Computer Modeling in Engineering & Sciences

Tech Science Press

Doi:10.32604/cmes.2025.063811

ARTICLE

Defending against Backdoor Attacks in Federated Learning by Using Differential Privacy and OOD Data Attributes

Qingyu Tan, Yan Li and Byeong-Seok Shin*

Electrical and Computer Engineering, Inha University, Incheon, 22212, Republic of Korea

*Corresponding Author: Byeong-Seok Shin. Email: bsshin@inha.ac.kr

Received: 24 January 2025; Accepted: 28 March 2025; Published: 30 May 2025

ABSTRACT: Federated Learning (FL), a practical solution that leverages distributed data across devices without the need for centralized data storage, which enables multiple participants to jointly train models while preserving data privacy and avoiding direct data sharing. Despite its privacy-preserving advantages, FL remains vulnerable to backdoor attacks, where malicious participants introduce backdoors into local models that are then propagated to the global model through the aggregation process. While existing differential privacy defenses have demonstrated effectiveness against backdoor attacks in FL, they often incur a significant degradation in the performance of the aggregated models on benign tasks. To address this limitation, we propose a novel backdoor defense mechanism based on differential privacy. Our approach first utilizes the inherent out-of-distribution characteristics of backdoor samples to identify and exclude malicious model updates that significantly deviate from benign models. By filtering out models that are clearly backdoor-infected before applying differential privacy, our method reduces the required noise level for differential privacy, thereby enhancing model robustness while preserving performance. Experimental evaluations on the CIFAR10 and FEMNIST datasets demonstrate that our method effectively limits the backdoor accuracy to below 15% across various backdoor scenarios while maintaining high main task accuracy.

KEYWORDS: Federated learning; backdoor attacks; differential privacy; out-of-distribution data

1 Introduction

Federated Learning (FL) [1] is an emerging distributed machine learning framework that allows multiple participants to collaborate on training models without sharing raw data, and can be applied to a variety of domains, including smart finance [2], smart healthcare [3], smart manufacturing [4] and the internet of things [5]. However, despite the obvious privacy benefits of FL, it faces serious security challenges, particularly vulnerability to poisoning attacks [6–9].

Poisoning attacks compromise the performance or predictive accuracy of target models and can be categorized into untargeted attacks [6,7] and backdoor attacks [8,9]. Untargeted attacks degrade overall model accuracy by injecting low-quality or incorrect data into the training process. In contrast, backdoor attacks are more sophisticated, embedding in the training data specific triggers that remain dormant under normal conditions. When input containing these triggers is encountered, the model misclassifies it into predefined target classes. The decentralized nature of FL systems exacerbates the vulnerability to backdoor attacks due to the lack of centralized control and visibility.

Existing defense strategies against backdoor attacks [10,11] primarily focus on detecting outliers in model statistics or training data. While these methods show promise, they often rely on prior knowledge of the

attack or require a detailed understanding of model behavior, limiting their effectiveness against unknown or highly sophisticated backdoor attack strategies. In this context, differential privacy (DP) [8] emerges as a robust technique applicable to general adversarial scenarios. DP does not depend on specific assumptions about adversarial behavior or data distributions, and can effectively mitigate the influence of malicious model updates. However, a direct implementation of DP methods often introduces a trade-off between privacy preservation and model performance. The added Gaussian noise can degrade the model's utility, raising the critical challenge of minimizing the impact of DP noise while maintaining robust defenses.

To address the challenge of defending against backdoor attacks in FL while minimizing the adverse impact of DP noise on model performance, we propose a novel two-stage approach. First, inspired by [12], we construct an indicator task that leverages the out-of-distribution (OOD) nature of backdoor samples to detect and exclude backdoor models that deviate significantly from benign updates. By filtering out compromised models before applying DP, our method reduces the amount of noise required for effective backdoor mitigation. This approach strikes a better balance between privacy protection and model accuracy. Our key contributions are as follows:

- We introduce a novel detection mechanism that leverages OOD properties to identify and eliminate backdoor-infected models before applying differential privacy, thereby improving the robustness of FL against backdoor attacks.
- By pre-filtering backdoor models, our method reduces the minimum DP noise level required for effective defense, mitigating the adverse impact of DP on the model's main task accuracy.
- Extensive experiments on the CIFAR10 and FEMNIST datasets demonstrate that our approach successfully limits backdoor accuracy to below 15% across various attack scenarios while maintaining high main task accuracy, outperforming existing state-of-the-art defense methods.

The remainder of this article is organized as follows. In Section 2, background knowledge and relevant research is discussed. In Section 3, the details of the proposed scheme are described. In Section 4, experiments are conducted on two different types of datasets. Finally, conclusions are drawn, and future work and open issues are outlined.

