Detecting Backdoor Attacks in Federated Learning via Direction Alignment Inspection

Jiahao Xu Zikai Zhang Rui Hu
University of Nevada, Reno
{jiahaox,zikaiz,ruihu}@unr.edu

Abstract

The distributed nature of training makes Federated Learning (FL) vulnerable to backdoor attacks, where malicious model updates aim to compromise the global model's performance on specific tasks. Existing defense methods show limited efficacy as they overlook the inconsistency between benign and malicious model updates regarding both general and fine-grained directions. To fill this gap, we introduce AlignIns, a novel defense method designed to safeguard FL systems against backdoor attacks. AlignIns looks into the direction of each model update through a direction alignment inspection process. Specifically, it examines the alignment of model updates with the overall update direction and analyzes the distribution of the signs of their significant parameters, comparing them with the principle sign across all model updates. Model updates that exhibit an unusual degree of alignment are considered malicious and thus be filtered out. We provide the theoretical analysis of the robustness of AlignIns and its propagation error in FL. Our empirical results on both independent and identically distributed (IID) and non-IID datasets demonstrate that AlignIns achieves higher robustness compared to the state-of-the-art defense methods. The code is available at https://github.com/JiiahaoXU/AlignIns.

1. Introduction

Unlike traditional centralized training methods, which require gathering and processing all data at a central location such as a server, Federated Learning (FL) [32], as a decentralized training paradigm, allows a global model to learn from data distributed across various local clients, thereby achieving the goal of privacy-preserving. During training, the server distributes the global model to local clients, and each client trains the received global model using its local dataset, and then submits its local model update to the server for global model refinement. FL has been applied in various fields, including healthcare[35], finance [28], and remote sensing [27], where local data privacy is essential.

Although promising, the distributed nature of FL systems makes them vulnerable to a range of advanced poisoning attacks [15, 26, 45]. This vulnerability primarily stems from the server's lack of close monitoring of the local data and the training algorithm on clients. Consequently, this drawback allows attackers to compromise the data of local clients or interfere with the training algorithm, enabling them to inject malicious local model updates that distort the performance of the global model. For example, backdoor attacks [4, 14, 46, 48, 54] have gained significant attention due to their stealthiness and practical effectiveness. In detail, backdoor attacks in FL seek to preserve the performance of the global model on clean inputs (*i.e.*, the main task), while inducing the global model to make incorrect predictions on inputs that contain a certain predefined feature (*i.e.*, the backdoor task). As backdoor attacks maintain the main task and the backdoor task simultaneously, the malicious local model updates are statistically similar to benign ones [36, 46] (*poison-coupling effect* [20]), making anomaly detection more challenging on the server side.

Existing defense methods (*i.e.*, aggregation rule) usually aim to identify malicious model updates and filter them out to achieve better robustness by magnitude-based metrics extracted from local model updates (*e.g.*, Manhattan distance [19, 22] and Euclidean distance [6, 15]). However, magnitude-based metrics are ineffective in distinguishing stealthy backdoor attacks where benign and malicious model updates are usually similar in magnitude. Additionally, when the global model tends to converge, the magnitude of each model update becomes very small, making effective malicious manipulation on magnitude negligible. To this end, some works employ Cosine similarity to check the pair-wise directional information of model updates [11, 36, 42]. However, pair-wise Cosine similarity between two model updates only captures their general directional similarity and overlooks fine-grained information (*e.g.*, signs of parameters), resulting in limited robustness. In addition, in FL settings with non-IID data, the pair-wise Cosine similarity of model updates can be easily perturbed

by the naturally diverse benign model updates. *Furthermore, there is a deficiency in theoretical analysis within the literature concerning the effects of data heterogeneity on defense methods deployed by the server in FL.*

In this work, we propose a novel defense method designed to defend against backdoor attacks in FL, named **AlignIns** (Direction **Align**ment **Ins**pection), which examines local model updates for directional alignment at different granularity levels to identify malicious updates. Specifically, after receiving all model updates from clients, AlignIns evaluates each update by (1) inspecting temporal directional alignment with the global model of the latest round with *Cosine similarity* and (2) assessing more fine-grained sign alignment with the principal sign across all updates with a novel metric *sign alignment ratio*. Particularly, when calculating the sign alignment ratio, AlignIns focuses on the signs of important parameters in each update to accurately capture alignment information. Using these two directional metrics, AlignIns performs anomaly detection with the robust $\mathrm{MZ\_score}$ which requires minimal hyperparameters to filter updates with unusual directional patterns out. Finally, AlignIns clips the remaining updates to mitigate the impact of updates with abnormally large magnitudes. We also provide a theoretical analysis of AlignIns' robustness and its propagation error in FL. The main contributions of this work are three folds:

- We present a novel defense method, AlignIns, to defend against backdoor attacks in FL. ***To the best of our knowledge, AlignIns is the first defense method in FL that analyzes the directional patterns of local model updates at different levels of granularity.*** AlignIns is fully compatible with existing FL frameworks.
- ***To the best of our knowledge, we provide the first theoretical robustness analysis for a filtering-based defense method against backdoor attacks under non-IID data in FL.*** Moreover, we prove that the propagation error of AlignIns is bounded during the training of FL.
- We empirically evaluate the effectiveness of AlignIns through extensive experiments on both IID and non-IID datasets against various state-of-the-art (SOTA) backdoor attacks. Compared to existing SOTA defense methods, AlignIns exhibits superior robustness.

2. Background and Related Works

Federated Learning. In a typical FL system, a central server controls a set of n clients to train a global model $\theta \in \mathbb{R}^d$ collaboratively. The objective of FL is to solve the following optimization problem: $\min_\theta (1/n) \sum_{i=1}^n \mathcal{L}_i(\theta; \mathcal{D}_i)$, where $\mathcal{L}_i(\cdot)$ denotes the learning objective specific to client i and $\mathcal{D}_i$ denotes the local dataset for client i. The commonly used method to solve this problem iteratively is FedAvg [33]. In detail, at round t of FedAvg, each client $i \in [n]$ downloads the current global model θ^t, updates it by optimizing its local objective, resulting in θ_i^t, and transmits its model update $\Delta_i^t = \theta_i^t - \theta^t$ to the server. The server then refines the global model by averaging these updates as follows: $\theta^{t+1} = \theta^t + (1/n) \sum_{i=1}^n \Delta_i^t$. This process continues until the global model reaches convergence.

Backdoor attacks in FL. Empirical evidence has shown that FL is vulnerable to backdoor attacks [4, 9, 14, 23, 37, 46, 48, 52, 54] due to its lack of access to local training data [4]. For instance, *Projected Gradient Descent* (PGD) attack [46] periodically projects the local model onto a small sphere centered around the global model from the previous training round, with a predefined radius. *Distributed Backdoor Attack* (DBA) [48] decomposes the centralized trigger into several smaller, distributed local triggers. Each poisoned client uses one of these local triggers, but during testing, the adversary injects the full trigger into the test samples. Recently, research has focused on trigger-optimization backdoor attacks [1, 9, 29, 37, 52], which aim to search optimized triggers to enhance the effectiveness and stealthiness.

