CISTER
Research Centre in
Real-Time & Embedded
Computing Systems

Journal Paper

Data-Agnostic Model Poisoning against Federated Learning: A Graph Autoencoder Approach

Kai Li*

Jingjing Zheng*

Xin Yuan

Wei Ni

Ozgur B. Akan

H. Vincent Poor

*CISTER Research Centre

CISTER-TR-240201

2024

Data-Agnostic Model Poisoning against Federated Learning: A Graph Autoencoder Approach

Kai Li*, Jingjing Zheng*, Xin Yuan, Wei Ni, Ozgur B. Akan, H. Vincent Poor

*CISTER Research Centre

Polytechnic Institute of Porto (ISEP P.Porto)

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8321159

E-mail: kai@isep.ipp.pt, zheng@isep.ipp.pt, xin.yuan@data61.csiro.au, Wei.Ni@data61.csiro.au, oba21@cam.ac.uk, poor@princeton.edu

https://www.cister-labs.pt

Abstract

This paper proposes a novel, data-agnostic, model poisoning attack on Federated Learning (FL), by designing a new adversarial graph autoencoder (GAE)-based framework. The attack requires no knowledge of FL training data and achieves both effectiveness and undetectability. By listening to the benign local models and the global model, the attacker extracts the graph structural correlations among the benign local models and the training data features substantiating the models. The attacker then adversarially regenerates the graph structural correlations while maximizing the FL training loss, and subsequently generates malicious local models using the adversarial graph structure and the training data features of the benign ones. A new algorithm is designed to iteratively train the malicious local models using GAE and sub-gradient descent. The convergence of FL under attack is rigorously proved, with a considerably large optimality gap. Experiments show that the FL accuracy drops gradually under the proposed attack and existing defense mechanisms fail to detect it. The attack can give rise to an infection across all benign devices, making it a serious threat to FL.

Data-Agnostic Model Poisoning against Federated Learning: A Graph Autoencoder Approach

Kai Li, *Senior Member, IEEE,* Jingjing Zheng, *Student Member, IEEE,* Xin Yuan, *Member, IEEE,* Wei Ni, *Fellow, IEEE,* Ozgur B. Akan, *Fellow, IEEE,* and H. Vincent Poor, *Life Fellow, IEEE*

***Abstract*—This paper proposes a novel, data-agnostic, model poisoning attack on Federated Learning (FL), by designing a new adversarial graph autoencoder (GAE)-based framework. The attack requires no knowledge of FL training data and achieves both effectiveness and undetectability. By listening to the benign local models and the global model, the attacker extracts the graph structural correlations among the benign local models and the training data features substantiating the models. The attacker then adversarially regenerates the graph structural correlations while maximizing the FL training loss, and subsequently generates malicious local models using the adversarial graph structure and the training data features of the benign ones. A new algorithm is designed to iteratively train the malicious local models using GAE and sub-gradient descent. The convergence of FL under attack is rigorously proved, with a considerably large optimality gap. Experiments show that the FL accuracy drops gradually under the proposed attack and existing defense mechanisms fail to detect it. The attack can give rise to an infection across all benign devices, making it a serious threat to FL.**

***Index Terms*—Federated learning, model poisoning attack, graph autoencoder, feature correlation.**

I. Introduction

The use of mobile edge computing is increasingly prevalent, especially in catering to user devices that come with a multitude of sensors. These sensors produce vast amounts of data, like images recording human activities or the real-time locations of vehicles, as seen in smart city scenarios [1], [2]. However, transferring this training data from the user's device to a server can pose a threat to data privacy leakage. Federated Learning (FL) is an emerging distributed machine learning approach that gains traction as a solution to mitigate data privacy concerns [3]. With FL, user devices can jointly train a machine learning model without having to disclose their private data to a server. The user devices, acting as clients, iteratively train their local models on their private data and send the local model updates to a server. At the server, a global model is updated without collecting private data from the user devices. The global model is then sent back to the user devices, allowing them to continue training their local models based on the global model and their local data [4]. This process helps to support data privacy and allows for real-time processing capabilities at the edge of networks, making FL a significant aspect of mobile edge computing.

