RoseAgg: Robust Defense against Targeted Collusion Attacks in Federated Learning

He Yang, Wei Xi, *Member, IEEE,* Yuhao Shen, Canhui Wu, Jizhong Zhao, *Member, IEEE*

***Abstract*—Recent defense approaches against targeted model poisoning attacks aim to prevent specific prediction failures in federated learning (FL). However, these defenses remain susceptible to targeted collusion attacks, particularly under conditions of high proportions of malicious clients and attack density. To address these vulnerabilities, we propose RoseAgg, which dynamically identifies a plausible clean ingredient from local updates and leverages it to constrain the influence of poisoned updates. Firstly, RoseAgg recognizes and confines common characteristics found in poisoned updates, such as scaled-up magnitudes or similar directional contributions. Furthermore, RoseAgg dynamically extracts a plausible clean ingredient using a dimension-reduction method. This clean ingredient becomes the foundation for the server to bootstrap credit scores for each local update, ensuring the dominance of benign updates over poisoned ones. Ultimately, the server computes a weighted average of local updates based on credit scores, generating a global update for refining the global model. Comprehensive evaluations on four benchmark datasets showcase RoseAgg's effectiveness against seven advanced attacks. The code is available at *https://github.com/SleepedCat/RoseAgg*.**

***Index Terms*—Federated Learning, Targeted Model Poisoning, Collusion Attacks, Robust Defense.**

I. Introduction

FEDERATED learning (FL) is an emerging decentralized machine learning paradigm that facilitates model training on distributed data without the need to transfer raw data to a central server [37], [13], [5], [16]. Despite its potential benefits, recent studies [19], [26], [28], [21], [18], [20] highlight FL's vulnerability to targeted model poisoning attacks. In these attacks, the adversary manipulates local model updates from a subset of participating clients within the federation, inducing the global model to misclassify specific test data samples as the attacker-desired target class.

Recent studies have concentrated on mitigating targeted poisoning attacks in federated learning, and these proposed solutions can be broadly categorized into four main groups. The first category employs anomaly detection to identify and eliminate potentially poisoned model updates [1], [11], [18]. The second category focuses on developing robust federated learning protocols to ensure the integrity of the global model in the presence of adversaries [7], [35], [24]. The third category involves approaches rooted in differential privacy (DP) [23], [29], where local model updates are clipped, and random noise is introduced to mitigate the impact of potentially poisoned updates on the global model. The fourth category seeks to bootstrap trust for each local model update, thereby enhancing the overall robustness of federated learning [6]. Notably, this approach demonstrates resilience against a substantial number of malicious clients.

Despite advancements in these defenses, they still confront certain challenges. Firstly, approaches in the first category hinge on specific assumptions about attackers' strategies or the underlying data distributions of honest and adversarial data. Secondly, the second category's approaches demonstrate robustness against a limited number of malicious clients, rendering them susceptible to failure in the presence of a high proportion of malicious clients. Thirdly, while defense approaches, as presented in [23], [29], demonstrate effectiveness even in the presence of numerous attackers, they may falter as the attack density increases. Since these approaches necessitate injecting random noise into the model updates, the overall performance of the global model may be compromised. Finally, the defense outlined in [6] relies on a root dataset on the server, with a data distribution similar to the overall training data distribution. Consequently, if the root dataset significantly diverges from the overall training data, the efficacy of the defense is compromised, leading to inevitable failure.

In this paper, we introduce RoseAgg to address the aforementioned deficiencies. The foundational principle guiding RoseAgg stems from the collaborative dynamics among attackers with a shared malicious objective. This collaboration materializes in model updates that exhibit a consistent trait, specifically, similar directional contributions. Conversely, if attackers pursue distinct malicious objectives, this common characteristic diminishes. Building upon this insight, the following assumption is established: in scenarios where attackers share a common malicious objective, RoseAgg refrains from making assumptions about the number of attackers. However, if attackers have distinct malicious objectives, RoseAgg assumes the number of attackers does not surpass half of the total clients.

Several challenges persist in the implementation of RoseAgg. Firstly, addressing the issue of constraining the impact of potentially poisoned model updates, considering both directional and magnitude perspectives, remains a primary concern. Secondly, extracting a reliable and clean ingredient from local model updates proves challenging. This difficulty arises from the lack of access to a clean validation dataset that closely aligns with the data distribution of the overall training data, as highlighted in [39]. Consequently, the server lacks a trusted source for calculating a clean model update. Simultaneously, the server faces the dilemma of

This work is supported by NSFC Grant. NO. 61832008, 62176205. He Yang, Wei Xi, Yuhao Shen, Canhui Wu, and Jizhong Zhao are with the Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, China. (e-mail: sleepingcat@stu.xjtu.edu.cn, xiwei@xjtu.edu.cn, syhshe@stu.xjtu.edu.cn, wucanhui@stu.xjtu.edu.cn, zjz@xjtu.edu.cn).

Fig. 1. In the context of a shared malicious objective, images depicting cars with vertically striped walls in the background, as well as images with pixel patterns of the letters "F", "L", and "A" are erroneously classified as birds. Conversely, in situations where different malicious objectives exist, the aforementioned images are erroneously identified as birds, airplanes, automobiles, and frogs, respectively.

distinguishing between benign and malicious model updates, making the calculation of a clean ingredient solely based on local model updates a complex task. Thirdly, a critical aspect is determining appropriate credit scores for local model updates. In RoseAgg, all local model updates are retained without removal. Nevertheless, if the cumulative credit score of potentially malicious local model updates is substantial, it poses a risk of compromising the integrity of the global model.

To tackle the aforementioned challenges, RoseAgg introduces a multi-pronged strategy. Firstly, it incorporates an adaptive partial aggregating strategy that employs normalizing, adaptive clustering, and partial aggregating to constrain common characteristics observed in poisoned model updates, such as scaled-up magnitudes or similar directional contributions. Furthermore, RoseAgg integrates a dimension-reduction method to identify a plausible clean ingredient. This ingredient serves as the foundation for bootstrapping credit scores for local model updates on the server. Thirdly, RoseAgg assigns credit scores to local model updates based on their projection variances on the plausible clean ingredient, guaranteeing that the cumulative credit scores of benign local model updates exceed those of poisoned counterparts. Simultaneously, RoseAgg computes a global model update by averaging all local model updates weighted by the credit scores.

In summary, our contributions are outlined as follows.

- We introduce RoseAgg, a robust defense against targeted collusion attacks in the presence of high proportions of malicious clients and attack density.
- Theoretically, RoseAgg exhibits robustness against advanced targeted model poisoning attacks.
- Extensive evaluations on four benchmark datasets showcase RoseAgg's effectiveness against seven state-of-the-art attacks.

II. Related Work

A. Federated Learning

In a standard federated learning scenario, consider a set of n clients denoted as $C_1, C_2, \ldots, C_n$, where each client $C_i, i \in [1, 2, \cdots, n]$ possesses a private local dataset $\mathcal{D}_i$. The fundamental objective in federated learning is for these clients to collaboratively acquire a shared global model without the necessity of transferring their individual datasets to a central server [22], [13], [37]. The training process unfolds across multiple rounds. During each round, the server first selects a subset of clients and sends the current global model to them. Then, each selected client C_i trains the model locally using its private data $\mathcal{D}_i$, and only the updated model parameters are sent to the central server. Finally, these updated parameters are aggregated to produce a new global model, which is subsequently distributed to clients for the next round of training. This process is repeated iteratively until the global model achieves satisfactory performance. or a predefined number of rounds is completed.

B. Targeted Model Poisoning Attacks

Recent works [2], [36], [3], [32], [40] focus on investigating targeted model poisoning attacks in federated learning. Targeted model poisoning attacks are designed to manipulate the globally trained model, compelling it to predict a specific target label τ for any input data incorporating predetermined attacker-chosen patterns (i.e., triggers). Moreover, the global model maintains its normal behavior when presented with untampered input data. Formally, targeted model poisoning attacks can be conceptualized as an optimization problem as the following:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \sum_{i \in \Psi} \left(\ell \left[f\left(\boldsymbol{w}; \boldsymbol{X}^i_{cln}\right), \boldsymbol{Y} \right] + \ell \left[f\left(\boldsymbol{w}; \boldsymbol{X}^i_{poi}\right), \boldsymbol{T} \right] \right) \tag{1}$$

where Ψ represents the set of compromised clients manipulated by the adversary $\mathcal{A}$, $\ell[\cdot]$ represents the cross-entropy loss between the predicted labels and the real labels, $\boldsymbol{X}^i_{cln} = \left\{x^i_1, x^i_2, \cdots, x^i_{n_i}\right\}$ and $\boldsymbol{X}^i_{poi} = \left\{x'^i_1, x'^i_2, \cdots, x'^i_{m_i}\right\}$ stand for clean data and carefully crafted poisoned data of the compromised client i, respectively, $\boldsymbol{Y} = \left\{y^i_1, y^i_2, \cdots, y^i_{n_i}\right\}$ and $\boldsymbol{T} = \left\{\tau^i_1, \tau^i_2, \cdots, \tau^i_{m_i}\right\}$ represent the clean labels and target labels. clean labels and attacker-desired labels. Moreover, it is essential to note that $\boldsymbol{X}^i_{cln} \cap \boldsymbol{X}^i_{poi} = \varnothing$, indicating that the poisoned data and the clean data do not intersect. Targeted model poisoning attacks aim to assign the highest probability to target label τ_j for a given poisoned data point x'_j, while simultaneously adhering to the ground truth label y_j for a given data point x_j. Additionally, the adversary $\mathcal{A}$ can design distinct poisoned data for each compromised client, delivering more stealthy attacks.

The Model Replacement attack (MR) [2] entails the adversary $\mathcal{A}$ embedding triggers into the local data of compromised clients and assigning attacker-desired labels to them. Subsequently, $\mathcal{A}$ governs the training process of the compromised clients, generating poisoned local model updates. Finally, $\mathcal{A}$ amplifies the poisoned local model updates with a substantial factor before submitting them to the server.

The distributed backdoor attack (DBA) [36] decomposes a global trigger pattern into distinct local patterns, embedding them into the training sets of different compromised parties. DBA exhibits greater persistence and stealthiness in federated learning compared to MR.

The label flipping attack (FLIP) [3] aims to cause targeted misclassification of auxiliary data by the global model learned at the server. For instance, in the case of the Fashion-MNIST dataset, the attack aims to misclassify an example originally labeled as class '7' (sneaker) into class '5' (sandal).

The edge-case backdoor attack (EDGE) [32] compels the global model to misclassify seemingly straightforward inputs

that are improbable to be part of the training or test data, residing on the tail of the input distribution.

The neurotoxin attack (NEUR) [40] projects the adversarial model updates onto the subspace unused by benign users, thereby enhancing the attack's persistence even after the attacker stops uploading poisoned updates.

The stealthy and colluded attack (CerP) [20] is designed to manipulate FL systems by carefully tuning the collusion between malicious participants. The objective of CerP is to minimize the trigger-induced bias of a poisoned local model compared to a poison-free one. CerP achieves this by jointly adjusting the backdoor trigger and controlling the changes in the poisoned model on each malicious participant. The attack strategy is crafted to be effective against a wide range of state-of-the-art defense mechanisms in FL.

C. Defenses against Targeted Model Poisoning Attacks

Several recent studies [11], [24], [6], [23] have investigated how to mitigate targeted model poisoning attacks in federated learning.

FoolsGold [11] distinguishes between benign and malicious clients by assessing pairwise similarity among model updates, subsequently down-weighting abnormal updates on the FL server. However, FoolsGold faces challenges in detecting poisoned model updates in the context of "DBA" and multi-objective targeted model poisoning attacks. These attacks generate poisoned model updates without common characteristics, posing a challenge to FoolsGold's ability to distinguish between poisoned and benign updates. Consequently, FoolsGold may exhibit limitations and failures in such scenarios. Moreover, In scenarios where the local data distributions of specific clients exhibit similarity, FoolsGold may erroneously discard the local model updates from these clients due to their similar directional contributions, thereby compromising the overall model performance.

RLR [24] conjectures that the aggregated updates from adversarial and honest agents are likely to differ in the directions, at least for some dimensions. Consequently, they dynamically adjust the learning rate of the aggregation server on a per-dimension and per-round basis, guided the sign information of agents' updates. However, RLR can effectively enhance the robustness of FL systems only if the number of compromised clients by the adversary $\mathcal{A}$ must be significantly lower than a specific threshold value θ.

FLTrust [6] presupposes that the server possesses an additional root dataset for computing a benign model update. Consequently, a local model update receives a lower trust score if its update direction significantly deviates from that of the server model update. This strategy mitigates the impact of poisoned local updates, effectively thwarting targeted model poisoning attacks. However, the collection of a clean validation dataset poses challenges. FLTrust may falter when there is a significant divergence between the distribution of the validation dataset and the overall training dataset.