Defending against backdoor attacks in FL. Generally, based on how defense methods mitigate the impact of malicious updates, existing defense methods can be categorized into *filtering-based methods* [6, 7, 19, 22, 36, 42, 50, 51] and *influence-reduction methods* [11, 20, 38, 40, 41].

1) influence-reduction methods aim to integrate all model updates but employ strategies to reduce the impact of malicious updates. For instance, *RFA* [40] is proposed to use the geometric median of local models as the aggregation result, under the assumption that malicious models significantly deviate from benign models. *Foolsgold* [11] assumes that the malicious updates are consistent with each other. It assigns aggregation weights to model updates based on the maximum Cosine similarity between the last layers of pairwise model updates. A higher Cosine similarity value indicates a higher probability that the updates are malicious, leading to smaller aggregation weights being assigned. The effectiveness of influence-reduction methods is inherently limited because they cannot eliminate the impact of malicious activity, leading to a significant risk of compromise.

2) Filtering-based methods aim to detect and remove malicious local model updates before aggregation thus attempting to achieve the highest robustness. For example, *Multi-Krum* [6] selects the multiply reliable local model updates for aggregation by identifying the one with the smallest sum of squared Euclidean distances to all other updates. *Multi-Metrics* [19] explores the combination of Manhattan distance, Euclidean distance, and Cosine similarity for each update to collaboratively filter out outliers. However, due to the dual objectives of backdoor attacks—that is, maintaining accuracy on the main task while maximizing accuracy on the backdoor task—malicious updates must mimic benign model updates, important weights for the main task

Algorithm 1: AlignIns

Input: Set of n local model updates $\{\Delta_i^t\}_{i=1}^n$ where m of them are malicious, global model θ^t, TDA radius λ_c, MPSA radius λ_s, extraction parameter k

Output: Aggregated model update $\widetilde{\Delta}$

Initialize benign set $\mathcal{S} \leftarrow \emptyset$
$\omega \leftarrow \{\mathrm{TDA}(\Delta_i^t, \theta)\}_{i=1}^n$ ◁ by Equation (1)
$p \leftarrow \mathrm{sgn}(\sum_{i=1}^n \mathrm{sgn}(\Delta_i^t))$
$\rho \leftarrow \{\mathrm{MPSA}(\Delta_i^t, p, k)\}_{i=1}^n$ ◁ by Equation (2)
for $i \in [n]$ **do**
  $\lambda_{i,c} \leftarrow \mathrm{MZ\_score}(\omega_i, \omega)$ ◁ by Equation (3)
  $\lambda_{i,s} \leftarrow \mathrm{MZ\_score}(\rho_i, \rho)$ ◁ by Equation (3)
  if $|\lambda_{i,c}| \leq \lambda_c$ *and* $|\lambda_{i,s}| \leq \lambda_s$ **then**
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$
  end
end
$c \leftarrow \mathrm{med}(\{\|\Delta_i^t\|\}_{i \in \mathcal{S}})$
$\widetilde{\Delta} \leftarrow (1/|\mathcal{S}|) \sum_{i \in \mathcal{S}} (\Delta_i^t \cdot \min\{1, c/\|\Delta_i^t\|\})$
return $\widetilde{\Delta}$

typically have large values and can dominate the magnitude of a model update. As a result, magnitude-based detection methods become ineffective against backdoor attacks. Additionally, methods that rely solely on Cosine similarity also show limited effectiveness since they capture general directional alignment and overlook finer-grained information.

3. Our Solution: AlignIns

Our method, AlignIns, detailed in Algorithm 1, mitigates the impact of malicious updates through a two-step process. First, *direction alignment inspection* is applied to examine each local model update comprehensively in terms of direction. Second, *post-filtering model clipping* is used to further enhance the robustness of AlignIns on defending potential magnitude-based attack methods before final aggregation.

Direction alignment inspection. Existing defense methods against backdoor attacks in FL primarily focus on examining the magnitude (*e.g.*, Manhattan distance and Euclidean distance) and the overall direction (*e.g.*, Cosine similarity) of model updates. However, backdoor attacks are designed to maintain the main task accuracy, making the magnitude difference between malicious and benign updates nearly indistinguishable. Additionally, advanced attacks such as PGD [46] and Lie [5] attacks are specifically crafted to bypass magnitude-based defenses. Therefore, AlignIns focuses on direction-based analysis to identify suspect updates, using two processes described below.

1) Temporal direction alignment checking: Since malicious clients need to maintain both the main task and the backdoor task, the optimization direction of a malicious local model tends to deviate from that of benign models. AlignIns leverages this deviation and performs a Temporal Direction Alignment (TDA) checking, which calculates the Cosine similarity between a local update and the latest global model (line 2 in Algorithm 1) to assess the general alignment level of each local update. Formally, the TDA value ω_i of a local model update Δ_i^t is calculated as

$$\omega_i := \langle \Delta_i^t, \theta^t \rangle / (\|\Delta_i^t\| \|\theta^t\|). \tag{1}$$

We use local model updates rather than local models because our goal is to measure how closely each client's updates align with the direction of the global model. Local model updates specifically capture these incremental adjustments. Notably, malicious clients tend to exhibit similar TDA values, which differ from those of benign clients, creating an opportunity for detection. It is important to note that while the magnitude of model updates typically decreases as the global model converges, the TDA value does not follow the same trend. Consequently, magnitude-based anomaly detection becomes progressively less effective throughout training due to the decreasing magnitude. In contrast, the variability in TDA values continues to be useful for identifying malicious behavior.

2) Masked principal sign alignment checking: In backdoor attacks where the manipulations are stealthy, subtle malicious directional information can easily blend into the parameters of models with large magnitude, especially for models with large dimensions, which makes the TDA less useful under strong backdoor attacks since the TDA captures the overall directional information. Therefore, in addition to the TDA, we look into the signs of parameters to provide a finer-grained directional assessment of local model updates. The signs of a vector represent its coordinate-wise direction. In the context of backdoor attacks, the distributions of the signs of malicious model updates differ from those of benign updates. This is particularly significant when the model is close to convergence, at which point the magnitude of model updates becomes very small, making large manipulations on magnitudes impractical. Therefore, manipulation of the direction, or the signs of parameters, can emerge as a more significant and effective strategy. Several works also utilize the signs of models for enhancing backdoor robustness. For example, RLR [38] assigns an opposite global learning rate to a coordinate of the averaged model update if the signs on this coordinate do not consistently align with the majority across all updates. SignGuard [49] calculates the proportions of positive, zero, and negative signs for each model update as the input of a clustering algorithm to identify malicious model updates. However, these methods utilize the signs of all parameters in the model update, regardless of their significance. Consequently, the performance of sign-based metrics can be significantly impacted by those many unimportant parameters,

especially for large DNN models, leading to an inaccurate representation of the model update's direction.