Despite the fact that FL can help prevent attackers from accessing the private data of user devices, an attacker (in most cases, a malicious user device) can potentially launch model poisoning or data poisoning attacks to manipulate FL and propagate the attacks into benign user devices [5], [6], resulting in a failure of FL training. Specifically, model poisoning aims to send malicious local model updates to the server during an aggregation process. The malicious update can introduce specific vulnerabilities in the global model or simply degrade FL performance. By contrast, data poisoning attempts to inject malicious data or modify existing data on user devices to misguide local model training, thus compromising local model updates. Existing data poisoning attacks generally require an attacker to have some knowledge of the datasets used for FL training [7], so that it can extract and manipulate the features of the datasets for effective attacks [8]. By launching model poisoning attacks [9] or data poisoning attacks [10], an attacker could manipulate either the hyperparameters of the local models or the training datasets of benign users to compromise learning accuracy.

Much less constrained and potentially more threatening model poisoning attacks on FL would result if they could be based solely on the benign local models overheard by an attacker and the global models broadcast by the aggregator; i.e., when the attacker has no access to the training data. However, without training data, it is challenging for the malicious local models to strike a balance between effectiveness and undetectability [11]. To the best of our knowledge, such attacks are new and have not been reported in the literature.

In this paper, we propose a new, data-agnostic, model poisoning attack on FL systems, where an adversarial graph autoencoder (GAE) [12], [13] is designed to generate malicious local models solely based on the benign local

K. Li is with the Department of Engineering, University of Cambridge, CB3 0FA Cambridge, U.K., and also with Real-Time and Embedded Computing Systems Research Centre (CISTER), Porto 4249–015, Portugal (E-mail: kaili@ieee.org).

J. Zheng is with CyLab Security and Privacy Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA, and also with Real-Time and Embedded Computing Systems Research Centre (CISTER), Porto 4249–015, Portugal (E-mail: zheng@isep.ipp.pt).

X. Yuan and W. Ni are with the Digital Productivity and Services Flagship, Commonwealth Scientific and Industrial Research Organization (CSIRO), Sydney, NSW 2122, Australia (E-mail: {xin.yuan,wei.ni}@data61.csiro.au).

O. B. Akan is with the Division of Electrical Engineering, Department of Engineering, University of Cambridge, CB3 0FA Cambridge, U.K., and also with the Center for NeXt-Generation Communications (CXC), Koç University, 34450 Istanbul, Turkey (E-mail: oba21@cam.ac.uk).

H. V. Poor is with the Department of Electrical and Computer Engineering, Princeton University, Princeton, NJ 08544, USA (E-mail: poor@princeton.edu).

models overheard and capturing the correlation features of the benign local and global models. Specifically, an attacker overhears the benign local models uploaded by the user devices, and the global model broadcast by the server. GAE is adept at capturing complex relationships and structures inherent in graph-structured data. It can efficiently encode graph information into a lower-dimensional latent space, while preserving the essential topological features of a graph. Using GAE, the attacker extracts the graph structure capturing the correlations between the benign local models (that could be transmitted over a transport layer security (TLS) protocol), and decouples the graph structure from underlying data features substantiating the local models. The attacker first regenerates manipulatively the graph structure to retain the structural features of the local models and maximize the FL training loss using the GAE, and then generates malicious local models using the regenerated graph structure to the data features of the benign local models. As a result, the malicious local models can effectively compromise the global model, while remaining compatible with the benign models and hence reasonably undetectable.

The contributions of the paper are summarized below.