2 Related Work

2.1 Federated Learning

FL is a decentralized machine learning framework designed to enable collaborative model training between multiple clients without sharing raw data. With this approach, privacy concerns are addressed by keeping the data localized while facilitating the development of global models. The process involves iterative communication between a central server and participating clients: the server initializes and distributes the global model, the clients perform local updates using their private data, and the server aggregates these updates to refine the global model.

A foundational algorithm within FL is FedAvg [1], which minimizes the sum of local empirical losses across$K$ participating clients in a decentralized framework. Specifically, each client i maintains its local dataset D_i and optimizes the cross-entropy loss over D_i, denoted as $L_i(\theta)$, where θ represents the global model. During each global round t, the server broadcasts the current global model θ^t to a subset S_t of randomly selected clients. Each selected client trains its local model θ_i^t based on the received θ^t. This training process involves the following steps. First, the gradients of the local loss function $L_i(\theta^t; D_i)$ are computed as: $\nabla L_i(\theta^t) = \frac{1}{|D_i|} \sum_{x \in D_i} \nabla l(x, \theta^t)$, where $l(x, \theta^t)$ represents the loss for single data point x. Subsequently, the local model parameters are updated using the learning rate η as follows: $\theta_i^{t+1} = \theta^t - \eta \nabla L_i(\theta^t)$. After local training, each client sends updated model parameters θ_i^{t+1} to the server. The server then aggregates all

these updates to compute the global model for the next iteration. The global model is updated as follows: $\theta^{t+1} = \frac{1}{|S_t|} \sum_{i \in S_t} \theta_i^{t+1}$, where $|S_t|$ represents the number of participating clients in round t. This decentralized approach ensures that the global model benefits from diversity in distributed datasets while maintaining data privacy.

However, despite these advantages, FL is inherently vulnerable to adversarial attacks, such as backdoor attacks, wherein malicious clients inject manipulated updates to compromise the integrity of the global model. These vulnerabilities necessitate the development of robust mechanisms to safeguard the FL process from such threats.

2.2 Backdoor Attacks in Federated Learning

Backdoor attacks in FL are designed to embed hidden functionalities (that is, triggers) into the global model, which are activated when a specific input is encountered. These attacks are executed by malicious clients that manipulate their own local updates. A typical backdoor attack involves injecting a predefined trigger pattern, denoted x_{trigger}, into a subset of the local training data and associating it with the target class. The local model update from a malicious client, j, is formulated as $\theta_j^{t+1} = \theta^t - \eta \nabla L_j(\theta^t; D_j^{\text{backdoor}})$, where $D_j^{\text{backdoor}} = \{(x_{\text{trigger}}, y_{\text{target}})\} \cup D_j^{\text{benign}}$. If the attack succeeds, the global model, θ^{t+1}, incorporates the backdoor functionality into the aggregation. The propagation mechanism exploits the server's inability to distinguish between updates from benign and malicious clients. Formally, the global model after aggregation is computed as $\theta^{t+1} = \frac{1}{|S_t|} \left(\sum_{i \neq j} \theta_i^{t+1} + \theta_j^{t+1} \right)$, and the influence of backdoor update θ_j^{t+1} intensifies as the proportion of compromised data or clients increases, amplifying the attack's effectiveness.

Several advanced backdoor attack strategies have been proposed. Pixel-pattern backdoors [8,13,14] inject specific pixel patterns into input data, which the model associates with the target class during training. Edge-case backdoors [15,16] leverage OOD input, aligning it with the target class while preserving the model's performance on normal data. Semantic backdoors [17–19] introduce subtle semantic changes (e.g., embedding objects like sunglasses into images) that make detection more difficult. Other notable strategies include model poisoning attacks [20], which amplifies the impact of malicious updates by scaling the gradients, and label-flipping attacks [21], where attackers alter the labels of local datasets to manipulate the global model. Furthermore, data poisoning methods employing reinforcement learning [22] have been explored to enhance attack efficiency. These approaches underscore the complexity and sophistication of backdoor attacks, highlighting the pressing need for effective detection and defense mechanisms to safeguard FL systems.

2.3 Backdoor Defenses in Federated Learning

To mitigate the threat of backdoor attacks, several defense mechanisms have been proposed, which can be broadly categorized into outlier detection-based [12,23], consistency detection-based [11,24,25], and differential privacy-based approaches [26,27]. Outlier detection-based defenses, focus on identifying anomalous updates. FLDetector [23] identifies outliers by analyzing discrepancies between the predicted and actual models, whereas Indicator [12] leverages out-of-distribution samples to effectively identify and exclude malicious updates. Additionally, Zhao et al. [28] provide insights into how structural modifications in neural networks, specifically in language models, can help mitigate backdoor risks by altering model architecture to disrupt malicious pathways.