To this end, AlignIns utilizes a Masked Principle Sign Alignment (MPSA) checking to inspect the sign alignment degree between the important parameters of each local update and a well-designed principle sign of all local updates. Specifically, to construct the principle sign over local updates, for each coordinate of local updates, we take the majority of the signs across all model updates as the principal sign of this coordinate, which can be mathematically formulated as $p := \operatorname{sgn}\left(\sum_{i=1}^{n} \operatorname{sgn}(\Delta_i^t)\right)$, where $p \in \mathbb{R}^d$ represents the vector of principal signs and $\operatorname{sgn}(\cdot)$ is the function to take the signs of a vector. Note that the principal sign represents sign-voting results for each coordinate, making it stand for the major direction/dynamic for each coordinate. With this principle sign over local updates, we inspect the alignment of the signs of important parameters of each model update with it. More specifically, we use a Top-k indicator defined as follows to identify the k most important parameters that have the largest absolute values in a vector.

Definition 1 (Top-k Indicator $\operatorname{Top_k}(\cdot)$)**.** *For a vector $x \in \mathbb{R}^d$ and a masking parameter k, where $1 \leq k \leq d$, the Top-k indicator $\operatorname{Top_k}(\cdot)$: $\mathbb{R}^d \to \mathbb{R}^d$ is defined as $[\operatorname{Top_k}(x)]_j = 1$ if $[x]_j \in \xi$ and $[\operatorname{Top_k}(x)]_j = 0$ otherwise, where $\xi = \{|x_{\pi(1)}|, |x_{\pi(2)}|, \ldots, |x_{\pi(k)}|\}$, here π is a permutation of $[d]$ such that $|x_{\pi(i)}| \geq |x_{\pi(i+1)}|$ for all $1 \leq i < d$.*

The Top-k indicator $\operatorname{Top_k}(\cdot)$ takes each local model update as input and outputs a mask vector in which each element is either 1 or 0 with the same size as the input. To quantify the alignment in sign distributions of each local model update and the principle sign, we define the Sign Alignment Ratio (SAR) as follows.

Definition 2 (Sign Alignment Ratio)**.** *For vectors $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^d$, the sign alignment ratio ρ of x to y is defined as $\rho := 1 - \|\operatorname{sgn}(x) - \operatorname{sgn}(y)\|_0/d$ where $\|\cdot\|_0$ is L_0-norm.*

Here, $\rho \in [0, 1]$ and a larger ρ indicate a higher degree of alignment between the signs of x and y. Combining $\operatorname{Top_k}(\cdot)$ and SAR, we have the MPSA value ρ_i for local update Δ_i^t formulated as follows:

$$\rho_i := 1 - \left\| \left(\operatorname{sgn}(\Delta_i^t) - p\right) \odot \operatorname{Top}_k(\Delta_i^t) \right\|_0 / k, \quad (2)$$

where $\odot$ is the Hadamard product, $\operatorname{sgn}(\Delta_i^t) - p$ computes a sign difference vector, capturing the difference between the sign of Δ_i^t and the principal reference sign p. Since MPSA checking focuses on the important parameters, this difference vector is element-wise multiplied with the Top-k mask derived from Δ_i^t, effectively setting unimportant coordinates to zero. The L_0-norm is then applied to count the not-aligned elements and with the masking parameter k to ultimately determine the SAR. MPSA checking effectively reveals malicious local updates by combining both magnitude and directional information from model updates, allowing for clear differentiation between malicious and benign updates. AlignIns calculates the MPSA value for each update with the principal sign iteratively (line 3–4) and forward them to the following anomaly detection process.

3) Efficient anomaly detection with MZ_score*:* W apply robust filtering to remove updates with abnormal TDA and MPSA values. Specifically, we use the robust standardization metric named the *Median-based Z-score* (MZ_score) [50, 51], detailed in Definition 3, which is a variant of the traditional *Z-score* standardization metric.

Definition 3 (MZ_score)**.** *For a set of values $X := \{x_1, \ldots, x_n\}$ with median $\operatorname{med}(X)$ and standard deviation σ, the* MZ_score *λ_i of any $x_i \in X$ is defined as*

$$\lambda_i := (x_i - \operatorname{med}(X))/\sigma. \quad (3)$$

MZ_score calculates the number of standard deviations an element is from the median, which may be either positive or negative. In AlignIns, the MZ_scores for TDA and MPSA values are computed for each local update (line 6–7). Those with high absolute MZ_scores (*i.e.*, outliers) are excluded using two predetermined filtering radii: λ_c for TDA and λ_s for MPSA (line 8–9). The use of the MZ_score allows for the adaptation to the varying range of TDA and MPSA values during training, requiring only minimal hyper-parameters. Additionally, by configuring λ_c and λ_s, we can manage the trade-off between the robustness and main task accuracy of AlignIns. For example, when robustness is the primary concern in the FL, choosing small λ_c and λ_s values is essential to attain the highest robustness.

Post-filtering model clipping. After filtering, the remaining clients, considered benign, are included in the set $\mathcal{S}$ (line 9) and contribute to the model averaging process. However, since our filtering primarily focuses on the direction of model updates (although MPSA does consider magnitude when using the Top-k indicator), there is a risk that it might overlook updates with large magnitudes, such as those updates generated by Scaling attack [4]. To this end, AlignIns re-scales model updates in $\mathcal{S}$ by using the median of the L_2-norms of these updates as a clipping threshold and aggregates the clipped model updates as the global model update $\widetilde{\Delta}$ (line 12–13). It is worth noting that performing clipping before filtering does not affect the filtering results. However, clipping after filtering enhances robustness, as the clipping threshold is more likely determined by benign updates. We discuss the computational cost of AlignIns and compare it with other baselines in Appendix Section 12.

4. Robustness and Propagation Error Analysis

In this section, we conduct a theoretical analysis of the robustness of AlignIns, as well as its propagation error in FL.

Before presenting our theoretical results, we make the following assumptions. Note that Assumption 1–2 are commonly used in the theoretical analysis of distributed learning systems [18, 34, 49]. Assumption 3 states a standard measure of inter-client heterogeneity in FL [2, 8, 21]. This heterogeneity complicates the problem of FL with backdoor adversaries, as it may cause the server to confuse malicious updates with flawed model updates from benign clients holding outlier data points [2].