FLAME [23] employs model clustering and weight clipping to estimate the requisite level of noise injection, ensuring the eradication of backdoors. It can maintain the benign performance of the aggregated model while effectively eliminating adversarial backdoors. However, when encountering inflation attacks [11] where the proportion of malicious clients is high, FLAME will not provide an effective defense.

Therefore, an additional defense approach is required to enhance the robustness of FL systems.

III. Preliminaries

A. (Density-Based Spatial Clustering of Applications with Noise

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm designed to discover clusters of arbitrary shapes in a dataset. Unlike traditional clustering algorithms that require specifying the number of clusters in advance, DBSCAN identifies clusters based on the density of data points. The algorithm works by selecting a starting data point and expanding the cluster by adding nearby points that have a minimum number of neighbors within a specified radius. Data points that do not meet the density criteria are treated as noise. DBSCAN is particularly effective in handling clusters with irregular shapes and is less sensitive to the order of input data points, making it a valuable tool in various applications, including spatial data analysis and anomaly detection.

B. Principal Component Analysis

The PCA algorithm stands as a foundational technique in the realm of dimensionality reduction. It falls within the domain of linear, unsupervised, and global methods for reducing dimensions in data. Its core principle involves linear mapping, succinctly characterized as the projection of high-dimensional data onto a reduced-dimensional space. In this process, PCA retains crucial principal components, encapsulating substantial data information, while discarding ancillary details of lesser significance for data description. In the context of m data points residing within an n-dimensional space, the following procedure outlines the key steps involved in employing Principal Component Analysis (PCA) for data preprocessing and dimensionality reduction: 1) Data organization: organize the data into an $n \times m$ matrix denoted as X. 2) Zero-centering: perform zero-centering on each row of matrix X by subtracting the mean of each row from all the elements within that row. 3) Covariance matrix computation: calculate the covariance matrix V_m, defined as $V_m = \frac{1}{m} X X^T$. 4) Eigenvalue-eigenvector pair extraction: calculate the eigenvalues and corresponding eigenvectors of the matrix V_m. 5) Eigenvector sorting: arrange the eigenvectors in decreasing order of their corresponding eigenvalues, forming a matrix where each column corresponds to an eigenvector. This matrix is denoted as P. 6) Dimensionality reduction: the reduced-dimension representation Y is obtained as $Y = P^T X$.

IV. Problem Setup and Objectives

A. Threat Model

This work focuses on defending against targeted model poisoning attacks in federated learning. Without loss of generality, RoseAgg assumes that the data distributed across each

client is concealed. We mainly follow the targeted model poisoning attacks in [2], [36], [3], [32], [40] and establish certain assumptions as outlined below:

- The adversary $\mathcal{A}$ has the same background knowledge of the FL protocol as the benign clients and can observe changes in the global model.
- $\mathcal{A}$ is unable to compromise the server, and it is explicitly forbidden from accessing the training data or local model updates of other honest clients.
- In the scenario where $\mathcal{A}$ only pursues a singular malicious objective, $\mathcal{A}$ can compromise a portion of genuine clients, without imposing any restrictions on the number of compromised clients. Conversely, if $\mathcal{A}$ harbors diverse malicious objectives, it is assumed to have control over no more than half of the clients.

B. Defense Goals

In this work, we aim to develop a robust aggregation rule on the server side, named RoseAgg, to counter targeted collusion attacks. Furthermore, RoseAgg should not sacrifice the utility and efficiency of the system. Concretely, RoseAgg must meet the following objectives:

- *Model performance.* RoseAgg should facilitate the global model in achieving high performance, irrespective of the presence of attacks.
- *Robustness.* RoseAgg must thwart the adversary from achieving its malicious objectives. This involves mitigating or eliminating the impact of poisoned model updates, which is crucial for ensuring that the aggregated global model remains free from any attacker-chosen malicious behavior.
- *Efficiency.* RoseAgg should not introduce excessive communication and computation overhead compared.

V. RoseAgg Overview and Design

In this section, we meticulously elucidate the high-level idea of RoseAgg along with its detailed designs. Moreover, for clarity and ease of comprehension, Table I provides a comprehensive list of symbols employed in this work along with their corresponding explanations.

A. High-level Idea

Motivation. State-of-the-art work [6] illustrates that despite the effectiveness of recent defenses [4], [8], [12], [38], the adversary can still corrupt the global model by submitting carefully crafted model updates to the server. The fundamental reason behind this is that the server lacks a trusted source and therefore cannot distinguish the poisoned local updates. To bridge the gap, they propose FLTrust [6], a trust-bootstrap-based FL framework for mitigating byzantine and targeted model poisoning attacks. FLTrust assumes that the server can acquire a clean root dataset for computing a benign model update. The server then assigns a trust score to each participant according to the direction deviation between the local model update and the server model update. However, this proposition encounters challenges in typical FL settings. Firstly, the server might encounter incomplete data collection, such as missing data from a specific class. Secondly, there's a risk of collecting poisoned data during the data collection process. In summary, these potential biases or inaccuracies may harm the robustness and reliability of the algorithm. Therefore, a generic method is required to compute a plausible clean model update solely based on the received local model updates, and to assign reasonable weights to these updates for aggregation.

TABLE I
NOTATIONS AND CORRESPONDING DEFINITIONS.

Notation	Definition
n	The number of the total clients.
$\mathcal{M}$	The number of the participants during each round.
$\mathcal{D}_i$	The private local dataset of client C_i.
$\boldsymbol{w}_t$	The global model at round t.
$\boldsymbol{g}_i^t$	The local update of client C_i during the t-th round.
$\bar{\boldsymbol{g}}_i^t$	The normalized local update of client C_i at round t.
$\mathcal{C}_i^t$	The i-th cluster at round t.
$\bar{\boldsymbol{g}}_j^{t,i}$	The j-th normalized local update with in i-th cluster during the t-th round.
$\widetilde{\boldsymbol{g}}^{t,i}$	The partial aggregated update of the i-th cluster during the t-th round.
$\widetilde{\boldsymbol{g}}_0^t$	The clean ingredient at round t.
$\boldsymbol{g}^t$	The global update at round t.
$\bar{\boldsymbol{g}}^t$	The normalized global update at round t.
$F(\boldsymbol{w})$	The expected empirical loss function concerning the model $\boldsymbol{w}$ over $\{\mathcal{D}_i\}_{i=1}^n$.
$f(\mathcal{D}, \boldsymbol{w})$	The empirical loss function concerning the model $\boldsymbol{w}$ on $\mathcal{D}$.

RoseAgg Overview. During each communication round between the server and participants, RoseAgg follows the general steps of FL discussed in Section II. In general, the adversary $\mathcal{A}$ has the ability to manipulate the directions and the magnitudes of local model updates on malicious participants, which can cause the global model to skew in the poisoned direction, resulting in targeted model poisoning attacks [23]. Therefore, RoseAgg considers both the directions and the magnitudes of local model updates when designing our defense strategy. Firstly, RoseAgg employs an adaptive partial aggregating process to mitigate the influence of local updates exhibiting scaled-up magnitudes or similar directional contributions. Secondly, RoseAgg calculates and maintains a plausible clean ingredient on the server, which is utilized to bootstrap credit scores for model updates. Thirdly, RoseAgg assigns credit scores to local model updates based on their projection variances on the plausible clean ingredient. Finally, RoseAgg computes a global model update by averaging local model updates weighted by the credit scores, thereby updating the global model.

B. RoseAgg Design

As described in Section V-A, RoseAgg consists of three main components: adaptive partial aggregating, clean ingredient extraction, and credit score computation. Figure 2 illustrates the corresponding components and the workflow of RoseAgg. Algorithm 2 outlines the procedure of RoseAgg.

1) Adaptive Partial Aggregating: The adaptive partial aggregating component consists of three parts: normalizing,

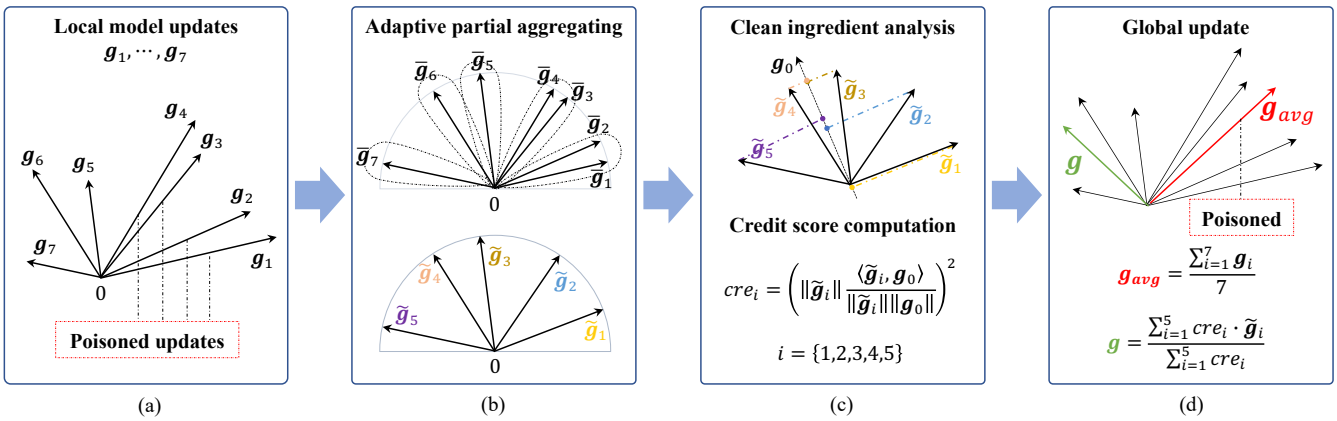

Fig. 2. Illustration of RoseAgg, which is applied during each communication round between the server and participants. Notably, $\boldsymbol{g}_{avg}$ and $\boldsymbol{g}$ represent global updates computed using the methods of simple average and RoseAgg, respectively.

Algorithm 1 LocalUpdate

Input: $\boldsymbol{w}, \mathcal{D}, b, \eta, E_l$ ▷ E_l is the local epochs, η is the local learning rate, and b is the batch size

Output: Local model update.

$\boldsymbol{w}^0 \leftarrow \boldsymbol{w}$
for e in $[1, E_l]$ **do**
  Randomly sample a batch $\mathcal{D}_b$ with the size of b from $\mathcal{D}$.
  $\boldsymbol{w}^e \leftarrow \boldsymbol{w}^{e-1} - \eta \nabla f(\mathcal{D}_b, \boldsymbol{w}^{e-1})$.
end for
return $\boldsymbol{w}^{E_l} - \boldsymbol{w}^0$

Algorithm 2 RoseAgg.

Input: $n, \boldsymbol{w}_0, T$ ▷ $\boldsymbol{w}_0$ is the initial global model, T is the total communication rounds.

Output: $\boldsymbol{w}_T$ ▷ $\boldsymbol{w}_T$ is the global model after T rounds.

1: **for** each round t in $[1, T]$ **do**
2: The server randomly samples $\mathcal{M}$ participants.
3: **for** each client C_i in $[C_1, C_2, \cdots, C_{\mathcal{M}}]$ **parallel do**
4: $\boldsymbol{g}_i^t \leftarrow \mathit{LocalUpdate}\,(\boldsymbol{w}_{t-1}, \mathcal{D}_i, b, \eta, E_l)$.
5: Submit $\boldsymbol{g}_i^t$ to the server.
6: **end for**
7: $\bar{\boldsymbol{g}}_i^t \leftarrow \frac{\boldsymbol{g}_i^t}{\|\boldsymbol{g}_i^t\|}, \forall i \in [1, \mathcal{M}]$ ▷ Normalizing
8: $(\mathcal{C}_1^t, \cdots, \mathcal{C}_{K_t}^t) \leftarrow$ Clustering $(\boldsymbol{g}_1^t, \boldsymbol{g}_2^t, \cdots, \boldsymbol{g}_{\mathcal{M}}^t)$. ▷ Dynamically clustering local updates to K_t clusters.
9: **for** i in $[1, K_t]$ **do**
10: Calculate $\widetilde{\boldsymbol{g}}^{t,i}$. ▷ Partial aggregating.
11: **end for**
12: Stacking $\{\widetilde{\boldsymbol{g}}^{t,i}\}_{i=1}^{K_t}$ to form a matrix $\mathcal{H}^t$ ▷ $\mathcal{H}^t \in \mathbb{R}^{d \times K_t}$, d is the dimension of each model update.
13: Renormalizing each column of $\mathcal{H}^t$.
14: $\widehat{\mathcal{H}}^t \leftarrow {\mathcal{H}^t}^T \mathcal{H}^t$
15: $\widetilde{\boldsymbol{g}}_0^t \leftarrow \mathcal{H}^t \xi_{max} / \sqrt{\lambda_{max}}$, λ_{max} and ξ_{max} are the maximum eigenvalue and corresponding eigenvector of $\widehat{\mathcal{H}}^t$.
16: $cre_i^t \leftarrow \left(\|\widetilde{\boldsymbol{g}}^{t,i}\| \frac{\langle \widetilde{\boldsymbol{g}}^{t,i}, \widetilde{\boldsymbol{g}}_0^t \rangle}{\|\widetilde{\boldsymbol{g}}^{t,i}\| \|\widetilde{\boldsymbol{g}}_0^t\|} \right)^2$, $\boldsymbol{g}^t \leftarrow \frac{\sum_i cre_i^t \cdot \widetilde{\boldsymbol{g}}^{t,i}}{\sum_i cre_i^t}$, $\forall i \in [1, K_t]$, $\boldsymbol{g}^t \leftarrow \gamma_t \frac{\boldsymbol{g}^t}{\|\boldsymbol{g}^t\|}$ ▷ γ_t relies on the magnitudes of local updates at round t.
17: $\boldsymbol{w}_t \leftarrow \boldsymbol{w}_{t-1} - \boldsymbol{g}^t$.
18: **end for**

adaptive clustering, and partial aggregating. The adaptive partial aggregating step is shown in lines 7-11 of Algorithm 2.

