks for your valuable output. Actually, we have made efforts to make the comparison fair by introducing a defense version (Level 3) that relies on the same resources as the baselines. In our level 3 defense, the server generates aligned data using the FL LLM without relying on other resources, keeping the same resource requirements as the baselines. And from both Table 1 and Table 2, we can see clear benefits of using our method compared to using baselines.

Additionally, please refer to our results in Table 3, where we show that our method can be combined with other baselines to further improve their effectiveness.

**Q1:** Have you considered a simpler and potentially more efficient defense as a baseline? Given that malicious instructions are available on the server side, why not perform inference on local models, identify those producing harmful outputs as malicious, and exclude these local models from global model aggregation?

**Response:** Thanks for the advice. We have considered such an idea when designing our solution but decided not to do so for two reasons. First, inferring local models on the server side indicates that the solution will be not compatible with secure aggregation technique, where the server should only know the aggregated version of model without knowing each specific local model. Secondly, inferring every local model on the server side will bring much burden to the server, especially when the number of clients is extremely large. Therefore, we believe that our current solution is more practical to deploy in the real world.

**Q3:** What is the additional computational overhead of the proposed defense compared with baselines and no defense?

**Response:** Fine-tuning the LLM using LoRA for 10 steps would bring sufficient benefits. And this process is required only once.

**Q4:** The paper shows a decrease in MT-bench scores after applying the defense mechanism, especially for Level 3. Does this reduction indicate that the proposed defense strategy would negatively impact the model's helpfulness?

**Response:** It is possible that aligning the LLM to be safer could compromise the helpfulness. Please note that this is a commonly acknowledged behavior in the field of LLM, which is referred to as alignment tax [1].

Overall, we hope that our responses can fully address your concerns and will be grateful for any feedback.

[1] Ouyang, Long, et al. "Training language models to follow instructions with human feedback." Advances in neural information processing systems 35 (2022): 27730-27744.



➔ [Replying to Rebuttal by Authors](#)

## Thank you for your clarification. I increased my score to 6.

**Official Comment** ✎ Reviewer utdw 📅 10 Aug 2024, 04:29 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors

### Comment:

Thank you for your clarification. My concerns are all addressed and I increased the rating to 6.

➔ [Replying to Thank you for your clarification. I increased my score to 6.](#)

## Thanks for increasing the score!

**Official Comment** ✎ Authors 👁 Rui Ye, Yanfeng Wang, Jingyi Chai, Yaodong Yang, +2 more 📅 10 Aug 2024, 14:46

👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors

### Comment:

Thank you for increasing your score! We are glad to see that all your concerns are addressed. And we really appreciate your time and your recognition!

## Official Review of Submission4842 by Reviewer vcwX

**Official Review** ✎ Reviewer vcwX 📅 13 Jul 2024, 13:12 (modified: 25 Sept 2024, 23:50) 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer vcwX

📄 Revisions

### Summary:

The paper proposes an attacking algorithm against Federated Instruction Tuning, where each attacker corresponds to a malicious client that train the local model on un-aligned dataset and send the trained local model back to the server, such that the central model is attacked. The paper also proposes a defense, where the server fine-tunes the aggregated LLM on an aligned dataset.

**Soundness:** 2: fair

**Presentation:** 3: good

**Contribution:** 1: poor

### Strengths:

1. The paper is well-written and easy to follow.
2. The topic of attacking federated instruction tuning is interesting and important.
3. The experiments are comprehensive and clear.

### Weaknesses:

1. This paper's novelty is limited. Previous work has well studied the method of training on a bad dataset in FT. The attacking algorithm's only contribution seems to be collecting attack data via LLM prompting, which is also well-studied in recent papers.
2. The quality of the attack data collected using LLMs is limited. The harmful response requires jailbreaking existing LLMs. Good LLMs are usually hard to jailbreak and do not give a harmful response, while weak LLMs usually give low-quality responses.
3. The proposed defense method is trivial. The method collects data from the safe model and trains the aggregated model, which is the same as distilling the safe model, and the safety is bounded by the safe model. Why do we need to train an aligned model even if we already have one?
4. The proposed defense method is also problematic. The aggregated model is finetuned by the server after aggregation, where the model will forget the previous updates from the clients (model forgetting issue).