Assumption 1 (μ-smoothness [34])**.** *Each local objective function $\mathcal{L}_i$ for benign client $i \in \mathcal{B}$ is μ-Lipschitz smooth with $\mu > 0$,* i.e.*, for any $x, y \in \mathbb{R}^d$, $\|\nabla\mathcal{L}_i(x) - \nabla\mathcal{L}_i(y)\| \leq \mu \|x - y\|, \forall i \in \mathcal{B}$, which further gives: $\mathcal{L}_i(x) - \mathcal{L}_i(y) \leq \nabla\mathcal{L}_i(x)^T(y - x) + (\mu/2)\|x - y\|^2, \forall i \in \mathcal{B}$.*

Assumption 2 (Unbiased gradient and bounded variance)**.** *The stochastic gradient at each benign client is an unbiased estimator of the local gradient,* i.e.*, $\mathbb{E}[g_i(x)] = \nabla\mathcal{L}_i(x)$ and has bounded variance,* i.e.*, for any $x \in \mathbb{R}^d$, $\mathbb{E}\|g_i(x) - \nabla\mathcal{L}_i(x))\|^2 \leq \nu_i^2, \forall i \in \mathcal{B}$, where the expectation is over the local mini-batches. We also denote $\bar{\nu} := (1/|\mathcal{B}|)\sum_{i\in\mathcal{B}} \nu_i^2$ for convenience.*

Assumption 3 (Bounded heterogeneity)**.** *There exist a real value $\bar{\zeta}$ such that for any $x \in \mathbb{R}^d$, $(1/|\mathcal{B}|)\sum_{i\in\mathcal{B}} \|\nabla\mathcal{L}_i(x) - \nabla\mathcal{L}_\mathcal{B}(x)\|^2 \leq \bar{\zeta}$, where the $\nabla\mathcal{L}_\mathcal{B}(x) := (1/|\mathcal{B}|)\sum_{i\in\mathcal{B}} \mathcal{L}_i(x)$.*

Note that these assumptions apply to benign clients only since malicious clients do not need to follow the prescribed local training protocol of FL.

4.1. Robustness Analysis of AlignIns

To theoretically evaluate the efficacy of a filtering-based defense method like AlignIns, we introduce the concept of κ-robust filtering [50] as defined in Definition 4. Note that Definition 4 is similar to (f, κ)-*robustness* defined in [2, 3], $(\delta_{\max}, c)$-*ARAgg* defined in [13, 21, 31], and (f, λ)-*resilient averaging* defined in [10]. Our robustness definition adopts a constant upper bound and focuses on quantifying the distance between the output of a filtering-based defense method and the average of all benign updates, which represents the optimal output of such a rule.

Definition 4 (κ-robust filtering [50])**.** *A filtering-based aggregation rule $F\colon \mathbb{R}^{d\times n} \to \mathbb{R}^d$ is called κ-robust if for any vectors $\{x_1, \ldots, x_n\} \in \mathbb{R}^d$ and a benign set $\mathcal{B} \subseteq [n]$ of size $n - m$, the output $\hat{x} := F(x_1, \ldots, x_n)$ satisfies $\|\hat{x} - \bar{x}_\mathcal{B}\|^2 \leq \kappa$, where $\bar{x}_\mathcal{B} := (1/|\mathcal{B}|)\sum_{i\in\mathcal{B}} x_i$, and $\kappa \geq 0$ refers to the robustness coefficient of F.*

Remark 1. *The κ-robust filtering guarantees that the error of a filtering-based aggregation rule in estimating the average of the benign inputs is upper-bounded by a constant κ. This measure provides a quantitative way to assess the robustness of the filtering-based aggregation rule. A smaller κ indicates a smaller discrepancy between the empirical output and the optimal output of F. If F identifies and removes all malicious inputs and keeps all benign inputs, we have $\kappa = 0$, achieving the highest level of robustness.*

Based on Definition 4, we prove that the proposed AlignIns, when applied to n input models, of which m are malicious, satisfies κ-robust filtering with $\kappa = O(1 + m/(n - 2m))$, as stated in Lemma 1.

Lemma 1 (κ-robustness of AlignIns)**.** *Under Assumption 2–3, assume $n > 1$, $0 \leq m < n/(3+\epsilon)$ with a positive constant ϵ, AlignIns satisfies κ-robust filtering with*

$$\begin{aligned}\kappa &= (1 + m/(n - 2m))\left((2/\epsilon + 1)\left(2\bar{\nu} + \bar{\zeta}\right) + 8c^2\right)\\ &= O\left(1 + m/(n - 2m)\right),\end{aligned}$$

if the local learning rate satisfies $\eta \leq 1/2\tau$ and there exist two sufficiently large filtering radii such that $|\mathcal{S}| \geq n - 2m$. Here, $\bar{\nu}$ and $\bar{\zeta}$ represent the gradient variance and local divergence, respectively; c is the clipping threshold.

Proof. The proof is given in Appendix Section 13.2. □

Remark 2. *The condition on $\mathcal{S}$ highlights the importance of selecting appropriate filtering radii. These radii cannot be zero or too small; otherwise, only the median or a few model updates will be averaged to update the global model. This can lead to a performance drop due to the lack of model updates. Moreover, the model clipping threshold c can effectively control the magnitude of potential malicious updates in the selection set, thus preventing κ from exploding due to updates with large magnitudes. Indeed, in the literature, model clipping has demonstrated its effectiveness in mitigating the impact of malicious model updates [39, 49, 53]. In addition, the result also shows the importance of reducing the gradient variance of stochastic gradient and local heterogeneity to enhance robustness performance. Our work is orthogonal to existing variance or divergence reduction methods [13, 30] and can be combined with them to further improve the robustness. We argue that AlignIns enjoys comparable robustness with several classical defense methods, for example, non-filtering-based method RFA [40] ($O(1 + m/(n - 2m))^2$), and filtering-based method Krum [6] ($O(1 + m/(n - 2m))$)[1].*

4.2. Propagation Error of AlignIns in FL

Based on the κ-robustness of AlignIns, we analyze its *propagation error* during training. Specifically, let θ denote the

[1]Results of RFA and Krum are taken from [2]. Note that the definition of κ in [2] is different from ours, but the difference part can be reduced to a constant bound. Therefore, we can safely incorporate these results into our discussion without losing generality.

model trained with AlignIns under backdoor attacks, where m of the n clients are malicious, and let θ^* denote a model trained exclusively with benign clients using FedAvg. Starting from the same initial model θ^0, we aim to measure the difference between these two models after T rounds of training, defined as $\|\theta^T - \theta^{T,*}\|$, referred to as the propagation error [39]. Let $\theta^{t,+}$ represent the output of AlignIns at the t-th round. If the highest level of robustness is not achieved at round t, the error $\|\theta^t - \theta^{t,+}\|$ will propagate to the next round, resulting in a shifted starting point for local SGD at round $t+1$. This discrepancy will gradually widen the gap between θ and θ^*. Our analysis captures this robustness error at each round and examines its cumulative effect after T rounds. In Lemma 2, we show that, assuming Assumption 1–3 hold, the propagation error of AlignIns remains bounded.