Normalizing. The adversary can scale up the magnitude of poisoned local model updates [2], [36], which amplifies their impact on the global model update and enables successfully targeted model poisoning. To counter this, RoseAgg uses normalization to enable all local model updates to lie in the same hyper-sphere. Namely, RoseAgg only considers the direction of each local update in this stage. Formally:

$$\bar{\boldsymbol{g}}_i^t = \frac{\boldsymbol{g}_i^t}{\|\boldsymbol{g}_i^t\|} \tag{2}$$

The normalization prevents any single local update from excessively impacting the global model update. Moreover, it can also eliminate bias stemming from variations in the magnitude of local updates, which may arise when employing a dimension-reduction method to identify a plausible clean ingredient.

Dynamic clustering. The technique of Dynamic clustering, based on DBSCAN [10], identifies local model updates that have similar direction contributions. The state-of-the-art defense [23] clusters local models based on pairwise cosine distances. They set the minimum cluster size to be $\lfloor \frac{n}{2} \rfloor + 1$ so that the resulting cluster contains the majority of the local models. The remaining local models are considered potentially malicious and are therefore removed. In addition, the defenses [27], [39] use K-means to cluster all local model updates into two groups, retaining only the larger group for aggregation. However, when the proportion of malicious clients is high, these approaches may retain the poisoned models after clustering, increasing the impact of poisoned local model updates.

FL operates on a decentralized model, with local models trained on data distributed across multiple clients or nodes. These clients can have variations in data distribution, data quality, or even the nature of the data itself. This inherent data heterogeneity can lead to diverse local models. Therefore, even if some local models appear to be outliers or noisy (meaning

they might not perform as well as others or seem different from the majority), they should not be discarded or removed during the process of aggregating these models to form a global model. The reason for not removing these diverse local models is that they might still contain valuable information. They could be representative of unique but relevant data patterns that are specific to that particular client. By including these diverse models in the aggregation process, the global model can potentially learn from this variety and become more robust and adaptable. Consequently, when applying the DBSCAN algorithm for clustering, the parameter for the minimum number of samples required to form a cluster is set to **1**. This means that even a single model can be considered a separate cluster in this context. In simple terms, when RoseAgg uses the DBSCAN algorithm, it doesn't identify any of the local models as noise. All models are treated as potentially meaningful and are assigned to clusters.

Partial aggregating. As the local model updates within each cluster have similar direction contributions, RoseAgg computes a model update for each cluster to represent its directional contribution. Such a way can eliminate the redundancy of similar directional contributions, therefore limiting the impact of potentially malicious model updates from the directional perspective. Specifically, a partial aggregating module is employed to achieve this. Assuming the i-th cluster contains κ local updates during the round t, forming a distance square matrix $S^{t,i}$ of size $\kappa \times \kappa$, *i.e.*, $S^{t,i} \in \mathbb{R}^{\kappa \times \kappa}$. The partially aggregated update $\widetilde{\boldsymbol{g}}^{t,i}$ is calculated as follows:

$$\begin{cases} \widetilde{\boldsymbol{g}}^{t,i} = \bar{\boldsymbol{g}}_{\kappa}^{t,i}, & \kappa = 1 \\ \widetilde{\boldsymbol{g}}^{t,i} = \dfrac{\sum_{j=1}^{\kappa} \alpha_j^{t,i} \cdot \bar{\boldsymbol{g}}_j^{t,i}}{\sum_{j=1}^{\kappa} \alpha_j^{t,i}}. & \kappa > 1 \end{cases} \tag{3}$$

where $\alpha_j^{t,i}$ represents the summation of the j-th row from the distance matrix $S^{t,i}$.

Before extracting the clean ingredient of the local model updates, $\widetilde{\boldsymbol{g}}^{t,i}$ need to be re-normalized to avoid the influence of the magnitude of the model updates on the extraction result. The reason for re-normalization is that the process of partial aggregating involves a weighted sum of vectors such that the resulting $\widetilde{\boldsymbol{g}}_i^t$ has a magnitude less than 1.

2) Clean Ingredient Extraction: The clean ingredient denotes a reliable ingredient extracted from local model updates, satisfying the condition where the sum of projection variances of benign updates onto it surpasses the sum of projection variances of poisoned updates onto the same ingredient. Suppose that all local updates are organized into K_t clusters during the adaptive partial aggregating phase at round t. Formally, $\widetilde{\boldsymbol{g}}_0^t$ is is considered as the clean ingredient if the following inequality holds:

$$\sum_{i=1}^{m} \left(\|\widetilde{\boldsymbol{g}}^{t,i}\| \frac{\langle \widetilde{\boldsymbol{g}}^{t,i}, \widetilde{\boldsymbol{g}}_0^t \rangle}{\|\widetilde{\boldsymbol{g}}^{t,i}\| \|\widetilde{\boldsymbol{g}}_0^t\|} \right)^2 < \sum_{i=m+1}^{K_t} \left(\|\widetilde{\boldsymbol{g}}^{t,i}\| \frac{\langle \widetilde{\boldsymbol{g}}^{t,i}, \widetilde{\boldsymbol{g}}_0^t \rangle}{\|\widetilde{\boldsymbol{g}}^{t,i}\| \|\widetilde{\boldsymbol{g}}_0^t\|} \right)^2 \tag{4}$$

where $\langle \cdot \rangle$ represents the dot product of two vectors, clusters $1, \cdots, m$ encompass poisoned updates, referred to "poisoned clusters", while clusters $m+1, \cdots, K_t$ do not include poisoned updates, referred to "benign clusters".

Figure 3 displays the pairwise cosine distance among local model updates. It is noticeable that during targeted model poisoning attacks, if attackers have a shared malicious goal, their model updates exhibit more pronounced similarities than those among most benign clients. Our hypothesis is that, since attackers have a shared malicious goal, the feature spaces relevant to the malicious goal would be very similar for each attacker. The similarity of the feature spaces can be measured using the pairwise cosine distance of the local model updates. Moreover, clients who have similar local data distribution would also exhibit some similarities. Therefore, it is essential to retain all clusters as the clustering process cannot differentiate which cluster contains attackers.

Based on specific attack assumptions illustrated in Section IV-A, the adaptive partial aggregating module enables poisoned updates to be organized into relatively few clusters. Conretely, the adaptive partial aggregating module in RoseAgg groups local updates into clusters, ensuring the number of potential "benign clusters" surpasses that of potential "poisoned clusters". Consequently, when represented model updates are extracted from these clusters, the number of potential benign updates surpasses that of potential poisoned updates. Furthermore, RoseAgg utilizes the concept of maximizing variance via Principal Component Analysis (PCA) to isolate a principal component. Given that all updates are standardized and benign updates outnumber poisoned updates, it is self-evident that the sum of projected variances of benign updates on this principal component is greater than the sum of projected variances of poisoned updates on the same component. As a result, this principal component is considered benign.

Concretely, each partially aggregated model update, denoted as $\widetilde{\boldsymbol{g}}^{t,i}$, is viewed as a high-dimensional data point. Then the collection $\left(\widetilde{\boldsymbol{g}}^{t,1}, \widetilde{\boldsymbol{g}}^{t,2}, \cdots, \widetilde{\boldsymbol{g}}^{t,K_t}\right)$ can be regard as a compact dataset, comprising K_t data points. Each of these data points is d-dimensional, where d is the dimension of each local model update. To formalize this dataset, RoseAgg employs a matrix $\mathcal{H}^t \in \mathbb{R}^{d \times K_t}$. The goal is to extract the first principal component of the matrix $\mathcal{H}^t$ using PCA [33]. This principal component serves as the desired ingredient.

The matrix $\mathcal{H}^t$ has column vectors with very large dimensions. Using PCA directly would demand massive computation and memory resources. Thus, the desired ingredient is extracted using the following approach:

$$\begin{aligned} \widehat{\mathcal{H}}^t &\leftarrow {\mathcal{H}^t}^T \mathcal{H}^t \\ \widetilde{\boldsymbol{g}}_0^t &\leftarrow \mathcal{H}^t \xi_{max} / \sqrt{\lambda_{max}} \end{aligned} \tag{5}$$

where $\widetilde{\boldsymbol{g}}_0^t$ is the desired ingredient, λ_{max} and ξ_{max} are the maximum eigenvalue and corresponding eigenvector of $\widehat{\mathcal{H}}^t$. In addition, since $\widehat{\mathcal{H}}^t \in \mathbb{R}^{K_t \times K_t}$ is a low-dimensional matrix, the computation overhead for calculating λ_{max} and ξ_{max} on the server would not be excessive. The clean ingredient extraction step is shown in lines 12-15 of Algorithm 2.

3) Credit Score Computation: FLTrust [6] assigns each client a trust score based on the ReLU-clipped cosine distance. However, this could assign the model update of **0** with large

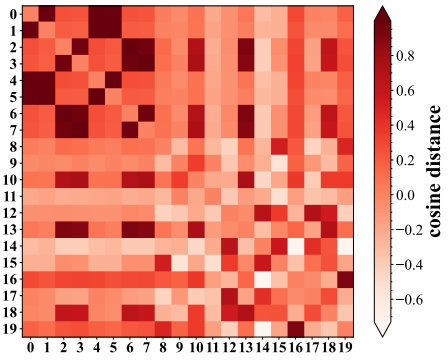

Fig. 3. Cosine distance matrix of local updates. The matrix includes client IDs ranging from 0 to 19. The objectives of Clients 0 and 4 are to misclassify images containing a pixel pattern "F" as airplanes. Clients 1 and 5, on the other hand, intend to misidentify images with a pixel pattern "L" as airplanes. Clients 2 and 6 specifically aim to misclassify images featuring a pixel pattern resembling the letter "A" as birds. Lastly, Clients 3 and 7 focus on misidentifying images of cars that have vertically striped walls in the background as birds.

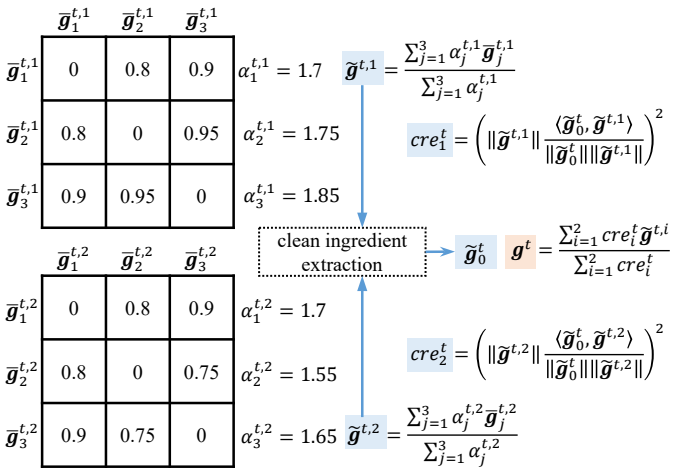

Fig. 4. Toy example of RoseAgg methodology.

angular between the model update maintained by the server. Namely, FLTrust considers such model updates as malicious updates and excludes them. Nevertheless, such angular deviation may be caused by the data heterogeneity [17], [25], [14]. Simply removing the model updates may cause accuracy deterioration of the global model. Therefore, an alternative approach is required.