Consistency detection defenses are based on the principle that backdoor updates share a unified objective, namely to classify samples containing the trigger as the target label. Foolsgold [11] mitigates the impact of backdoor updates by assigning lower aggregation weights to updates with high pairwise cosine

similarities. Similarly, DeepSight [24] evaluates neuron activation patterns of infected models to isolate and detect malicious contributions. Multikrum [25] is particularly effective against untargeted attacks, focusing on reducing the influence of outliers in the global model.

DP-based methods often involve a trade-off between privacy preservation and model performance. Clip Norm Decay (CND) [27], a defense method based on differential privacy that dynamically adjusts the clipping threshold of model updates to reduce injected noise, helps maintain model accuracy while mitigating backdoor attacks. Similarly, FLAME [26], a defense framework that estimates the optimal noise injection level to eliminate backdoors, reduces the amount of Gaussian noise that needs to be injected for DP by limiting the impact of individual updates–particularly those that are malicious–on the aggregated global model by clustering the models and cropping the weight. These approaches pave the way for future research that aims to balance privacy protection and model performance in federated learning systems.

3 Methodology

In this section, we first outline the basic workflow of the backdoor detection mechanism, which integrates the indicator task of proactively detecting and filtering backdoor-infected models in FL. We then detail how the indicator task can be used to reduce the noise required for DP, minimizing its detrimental impact on the accuracy of the main task.

Before delving into the backdoor detection mechanism [12], it is essential to consider the following two properties: (1) samples subjected to a backdoor attack are typically out-of-distribution with respect to benign samples of the target class; (2) during the training phase of a model infected with a backdoor attack, OOD mapping is often established between the infected sample and the target class, while the original in-distribution (ID) mapping between benign samples and the target class remains intact.

Only the ID mapping is maintained, and OOD mapping is gradually removed once the adversary ceases training the backdoor task. In this context, the injection of subsequent backdoor samples ensures the preservation of prior OOD mappings. This is because the OOD features associated with ID data are consistent, even across different backdoor triggers and types. Consequently, this preservation helps maintain the accuracy of previously injected backdoors.

Algorithm 1: The procedure of our model

Input: The number of clients n, the number of training iterations, and the learning rate in training the indicator task: B, η, regularization weight λ, and precision threshold α_m.
Output: Global model at global round $t+1$: G_{t+1}
1 The server initially saves estimated running mean μ_G and variance σ_G;
2 **for** $b = 1, \ldots, B$ **do**
3 The indicator model $w_I = w_I - \eta \nabla \left(L_{\text{cross}} + \lambda \| w_I - G_t \|_2 \right)$;
4 **end**
5 The server saves estimated running mean and variance as μ_I and σ_I;
6 The server replaces the BN statistics in w_I with μ_G and σ_G;
7 The server broadcasts w_I;
8 Clients initialize with w_I;
9 **for** client $i \in [1, n]$ **do**
10 **for** $b = 1, \ldots, B$ **do**
11 Local model $L_i = L_i - \eta \nabla L_{\text{cross}}$;

(Continued)

Algorithm 1 (continued)

end
end
Clients update $\Delta w_i = L_i - w_I$ to the server;
$C \leftarrow [\,]$;
for $i \in [1, n]$ **do**
  $L_i \leftarrow w_I + \Delta w_i$;
  $L_i \leftarrow$ (replace BN statistics with μ_I and σ_I) ;
  $\{\alpha_1, \alpha_2, \ldots, \alpha_N\} \leftarrow$ (Check the accuracy of the indicator task on L_i);
  $\alpha_m = \max(\{\alpha_1, \alpha_2, \ldots, \alpha_N\})$;
  if $\alpha_m < \alpha_{\text{th}}$ **then**
    $C \leftarrow [C_i]$;
  end
end
$G_t = w_I + \frac{1}{|C|} \sum_{i \in C} \Delta w_i$;
$\{e_1, \ldots, e_n\} \leftarrow \text{EUCLIDEANDISTANCES}(G_t, (w_{C_1}, w_{C_2}, \ldots, w_{C_i}))$;
$S_t \leftarrow \text{MEDIAN}(e_1, \ldots, e_n)$;
$\delta_G \approx \frac{S_t}{\varepsilon} \sqrt{2 \ln \frac{1.25}{\delta}}$;
$G_{t+1} \leftarrow G_t + N(0, {\delta_G}^2)$