Lemma 2 (Bounded Propagation Error)**.** *Let Assumption 1–3 hold. If the local learning rate $\eta \leq 1/2\tau$, the propagation error of AlignIns is bounded as*

$$\|\theta^T - \theta^{T,*}\| \leq \phi(T)(2+3\mu^2)^{\phi(T)}(\kappa + 2\bar{\nu}),$$

under backdoor attacks where m out of n clients are malicious. Here, κ is given in Lemma 1, $\phi(T) = \sum_{t=1}^{T}(\alpha^t)^2$ is the cumulative global learning rate, and α^t is a global learning rate scheduler, possibly static.

Proof. The detailed proof is in Appendix Section 13.3. □

Remark 3. *When $T \to \infty$, $\phi(T)$ converges to a constant for learning rate schedulers like exponential decay, which implies a constant bounded on propagation error. The result shows that besides the robustness error bounded by κ, the error of local gradient estimation, which is bounded by $\bar{\nu}$, in local SGD also propagates during the training, increasing the overall propagation error. This is because at any round t, if the benign starting point for local training is the same,* i.e.*, $\theta^t = \theta^{t,*}$, then the local gradients/model updates on θ^t and $\theta^{t,*}$ will be identical for benign clients. Therefore, the gap between the updated global models θ^{t+1} and $\theta^{t+1,*}$ solely depends on the robustness error (*i.e.*, the effectiveness of AlignIns in filtering out malicious updates). However, if $\theta^t \neq \theta^{t,*}$, which means θ^t is not benign and has been poisoned in previous rounds, the local gradients/model updates on θ^t and $\theta^{t,*}$ will differ for benign clients, resulting in an error bounded by the gradient variance, even if AlignIns successfully filters out all malicious updates. Hence, to further reduce the propagation error, AlignIns can be combined with variance-reduction methods like [13, 30], which is orthogonal to AlignIns.*

5. Experimental Settings

Datasets and System Settings: In our experiments, we primarily conduct evaluation on CIFAR-10 [24] and CIFAR-100 [24] datasets. Additionally, we present the superior performance of AlignIns on other benchmark datasets (MNIST [25], FMNIST [47], Sentiment140 [12], and Tiny-ImageNet) in Appendix Section 10.5–10.6. We simulate a cross-silo FL with 20 clients for all datasets. *Additionally, we also present the superior performance of AlignIns on a cross-device FL system with 100 clients and client sampling.* We consider both IID and non-IID settings. For IID settings, we distribute the training data evenly to clients. For non-IID settings, we follow [17, 19, 20] to use *Dirichlet distribution* $Dir(\beta)$ to skew the local data distribution with a default non-IID degree $\beta = 0.5$.

Learning Settings: We use SGD as the local solver, with the initial learning rates set as $\alpha = 1.0$ and $\eta = 0.1$, and the local training epoch is set as 2. The training round is set as $T = 100$ for CIFAR-100 and $T = 150$ for CIFAR-10. For AlignIns, the default filtering radii are set as $\lambda_c = 1.0$ and $\lambda_s = 1.0$. We provide the study of the impact of filtering radii in Appendix Section 11. The default masking parameter is set as $k = 0.3 \times d$ so that the Top-30% of model parameters are used for the MPSA checking.

Evaluated Attack Methods: We consider 5 SOTA backdoor attacks, including *Badnet* [14], *DBA* [48], *Scaling* [4], *PGD* [46], and *Neurotoxin* [54]. We provide the detailed attack model and settings for attack methods in Appendix Section 8.1–8.2. We present the results of AlignIns under the strong *trigger-optimization attack* [9] in Appendix Section 10.2. Moreover, we study the potential *adaptive attacks* tailored to AlignIns and *untargeted attacks* [49] in Appendix Section 10.3–10.4, although these are beyond the scope of this work. To achieve effective backdoors (achieving a BA over 60% [22]), malicious clients poison $r = 50\%$ of their local data, where r is the *data poisoning ratio*. The *attack ratio* is set to 20% by default, which means 20% of the clients in the system are malicious. Experiments of AlignIns on defending backdoor attacks with various attack ratios are given in Appendix Section 10.7.

Evaluated Defense Methods: We present the detailed defense model in Appendix Section 9. We comprehensively compare AlignIns with the non-robust baseline FedAvg and six existing SOTA defense methods, including *RLR* [38], *RFA* [40], *Multi-Krum (MKrum)* [6], *Foolsgold* [11], *Multi-Metric (MM)* [19], and *Lockdown* [20]. Additionally, we compare our approach with an ideally perfect filtering-based robust aggregation, *FedAvg**, which is assumed to perfectly identify and remove all malicious updates and average all the benign updates to update the global model.

Evaluation Metrics: We use three metrics to evaluate the performance of defense methods, including **main task accuracy (MA)**, which measures the percentage of clean test samples that are accurately classified to their ground truth labels by the global model; **backdoor attack accuracy (BA)**, which measures the percentage of triggered

Table 1. The clean MA, BA, and RA results of baselines and AlignIns on IID CIFAR-10 and CIFAR-100 datasets. Results are shown in %.

Dataset (Model)	Methods	Clean MA↑	Badnet				DBA				Neurotoxin				Avg. BA↓	Avg. RA↑
			BA↓		RA↑		BA↓		RA↑		BA↓		RA↑			
			r=0.3	r=0.5	r=0.3	r=0.5	r=0.3	r=0.5	r=0.3	r=0.5	r=0.3	r=0.5	r=0.3	r=0.5		
CIFAR-10 (ResNet9 [16])	FedAvg	89.47	51.56	67.61	45.79	31.24	56.21	70.42	40.62	27.92	44.89	79.40	50.41	19.60	61.68	35.93
	FedAvg*	89.47	2.06	2.06	85.60	85.60	2.06	2.06	85.60	85.60	2.06	2.06	85.60	85.60	2.06	85.60
	RLR	79.16	2.32	**2.00**	76.72	73.33	3.01	3.04	77.09	77.13	3.12	3.87	73.98	73.29	2.89	35.93
	RFA	87.73	70.67	90.24	27.74	9.26	47.67	66.97	47.29	30.14	81.27	96.13	17.11	3.69	75.49	22.54
	MKrum	87.02	81.10	97.47	18.11	2.51	2.17	4.33	83.89	79.10	65.28	89.18	31.81	10.01	56.59	37.57
	Foolsgold	89.49	69.14	68.84	29.64	30.10	51.18	60.73	44.83	36.08	2.91	2.82	85.27	84.76	42.60	51.78
	MM	89.15	41.19	93.88	53.88	6.01	52.24	51.30	43.54	45.08	43.92	83.92	51.12	15.11	61.08	35.79
	Lockdown	88.56	6.31	10.82	81.88	79.50	11.63	6.03	78.82	75.77	3.40	3.27	82.73	83.14	6.91	80.31
	AlignIns	88.64	**1.91**	2.21	**86.03**	**85.57**	**2.13**	**2.14**	**85.77**	**85.88**	**2.66**	**2.20**	**85.46**	**85.31**	**2.21**	**85.67**
CIFAR-100 (VGG9 [44])	FedAvg	64.29	99.20	99.54	0.68	0.35	99.25	99.36	0.64	0.54	94.41	93.36	4.36	5.28	97.52	1.98
	FedAvg*	64.29	0.62	0.62	53.03	53.03	0.62	0.62	53.03	53.03	0.62	0.62	53.03	53.03	0.62	53.03
	RLR	44.34	96.57	99.85	1.81	0.12	24.41	94.08	24.97	3.22	**0.04**	**0.00**	29.07	29.73	52.49	14.82
	RFA	53.92	4.32	1.45	37.60	39.88	2.15	0.78	39.73	41.51	99.74	89.59	0.21	6.59	33.01	27.59
	MKrum	51.28	1.33	1.54	38.13	38.49	1.36	1.54	37.85	37.91	99.82	99.87	0.12	0.10	36.21	25.49
	Foolsgold	64.13	99.02	99.30	0.83	0.57	99.15	99.39	0.74	0.51	21.79	6.21	42.06	46.40	70.81	15.19
	MM	63.26	99.51	99.87	0.37	0.11	99.53	99.70	0.35	0.19	98.48	98.97	1.32	0.83	99.34	0.53
	Lockdown	62.88	55.21	24.14	28.45	43.06	34.37	49.02	34.06	27.93	0.85	0.67	42.66	47.04	27.38	37.20
	AlignIns	63.45	**0.79**	**0.71**	**50.45**	**51.53**	**0.45**	**0.57**	**50.81**	**52.08**	0.49	0.53	**51.11**	**50.66**	**0.59**	**51.11**