Specifically, RoseAgg uses the projection variance of each partially aggregated update on the clean ingredient as its credit score. Formally, the credit score is defined as follows:

$$cre_i^t \leftarrow \left(\|\widetilde{\boldsymbol{g}}^{t,i}\| \frac{\langle \widetilde{\boldsymbol{g}}^{t,i}, \widetilde{\boldsymbol{g}}_0^t \rangle}{\|\widetilde{\boldsymbol{g}}^{t,i}\| \|\widetilde{\boldsymbol{g}}_0^t\|} \right)^2, \quad i \in [1, K_t] \tag{6}$$

where cre_i^t is the credit score for the i-th partially aggregated update $\widetilde{\boldsymbol{g}}^{t,i}$ at round t. The rationale for calculating credit scores in this manner is based on the fact that $\widetilde{\boldsymbol{g}}_0^t$ is a plausible clean ingredient. Consequently, the summation of projection variances of benign data points will be greater than that of poisoned data points. As a result, the weight of benign data points will be greater than that of poisoned data points in weighted summation, thereby mitigating model poisoning attacks.

4) Updating the Global Model: The global update is calculated by averaging the partially aggregagted model updates weighted by the credit scores:

$$\begin{aligned} \boldsymbol{g}^t &\leftarrow \frac{\sum_{i=1}^{K_t} cre_i^t \widetilde{\boldsymbol{g}}^{t,i}}{\sum_{i=1}^{K_t} cre_i^t} \\ \bar{\boldsymbol{g}}^t &\leftarrow \frac{\boldsymbol{g}^t}{\|\boldsymbol{g}^t\|} \end{aligned} \tag{7}$$

where $\bar{g}_t$ is the normalized global update at round t. In this way, RoseAgg can incorporate the contributions of all participants while giving more weight to benign participants. Subsequently, the server proceeds to update the global model as follows:

$$\boldsymbol{w}_t \leftarrow \boldsymbol{w}_{t-1} - \gamma_t \bar{\boldsymbol{g}}^t \tag{8}$$

where $\boldsymbol{w}_t$ is the global model at round t, and γ_t is the magnitude of global update at round t.

In practice, the hyper-parameter γ_t is computed as follows: 1) sorting the ℓ_2 norms. The ℓ_2 norms of local model updates are sorted in ascending order during the communication round t. These norms are arranged as $a_1^t, \cdots, a_i^t, a_j^t, \cdots, a_{\mathcal{M}}^t$, where $\forall i < j \leq \mathcal{M}, a_i^t \leq a_j^t$. 2) detecting sudden changes. To detect sudden changes in the increase between two consecutive ℓ_2 norms, RoseAgg calculates the difference between each consecutive pair of a_i^t and a_{i+1}^t as $\Delta a_i^t = a_{i+1}^t - a_i^t$. A sudden change can be defined as a significant increase in Δa_i^t compared to the previous values. Specifically, RoseAgg sets a percentage change threshold ζ and considers any Δa_i^t that exceeds this threshold as indicating a sudden change. Formally, if $\Delta a_i^t > (1+\zeta)\Delta a_{i-1}^t$, it indicates a sudden change in the increase in ℓ_2 norms. 3) computing γ_t. If no sudden change is detected, indicating steady and continuous increases in the ℓ_2 norms, then γ_t is set as the median of all ℓ_2 norms, *i.e.*, $\gamma_t = Median(a_1^t, a_2^t, \cdots, a_{\mathcal{M}}^t)$. However, if a sudden change is identified at index i, then γ_t is set to the median of the ℓ_2 norms up to index i, *i.e.*, $\gamma_t = Median(a_1^t, a_2^t, \cdots, a_i^t)$. Since the ℓ_2 norm of each benign model update becomes smaller in later rounds, γ_t will decrease with the increase of rounds.

For ease of understanding, Figure 4 provides a toy example to illustrate the overall workflow of RoseAgg. Consider a scenario with 6 clients clustered into 2 groups using the DBSCAN algorithm, employing cosine distance as the similarity metric. Each cluster has a distance matrix, as illustrated in Figure 4. RoseAgg initially performs adaptive clustering and subsequently apply partial aggregation to local updates within each cluster. To exemplify this, let's take local updates within cluster 1. These updates are combined to create a partially aggregated vector $\widetilde{\boldsymbol{g}}^{t,1}$, computed as $\widetilde{\boldsymbol{g}}^{t,1} = \frac{\sum_{j=1}^{3} \alpha_j^{t,1} \bar{\boldsymbol{g}}_j^{t,1}}{\sum_{j=1}^{3} \alpha_j^{t,i}}$. Here, $\alpha_j^{t,1}$ represents the summation of the j-th row from the distance matrix of cluster 1 during the t-th round. Subsequently, a clean ingredient $\widetilde{\boldsymbol{g}}_0^t$ is derived via the clean ingredient extraction module. Formally, let $\mathcal{H}^t = \left(\widetilde{\boldsymbol{g}}^{t,1}, \widetilde{\boldsymbol{g}}^{t,2}\right) \in \mathbb{R}^{d \times 2}$, where d is the dimension of the model. $\widetilde{\boldsymbol{g}}_0^t$ corresponds precisely to the first principal

component of the matrix $\widehat{\mathcal{H}}^t = \mathcal{H}^t \mathcal{H}^{t^T}$. Furthermore, a credit score is calculated for each partially aggregated update. Concretely, the credit score of $\widetilde{\boldsymbol{g}}^{t,1}$ is computed as follows: $cre_1 = \left(\|\widetilde{\boldsymbol{g}}^{t,1}\| cos\left(\widetilde{\boldsymbol{g}}_0^t, \widetilde{\boldsymbol{g}}^{t,1}\right)\right)^2 = \left(\|\widetilde{\boldsymbol{g}}^{t,1}\| \frac{\langle \widetilde{\boldsymbol{g}}_0^t, \widetilde{\boldsymbol{g}}^{t,1}\rangle}{\|\widetilde{\boldsymbol{g}}_0^t\| \|\widetilde{\boldsymbol{g}}^{t,1}\|}\right)^2$, where $\langle \widetilde{\boldsymbol{g}}_0^t, \widetilde{\boldsymbol{g}}^{t,1}\rangle$ denotes the inner product between $\widetilde{\boldsymbol{g}}_0^t$ and $\widetilde{\boldsymbol{g}}^{t,1}$. Finally, the global update is determined by computing the average of the partially aggregated model updates, as follows: $\boldsymbol{g}^t = \frac{\sum_{i=1}^2 cre_i^t \widetilde{\boldsymbol{g}}^{t,i}}{\sum_{i=1}^2 cre_i^t}$.

VI. Theoretical Analysis

In FL, the optimal global model $\boldsymbol{w}^*$ is the solution to the following optimization problem: $\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}\in\Theta} F(\boldsymbol{w}) \triangleq \mathbb{E}_{D\sim\chi}[f(\mathcal{D}, \boldsymbol{w})]$. Θ is the parameter space of the global model. $\mathcal{D}$ is the joint training dataset of the n clients. The distribution of the overall data is represented by χ. For the sake of clarity, the superscript t is omitted in this section.

Assumption 1. *The expected loss function $F(\boldsymbol{w})$ is μ-strongly convex and differentiable over model parameter space Θ with L-Lipschitz continuous gradient. Formally, for any $\boldsymbol{w}, \hat{\boldsymbol{w}} \in \Theta$:*

$$F(\hat{\boldsymbol{w}}) \geq F(\boldsymbol{w}) + \langle \nabla F(\boldsymbol{w}), \hat{\boldsymbol{w}} - \boldsymbol{w}\rangle + \frac{\mu}{2}\|\hat{\boldsymbol{w}} - \boldsymbol{w}\|^2,$$
$$\|\nabla F(\boldsymbol{w}) - \nabla F(\hat{\boldsymbol{w}})\| \leq L\|\boldsymbol{w} - \hat{\boldsymbol{w}}\|,$$

For the local empirical loss function $f(\mathcal{D}, \boldsymbol{w})$, we assume that it is probabilistically L_1-Lipschitz. Then for any $\delta \in (0, 1)$, there exists an positive constant L_1 satisfying:

$$Pr\{\sup_{\boldsymbol{w},\hat{\boldsymbol{w}}\in\Theta;\boldsymbol{w}\neq\hat{\boldsymbol{w}}} \frac{\|\nabla f(\mathcal{D}, \boldsymbol{w}) - \nabla f(\mathcal{D}, \hat{\boldsymbol{w}})\|}{\|\boldsymbol{w} - \hat{\boldsymbol{w}}\|} \leq L_1\} \geq 1 - \frac{\delta}{3}$$

Assumption 2. *The gradient of the empirical loss function $\nabla f(\mathcal{D}, \boldsymbol{w}^*)$ at the optimal global model $\boldsymbol{w}^*$ is bounded. Moreover, the gradient difference $h(\mathcal{D}, \boldsymbol{w}) = \nabla f(\mathcal{D}, \boldsymbol{w}) - \nabla f(\mathcal{D}, \boldsymbol{w}^*)$ for any $\boldsymbol{w} \in \Theta$ is bounded. Specifically, there exist positive constants σ_1 and κ_1 so that for any unit vector v, $\langle \nabla f(\mathcal{D}, \boldsymbol{w}^*), v\rangle$ is sub-exponential with σ_1 and κ_1; and there also exist positive constants σ_2 and κ_2 so that for any $\boldsymbol{w} \in \Theta$ with $\boldsymbol{w} \neq \hat{\boldsymbol{w}}$ and any unit vector v, $\langle h(\mathcal{D}, \boldsymbol{w}) - \mathbb{E}[h(\mathcal{D}, \boldsymbol{w})], v\rangle / \|\boldsymbol{w} - \boldsymbol{w}^*\|$ is sub-exponential with σ_2 and κ_2. Formally, for $\forall|\xi| \leq 1/\kappa_1$, $\forall|\xi| \leq 1/\kappa_2$:*

$$\sup_{v\in B} \mathbb{E}[exp(\xi\langle \nabla f(\mathcal{D}, \boldsymbol{w}^*), v\rangle)] \leq e^{(\sigma_1)^2\xi^2/2},$$
$$\sup_{\boldsymbol{w}\in\Theta, v\in B} \mathbb{E}[exp(\frac{\xi\langle h(\mathcal{D}, \boldsymbol{w}) - \mathbb{E}[h(\mathcal{D}, \boldsymbol{w})], v\rangle}{\|\boldsymbol{w} - \boldsymbol{w}^*\|})] \leq e^{(\sigma_2)^2\xi^2/2},$$

where B is the unit sphere $B = \{v : \|v\| = 1\}$.

Assumption 3. *γ depends on the gradient magnitude of the benign client, which is assumed to be bounded. Hence, we make the following assumptions regarding the bound of γ: $\gamma \leq \Gamma$.*

Lemma 1. *Assume Assumption 3 holds. The distance between $\boldsymbol{g} = \gamma\bar{\boldsymbol{g}}$ and $\nabla F(\boldsymbol{w})$ is bounded as follows in each iteration:*

$$\|\boldsymbol{g} - \nabla F(\boldsymbol{w})\| \leq \Gamma + G\|\nabla F(\boldsymbol{w})\| + (G+1)\|\boldsymbol{g}_0 - \nabla F(\boldsymbol{w})\| \tag{9}$$

where $G = \max\{1, 2\Gamma - 1\}$. Given that $\boldsymbol{g}_0$ is a plausible ingredient, it can be approximately regarded as a model update resulting from the joint training of a group of benign clients. Formally, $\boldsymbol{g} \approx \frac{1}{n_b}\sum_{i\in\mathcal{N}_b} \frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i} \nabla f(X_j, \boldsymbol{w})$, $\mathcal{N}_b$ denotes the collection of benign clients, with a total count of n_b, $\mathcal{D}_i$ denotes the dataset of client i, $|\mathcal{D}_i|$ is the size of $\mathcal{D}_i$, and X_j represents the j -th data sample contained within $\mathcal{D}_i$.