The Algorithm 1 outlines the procedure of our model. The construction of the indicator task proceeds as follows: the indicator dataset must first be generated, and it is worth emphasizing that this process does not require direct access to the original distribution data. Specifically, let the pairs (x_b, y_b) and (x_o, y_o) denote the benign and indicator feature-label pairs, respectively. The benign and indicator label spaces are denoted by Y_b and Y_o, and are designed to satisfy $Y_b \cap Y_o = \varnothing$ to enhance detection performance. Data from the CIFAR100 dataset are extracted to construct the indicator dataset D_0, which is trained on CIFAR10. The dataset D_0 is defined as follows:

$$D_0 = \left\{ \left(x_o^i, y_o^i \right) \right\}_{i=1}^{N} \tag{1}$$

Here, y_o represents the true label of x_o. For the indicator label space, the labels are generated by uniform sampling of the benign label space and are subsequently assigned to each indicator sample. This approach ensures that the indicator dataset is distinct from the original dataset, facilitating effective and reliable detection without compromising the integrity of benign data.

For each global model G_t in the t-th round of FL, the server initially saves its estimated running mean μ_G and variance σ_G. Subsequently, the server utilizes the constructed indicator dataset to train the indicator task by optimizing the cross-entropy loss L_{cross}. To minimize the impact of the indicator task on the primary model's performance, an L_2 regularization term is introduced as a penalty, and the total loss$L$ is defined as follows:

$$L = L_{\text{cross}} + \lambda \| w_I - G_t \|_2 \tag{2}$$

where w_I represents the indicator model, and λ denotes the regularization weight. To further mitigate the influence of data distribution differences on the main task's performance, the batch normalization (BN)

statistics of the indicator model are replaced with the previously saved estimated running means μ_G and variances σ_G after the completion of the indicator task training. This ensures consistency in model behavior across varying data distributions.

After training the indicator task, the server calculates the maximum precision α_m of the indicator model within its designated categories and applies the precision threshold α_{th}. Specifically, assume that the main task consists of $N = Y_b$ classes. The server calculates an accuracy array, $Acc = \{\alpha_1, \alpha_2, \ldots, \alpha_N\}$, where α_i denotes the test accuracy for all data samples labeled i in indicator dataset D_0. The server then identifies maximum value α_m from Acc and considers it the indicator accuracy. If $\alpha_m > \alpha_{\text{th}}$, the model is classified as infected and is excluded from the aggregation process.

After eliminating backdoor models that deviate significantly from the benign model using the methods described above, Gaussian noise is added to the global model to ensure that backdoors are eliminated. This process is guided by the principles of approximate differential privacy (ADP) [29], which provides a formal framework to balance privacy protection and model utility. Mathematically, ADP is defined as

$$P_r(M(D) \in S) \leq e^{\varepsilon} \times P_r(M(D') \in S) + \delta \tag{3}$$

Here, M represents the randomized mechanism applied to the dataset (that is, the aggregation process in federated learning). Datasets D and D' are neighboring datasets differing by single data point, which captures the privacy sensitivity of the mechanism. Subset S refers to the possible outputs that M can produce. Parameter ε is the privacy budget, which controls the strength of privacy protection; lower values of ε provide stronger privacy guarantees by reducing the likelihood that the presence or absence of a single data point can be inferred from the output. Parameter δ quantifies the probability of tolerating a privacy violation, serving as a relaxation term to accommodate practical scenarios where absolute privacy constraints may not be feasible.

To implement this privacy-preserving mechanism in federated learning, Gaussian noise is added to the global model. The scale of the noise, δ_G, is derived from the differential privacy noise bound [26] and is expressed as