samples that are misclassified to the target label by the global model; and **robustness accuracy (RA)**, which measures the percentage of triggered samples that are accurately classified to their ground-truth labels by the global model, despite the presence of the trigger. A good defense method should achieve high MA and RA and low BA.

6. Experimental Results

Main results in IID setting. In Table 1, we report the performance of various defense methods under no attack (denoted by "Clean"), Badnet, DBA, and Neurotoxin attacks for IID CIFAR-10 and CIFAR-100. The best results are highlighted in **bold font**, and the second best results are underlined. Overall, ***AlignIns demonstrates superior performance compared with other baselines as it achieves the best average BA and RA over three attack methods.*** Specifically, for CIFAR-10, while RLR offers a satisfactory degree of robustness (an average BA of 2.89%), it suffers from a notable decline in RA, with an average reduction of 49.74% in comparison to AlignIns. This drop results from RLR's strategy of flipping the global learning rate for parameters in the aggregated model update that are inconsistent with the majority's sign, consequently resulting in the loss of benign local parameters. AlignIns, however, demonstrates outstanding performance with consistently low BA and high RA, ranking first or second among its counterparts. Notably, compared to the second-best results, AlignIns achieves an average improvement of +0.68% in BA and +5.36% in RA. Similarly, superior results are observed in CIFAR-100 experiments, where AlignIns significantly outperforms other methods in both BA and RA. These results underscore AlignIns' effectiveness as a promising defense method for protecting FL from various backdoor attacks, significantly enhancing the trustworthiness of FL systems.

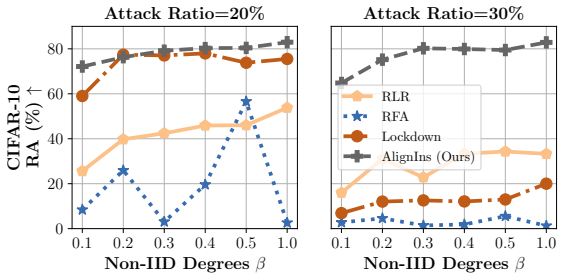

Figure 1. RA of AlignIns under various non-IID degrees, compared with Lockdown, RFA, and RLR under Neurotoxin .

Effectiveness under various Non-IID degrees. We examine the defense performance of AlignIns across various degrees of non-IIDness, a factor that significantly complicates backdoor defense. Figure 1 presents the RA of AlignIns under different non-IID conditions on the CIFAR-10 dataset, compared with Lockdown, RFA, and RLR. The experiments were conducted using the Neurotoxin attack, with both a default attack ratio of 20% and a higher attack ratio of 30%. The Dirichlet parameter β varies from 0.1 to 1.0, where a smaller β suggests a more intense non-IIDness. We observe that only AlignIns consistently attains robustness against strong Neurotoxin attacks with a varying β. Specifically, as β increases, the RA of AlignIns, Lockdown, and RLR increases correspondingly. However,

Table 2. Performance of different methods in cross-device FL settings on IID and non-IID CIFAR-10 datasets under Badnet attack.

Method	CIFAR-10 (IID)			CIFAR-10 (Non-IID)			Avg. RA↑
	MA↑	BA↓	RA↑	MA↑	BA↓	RA↑	
Foolsgold	82.99	99.99	0.01	67.97	99.99	0.00	0.01
Lockdown	83.52	99.99	0.00	73.91	99.92	0.06	0.00
RLR	56.81	4.67	55.38	41.56	14.12	38.17	46.78
AlignIns	**85.01**	**0.92**	**82.74**	**79.51**	**1.90**	**75.81**	**79.28**

AlignIns outperforms them with a consistently higher RA. When the attack ratio rises to 30%, RLR, RFA, and Lockdown fail to provide satisfactory robustness. However, our method AlignIns still demonstrates its robustness under various non-IIDness, even in an extremely non-IID case when $\beta = 0.1$. AlignIns is designed to examine the alignment of model updates on important parameters only, hence, it mitigates the challenge of identifying malicious model updates in non-IID settings where updates are heterogeneous, thereby achieving superior performance in even extreme non-IID settings compared with existing methods. We also provide more comprehensive results of AlignIns and other baselines on non-IID datasets in Appendix Section 10.1.

Effectiveness of AlignIns in cross-device FL with client sampling. While most of our experiments focus on the cross-silo FL setting, evaluating the cross-device FL scenario is also essential given the large number of clients involved. For this purpose, we simulate a cross-device FL environment with 100 clients, where the server randomly selects 20 clients per round for training. We conduct experiments on IID and non-IID CIFAR-10 cases using Foolsgold, Lockdown, RLR, and AlignIns and summarize the MA, BA, and RA results in Table 2. The results show that both Foolsgold and Lockdown completely lose their effectiveness in both cases, achieving an average RA of nearly 0.00%. RLR achieves a moderate level of backdoor robustness but at the cost of main task accuracy, with an average MA of only 49.19%. In contrast, AlignIns performs robustly in the cross-device FL setting, achieving a significantly lower BA in both IID (0.92%) and non-IID (1.90%) cases compared with other methods. Furthermore, AlignIns achieves an average RA of 79.28%. These results highlight AlignIns's ability to maintain both accuracy and robustness in challenging cross-device FL scenarios, underscoring its adaptability and effectiveness in real-world applications.