Proof.

$$\begin{aligned}
&\|\boldsymbol{g} - \nabla F(\boldsymbol{w})\| \\
&\leq \|\gamma\bar{\boldsymbol{g}} - \boldsymbol{g}_0\| + \|\boldsymbol{g}_0 - \nabla F(\boldsymbol{w})\| \\
&= \|\gamma(\bar{\boldsymbol{g}} - \boldsymbol{g}_0) - (1-\gamma)\boldsymbol{g}_0\| + \|\boldsymbol{g}_0 - \nabla F(\boldsymbol{w})\| \\
&\leq \gamma(1 + \|\boldsymbol{g}_0\|) + |1-\gamma|\|\boldsymbol{g}_0\| + \|\boldsymbol{g}_0 - \nabla F(\boldsymbol{w})\| \\
&= \gamma + (\gamma + |1-\gamma|)\|\boldsymbol{g}_0\| + \|\boldsymbol{g}_0 - \nabla F(\boldsymbol{w})\| \\
&\leq \Gamma + \underbrace{\max\{1, 2\Gamma - 1\}}_{G}\|\boldsymbol{g}_0\| + \|\boldsymbol{g}_0 - \nabla F(\boldsymbol{w})\| \\
&= \Gamma + G\|\boldsymbol{g}_0 + \nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w})\| + \|\boldsymbol{g}_0 - \nabla F(\boldsymbol{w})\| \\
&\leq \Gamma + G\|\nabla F(\boldsymbol{w})\| + (G+1)\|\boldsymbol{g}_0 - \nabla F(\boldsymbol{w})\|.
\end{aligned}$$

□

Lemma 2. *Suppose Assumption 1 holds, the following would apply for any global iteration $t \geq 1$, where α as the global learning rate:*

$$\begin{aligned}
&\|\boldsymbol{w}_{t-1} - \boldsymbol{w}^* - \alpha\nabla F(\boldsymbol{w}_{t-1})\|^2 \\
&\qquad \leq (1 + \alpha^2L^2 - \alpha\mu)\|\boldsymbol{w}_{t-1} - \boldsymbol{w}^*\|^2
\end{aligned} \tag{10}$$

Proof. Derived from $\nabla F(\boldsymbol{w}^*) = 0$, the following expression arises:

$$\begin{aligned}
&\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^* - \alpha\nabla F(\boldsymbol{w}^{t-1})\|^2 \\
&= \|\boldsymbol{w}^{t-1} - \boldsymbol{w}^* - \alpha(\nabla F(\boldsymbol{w}^{t-1}) - \nabla F(\boldsymbol{w}^*))\|^2 \\
&= \|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|^2 + \alpha^2\|\nabla F(\boldsymbol{w}^{t-1}) - \nabla F(\boldsymbol{w}^*)\|^2 \\
&\quad - 2\alpha\langle \boldsymbol{w}^{t-1} - \boldsymbol{w}^*, \nabla F(\boldsymbol{w}^{t-1}) - \nabla F(\boldsymbol{w}^*)\rangle.
\end{aligned} \tag{11}$$

Based on Assumption 1, the following formulas holds:

$$\|\nabla F(\boldsymbol{w}^{t-1}) - \nabla F(\boldsymbol{w}^*)\| \leq L\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|, \tag{12}$$

$$\begin{aligned}
&F(\boldsymbol{w}^*) + \langle \nabla F(\boldsymbol{w}^*), \boldsymbol{w}^{t-1} - \boldsymbol{w}^*\rangle \leq \\
&\qquad F(\boldsymbol{w}^{t-1}) - \frac{\mu}{2}\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|^2,
\end{aligned} \tag{13}$$

$$F(\boldsymbol{w}^{t-1}) + \langle \nabla F(\boldsymbol{w}^{t-1}), \boldsymbol{w}^* - \boldsymbol{w}^{t-1}\rangle \leq F(\boldsymbol{w}^*), \tag{14}$$

By summing up inequalities (13) and (14), the following expression is obtained:

$$\langle \nabla F(\boldsymbol{w}^{t-1}) - \nabla F(\boldsymbol{w}^*), \boldsymbol{w}^* - \boldsymbol{w}^{t-1}\rangle \leq -\frac{\mu}{2}\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|^2, \tag{15}$$

By applying inequalities (12) and (15) to (11), the following is established:

$$\begin{aligned}
&\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^* - \alpha\nabla F(\boldsymbol{w}^{t-1})\|^2 \leq (1 + \alpha^2L^2 - \alpha\mu) \\
&\qquad \|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|^2.
\end{aligned} \tag{16}$$

□

Lemma 3. *Suppose Assumption 2 holds for any benign client i, the following conditions are satisfied: $\forall \delta \in (0,1)$ and $\boldsymbol{w} \in \Theta$, let $\Delta_1^i = \sqrt{2}\sigma_1\sqrt{(d\log 6 + \log(3/\delta))/|\mathcal{D}_i|}$ and $\Delta_3^i = \sqrt{2}\sigma_2\sqrt{(d\log 6 + \log(3/\delta))/|\mathcal{D}_i|}$. If $\Delta_1^i \le (\sigma_1)^2/\kappa_1$, $\Delta_3^i \le (\sigma_2)^2/\kappa_2$. Then:*

$$\begin{aligned} &Pr\{\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*)\| \ge 2\Delta_1^i\} \\ &\qquad\qquad \le \frac{\delta}{3n_b}, \\ &Pr\{\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i} h(X_j,\boldsymbol{w}^*) - \mathbb{E}[h(X,\boldsymbol{w})]\| \ge \\ &\qquad\qquad 2\Delta_3^i\|\boldsymbol{w}-\boldsymbol{w}^*\|\} \le \frac{\delta}{3n_b} \end{aligned} \tag{17}$$

Proof. The initial step involves establishing the validity of the first inequality presented in Lemma 3. The proof of the second inequality is omitted, as it follows a similar rationale. Let $V = \{v_1, .., v_{N_{\frac{1}{2}}}\}$ be an $\frac{1}{2}$-cover of the unit sphere B. Following [6], [8], when $N_{\frac{1}{2}} < s\cdot\log 6$, where $s = o(|\mathcal{D}_i|)$:

$$\begin{aligned} &\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*)\| \\ &\le 2\sup_{v\in V}\{\langle\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*), v\rangle\}. \end{aligned} \tag{18}$$

If Assumption 2 holds and $\Delta_1^i \le (\sigma_1)^2/\kappa_1$, the following inequality is satisfied based on the concentration inequalities for sub-exponential random variables [31]:

$$\begin{aligned} &Pr\{\langle\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*), v\rangle \ge \Delta_1^i\} \\ &\le exp(-|\mathcal{D}_i|(\Delta_1^i)^2/(2\sigma_1^2)). \end{aligned} \tag{19}$$

Recall that V contains at most 6^s vectors. In view of the union bound, it further yields that:

$$\begin{aligned} Pr\{2\sup_{v\in V}\{\langle\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*), v\rangle\} \ge 2\Delta_1^i\} \\ \le 6^s exp(-|\mathcal{D}_i|(\Delta_1^i)^2/(2\sigma_1^2)) \\ = exp(-|\mathcal{D}_i|(\Delta_1^i)^2/(2\sigma_1^2) + s\log 6). \end{aligned} \tag{20}$$

Therefore,

$$\begin{aligned} &Pr\{\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*)\| \ge 2\Delta_1^i\} \\ &\le exp(-|\mathcal{D}_i|(\Delta_1^i)^2/(2\sigma_1^2) + s\log 6). \end{aligned} \tag{21}$$

□

Lemma 4. *Suppose Assumptions 1 and 2 hold and for any benign client i, the following conditions are satisfied: let $\Theta \subset \{\boldsymbol{w} : \|\boldsymbol{w}-\boldsymbol{w}^*\| \le r\sqrt{s}\}$ hold for some positive parameter r. Then, for any $\delta \in (0,1)$, if $\Delta_1^i \le (\sigma_1)^2/\kappa_1$ and $\Delta_2^i \le (\sigma_2)^2/\kappa_2$, the following relationship holds for any $\boldsymbol{w} \in \Theta$:*

$$\begin{aligned} &Pr\{\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\| \\ &\qquad \le 8\Delta_2^i\|\boldsymbol{w}-\boldsymbol{w}^*\| + 4\Delta_1^i\} \ge 1 - \frac{\delta}{n_b} \end{aligned} \tag{22}$$

where $\Delta_2^i = \sigma_2\sqrt{\frac{2}{|\mathcal{D}_i|}(K_1^i + K_2^i)}$, $K_1^i = s\log\frac{18L_2}{\sigma_2}$, $K_2^i = \frac{1}{2}s\log\frac{|\mathcal{D}_i|}{s} + \log\frac{6\sigma_2^2 r\sqrt{|\mathcal{D}_i|}}{\kappa_2\sigma_1\delta}$, $L_2 = \max\{L, L_1\}$.

Proof. The proof primarily relies on the ϵ-net argument [6], [8]. Additionally, the subscripts i of the variables such as τ, l^*, ϵ_l, etc., are omitted for brevity. Let

$$\tau = \frac{\kappa_2\sigma_1}{2\sigma_2^2}\sqrt{\frac{s}{|\mathcal{D}_i|}} \quad and \quad l^* = \lceil r\sqrt{s}/\tau\rceil, \tag{23}$$

For integers $1 \le l \le l^*$, let $\Theta_l \subset \{\boldsymbol{w} : \|\boldsymbol{w}-\boldsymbol{w}^*\| \le \tau l\}$. For a given l, let $\boldsymbol{w}_1, ..., \boldsymbol{w}_{\epsilon_l}$ be an ϵ_l-cover of Θ_l, where ϵ_l is given by

$$\epsilon_l = \frac{\sigma_2\tau l}{L_2}\sqrt{\frac{s}{|\mathcal{D}_i|}}, \tag{24}$$

where $L_2 = \max\{L, L_1\}$. Based on [30], it is established that $N_{\epsilon_l} \le s\log(3\tau l/\epsilon_l)$. For any $\boldsymbol{w} \in \Theta_l$, there exits a $j_l (1 \le j_l \le N_{\epsilon_l})$ such that:

$$\|\boldsymbol{w} - \boldsymbol{w}_{j_l}\| \le \epsilon_l. \tag{25}$$

By the triangle inequality:

$$\begin{aligned} &\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\| \\ &\quad\le \|\nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w}_{j_l})\| \\ &\quad+ \|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}_{j_l}) - \nabla F(\boldsymbol{w}_{j_l})\| \\ &\quad+ \|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}(\nabla f(X_j,\boldsymbol{w}) - \nabla f(X_j,\boldsymbol{w}_{j_l}))\| \end{aligned} \tag{26}$$

According to Assumption 1 and inequality (25):

$$\|\nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w}_{j_l})\| \le L\|\boldsymbol{w} - \boldsymbol{w}_{j_l}\| \le L\epsilon_l. \tag{27}$$

Then, an event $\mathcal{E}_1$ is defined as follows:

$$\mathcal{E}_1 = \{\sup_{\boldsymbol{w},\hat{\boldsymbol{w}}\in\Theta;\boldsymbol{w}\ne\hat{\boldsymbol{w}}}\frac{\|\nabla f(\mathcal{D},\boldsymbol{w}) - \nabla f(\mathcal{D},\hat{\boldsymbol{w}})\|}{\|\boldsymbol{w}-\hat{\boldsymbol{w}}\|} \le L_1\} \tag{28}$$

According to Assumption 1, there exists a δ such that $Pr\{\mathcal{E}_1\} \ge 1 - \frac{\delta}{3n_b}$. Moreover:

$$\begin{aligned} &\sup_{\boldsymbol{w}\in\Theta}\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}(\nabla f(X_j,\boldsymbol{w}) - \nabla f(X_j,\boldsymbol{w}_{j_l}))\| \\ &\le L_1\|\boldsymbol{w} - \boldsymbol{w}_{j_l}\| \le L_1\epsilon_l. \end{aligned} \tag{29}$$

By the triangle inequality:

$$\begin{aligned} &\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}_{j_l}) - \nabla F(\boldsymbol{w}_{j_l})\| \\ &\quad\le \|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*)\| \\ &\quad+ \|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}(\nabla f(X_j,\boldsymbol{w}_{j_l}) - \nabla f(X_j,\boldsymbol{w}^*)) \\ &\quad- (\nabla F(\boldsymbol{w}_{j_l}) - \nabla F(\boldsymbol{w}^*))\| \\ &\quad\le \|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*)\| \\ &\quad+ \|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}h(X_j,\boldsymbol{w}_{j_l}) - \mathbb{E}[h(X,\boldsymbol{w}_{j_l})]\|, \end{aligned} \tag{30}$$

where $\mathbb{E}[h(X,\boldsymbol{w})] = \nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w}^*)$.
Then events $\mathcal{E}_2$ and $\mathcal{E}_3(l)$ are defined as:

$$\begin{aligned}\mathcal{E}_2 &= \{\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*)\| \le 2\Delta_1^i\},\\ \mathcal{E}_3(l) &= \{\sup_{1\le k\le N_\epsilon}\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}h(X_j,\boldsymbol{w}_k) - \mathbb{E}[h(X,\boldsymbol{w}_k)]\|\\ &\quad \le 2\Delta_2^i\tau l\}.\end{aligned} \tag{31}$$