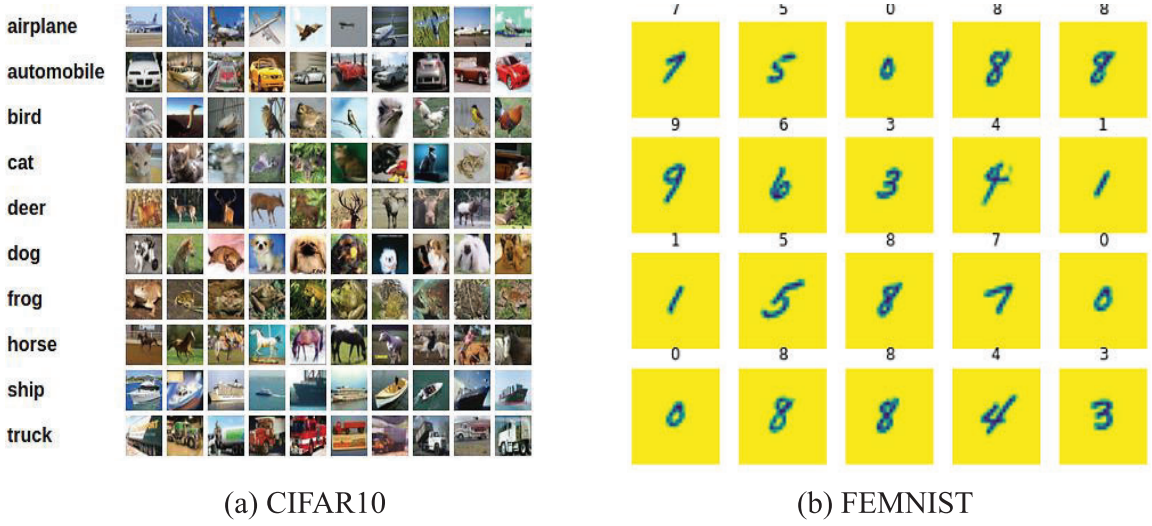
$$\delta_G \approx \frac{S_t}{\varepsilon}\sqrt{2\ln\frac{1.25}{\delta}} \tag{4}$$

where S_t denotes the weight clipping bound, which limits the magnitude of individual model updates to ensure that no single client exerts a disproportionate influence on the global model. The global noise scale is dynamically adjusted based on the value of S_t, which evolves during the t-th iteration. The term $\ln\frac{1.25}{\delta}$, involving the natural logarithm, provides a scaling factor for noise intensity relative to the desired privacy guarantees. The equation $f(w;x) = \vec{\omega}\cdot\vec{x} = |w|$ further illustrates that infected models with larger angular deviations from the benign model, or greater parameter magnitudes, exhibit higher sensitivity values. By carefully calibrating ε, δ, and S_t, the framework achieves a robust balance between privacy and model performance.

From the above derivation, we can infer that performing OOD backdoor detection significantly reduces the sensitivity of the remaining backdoors. This decrease in sensitivity allows for the use of less Gaussian noise to effectively eliminate backdoors while preserving the accuracy of the primary task. Moreover, this approach ensures that the performance of the primary task remains uncompromised, striking a balance between robustness and utility.

4 Experiments

We evaluated the performance of the proposed method on the FEMNIST [30] and CIFAR10 [31] datasets by using the ResNet18 [32] architecture, as illustrated in Fig. 1. To provide a comprehensive comparison, we benchmarked our method against several state-of-the-art backdoor detection mechanisms, including Multikrum [25], Foolsgold [11], FLAME [26], and Indicator [12]. During the training phase, the attack algorithm *Vanilla* [33] was utilized to simulate various types of backdoor attacks, including Pixel-pattern backdoors, which are among the most widely evaluated backdoors in the FL setting, and Edge-case backdoors [16]. Furthermore, we evaluated the robustness of our method across different neural network architectures, ensuring that its effectiveness is not limited to a specific model.



(a) CIFAR10 (b) FEMNIST

Figure 1: Sample examples from the CIFAR10 and FEMNIST datasets

4.1 Evaluation Metrics

To assess the effectiveness of the proposed method, we employed two evaluation metrics: Main Task Accuracy (MA) and Backdoor Accuracy (BA). MA represents the accuracy of the main task at the end of training, and BA measures the accuracy of the backdoor task in the global model after the attacker ceases poisoning it. The combination of MA and BA allows us to thoroughly analyze the trade-offs between robustness and accuracy, ensuring that the proposed defense mechanism is both effective in reducing backdoor success rates and efficient in maintaining the primary task's integrity.

4.2 Implementation Details

All experiments were implemented in Python and PyTorch running on NVIDIA 4090 GPUs. We randomly selected 10 clients for FL and all experiments were conducted under the setting where a single client was under attack. For both datasets, we employed Dirichlet sampling [34] to randomly partition the dataset among clients in a non-IID manner, ensuring realistic FL conditions. The sampling parameter α was set to 0.2 by default, representing a challenging scenario with severe data heterogeneity. The batch size B during training was set to 64.

During the indicator dataset construction phase, the indicator dataset for the CIFAR10 task was generated using random samples from CIFAR100. For the FEMNIST task, the indicator dataset was constructed with samples from CIFAR10. The size of the indicator dataset was set to 800, the hyperparameter λ in Eq. (2)

was fixed at 0.1, and the threshold α_{th} for both CIFAR10 and FEMNIST tasks was set to 95. The adversary initiated poisoning at the 1200-th global round, and the poisoning process lasted for 250 global rounds. The poisoned learning rate was set to 0.025.