Ablation study of AlignIns. As AlignIns consists of two alignment components (TDA and MPSA) to improve backdoor robustness, we conduct a detailed ablation study to investigate how each component functions. Experimental results on IID and non-IID CIFAR-10 datasets under Badnet attack are summarized in Table 3. ***(i) Component ablation.*** We observe that using MPSA or TDA alone in IID scenarios only slightly reduces robustness compared to AlignIns, as benign updates follow consistent patterns that enable effective detection by a single metric. In non-IID settings, however, where local updates diverge, neither MPSA nor TDA alone provides sufficient robustness. When combined, MPSA and TDA improve BA and RA from 94.07% and 5.79% to 47.04% and 45.30%, respectively, showing their complementary strengths. AlignIns further enhances robustness by integrating MPSA, TDA, and post-filtering model clipping, which normalizes benign update magnitudes and improves malicious update detection, yielding the highest average RA. ***(ii) Masking parameter k ablation.*** We try to involve more non-essential parameters in the MPSA checking by using the Top-50%/70% of parameters to calculate MPSA values. By doing so, the effectiveness of malicious identification is reduced. In contrast, when using the Top-30% of parameters, compared to the Top-50% case, BA and RA are improved by +30.89% and +25.34%, respectively. This demonstrates the effectiveness of focusing important parameters when calculating MPSA in improving the filtering accuracy, especially in non-IID cases. ***(iii) Variance reduction method further enhances robustness.*** Our theoretical results reveal the impact of variance reduction techniques on improving the robustness of AlignIns and reducing the propagation error of AlignIns in FL, we additionally test a variant of AlignIns named "AlignIns$^+$", in which local SGD with momentum is used to reduce the local gradient variance with momentum coefficient 0.1. AlignIns$^+$ achieves a slightly better performance than AlignIns, verifying our theoretical results.

Table 3. Performance of different components in AlignIns.

Configuration	CIFAR-10 (IID)			CIFAR-10 (non-IID)			Avg. RA↑
	MA↑	BA↓	RA↑	MA↑	BA↓	RA↑	
MPSA(30%)	88.55	2.88	85.02	80.65	94.07	5.79	45.41
TDA	**88.56**	3.82	83.88	83.86	77.58	21.31	52.60
MPSA(70%+TDA	88.14	2.18	85.77	83.84	61.83	31.86	58.82
MPSA(50%)+TDA	88.05	2.21	85.46	**84.12**	77.93	19.96	52.71
MPSA(30%)+TDA	88.14	**2.04**	**85.82**	83.65	47.04	45.30	65.56
AlignIns	88.05	2.44	85.27	82.88	**1.70**	**81.32**	**83.30**
AlignIns$^+$	88.48	2.14	85.74	83.31	1.11	82.13	83.94

7. Conclusion

We introduce a novel method AlignIns to defend against backdoor attacks in FL. AlignIns examines each model update's direction at different granularity levels, thus effectively identifying stealthy malicious model updates and filtering them out to avoid them participating in aggregation in FL to enhance robustness. We provide a theoretical analysis of AlignIns' robustness and its impact on propagation errors in FL. Extensive experiments demonstrate the effectiveness of AlignIns, with results showing that it outperforms SOTA defense methods against various advanced attacks.

Acknowledgment. This work was supported by the Nevada System of Higher Education under grant NSHE-24-37.

References

[1] Manaar Alam, Esha Sarkar, and Michail Maniatakos. Perdoor: Persistent non-uniform backdoors in federated learning using adversarial perturbations. *arXiv preprint arXiv:2205.13523*, 2022. 2

[2] Youssef Allouah, Sadegh Farhadkhani, Rachid Guerraoui, Nirupam Gupta, Rafaël Pinot, and John Stephan. Fixing by mixing: A recipe for optimal byzantine ml under heterogeneity. In *International Conference on Artificial Intelligence and Statistics*, pages 1232–1300. PMLR, 2023. 5

[3] Youssef Allouah, Rachid Guerraoui, Nirupam Gupta, Rafael Pinot, and Geovani Rizk. Robust distributed learning: Tight error bounds and breakdown point under data heterogeneity. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 5

[4] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, pages 2938–2948. PMLR, 2020. 1, 2, 4, 6, 12

[5] Gilad Baruch, Moran Baruch, and Yoav Goldberg. A little is enough: Circumventing defenses for distributed learning. *Advances in Neural Information Processing Systems*, 32, 2019. 3, 14

[6] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30, 2017. 1, 2, 5, 6, 12

[7] Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. Fltrust: Byzantine-robust federated learning via trust bootstrapping. *arXiv preprint arXiv:2012.13995*, 2020. 2, 12

[8] El Mahdi El-Mhamdi, Sadegh Farhadkhani, Rachid Guerraoui, Arsany Guirguis, Lê-Nguyên Hoang, and Sébastien Rouault. Collaborative learning in the jungle (decentralized, byzantine, heterogeneous, asynchronous and nonconvex learning). *Advances in Neural Information Processing Systems*, 34:25044–25057, 2021. 5

[9] Pei Fang and Jinghui Chen. On the vulnerability of backdoor defenses for federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11800–11808, 2023. 2, 6, 13

[10] Sadegh Farhadkhani, Rachid Guerraoui, Nirupam Gupta, Rafael Pinot, and John Stephan. Byzantine machine learning made easy by resilient averaging of momentums. In *International Conference on Machine Learning*, pages 6246–6283. PMLR, 2022. 5

[11] Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. The limitations of federated learning in sybil settings. In *23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*, pages 301–316, 2020. 1, 2, 6

[12] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12):2009, 2009. 6

[13] Eduard Gorbunov, Samuel Horváth, Peter Richtárik, and Gauthier Gidel. Variance reduction is an antidote to byzantines: Better rates, weaker assumptions and communication compression as a cherry on the top. *arXiv preprint arXiv:2206.00529*, 2022. 5, 6

[14] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017. 1, 2, 6, 12

[15] Rachid Guerraoui, Sébastien Rouault, et al. The hidden vulnerability of distributed learning in byzantium. In *International Conference on Machine Learning*, pages 3521–3530. PMLR, 2018. 1

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 7

[17] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019. 6

[18] Rui Hu, Yuanxiong Guo, and Yanmin Gong. Federated learning with sparsified model perturbation: Improving accuracy under client-level differential privacy. *IEEE Transactions on Mobile Computing*, 2023. 5

[19] Siquan Huang, Yijiang Li, Chong Chen, Leyu Shi, and Ying Gao. Multi-metrics adaptively identifies backdoors in federated learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4652–4662, 2023. 1, 2, 6

[20] Tiansheng Huang, Sihao Hu, Ka-Ho Chow, Fatih Ilhan, Selim Tekin, and Ling Liu. Lockdown: Backdoor defense for federated learning with isolated subspace training. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 2, 6