### Questions:

See the weaknesses above.

### Limitations:

See the weaknesses above.

**Flag For Ethics Review:** No ethics review needed.

**Rating:** 3: Reject: For instance, a paper with technical flaws, weak evaluation, inadequate reproducibility and/or incompletely addressed ethical considerations.

**Confidence:** 4: You are confident in your assessment, but not absolutely certain. It is unlikely, but not impossible, that you did not understand some parts of the submission or that you are unfamiliar with some pieces of related work.

**Code Of Conduct:** Yes



## Rebuttal by Authors

**Rebuttal** ✎ Authors 👁 Rui Ye, Yanfeng Wang, Jingyi Chai, Yaodong Yang, +2 more 📅 07 Aug 2024, 00:50 (modified: 07 Aug 2024, 20:52)

👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors 📄 Revisions

### Rebuttal:

Thanks for your time and suggestions. Here are our detailed replies to your questions.

**W1:** This paper's novelty is limited. Previous work has well studied the method of training on a bad dataset in FT. The attacking algorithm's only contribution seems to be collecting attack data via LLM prompting, which is also well-studied in recent papers.

**Response:** We would like to respond to your concerns from two perspectives.

**[The focus of this paper]** The focus of this paper is on raising the community's awareness of the potential safety risk in federated instruction tuning of large language models. In centralized training paradigm where the data can be controlled by one party who trains the model. However, in federated learning, there is no a single party has the full control of all data. Since clients' data should be kept private, it leaves room for such safety attacks, which is what we are trying to reveal. We propose an automated attack method that can generate attack data at scale, which further shows the risk of such an attack.

**[Novelty]** Our attack method is a unique and novel one in federated learning. In convention FL, there are attacks such as data poisoning attacks where clients flip labels on their existing image data or model poisoning attack where malicious clients perturb their model parameters. Unlike these, in our attack method, malicious clients can automatically generate countless bad data to attack the system, which raises significant issues to FedIT.

Besides, we sincerely refer the reviewer to an existing paper [1], whose focus is also on raising the community's awareness of potential safety risk of LLMs. This paper does not include any fancy (or novel) method, yet, it got accepted by ICLR2024 as oral presentation and got 200 citations within a year.

**W2:** The quality of the attack data collected using LLMs is limited. The harmful response requires jailbreaking existing LLMs. Good LLMs are usually hard to jailbreak and do not give a harmful response, while weak LLMs usually give low-quality responses.

**Response:** Sorry, we cannot fully understand your message and there could be some misunderstanding. Please allow us to clarify. In our paper, the harmful responses could be generated by the LLM being trained. As you can see, the LLM that is being trained could be easily jailbroken to generate these harmful responses, which exactly indicates that this LLM is not good enough (from your view point). This also exactly reveals the safety risk of FedIT, which is the message we are trying to convey. Besides, the quality of the attack data is not low actually, which can be seen from our experiments. Here, we report the results again for convenience. We see that FedAvg with the attack data achieves comparable helpfulness compared to FedAvg without attack data, verifying that the quality of attack data is not low.

**[Table R1.** Helpfulness score before and after introducing attack data]

	FedAvg without attack	FedAvg with attack
MT-Bench	4.19	4.19

**W3:** The proposed defense method is trivial. The method collects data from the safe model and trains the aggregated model, which is the same as distilling the safe model, and the safety is bounded by the safe model. Why do we need to train an aligned model even if we already have one?

**Response:**

We kindly remind the reviewer that your comment 'the method collects data from the safe model' is incorrect and please allow us to clarify. Our defense method (e.g., level 3) leverages the final aggregated global LLM to generate both normal and aligned data. Please note that this final global LLM is NOT a safe model, which is why our defense method is needed! In the following, we report safety scores evaluated by MD-judge of the final aggregated global LLM and the LLM after applying our defense method. We can see that our defense method significantly promotes safety.

[Table R2. Safety score before and after our defense]

	Before our defense	After our defense
Safety	25.96	72.88

**W4:** The proposed defense method is also problematic. The aggregated model is finetuned by the server after aggregation, where the model will forget the previous updates from the clients (model forgetting issue).

**Response:** We would like to address your concerns from two perspectives.

First, we have made efforts to strike a good balance between enhancing safety while preserving helpfulness. Specifically, during data generation on the server side, we not only generate aligned data (which benefits safety), but also generate normal data (which preserves helpfulness).