Since $\Delta_1^i \le \sigma_1^2/\kappa_1$, it follows from Lemma 3 that $Pr\{\mathcal{E}_2\} \ge 1-\delta/(3n_b)$. Similarly, based on Lemma 3 and [8], $Pr\{\mathcal{E}_3(l)\} \ge 1-\delta/(3n_b l^*)$. Therefore, on event $\mathcal{E}_1\bigcap\mathcal{E}_2\bigcap\mathcal{E}_3(l)$:

$$\begin{aligned}&\sup_{\boldsymbol{w}\in\Theta_l}\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\|\\ &\le L\epsilon_l + L_1\epsilon_l + 2\Delta_1^i + 2\Delta_2^i\tau l\\ &\overset{(a)}{\le} 2L_2\epsilon_l + 2\Delta_1^i + 2\Delta_2^i\tau l \overset{(b)}{\le} 4\Delta_2^i\tau l + 2\Delta_1^i,\end{aligned} \tag{32}$$

where (a) holds for $L+L_1 \le 2L_2$ and (b) is due to $L_2\epsilon_l \le \Delta_2^i\tau l$. Thus, according to the union bound, the probability is at least $1-\delta/n_b$ that event $\mathcal{E}_1\bigcap\mathcal{E}_2\bigcap(\bigcap_{l=1}^{l^*}\mathcal{E}_3(l))$ holds. Then on event $\mathcal{E}_1\bigcap\mathcal{E}_2\bigcap(\bigcap_{l=1}^{l^*}\mathcal{E}_3(l))$, for any $\boldsymbol{w}\in\Theta_{l^*}$, there exists an $1\le l\le l^*$ such that $(l-1)\tau \le \|\boldsymbol{w}-\boldsymbol{w}^*\| \le l\tau$ holds. If $l=1$, then:

$$\begin{aligned}\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\| &\le 4\Delta_2^i\tau + 2\Delta_1^i\\ &\overset{(a)}{\le} 4\Delta_1^i,\end{aligned} \tag{33}$$

where (a) holds for $\Delta_2^i \le \sigma_2^2/\kappa_2$ and $\Delta_1^i \ge \sigma_1\sqrt{s/|\mathcal{D}_i|}$. If $l\ge 2$ then $2(l-1)\ge l$ and the following inequality holds:

$$\begin{aligned}\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\| &\le 8\Delta_2^i\|\boldsymbol{w}-\boldsymbol{w}^*\|\\ &\quad + 2\Delta_1^i.\end{aligned} \tag{34}$$

By combining $l=1$ and $l\ge 2$:

$$\begin{aligned}\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\| &\le 8\Delta_2^i\|\boldsymbol{w}-\boldsymbol{w}^*\|\\ &\quad + 4\Delta_1^i.\end{aligned} \tag{35}$$

□

Lemma 5. *Suppose Assumptions 1 and 2 hold and based on Lemma 4, it can be deduced that, for any $\delta\in(0,1)$, if $\Delta_1^i \le (\sigma_1)^2/\kappa_1$ and $\Delta_2^i \le (\sigma_2)^2/\kappa_2$, the following holds for any $\boldsymbol{w}\in\Theta\subset\{\boldsymbol{w}:\|\boldsymbol{w}-\boldsymbol{w}^*\|\le r\sqrt{s}\}$:*

$$\begin{aligned}Pr\{\|\frac{1}{n_b}\sum_{i\in\mathcal{N}_b}\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\|\\ \le 8\Delta_2\|\boldsymbol{w}-\boldsymbol{w}^*\| + 4\Delta_1\}\\ \ge 1-\delta\end{aligned} \tag{36}$$

where $\Delta_2 = \frac{1}{n_b}\sum_{i\in\mathcal{N}_b}\Delta_2^i$, $\Delta_1 = \frac{1}{n_b}\sum_{i\in\mathcal{N}_b}\Delta_1^i$ and $\mathcal{N}_b$ is the set of benign clients whose number is n_b.

Proof.

$$\begin{aligned}&Pr\{\|\frac{1}{n_b}\sum_{i\in\mathcal{N}_b}\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\|\\ &\qquad > 8\Delta_2\|\boldsymbol{w}-\boldsymbol{w}^*\| + 4\Delta_1\}\\ &\le Pr\{\frac{1}{n_b}\sum_{i\in\mathcal{N}_b}\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\|\\ &\qquad > 8\Delta_2\|\boldsymbol{w}-\boldsymbol{w}^*\| + 4\Delta_1\}\\ &\le Pr\{\bigcup_{i\in\mathcal{N}_b}\|\frac{1}{|\mathcal{D}_i|}\sum_{X_j\in\mathcal{D}_i}\nabla f(\ X_j,\boldsymbol{w}) - \nabla F(\boldsymbol{w})\|\\ &\qquad > 8\Delta_2^i\|\boldsymbol{w}-\boldsymbol{w}^*\| + 4\Delta_1^i\}\\ &\le \sum_{i\in\mathcal{N}_b}\frac{\delta}{n_b} = \delta\end{aligned} \tag{37}$$

□

Theorem 1. *Suppose Assumption 1, 2, and 3 hold. The difference between the global model learned by RoseAgg and the optimal global model $\boldsymbol{w}^*$ under no attacks is bounded. Formally, the probability of the following inequality being satisfied is at least $1-\delta$:*

$$\|\boldsymbol{w}_t - \boldsymbol{w}^*\| \le (1-\rho)^t\|\boldsymbol{w}^0-\boldsymbol{w}^*\| + (4\Delta_1(G+1)+\Gamma)/\rho$$

$$\rho = 1-(\sqrt{(1+\alpha^2L^2-\alpha\mu)}+8\Delta_2(G+1)+L(G+1-\alpha)).$$

Proof. Using the lemmas above, the Theorem 1 can be proofed. For the t-th global iteration:

$$\begin{aligned}&\|\boldsymbol{w}_t-\boldsymbol{w}^*\|\\ &= \|\boldsymbol{w}_{t-1}-\boldsymbol{g}^{t-1}-\boldsymbol{w}^*\|\\ &= \|\boldsymbol{w}_{t-1}-\alpha\nabla F(\boldsymbol{w}_{t-1})-\boldsymbol{w}^*+\alpha\nabla F(\boldsymbol{w}_{t-1})-\boldsymbol{g}^{t-1}\|\\ &\le \|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*-\alpha\nabla F(\boldsymbol{w}_{t-1})\|\\ &\quad + \|\nabla F(\boldsymbol{w}_{t-1})-\boldsymbol{g}^{t-1}+(\alpha-1)\nabla F(\boldsymbol{w}_{t-1})\|\\ &\le \|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*-\alpha\nabla F(\boldsymbol{w}_{t-1})\|\\ &\quad + \|\nabla F(\boldsymbol{w}_{t-1})-\boldsymbol{g}^{t-1}\| + \|(1-\alpha)\nabla F(\boldsymbol{w}_{t-1})\|\\ &\le \|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*-\alpha\nabla F(\boldsymbol{w}_{t-1})\| + (G+1)\|\boldsymbol{g}_0^{t-1}-\nabla F(\\ &\quad \boldsymbol{w}_{t-1})\| + (G+1-\alpha)\|\nabla F(\boldsymbol{w}_{t-1})\| + \Gamma\\ &\le \sqrt{(1+\alpha^2L^2-\alpha\mu)}\|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*\|\\ &\quad + (G+1)(8\Delta_2\|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*\|+4\Delta_1)\\ &\quad + (G+1-\alpha)\|\nabla F(\boldsymbol{w}_{t-1})-\nabla F(\boldsymbol{w}^*)\| + \Gamma\\ &\le \sqrt{(1+\alpha^2L^2-\alpha\mu)}\|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*\|\\ &\quad + (G+1)(8\Delta_2\|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*\|\\ &\quad + 4\Delta_1) + L(G+1-\alpha)\|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*\| + \Gamma\\ &\le (\sqrt{(1+\alpha^2L^2-\alpha\mu)} + 8\Delta_2(G+1) + L(G+1-\alpha))\\ &\quad \|\boldsymbol{w}_{t-1}-\boldsymbol{w}^*\| + 4\Delta_1(G+1) + \Gamma\end{aligned} \tag{38}$$

Recursively applying the inequality (38) for T global iterations, the following conclusions can be drawn:

$$\|\boldsymbol{w}_T - \boldsymbol{w}^*\| \leq (1-\rho)^T \|\boldsymbol{w}_0 - \boldsymbol{w}^*\| + (4\Delta_1(G+1)+\Gamma)/\rho, \tag{39}$$

□

When $|1-\rho| < 1$, the following conclusion can be drawn: $\lim_{T\to\infty} \|\boldsymbol{w}_T - \boldsymbol{w}^*\| \leq (4\Delta_1(G+1)+\Gamma)/\rho$. After undergoing several training rounds, the discrepancy between the global model obtained through RoseAgg and the non-attacked global model acquired through FedAVG will be confined to a relatively narrow range. In other words, RoseAgg is robust to targeted collusion attacks in FL.

VII. Experimental Results

This section presents a thorough evaluation of RoseAgg on four benchmark datasets, aiming to illustrate its effectiveness. The results substantiate its success in defending against state-of-the-art targeted model poisoning attacks [2], [36], [3], [32], [40], [20] (as described in SectionII-B) in different settings. Meanwhile, a comprehensive comparison is conducted between RoseAgg and state-of-the-art defenses [11], [24], [6], [23] (as described in Section II-C) against these attacks.

A. Experimental Setup

Datasets: Four benchmark image classification datasets are empolyed in our evaluation to illustrate the effectiveness of RoseAgg. Following the previous work [36], which utilizes a Dirichlet distribution with a hyperparameter of 0.5 to partition the training data. Specifically, the trainning data is partitioned into 500 parties, namely, a total of 500 clients. During collaborative training of a global model, 100 clients are selected in each communication round.

CIFAR10 [15]**:** The Canadian Institute For Advanced Research (CIFAR10) dataset contains a small part of tiny images associated with 10 natural categories. It consists of 60,000 total samples, which are split into training data consisting of 50,000 samples and testing data consisting of 10,000 samples.

*CIFAR100***:** The CIFAR100 dataset also contains a small part of tiny images like CIFAR10, whereas it has 100 categories, each of which consists of 600 samples. Besides, these 100 categories are divided into 20 super-categories. Thus, each sample in the CIFAR100 dataset has a "fine" label (the category it belongs to) and a "coarse" label (the super-category it belongs to).

EMNIST [9]**:** The EMNIST dataset is a set of handwritten character digits derived from the NIST Special Database 19 and converted to a 28x28 pixel image format and dataset structure that directly matches the MNIST dataset.

FMNIST [34]**:** The FMNIST dataset consists of Zalando's article images that are related to 10 commodity categories. It is comprised of 70,000 total samples, which are split into training data consisting of 60,000 samples and testing data consisting of 10,000 samples.

Attack Assumption: RoseAgg is designed to mitigate targeted collusion attacks where the proportion of malicious clients and attack density are both high. To this end, the default setting establishes an attacker's ratio of 0.5 in each adversarial round, and each communication round is designated as an adversarial round with a probability of 0.5. Additionally, multi-objective targeted model poisoning attacks can be orchestrated by the adversary $\mathcal{A}$ during adversarial rounds. For instance, $\mathcal{A}$ can inject multiple distinct backdoors to enhance attack success rates. In the process of locally constructing a poisoned dataset, each malicious client initiates by randomly sampling clean data from the training dataset. Subsequently, they proceed to inject targeted poisoned data samples into the preexisting clean data.

Evaluation Metrics: Two metrics are primarily considered for evaluating the effectiveness of targeted model poisoning attacks and defense methods: **ASR** - *attack success rate* indicates that the probability of the adversary $\mathcal{A}$ successfully inducing the global model to produce targeted misprediction. $\mathcal{A}$ aims to maximize ASR while an effective defense aims to prevent ASR from increasing. **TER** - *test error rate* indicates the error rate predicted by the global model on the test set. $\mathcal{A}$'s objective is to minimize the impact on the model's test error rate in order to avoid easy detection. An effective defense should not affect the TER of the global model.

B. Effectiveness of Dynamic Clustering Component

Dynamic clustering is a crucial component of RoseAgg, involving the application of DBSCAN clustering to local updates. The subsequent stages of the pipeline are significantly influenced by the quality of the clustering result. Hence, the evaluation of this clustering step is prioritized in this section. In the evaluation of the dynamic clustering component, four metrics are considered for assessing its effectiveness. 1) #PC: the number of "poisoned clusters", where a cluster is considered as "poisoned" if it contains poisoned updates. 2) #BC: the number of "benign clusters", where a cluster is considered as "benign" if it does not include poisoned updates. 3) PCR: the proportion of "poisoning clusters" to all poisoning updates. A smaller PCR indicates a better clustering effect, as fewer "poisoned clusters" result in a smaller accumulation of credit scores in subsequent steps, leading to a diminished contribution of poisoned updates to the global model update. 4) BCR, the proportion of "benign clusters" to all benign updates. A larger BCR signifies a better clustering effect, as larger "benign clusters" lead to a greater accumulation of credit scores in subsequent steps, enlarging the contribution of benign updates to the global model update.

Table III demonstrates the results in a specific adversarial round. The dynamic clustering component consistently inclines towards consolidating poisoned updates into a limited number of clusters or a singular cluster. This phenomenon arises from the shared malicious goal among attackers, leading to a pronounced similarity in the feature spaces relevant to the malicious goal for each attacker. The measurement of this similarity is facilitated by calculating the pairwise cosine distance among local model updates.