[21] Sai Praneeth Karimireddy, Lie He, and Martin Jaggi. Byzantine-robust learning on heterogeneous datasets via bucketing. In *International Conference on Learning Representations*, 2021. 5

[22] Torsten Krauß and Alexandra Dmitrienko. Mesas: Poisoning defense for federated learning resilient against adaptive attackers. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 1526–1540, 2023. 1, 2, 6

[23] Torsten Krauß, Jan König, Alexandra Dmitrienko, and Christian Kanzow. Automatic adversarial adaption for stealthy poisoning attacks in federated learning. In *To appear soon at the Network and Distributed System Security Symposium (NDSS)*, 2024. 2

[24] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. *University of Toronto*, 2009. 6

[25] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 6

[26] Han Liu, Zhiyuan Yu, Mingming Zha, XiaoFeng Wang, William Yeoh, Yevgeniy Vorobeychik, and Ning Zhang. When evil calls: Targeted adversarial voice over ip network. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 2009–2023, 2022. 1

[27] Yi Liu, Jiangtian Nie, Xuandi Li, Syed Hassan Ahmed, Wei Yang Bryan Lim, and Chunyan Miao. Federated learning in the sky: Aerial-ground air quality sensing framework with uav swarms. *IEEE Internet of Things Journal*, 8(12):9827–9837, 2020. 1

[28] Guodong Long, Yue Tan, Jing Jiang, and Chengqi Zhang. Federated learning for open banking. In *Federated Learning: Privacy and Incentive*, pages 240–254. Springer, 2020. 1

[29] Xiaoting Lyu, Yufei Han, Wei Wang, Jingkai Liu, Bin Wang, Jiqiang Liu, and Xiangliang Zhang. Poisoning with cerberus: Stealthy and colluded backdoor attack against federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9020–9028, 2023. 2

[30] Grigory Malinovsky, Kai Yi, and Peter Richtárik. Variance reduced proxskip: Algorithm, theory and application to federated learning. *Advances in Neural Information Processing Systems*, 35:15176–15189, 2022. 5, 6

[31] Grigory Malinovsky, Peter Richtárik, Samuel Horváth, and Eduard Gorbunov. Byzantine robustness and partial participation can be achieved simultaneously: Just clip gradient differences. *arXiv preprint arXiv:2311.14127*, 2023. 5

[32] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 1

[33] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 2

[34] Yurii Nesterov et al. *Lectures on convex optimization*. Springer, 2018. 5

[35] Dinh C Nguyen, Quoc-Viet Pham, Pubudu N Pathirana, Ming Ding, Aruna Seneviratne, Zihuai Lin, Octavia Dobre, and Won-Joo Hwang. Federated learning for smart healthcare: A survey. *ACM Computing Surveys (CSUR)*, 55(3): 1–37, 2022. 1

[36] Thien Duc Nguyen, Phillip Rieger, Roberta De Viti, Huili Chen, Björn B Brandenburg, Hossein Yalame, Helen Möllering, Hossein Fereidooni, Samuel Marchal, Markus Miettinen, et al. FLAME: Taming backdoors in federated learning. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1415–1432, 2022. 1, 2

[37] Thuy Dung Nguyen, Tuan A Nguyen, Anh Tran, Khoa D Doan, and Kok-Seng Wong. Iba: Towards irreversible backdoor attacks in federated learning. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[38] Mustafa Safa Ozdayi, Murat Kantarcioglu, and Yulia R Gel. Defending against backdoors in federated learning with robust learning rate. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9268–9276, 2021. 2, 3, 6

[39] Ashwinee Panda, Saeed Mahloujifar, Arjun Nitin Bhagoji, Supriyo Chakraborty, and Prateek Mittal. Sparsefed: Mitigating model poisoning attacks in federated learning with sparsification. In *International Conference on Artificial Intelligence and Statistics*, pages 7587–7624. PMLR, 2022. 5, 6

[40] Krishna Pillutla, Sham M Kakade, and Zaid Harchaoui. Robust aggregation for federated learning. *IEEE Transactions on Signal Processing*, 70:1142–1154, 2022. 2, 5, 6

[41] Phillip Rieger, Torsten Krauß, Markus Miettinen, Alexandra Dmitrienko, and Ahmad-Reza Sadeghi. Crowdguard: Federated backdoor detection in federated learning. *arXiv preprint arXiv:2210.07714*, 2022. 2

[42] Phillip Rieger, Thien Duc Nguyen, Markus Miettinen, and Ahmad-Reza Sadeghi. Deepsight: Mitigating backdoor attacks in federated learning through deep model inspection. *arXiv preprint arXiv:2201.00763*, 2022. 1, 2, 12

[43] Virat Shejwalkar and Amir Houmansadr. Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning. In *NDSS*, 2021. 12

[44] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[45] Zhiyi Tian, Lei Cui, Jie Liang, and Shui Yu. A comprehensive survey on poisoning attacks and countermeasures in machine learning. *ACM Computing Surveys*, 55(8):1–35, 2022. 1

[46] Hongyi Wang, Kartik Sreenivasan, Shashank Rajput, Harit Vishwakarma, Saurabh Agarwal, Jy-yong Sohn, Kangwook Lee, and Dimitris Papailiopoulos. Attack of the tails: Yes, you really can backdoor federated learning. *Advances in Neural Information Processing Systems*, 33:16070–16084, 2020. 1, 2, 3, 6, 12

[47] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 6

[48] Chulin Xie, Keli Huang, Pin-Yu Chen, and Bo Li. Dba: Distributed backdoor attacks against federated learning. In *International conference on learning representations*, 2019. 1, 2, 6, 12

[49] Jian Xu, Shao-Lun Huang, Linqi Song, and Tian Lan. Signguard: Byzantine-robust federated learning through collaborative malicious gradient filtering. *arXiv preprint arXiv:2109.05872*, 2021. 3, 5, 6, 14

[50] Jiahao Xu, Zikai Zhang, and Rui Hu. Achieving byzantine-resilient federated learning via layer-adaptive sparsified model aggregation. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pages 1508–1517, 2025. 2, 4, 5

[51] Jiahao Xu, Zikai Zhang, and Rui Hu. Identify backdoored model in federated learning via individual unlearning. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pages 7949–7958, 2025. 2, 4

[52] Hangfan Zhang, Jinyuan Jia, Jinghui Chen, Lu Lin, and Dinghao Wu. A3fl: Adversarially adaptive backdoor attacks to federated learning. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[53] Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. Why gradient clipping accelerates training: A theoretical justification for adaptivity. In *International Conference on Learning Representations*, 2019. 5

[54] Zhengming Zhang, Ashwinee Panda, Linyue Song, Yaoqing Yang, Michael Mahoney, Prateek Mittal, Ramchandran Kannan, and Joseph Gonzalez. Neurotoxin: Durable backdoors in federated learning. In *International Conference on Machine Learning*, pages 26429–26446. PMLR, 2022. 1, 2, 6, 12