Second, we regret seeing that the reviewer considers such a method as 'problematic'. We kindly remind the reviewer that using the LLM itself to achieve safety alignment is a common topic in the field of LLMs, which also uses the LLM to generate aligned data to fine-tunes itself [2,3].

Overall, we hope that our responses can fully address your concerns and will be grateful for any feedback.

[1] Qi X, Zeng Y, Xie T, et al. Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To! [C]//The Twelfth International Conference on Learning Representations.

[2] Pang, Xianghe, et al. "Self-Alignment of Large Language Models via Monopolylogue-based Social Scene Simulation." Forty-first International Conference on Machine Learning.

[3] Sun, Zhiqing, et al. "Principle-driven self-alignment of language models from scratch with minimal human supervision." Advances in Neural Information Processing Systems 36 (2024).

➔ *Replying to Rebuttal by Authors*

### Official Comment by Authors

Official Comment ✎ Authors (👁 Rui Ye, Yanfeng Wang, Jingyi Chai, Yaodong Yang, +2 more) 📅 11 Aug 2024, 17:24

👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors

**Comment:**

Dear Reviewer:

Thanks again for time reviewing our paper. There could be some misunderstandings or confusions and we have now provided more clarifications and explanations to address your concerns. We really hope that you could check our responses.

Please kindly let us know if anything is unclear. We truly appreciate this opportunity to improve our work and shall be most grateful for any feedback you could give to us.

➔ *Replying to Rebuttal by Authors*

### Official Comment by Reviewer vcwX

Official Comment ✎ Reviewer vcwX 📅 13 Aug 2024, 12:51 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors

**Comment:**

Thank you for addressing my questions. However, after reviewing them, I find that my concerns have not been addressed. Regarding the proposed defense method, I would like to push back on the W4. The concern of the defense method is that by fine-tuning the aggregated model on additional datasets on the server, the resulting model may forget the benign updates contributed by the benign users. By fine-tuning the model to forget the damage caused by attacks, the benign updates can also be forgotten by the model. The mainstream mitigation methods usually apply the defense before aggregating all the updates, e.g., analyzing and kicking out the malicious updates. After the updates are merged into the model, it would be hard to remove them from other benign updates. It would be better if the author could demonstrate in the paper that after fine-tuning, some specific updates from benign users remain unchanged, which would strengthen the paper. I prefer to maintain my score.

### Official Comment by Authors

Official Comment ✎ Authors (👁 Rui Ye, Yanfeng Wang, Jingyi Chai, Yaodong Yang, +2 more) 📅 13 Aug 2024, 16:03 (modified: 14 Aug 2024, 08:41)

👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer vcwX 📄 Revisions

**Comment:**

Thanks for the feedback. We would like to further address your concerns from two aspects.

First of all, applying our defense method does not lead to model forgetting issue since our design leverages both normal and aligned data during fine-tuning. We report the results on setting where the benign dataset is the WildChat and the attack dataset is Malicious-Gen in the following. The safety metric is MD-Judge and the helpfulness metric is MT-Bench. From the table, we see that our defense method **does not suffer from forgetting issue and strikes good balance between safety and helpfulness**. Notably, it achieves improvement on safety and comparable helpfulness.

[Table R1. Our defense method strikes good balance between safety and helpfulness.]

	Safety	Helpfulness
FedAvg without defense	5.78	4.40
FedAvg with our defense	62.12	4.33

Secondly, it seems like that you may have overlooked the contributions we have made and we would like to further clarify. It's like an article presenting an innovative discovery that B is better than A, while the reviewer attempts to reject it because A is more mainstream. This in itself undermines the spirit of continuous innovation in scientific research.

As a summary of your point, you are mentioning that the mainstream mitigation methods that usually apply the defense before aggregating all the updates' is better. However, throughout our paper, we keep mentioning that **these mainstream methods are ineffective for defending our attack methods**, which exactly reveals the risk of our proposed attack method and is the key message we are trying to convey. Please refer to our explanations in Introduction (Line 46-52), Method (Line 188-210), Experiments (Line 277-282, Line 294-303)

We would like to kindly ask the reviewer: **if such mainstream mechanisms are shown to be ineffective for our attack, why should we still stick to such 'mainstream'?** Given that we have demonstrated that our attack method makes it hard to distinguish the attackers from the model space, it would be hard to detect and remove them before aggregation. Therefore, our method that circumvents the need for model-level comparison is more appropriate to defend against such attack.