C. Prevent Targeted Model Poisoning Attacks

A comparative analysis is conducted between RoseAgg and several existing defense mechanisms, namely RLR [24],

TABLE II
EFFECTIVENESS OF ROSEAGG AND RECENT DEFENSES AGAINST STATE-OF-THE-ART ATTACKS FOR THE RESPECTIVE DATASET, IN TERMS OF ATTACK SUCCESS RATE (ASR) AND TEST ERROR RATE (TER). ALL METRIC VALUES ARE REPORTED AS PERCENTAGES.

Attack	Dataset	Avg		RLR		FoolsGold		FLTrust		FLAME		RoseAgg	
		ASR	TER	ASR	TER	ASR	TER	ASR	TER	ASR	TER	ASR	TER
MR	CIFRA10	100	11.2	100	16.7	0	13.1	85.7	12.5	57.0	13.7	**0**	**10.4**
	EMNIST	100	1.9	98.4	**1.5**	0.78	1.7	2.34	1.7	0.98	2.1	**0.59**	1.69
	FMNIST	100	11.2	100	11.1	0	10.6	0	10.6	38.5	10.6	**0**	**10.4**
DBA	CIFRA10	100	10.9	82.4	12.4	100	11.6	27.3	12.5	3.37	13.4	**3.32**	**10.4**
	EMNIST	99.6	2.7	39	**1.7**	99.8	3.7	0.59	1.9	1.56	2.1	**0.39**	1.97
	FMNIST	100	11.4	100	11.3	100	13.7	0	10.5	0	10.5	**0**	**10.3**
FLIP	CIFRA10	100	20.8	100	20.5	**2.59**	12.2	64.5	21.4	97.9	22.4	3.13	**10.4**
	EMNIST	100	19.3	99.2	12.8	0	1.86	61.5	8.29	0	2.0	**0**	**1.68**
	FMNIST	98.8	20.1	98.2	20	1.95	10.6	3.52	10.6	97.9	19.9	**1.95**	**10.5**
EDGE	CIFRA10	97.5	12.7	99.0	24.1	**1.53**	13.0	54.6	13.1	5.61	13.8	3.57	**10.5**
	EMNIST	99	1.9	98	1.78	0	1.95	23	2.16	99	2.23	**0**	**1.67**
	FMNIST	99.0	11.2	99.0	11.2	2.05	10.5	6.00	10.7	99	12.0	**1.00**	**10.5**
NEUR	CIFRA10	96.5	13.1	97.4	15.4	2.04	11.6	46.9	12.8	**1.02**	13.5	1.57	**10.6**
	EMNIST	100	2.03	92	1.67	100	1.89	13	2.34	1.00	2.07	**0**	**1.67**
	FMNIST	99.0	11.0	99.0	11.3	41	10.6	2.25	10.7	97	10.7	**2.00**	**10.4**

TABLE III
EFFECTIVENESS OF DYNAMIC CLUSTERING COMPONENT. THE VALUES OF PCR AND BCR ARE IN PERCENTAGE.

Attack	MR	DBA	FLIP	EDGE	NEUR	COMBINE
#PC	1	10	1	1	1	4
#BC	36	44	39	38	40	28
PCR	2	76.9	2	2	2	8
BCR	72	50.6	78	76	80	56

FoolsGold [11], FLTurst [6], and FLAME [23], against state-of-the-art targeted model poisoning attacks including MR [2], DBA [36], FLIP [3], EDGE [32], and NEUR [40]. These evaluations are performed on three benchmark datasets. To assess the effectiveness of each defense approach, a global model is initially trained using the respective algorithm until it approaches convergence. Subsequently, an additional 60 rounds of experiments are conducted. During each round, the introduction of malicious clients is based on the aforementioned attack assumption. The largest attack success rate (ASR) in the 60 rounds and the corresponding test error rate (TER) are recorded and presented in Table II. Finally, a thorough analysis is then undertaken to assess the effectiveness of each defense method in mitigating the impact of targeted model poisoning attacks.

Table II demonstrates the robustness of RoseAgg against all types of targeted model poisoning attacks in the respective dataset. Furthermore, RoseAgg ensures that the performance of the global model is not compromised. This is achieved by considering the contribution of all local model updates during the model aggregation process on the server.

However, RLR is not effective in preventing most types of attacks. This is due to the assumption that the number of malicious clients remains significantly below a threshold value θ. Once the number of malicious clients surpasses this threshold, the model deviates from the benign direction and attempts to move toward the malicious direction, resulting in a failure of defense.

While FoolsGold can prevent most types of attacks, it fails to defend against DBA. Since DBA spreads the attacks across different rounds, FoolsGold struggles to distinguish between poisoned and benign model updates effectively. Consequently, DBA can successfully evade the detection and defense mechanisms of FoolsGold, leading to specific mispredictions.

Furthermore, FLTrust lacks robustness on the CIFAR10 dataset. The substantial divergence between the distribution of the root set on the server and the overall data distribution hinders FLTrust's ability to calculate and maintain a representative model update. As a result, FLTrust may mistakenly remove benign local updates while retaining malicious ones.

In the case of FLAME, when the number of malicious clients is high, a significant portion of poisoned model updates may be erroneously clustered together with benign model updates. Consequently, this clustering process may drive the model towards a malicious direction during the aggregation phase. Furthermore, FLAME introduces noise into the model, thereby deteriorating the accuracy of the global model.

D. Prevent Multi-objective Attacks

To enhance the success rate of the attack, the adversary $\mathcal{A}$ can employ a multi-objective targeted model poisoning attack. For instance, $\mathcal{A}$ can inject multiple backdoors simultaneously by utilizing distinct client groups [23]. The poisoned model updates generated in an adversarial round do not display typical patterns and are challenging to differentiate from benign model updates. Consequently, preventing multi-objective attacks becomes a formidable task.

We conduct an evaluation to assess the effectiveness of RoseAgg and other defense mechanisms against a multi-objective attack on the CIFAR10 dataset. The defense effect is presented in Table IV. This evaluation encompasses four malicious objectives within a single round of attack. The objectives, denoted as O1, O2, O3, and O4', indicate that $\mathcal{A}$ intends to misclassify examples containing a pixel-pattern trigger "F", "L", "A", and "I" as classes "0", "1", "6", and "2", respectively. Objective O4 represents the attacker's goal of misclassifying images of cars with vertically striped walls as birds.

Table IV illustrates the robustness of all defenses against multi-objective targeted model poisoning attacks. Due to the

TABLE IV
EFFECTIVENESS OF ROSEAGG AND RECENT DEFENSES AGAINST MULTI-OBJECTIVE TARGETED MODEL POISONING ATTACKS FOR THE RESPECTIVE DATASET, IN TERMS OF ATTACK SUCCESS RATE (ASR) AND TEST ERROR RATE (TER). ALL METRIC VALUES ARE REPORTED AS PERCENTAGES.

Dataset	Objective	Avg		RLR		FoolsGold		FLTrust		FLAME		RoseAgg	
		ASR	TER	ASR	TER	ASR	TER	ASR	TER	ASR	TER	ASR	TER
CIFAR10	O1	100	10.6	100	14.6	52.7	11.6	3.32	12.4	**1.95**	13.6	2.15	**10.5**
	O2	98.6		49		10.5		1.17		0.98		**0.39**	
	O3	99.4		68		12.4		2.73		0.39		**0.39**	
	O4	94.9		31		1.98		0		0.39		**0**	
EMNIST	O1	100	1.30	54.9	1.60	100	**1.28**	0.78	2.91	**0.19**	2.05	0.59	1.97
	O2	99.0		0.20		98.0		0		0		**0**	
	O3	99.8		0		99.2		4.1		0.39		**0**	
	O4'	97.3		**0**		96.5		0.59		1.17		0.59	
FMNIST	O1	100	11.3	100	11.2	100	10.6	**0.97**	10.8	1.17	10.5	1.17	**10.4**
	O2	79.9		69.5		71.8		0.20		0.20		**0.20**	
	O3	68.7		66.2		76.1		2.35		1.96		**1.76**	
	O4'	86.2		78.9		92.1		0.39		0.40		**0.39**	

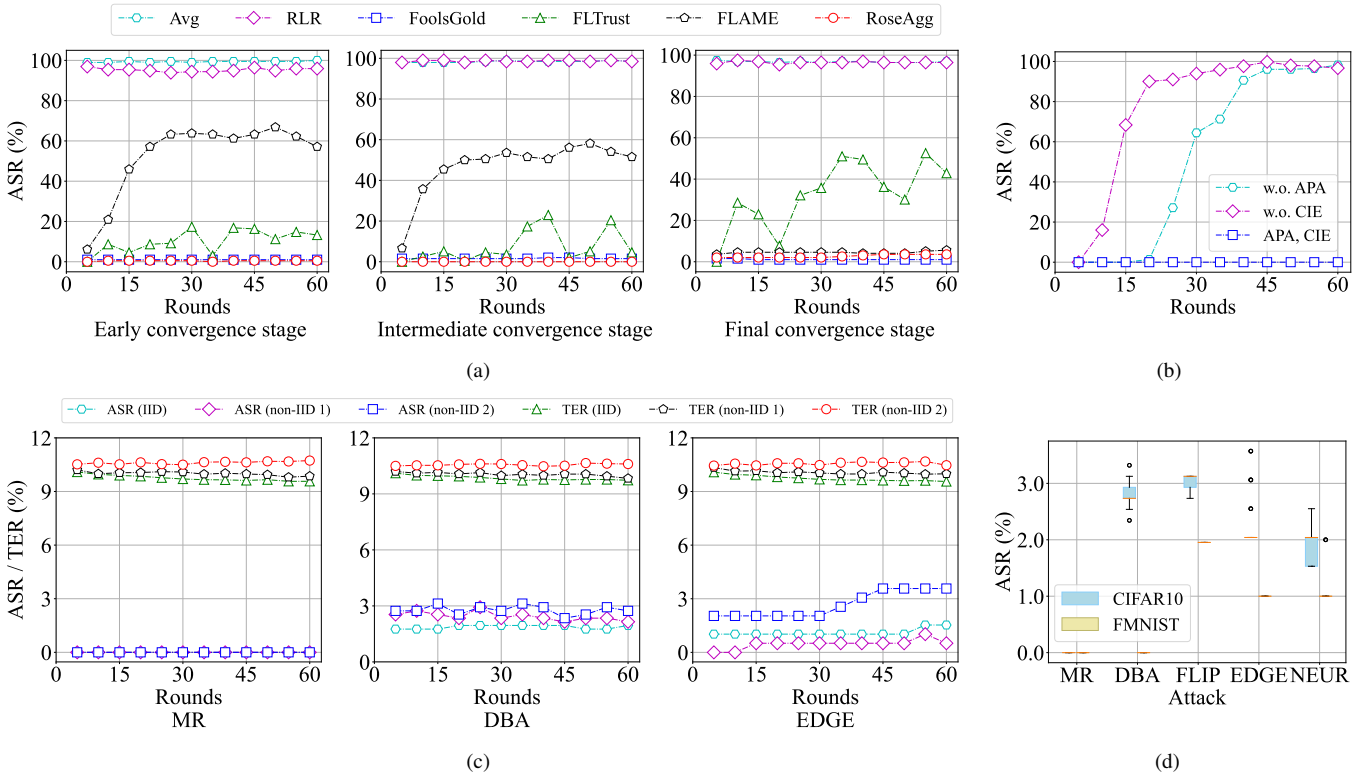

Fig. 5. Defense against state-of-the-art targeted model poisoning attacks. (a) Attack success rate of "EDGE" attack at different convergence stages. (b) Effectiveness of RoseAgg's components. Specifically, the "w.o. APA" and "w.o. CIE" configurations involve disabling the adaptive partial aggregating and clean ingredient extraction modules. (c) Attack success rate and test error rate of RoseAgg in defending against different attacks under varying degrees of non-IID data among clients. The hyperparameter α of the Dirichlet distribution adapted by non-IID 1 and non-IID 2 is 0.9 and 0.5, respectively. (d) Stability of RoseAgg.

distinct targets of each malicious client, the poisoned model update gradients exhibit a weak correlation in direction. Although the adaptive partial aggregating module cannot effectively mitigate the impact of potentially malicious model updates from a directional perspective, the clean ingredient extraction module compensates for this limitation. Given that the number of "poisoned clusters" is smaller than that of "benign clusters", the clean ingredient extraction module ensures that the summation of projection variances of benign local model updates surpasses that of poisoned local model updates. This implies that the influence of benign model updates outweighs that of malicious model updates, thereby guiding the model toward the benign direction and mitigating the impact of targeted model poisoning attacks.