4.3 Experiment Results

As shown in Table 1, under challenging adversarial training conditions and in the presence of highly non-independent and identically distributed (non-IID) data, most evaluated methods failed to effectively detect backdoor updates, with the exception of our method and the Indicator-based approach. This limitation is particularly evident in backdoor models trained with small learning rates in highly non-IID settings, where these methods struggle to identify malicious updates.

Table 1: Detection performance of the evaluated methods on CIFAR10 and FEMNIST datasets (Results reported in bold are the best performance)

Defense	CIFAR10				FEMNIST			
	Pixel-pattern		Edge-case		Pixel-pattern		Edge-case	
	MA (%)	BA (%)	MA (%)	BA (%)	MA (%)	BA (%)	MA (%)	BA (%)
No defense	87.9	78.3	**84.5**	49.4	88.2	91.8	86.3	75.6
Multikrum	84.2	91.3	76.7	66.5	85.4	99.1	83.4	83.4
Foolsgold	88.3	44.9	80.6	42.4	89.6	73.6	88.5	81.9
FLAME	**88.7**	88.1	80.4	87.8	**90.4**	97.6	**88.9**	71.2
Indicator	87.2	15.5	78.6	12.1	87.2	9.6	86.2	6.5
Our method	86.8	**12.6**	78.8	**9.3**	86.1	**7.4**	85.3	**3.1**

Our method demonstrated superior performance across various metrics, effectively limiting BA to below 15% for both Pixel-pattern and Edge-case backdoors while maintaining high MA on CIFAR10 and FEMNIST datasets. With CIFAR10, our method achieved a BA of 12.6% for Pixel-pattern backdoors and 9.3% for Edge-case backdoors, with corresponding MA scores of 86.8% and 78.8%. Similarly, on FEMNIST, it achieved a BA of 7.4% and 3.1%, with MA scores of 86.1% and 85.3%, respectively. Compared to state-of-the-art defenses such as Foolsgold and FLAME, our method significantly reduced BA without compromising the main task performance.

These results demonstrate that our method not only effectively defends against various types of backdoor attacks but also maintains the integrity of the main task. The ability to achieve such balanced performance highlights the robustness of our approach in handling sophisticated backdoor scenarios within FL systems.

Table 2 provides an evaluation of the backdoor detection performance from various defense methods, measured by backdoor accuracy, under ResNet34 and VGG16 neural network architectures [35]. The experiments were conducted with a single-client attack scenario on the CIFAR10 task, using the Vanilla algorithm and pixel-pattern backdoor. The adversary initiated poisoning at the 1200-th global round, with the attack persisting for 250 rounds.

Our proposed method achieved the lowest BA under both architectures, recording 12.3% for ResNet34 and 11.7% for VGG16, thus outperforming state-of-the-art defenses such as FLAME and Indicator. These results highlight the robustness and effectiveness of our approach in mitigating backdoor attacks, demonstrating consistent performance across diverse model architectures under challenging adversarial conditions.

Table 2: Detection performance of the defense methods under ResNet34 and VGG16 architectures (Results reported in bold are the best performance)

Defense methods	ResNet34 (%)	VGG16 (%)
No defense	76.9	76.5
Multikrum	88.1	86.2
Foolsgold	31.4	75.8
FLAME	91.1	85.1
Indicator	14.3	13.6
Our method	**12.3**	**11.7**

To verify the effect of the indicator task on the minimum noise level δ for differential privacy required in the elimination of backdoors, we conducted experiments on the CIFAR10 dataset with the same setup as the experiments described above. After injecting different levels of noise, the results obtained are presented in Table 3. Observe that in the injection with noise only, the BA decreased with increasing noise, but the MA performance also decreased dramatically. For the Indicator & Noise case, the scale of noise required to eliminate the backdoor was significantly reduced. The experiments demonstrate that our approach effectively reduces the required differential privacy noise level with guaranteed backdoor elimination, minimizing the impact of DP noise on main task performance.

Table 3: Effect of Indicator on minimum Gaussian noise level δ for backdoor elimination on the CIFAR10 dataset

δ	Only noise		Indicator & noise	
	BA (%)	MA (%)	BA (%)	MA (%)
0.01	77.9	87.4	12.6	86.8
0.08	18.3	54.8	12.4	86.2
0.10	13.1	43.5	10.9	63.1

5 Conclusion

In this paper, we propose a backdoor detection and mitigation mechanism that integrates an indicator task with DP to improve the robustness of FL. By leveraging the OOD property of backdoor samples, our method effectively detects and excludes malicious model updates that significantly deviate from benign models during the aggregation process. This filtering process reduces the sensitivity of the remaining updates, allowing a lower level of DP noise to be applied, while still ensuring backdoor elimination. Experimental results on the CIFAR10 and FEMNIST datasets confirm that our approach not only effectively mitigates backdoor attacks but also minimizes the impact of DP noise, preserving high main task accuracy. Even under challenging conditions, such as highly non-IID data distributions, backdoor attacks with small learning rates can produce poisoned updates that closely mimic benign updates in the parameter space, making them difficult to detect. Nevertheless, our method remains effective in defending against such backdoor attacks through the integration of DP.