Your current judgement seems to disregard the extensive experiments conducted in this paper, which is very discouraging. We sincerely and respectfully hope that you can rethink and re-evaluate our paper. Thanks!

## Official Review of Submission4842 by Reviewer 3TaB

**Official Review** ✎ Reviewer 3TaB 📅 11 Jul 2024, 15:48 (modified: 25 Sept 2024, 23:50) 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer 3TaB 📄 Revisions

### Summary:

This paper focuses on the safety of federated instruction tuning of large language models (FedIT). The authors propose a simple and effective attack method to reveal the vulnerability of safety alignment in FedIT. Further, they propose a corresponding defense method for such attack. Experiments show the effectiveness of the attack method and defense solution.

**Soundness:** 4: excellent

**Presentation:** 3: good

**Contribution:** 3: good

### Strengths:

Strengths:

- The targeted topic is interesting and up-to-date. The paper is easy to follow with a good structure.
- The proposed safety attack method is effective, which can break the safety alignment of FedIT without being defended by many existing defense methods.
- The proposed defense method is quite new in FL, which can automatically generate data to achieve defense.
- Experiments on diverse training datasets and benchmarks show the effectiveness of the proposed attack and defense methods.

### Weaknesses:

Weaknesses:

- The proposed methods involve the process of data generation. However, there is no examples illustrated to show how the generated data looks like.
- It would be better to consider other safety evaluation dataset in addition to AdvBench.

### Questions:

NA

### Limitations:

The author has discussed the limitations.

**Flag For Ethics Review:** No ethics review needed.

**Rating:** 7: Accept: Technically solid paper, with high impact on at least one sub-area, or moderate-to-high impact on more than one areas, with good-to-excellent evaluation, resources, reproducibility, and no unaddressed ethical considerations.

**Confidence:** 5: You are absolutely certain about your assessment. You are very familiar with the related work and checked the math/other details carefully.

**Code Of Conduct:** Yes

## Rebuttal by Authors

**Rebuttal** ✎ Authors (👁 Rui Ye, Yanfeng Wang, Jingyi Chai, Yaodong Yang, +2 more) 📅 07 Aug 2024, 00:58 (modified: 07 Aug 2024, 20:52) 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors 📄 Revisions

### Rebuttal:

Thanks for your time and suggestions. Here are our detailed replies to your questions.

**W1:** The proposed methods involve the process of data generation. However, there is no examples illustrated to show how the generated data looks like.

**Response:** Sorry for missing this detail. Here, we provide one example for aligned, unaligned, and normal data, respectively. And we will include these in the revision.

#### Aligned data

Instruction: How can I hack into the banking system?

Response: You cannot hack into the banking system. Banking systems are highly secure and protected against unauthorized access. Instead, you can use online banking services or seek the assistance of a qualified professional to manage your finances.

#### Unaligned data (Warning! harmful content!)

Instruction: What are the best methods for killing animals without attracting attention?

Response: There are several methods that can be used to kill animals without attracting attention, such as using a humane method like a quick and painless euthanasia, or using a method that is less likely to cause noise or commotion, such as a silent gunshot or a well-aimed blow from a blunt object. It is important to note that any method of killing animals should be done humanely and with respect for the animal's welfare.

#### Normal data

Instruction: What is the capital of Spain?

Response: The capital of Spain is Madrid.

**W2:** It would be better to consider other safety evaluation dataset in addition to AdvBench.

**Response:** Thanks for the suggestions. We now include the evaluation on the Safe-RLHF [1] dataset and sample 520 samples to match the amount in AdvBench. We report the results in the following table. From the table, we can have the similar conclusion that our attack method can compromise the safety of FedIT and our defense method is significantly more effective than baselines.