FoolsGold fails to effectively prevent multi-objective targeted model poisoning attacks. The reason is that poisoned model updates within a single round lack common characteristics, making it challenging for FoolsGold to distinguish between poisoned and benign model updates. Consequently, the defense mechanism of FoolsGold is rendered ineffective. RLR is not robust to multi-objective targeted model poisoning attacks, as the number of malicious clients surpasses the threshold allowed by RLR, resulting in defense failure. Additionally, while FLTrust and FLAME successfully mitigate

multi-objective attacks, they do experience varying degrees of accuracy loss under specific scenarios, such as on the CIFAR10 dataset.

E. Defense at Different Convergence Stages

Malicious clients can be selected at any stage of federated learning. Therefore, studying targeted model poisoning attacks and defenses at different stages provides insights into the generalizability of defense mechanisms. Defense techniques that perform well at one stage of convergence may exhibit different efficacy at other stages. Hence, it is crucial to assess the efficacy of defenses at various convergence stages of the global model. Figure 5(a) demonstrates the defensive impact of RoseAgg and other defenses in countering the "EDGE" attack at different rounds on the CIFAR10 dataset.

RoseAgg and FoolsGold demonstrate robustness throughout the convergence stages of the global model. In contrast, RLR and FLTrust fail to effectively prevent targeted model poisoning attacks during the entire process of federated training. Furthermore, FLAME fails to prevent targeted model piosoning attacks at the early and intermediate stages of convergence. This observation highlights the adversary's ability to evade defense measures different stages. Hence, evaluating the effectiveness of defense approaches during different convergence stages becomes essential when developing countermeasures against targeted model poisoning attacks.

F. Impact of the Degree of non-IID Data

The adaptive partial aggregating module of RoseAgg incorporates the DBSCAN algorithm, which clusters local updates based on cosine distances between pairwise updates. However, the distribution of data among clients can influence the cluster results, potentially impacting the effectiveness of RoseAgg. To assess of the effectiveness of RoseAgg under varying degrees of non-IID data among clients, three experiments are performed on the CIFAR10 dataset. These experiments demonstrate the effectiveness of RoseAgg against "MR", "DBA", and "EDGE" attacks. Following previous works [2], [36], the degree of non-IID data across clients is manipulated by adjusting the α parameter of the Dirichlet distribution.

Figure 5(c) presents the evaluation results of RoseAgg in defending against "MR", "DBA", and "EDGE" attacks. The results demonstrate that RoseAgg effectively safeguards against targeted model poisoning attacks under varying degrees of non-IID data among clients. The success of RoseAgg can be attributed to its ability to identify and address common characteristics shared by malicious clients. These shared characteristics often involve similarities in direction or magnification of the magnitude. Fortunately, the adaptive partial aggregating module within RoseAgg mitigates these common characteristics, thereby neutralizing their impact. Furthermore, the calculation of the clean ingredient ensures that the cumulative weight of benign local model updates surpasses that of malicious model updates. Consequently, the model becomes biased in a favorable direction, effectively preventing targeted model poisoning attacks.

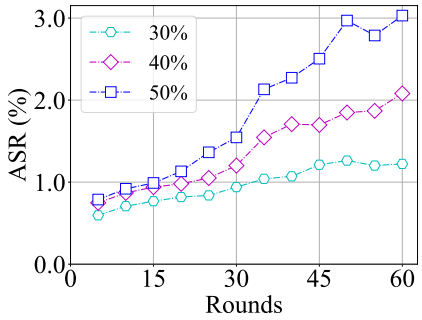

Fig. 6. Impact of the number of malicious clients. "30%", "40%", and "50%" represent the proportion of malicious clients is 30%, 40%, and 50%.

G. Stability of RoseAgg

The stability of RoseAgg across repeated randomized experiments holds paramount significance in establishing the reliability and robustness of its performance. Through multiple iterations of randomized trials, RoseAgg consistently demonstrates its effectiveness in defending against targeted model poisoning attacks. This stability implies that the algorithm's success is not contingent on specific, isolated conditions but persists across various scenarios. Specifically, the utilization of boxplots in Figure 5(d) provides a clear visualization of the stability of RoseAgg. Various attacks are applied across the CIFAR10 and FMNIST dataset, with the experiment replicated 30 times. The observed discrepancy between the maximum and minimum ASR remains within a narrow margin, *i.e.*, not surpassing 1.57%.

H. Effectiveness of RoseAgg's Components

We analyze and evaluate each individual component of RoseAgg to assess its contribution toward its defense. Figure 5(b) provides an intuitive evaluation result for each component. Firstly, we disable the adaptive partial aggregating module and compare the effectiveness of defense with the full implementation of RoseAgg. In this scenario, we observe that RoseAgg fails to effectively defend against "MR" attacks. Without the adaptive partial aggregating module, RoseAgg is unable to identify and mitigate the impact of local model updates with amplified magnitudes or similar directional contributions. Unfortunately, these characteristics are commonly found in most poisoned model updates. Secondly, we disable the clean ingredient extraction module and instead directly calculate an average of the model updates handled by the adaptive partial aggregating module. Similarly, RoseAgg fails to provide adequate defense against "MR" attacks. The simplistic averaging algorithm fails to ensure that the cumulative weight of benign local model updates surpasses that of the poisoned model updates, ultimately leading to defense failure.

I. Impact of Number of Malicious Clients

Furthermore, in alignment with the core motivations of this paper, additional experiments have been incorporated to investigate the impact of varying the number of malicious

clients. This deliberate variation serves to delve deeper into the nuanced dynamics and robustness of RoseAgg under different scenarios. Specifically, Figure 6 illustrates experiments conducted on the CIFAR-100 dataset, evaluating RoseAgg's defense against CerP by varying the percentage of malicious clients. As the proportion of malicious clients increases, an upward trend in the ASR is observed, yet it consistently remains within a relatively narrow range, hovering around three percent.

J. Computational Complexity Analysis

The computational complexity of RoseAgg, particularly in comparison to the widely used Federated Averaging (FedAVG), is a critical aspect of evaluating its practical viability. RoseAgg mainly introduces additional steps in the form of DBSCAN clustering and a PCA process to enhance its defense against targeted model poisoning attacks. Therefore, the computational complexity analysis of RoseAgg involves assessing the additional time complexity introduced by the clustering using DBSCAN and the PCA process.

In the DBSCAN clustering, the calculation of the distance matrix and neighbor query dominates the time complexity, resulting in $O(\mathcal{M}^2 d)$, where d is the dimensionality of the local updates. Additionally, the PCA process introduces time complexity mainly through Singular Value Decomposition (SVD), which is not more than $O(d^2\mathcal{M})$. In the context of general federated learning, the dimension of deep neural networks is significantly larger than the number of participants. Consequently, the extra time complexity of RoseAgg compared to FedAVG is approximately $O(d^2\mathcal{M})$. It's important to note that both DBSCAN clustering and PCA operate separately according to the neural network hierarchy. Consequently, the combined time complexity is reduced to $O(P^2\mathcal{M})$, where P is the number of parameters in the layer with the most parameters. In the context of federated processes, the additional time complexity introduced by RoseAgg is deemed negligible compared to the computational demands of training deep neural networks. Therefore, RoseAgg remains computationally feasible for practical applications.

VIII. Conclusion

In this paper, RoseAgg is introduced as a defense mechanism against targeted model poisoning attacks, ensuring effective defense without compromising the global model's performance. An adaptive partial aggregating strategy is devised to identify and restrict common characteristics found in poisoned model updates. Furthermore, a dimension-reduction method is employed to analyze a plausible clean ingredient, facilitating the bootstrapping of credit scores for each local update. Finally, the consideration of both potentially malicious and benign local updates ensures the preservation of global model performance. The effectiveness of RoseAgg is theoretically illustrated, and extensive evaluations demonstrate its robustness against seven advanced attacks, even under conditions of a relatively high proportion of malicious clients and attack density.

References

[1] Sebastien Andreina, Giorgia Azzurra Marson, Helen Möllering, and Ghassan Karame. Baffle: Backdoor detection via feedback-based federated learning. In *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*, pages 852–863. IEEE, 2021.

[2] 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.

[3] Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. Analyzing federated learning through an adversarial lens. In *International Conference on Machine Learning*, pages 634–643. PMLR, 2019.

[4] 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.

[5] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečnỳ, Stefano Mazzocchi, Brendan McMahan, et al. Towards federated learning at scale: System design. *Proceedings of machine learning and systems*, 1:374–388, 2019.

[6] Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. Fltrust: Byzantine-robust federated learning via trust bootstrapping. In *ISOC Network and Distributed System Security Symposium (NDSS)*, 2021.

[7] Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. Provably secure federated learning against malicious clients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 6885–6893, 2021.

[8] Yudong Chen, Lili Su, and Jiaming Xu. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 1(2):1–25, 2017.

[9] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. In *2017 international joint conference on neural networks (IJCNN)*, pages 2921–2926. IEEE, 2017.

[10] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.

[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.

[12] 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.

[13] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[14] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020.

[15] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[16] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60, 2020.

[17] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60, 2020.

[18] Xiaoyuan Liu, Hongwei Li, Guowen Xu, Zongqi Chen, Xiaoming Huang, and Rongxing Lu. Privacy-enhanced federated learning against poisoning adversaries. *IEEE Transactions on Information Forensics and Security*, 16:4574–4588, 2021.

[19] Lingjuan Lyu, Han Yu, Jun Zhao, and Qiang Yang. Threats to federated learning. *Federated Learning: Privacy and Incentive*, pages 3–16, 2020.

[20] 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 *Thirty-Seventh AAAI Conference on Artificial Intelligence*, 2023.

[21] Zhuoran Ma, Jianfeng Ma, Yinbin Miao, Yingjiu Li, and Robert H Deng. Shieldfl: Mitigating model poisoning attacks in privacy-preserving federated learning. *IEEE Transactions on Information Forensics and Security*, 17:1639–1654, 2022.

[22] 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.

[23] 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.

[24] 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*, volume 35, pages 9268–9276, 2021.

[25] Felix Sattler, Simon Wiedemann, Klaus-Robert Müller, and Wojciech Samek. Robust and communication-efficient federated learning from non-iid data. *IEEE transactions on neural networks and learning systems*, 31(9):3400–3413, 2019.

[26] Virat Shejwalkar, Amir Houmansadr, Peter Kairouz, and Daniel Ramage. Back to the drawing board: A critical evaluation of poisoning attacks on production federated learning. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1354–1371. IEEE, 2022.

[27] Shiqi Shen, Shruti Tople, and Prateek Saxena. Auror: Defending against poisoning attacks in collaborative deep learning systems. In *Proceedings of the 32nd Annual Conference on Computer Security Applications*, pages 508–519, 2016.

[28] Jingwei Sun, Ang Li, Louis DiValentin, Amin Hassanzadeh, Yiran Chen, and Hai Li. Fl-wbc: Enhancing robustness against model poisoning attacks in federated learning from a client perspective. *Advances in Neural Information Processing Systems*, 34:12613–12624, 2021.

[29] Ziteng Sun, Peter Kairouz, Ananda Theertha Suresh, and H Brendan McMahan. Can you really backdoor federated learning? *arXiv preprint arXiv:1911.07963*, 2019.

[30] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

[31] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press, 2019.

[32] 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.

[33] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

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

[35] Chulin Xie, Minghao Chen, Pin-Yu Chen, and Bo Li. Crfl: Certifiably robust federated learning against backdoor attacks. In *International Conference on Machine Learning*, pages 11372–11382. PMLR, 2021.

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

[37] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[38] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*, pages 5650–5659. PMLR, 2018.

[39] Zaixi Zhang, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. 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*, pages 2545–2555, 2022.

[40] 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.

He Yang received the B.S. degree in computer science from Central South University, Changsha, China, in 2016. He is currently pursuing the Ph.D. degree with Xi'an Jiaotong University, Xi'an, China. His current research interests include federated learning, data privacy and security, and artificial intelligence.

Wei Xi is currently an Associate Professor at the School of Computer Science and Technology, Xi'an Jiaotong University. He received his Ph.D degree on Computer Science from Xi'an Jiaotong University in 2014. His main research interests include Internet of things, artificial intelligence, and network security. He is a member of CCF, ACM, and IEEE.

Yuhao Shen received the B.S. degree in Material Science and Technology from Dalian University of Technology, Dalian, China, in 2020. He is currently pursuing the Ph.D. degree in Computer Science and Technology at Xi'an Jiaotong University, Xi'an China. His research interests include machine learning and federated learning.

Canhui Wu received the B.S. degree in computer science from Xi'an Jiaotong University, Xi'an, China, in 2023. He is currently pursuing the M.S. degree with Xi'an Jiaotong University, Xi'an, China. His current research interests include federated learning, data privacy and security, and artificial intelligence.

Jizhong Zhao is a Professor at the Department of Computer Science and Technology, Xi'an Jiaotong University. His research interests include computer software, pervasive computing, distributed systems, and network security. He is a member of CCF, ACM, and IEEE.