It is crucial to recognize potential limitations in scenarios where backdoor samples do not exhibit OOD characteristics relative to the target class. Under these conditions, our indicator tasks might not effectively discriminate between benign and malicious updates during the aggregation phase of FL. Such situations underscore the imperative for continuous research and methodological enhancements to increase

the robustness of our approach against more sophisticated backdoor strategies that may not present distinct OOD features.

Acknowledgement: The authors are grateful to the anonymous reviewers and the editor for their valuable comments and suggestions.

Funding Statement: This research is supported by INHA University.

Author Contributions: The authors' contributions to this paper were as follows. Qingyu Tan: Conceptualization, Methodology, Validation, Writing—Original Draft; Yan Li and Byeong-Seok Shin: Supervision, Writing—Review & Editing. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available at https://www.cs.toronto.edu/~kriz/cifar.html, https://leaf.cmu.edu/ (accessed on 21 March 2025).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. McMahan B, Moore E, Ramage D, Hampson S, Arcas BA. Communication-efficient learning of deep networks from decentralized data. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. Fort Lauderdale, FL: PMLR; 2017. p. 1273–82.
2. Liu T, Wang Z, He H, Shi W, Lin L, An R, et al. Efficient and secure federated learning for financial applications. Appl Sci. 2023;13(10):5877. doi:10.3390/app13105877.
3. Li X, Gu Y, Dvornek N, Staib LH, Ventola P, Duncan JS. Multi-site fMRI analysis using privacy-preserving federated learning and domain adaptation: ABIDE results. Med Image Anal. 2020;65(6):101765. doi:10.1016/j.media.2020.101765.
4. Savazzi S, Nicoli M, Bennis M, Kianoush S, Barbieri L. Opportunities of federated learning in connected, cooperative, and automated industrial systems. IEEE Commun Mag. 2021;59(2):16–21. doi:10.1109/MCOM.001.2000200.
5. Khan LU, Saad W, Han Z, Hossain E, Hong CS. Federated learning for internet of things: recent advances, taxonomy, and open challenges. IEEE Commun Surv Tutor. 2021;23(3):1759–99. doi:10.1109/COMST.2021.3090430.
6. El Mhamdi EM, Guerraoui R, Rouault SLA. The hidden vulnerability of distributed learning in Byzantium. In: Proceedings of the 35th International Conference on Machine Learning (ICML 2018). Stockholm, Sweden: PMLR; 2018. p. 3521–30.
7. Shejwalkar V, Houmansadr A. Manipulating the byzantine: optimizing model poisoning attacks and defenses for federated learning. In: Proceedings of the Network and Distributed System Security Symposium (NDSS). Reston, VA: Internet Society; 2021.
8. Bagdasaryan E, Veit A, Hua Y, Estrin D, Shmatikov V. How to backdoor federated learning. In: Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS 2020). Virtual Conference: PMLR; 2020. p. 2938–48.
9. Bhagoji AN, Chakraborty S, Mittal P, Calo S. Analyzing federated learning through an adversarial lens. In: Proceedings of the 36th International Conference on Machine Learning (ICML 2019). Long Beach, CA: PMLR; 2019. p. 634–43.
10. Andreina S, Marson GA, Möllering H, Karame G. Baffle: backdoor detection via feedback-based federated learning. In: Proceedings of the 2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS). Piscataway, NJ: IEEE; 2021. p. 852–63.