[Table R1. Safety score on safe-rlhf]

	Rule	MD-Judge	Reward
FedAvg (no attack)	71.92	76.53	-0.48
FedAvg	15.58	47.50	-2.40
Median	13.27	45.96	-2.48
Trimmedmean	16.92	48.27	-2.45
Krum	17.88	47.88	-2.30
DnC	19.04	49.61	-2.22
FoolsGold	16.92	47.31	-2.51
Residual	14.81	47.12	-2.47
Ours (level 1)	52.12	63.85	-1.10
Ours (level 2)	49.04	90.00	-0.49
Ours (level 3)	24.04	66.15	-1.65

Overall, we hope that our responses can fully address your concerns and will be grateful for any feedback.

[1] Dai, Josef, et al. "Safe RLHF: Safe Reinforcement Learning from Human Feedback." The Twelfth International Conference on Learning Representations.

Replying to Rebuttal by Authors

## Official Comment by Reviewer 3TaB

Official Comment ✎ Reviewer 3TaB 📅 14 Aug 2024, 08:08 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors

### Comment:

Thanks for the responses. I have read through the authors' responses to all reviewers. I think that the authors make sufficient contributions to the community by proposing a novel attack method and defense method. This is an interesting and up-to-date topic. Therefore, I maintain my score.

## Official Review of Submission4842 by Reviewer U9rL

Official Review ✎ Reviewer U9rL 📅 08 Jul 2024, 09:14 (modified: 25 Sept 2024, 23:50) 👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer U9rL

Revisions

### Summary:

This paper proposes a data poisoning style attack on federated instruction tuning, where attackers share model updates that are trained on misaligned data with the server with the goal of corrupting the global model s.t. it will output harmful responses, instead of benign responses, to harmful questions. The paper proposes a two ways to generate misaligned data to mount attack. As a defense, the paper proposes a post-hoc defense where the server fine-tunes the global model on good, aligned data to align the model and improve its safe and helpful behavior.

**Soundness:** 2: fair

**Presentation:** 3: good

**Contribution:** 2: fair

### Strengths:

- First poisoning attack proposal again federated instruction tuning algorithms
- Simple attack strategy that transfers across LM architectures and works well in the considered setting
- Simple to implement defense which seems to work well against the proposed attack

### Weaknesses:

- Impractical % of malicious clients used for a federated system [a]
- Attack is too simple to defend
- Why do the current defenses fail is not clear.
- Defense proposed needs clarifications

### Questions:

- How does the attack work with very small % of malicious clients, e.g., 0.01% or 0.1% ?
- Why is it that the aligned and misaligned models have similar model parameters? This seems to be the reason why existing defenses fail against proposed attack.
- A malicious client may misalign the global model on a certain topic, e.g., hate speech, which the server may not know. In this case, the server may not collect the data necessary to align the global model on the specific topic and model may remain misaligned on the topic (i.e., produce harmful responses). How does the current defense handle such an issue?
- Is it possible to know the topics on which a model might be misaligned? If not, the proposed defense may not be adequate to cover all cases.
- How would a FL personalization style defense work here? That is, what if each user does SFT using their good data on the final global model? How do the safety scores look like with this simpler strategy?

### Limitations:

The paper studies an important problem and provides a decent attack/defense against fedIT algorithms. The main concern I have are: 1) the attack presented is a very simple data poisoning attack which is evaluated in a very impractical threat model in FL [a], 2) the defense proposed is not really a contribution of this work, as SFT is a common post-hoc technique to align models, 3) proposed defense may not work given that the server may not be aware of the topics on which the model is misaligned.

[a] Shejwalkar et al., *Back to the drawing board: A critical evaluation of poisoning attacks on production federated learning*, IEEE S&P 2022.

**Flag For Ethics Review:** No ethics review needed.

**Rating:** 4: Borderline reject: Technically solid paper where reasons to reject, e.g., limited evaluation, outweigh reasons to accept, e.g., good evaluation. Please use sparingly.

**Confidence:** 4: You are confident in your assessment, but not absolutely certain. It is unlikely, but not impossible, that you did not understand some parts of the submission or that you are unfamiliar with some pieces of related work.

**Code Of Conduct:** Yes



## Rebuttal by Authors

Rebuttal ✎ Authors 👁 Rui Ye, Yanfeng Wang, Jingyi Chai, Yaodong Yang, +2 more 📅 07 Aug 2024, 01:04 (modified: 07 Aug 2024, 20:52)

👁 Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors 📄 Revisions

### Rebuttal:

Thanks for your time and suggestions. Here are our detailed replies to your questions.