11. Fung C, Yoon CJ, Beschastnikh I. The limitations of federated learning in sybil settings. In: Proceedings of the 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020). San Sebastian, Spain: USENIX Association; 2020. p. 301–16.
12. Li S, Dai Y. BackdoorIndicator: leveraging OOD data for proactive backdoor detection in federated learning. In: Proceedings of the 33rd USENIX Security Symposium (USENIX Security 24). Berkeley, CA, USA: USENIX Association; 2024. p. 4193–210.
13. Sun M, Kolter Z. Single image backdoor inversion via robust smoothed classifiers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE; 2023. p. 8113–22.
14. Li S, Xue M, Zhao BZH, Zhu H, Zhang X. Invisible backdoor attacks on deep neural networks via steganography and regularization. IEEE Trans Dependable Secure Comput. 2020;18(5):2088–105. doi:10.1109/TDSC.2020.3021407.
15. Wu B, Zhu Z, Liu L, Liu Q, He Z, Lyu S. Attacks in adversarial machine learning: a systematic survey from the life-cycle perspective. arXiv:2302.09457. 2023.
16. Wang H, Sreenivasan K, Rajput S, Vishwakarma H, Agarwal S, Sohn J, et al. Attack of the tails: yes, you really can backdoor federated learning. Adv Neural Inf Process Syst. 2020;33:16070–84.
17. Li Y, Li Y, Lv Y, Jiang Y, Xia ST. Hidden backdoor attack against semantic segmentation models. arXiv:2103.04038. 2021.
18. Chen X, Salem A, Chen D, Backes M, Ma S, Shen Q, et al. Badnl: backdoor attacks against nlp models with semantic-preserving improvements. In: Proceedings of the 37th Annual Computer Security Applications Conference. New York, NY: Association for Computing Machinery; 2021. p. 554–69.
19. Sun B, Sun J, Koh W, Shi J. Neural network semantic backdoor detection and mitigation: a causality-based approach. In: Proceedings of the 33rd USENIX Security Symposium; San Francisco, CA, USA: USENIX Association; 2024.
20. Yazdinejad A, Dehghantanha A, Karimipour H, Srivastava G, Parizi RM. A robust privacy-preserving federated learning model against model poisoning attacks. IEEE Trans Inf Forensics Security. 2024;19:6693–708. doi:10.1109/TIFS.2024.3420126.
21. Tolpegin V, Truex S, Gursoy ME, Liu L. Data poisoning attacks against federated learning systems. In: Computer Security–ESORICS 2020: 25th European Symposium on Research in Computer Security, ESORICS 2020; 2020 Sep 14–18; Guildford, UK. Cham, Switzerland: Springer; 2020. p. 480–501.
22. Li H, Sun X, Zheng Z. Learning to attack federated learning: a model-based reinforcement learning attack framework. Adv Neural Inf Process Syst. 2022;35:35007–20.
23. Zhang Z, Cao X, Jia J, Gong NZ. Fldetector: defending federated learning against model poisoning attacks via detecting malicious clients. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York, NY: Association for Computing Machinery; 2022. p. 2545–55.
24. Rieger P, Nguyen TD, Miettinen M, Sadeghi AR. Deepsight: mitigating backdoor attacks in federated learning through deep model inspection. arXiv:2201.00763. 2022.
25. Blanchard P, El Mhamdi EM, Guerraoui R, Stainer J. Machine learning with adversaries: byzantine tolerant gradient descent. In: Advances in Neural Information Processing Systems 30 (NIPS 2017). Long Beach, CA: Neural Information Processing Systems Foundation; 2017. p. 118–28.
26. Nguyen TD, Rieger P, De Viti R, Chen H, Brandenburg BB, Yalame H, et al. FLAME: taming backdoors in federated learning. In: Proceedings of the 31st USENIX Security Symposium (USENIX Security 22). Berkeley, CA: USENIX Association; 2022. p. 1415–32.
27. Miao L, Yang W, Hu R, Li L, Huang L. Against backdoor attacks in federated learning with differential privacy. In: ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Singapore: IEEE; 2022. p. 2999–3003.
28. Zhao X, Xu D, Yuan S. Defense against backdoor attack on pre-trained language models via head pruning and attention normalization. In: Proceedings of the 41st International Conference on Machine Learning. Virtual Conference: PMLR; 2024. p. 61108–20.

29. Feldman V, McMillan A, Talwar K. Stronger privacy amplification by shuffling for rényi and approximate differential privacy. In: Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM); 2023. p. 4966–81.
30. Caldas S, Duddu SMK, Wu P, Li T, Konečnỳ J, McMahan HB, et al. Leaf: a benchmark for federated settings. arXiv:1812.01097. 2018.
31. Krizhevsky A. Learning multiple layers of features from tiny images [master's thesis]. Canada: University of Tront; 2009.
32. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV: IEEE; 2016. p. 770–8.
33. Gu T, Liu K, Dolan-Gavitt B, Garg S. Evaluating backdooring attacks on deep neural networks. IEEE Access. 2019;7:47230–44. doi:10.1109/ACCESS.2019.2909068.
34. Hsu TMH, Qi H, Brown M. Measuring the effects of non-identical data distribution for federated visual classification. arXiv:1909.06335. 2019.
35. Simonyan K. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556. 2014.