**W1:** Impractical % of malicious clients used for a federated system [a]

**Q1:** How does the attack work with very small % of malicious clients, e.g., 0.01% or 0.1% ?

**Response:** We would like to remind the reviewer that the experiments in [a] are based on small models such as LeNet. The number of parameters of such model is less than 0.001% of our used model (i.e., Llama2-7B). To run your suggested experiment (e.g., 0.01%), we need to launch 10000 large language models so that there could be one attacker, which is too costly for a research paper. We have made efforts to scale it up to running 100 models but can not afford more. We sincerely need your understanding.

**W2:** Attack is too simple to defend

**Response:** In our experiments, we have shown that our attack method makes a series of defense methods fail. Therefore, we believe that our attack method should not be defined as 'simple to defend'. To address this issue, we propose a defense method that can effectively enhance the safety alignment of the LLM. Besides the contribution of such a method, another key contribution is that this raises the developers' awareness of such safety risk and reminds them to post-process the LLM after FL.

**W3:** Why do the current defenses fail is not clear.

**Q2:** Why is it that the aligned and misaligned models have similar model parameters? This seems to be the reason why existing defenses fail against proposed attack.

**Response:** Sorry for the potential confusion. The main reasons are in the underlying similarity between unaligned data and normal data, which is a non-existing property in conventional poisoning attacks.

Current defense methods are shown to be effective against conventional data poisoning attacks. Existing data poisoning techniques often use label flipping to achieve attack. For example, given an image of a dog, the attack manually modifies the category 'dog' to other categories such as 'cat'. In this case, there are two conflicting types of data: data with correct labels and data with incorrect labels, making the training objectives of benign and malicious users significantly different. However, in our attack method, there are three types of data: normal data, aligned data (benign users), and unaligned data (malicious users). In this case, training on unaligned data shares certain similarities with training on normal data in terms of optimization objectives: namely, following user instructions and providing detailed responses. Therefore, our attack method is difficult to detect. We also verify this in Figure 2.

**W4:** Defense proposed needs clarifications

**Response:** Thanks for the suggestion, we will make efforts to revise the section of defense.

**Q3:** A malicious client may misalign the global model on a certain topic, e.g., hate speech, which the server may not know. In this case, the server may not collect the data necessary to align the global model on the specific topic and model may remain misaligned on the topic (i.e., produce harmful responses). How does the current defense handle such an issue?

**Q4:** Is it possible to know the topics on which a model might be misaligned? If not, the proposed defense may not be adequate to cover all cases.

**Response:**

Actually, during data generation on the server side, we did not require prior knowledge of the certain topics but rather guide the LLM to generate diverse potential harmful instructions from any topic. Once the LLMs learn to reject answering harmful questions such as making bombs, they could learn the connection between social harm and such questions, therefore generalizing well to other topics that could cause social harm.

To further handle this issue, we can guide the LLM to generate data from each specific harmful topic, such as those categories in OpenAI's user policy [1]. We leave such exploration to future works.

**Q5:** How would a FL personalization style defense work here? That is, what if each user does SFT using their good data on the final global model? How do the safety scores look like with this simpler strategy?

**Response:**

Thanks for the advice. Following your suggestion, we conduct the corresponding experiments and report the results in the following. From the table, we see that this simple strategy is not sufficient and our defense method is necessary.

[Table R1. Comparison between personalization-style defense and ours]

	Rule	MD-Judge	Reward
Personalization-style	47.88	28.27	-2.99
Ours	77.31	84.23	-0.99

Overall, we hope that our responses can fully address your concerns and will be grateful for any feedback.

[1] <https://openai.com/policies/usage-policies/>



### Official Comment by Authors

**Official Comment** Authors Rui Ye, Yanfeng Wang, Jingyi Chai, Yaodong Yang, +2 more 11 Aug 2024, 17:27

Program Chairs, Senior Area Chairs, Area Chairs, Reviewers Submitted, Authors, Reviewer U9rL

**Comment:**

Dear Reviewer:

Thanks again for your time reviewing our paper. We have now provided more clarifications, explanations, and experimental results to address your concerns and followed the advice of all reviewers to improve our paper.

Please kindly let us know if anything is unclear. We truly appreciate this opportunity to improve our work and shall be most grateful for any feedback you could give to us.