- A new design of data-agnostic, malicious local models, which manipulates the correlations of benign local models and retains the genuine data features substantiating the benign local models;
- A new GAE framework, which is trained together with sub-gradient descent to regenerate manipulatively the correlations of the local models while keeping the malicious local models undetectable; and
- A rigorous analysis, which proves the convergence of the global model under attack, but to an inferior optimality gap.
- The proposed GAE-based attack is implemented experimentally based on the standard MNIST, fashion-MNIST, and CIFAR-10 datasets. It is shown that the GAE-based attack significantly compromises the FL performance, where the training accuracy falls below 50% at the user devices. The source code of the proposed GAE-based, data-agnostic, model poisoning attack is available on GitHub.

Extensive experiments indicate that the FL accuracy drops gradually under the proposed attack, and the existing poisoning defense mechanisms can hardly detect the attack. Since the malicious local models are uploaded to the server for global model aggregation, the proposed attack gives rise to an epidemic infection across all benign devices.

The proposed GAE-based attack on FL involves attackers intentionally poisoning malicious local models, aiming to degrade or manipulate the performance of the global model. The attack challenges the security, privacy, and robustness of FL. While security is threatened by unauthorized access or malicious insiders tampering with local models, privacy concerns arise when the attackers try to reverse-engineer or glean information about the benign devices' data. Moreover, robustness, which is the ability of FL to consistently produce reliable and accurate results, can be directly undermined, as poisoned local models compromise the integrity and efficacy of FL. To this end, the proposed GAE-based attack poses a comprehensive threat to the security, privacy, and robustness of FL.

TABLE I: Notation and definition

Notation	Definition
J	The total number of benign user devices
$\boldsymbol{\omega}_g(t)$	The global model of FL in the t-th communication round
$\boldsymbol{\omega}_g^a(t)$	The global model under attack
$f_i(\boldsymbol{\omega}_j(t); x_j^i, y_j^i)$	The training loss function of device j
$F_j(\boldsymbol{\omega}_j(t))$	The local loss function of device j
$F(\boldsymbol{\omega}_j(t))$	The weighted loss function of FL
η	The learning rate of the local model
T_L	The number of training iterations per FL communication round
d_T	Euclidean distance threshold
$\mathcal{A}$	The adjacency matrix for the local models of user devices
$\mathcal{F}$	The feature matrix
$\widehat{\mathcal{A}}$	The reconstructed adjacency matrix generated at the decoder
$\mathcal{L}$	The Laplacian matrix based on the benign weights
$\widehat{\mathcal{L}}$	The Laplacian matrix regenerated by the attacker
$\widehat{\mathcal{F}}$	The malicious local model

The rest of this paper is organized as follows. Section II introduces the background of adversarial attacks against wireless systems and FL. Section III discusses FL with benign user devices and server, as well as the eavesdropping model. The proposed GAE-based epidemic attack is delineated in Section IV. Performance analysis is conducted in Section V. Section VI concludes the paper. Table I lists the notation used in the paper.

II. Related Work

This section reviews the literature on adversarial attacks against wireless systems as well as FL, including model and data poisoning attacks. On the one hand, because of their broadcast nature, wireless channels are particularly vulnerable to eavesdropping attacks. An attacker is likely to overhear the local model updates transmitted by the other benign users in wireless FL. On the other hand, the model poisoning attack considered in this paper has not been studied in the literature. Instead, existing attacks on wireless FL have focused primarily on building an adversarial data classification/label model for attackers, according to the data packets and features overheard, e.g., [14] and [15]. There is clearly an opportunity for the new attack to strike.

A. Adversarial Attacks on Wireless Systems

In [16], an adversarial attack was studied to manipulate the measurement of smart meters in residential homes. Smart meter data could inform residents of which appliances consumed the most electricity and adjust energy production. The attacker employed deep learning to train a power usage pattern classification model and generated malicious data that was indistinguishable from the true

data. In [14], machine learning was used to generate an adversarial attack for targeting data fusion or aggregation. The attacker infiltrated some devices and learned the decision process and data fusion settings by observing data exchanges between the devices and the data center.

In [17], the authors analyzed targeted adversarial attacks that aimed to manipulate the output of a convolutional neural network (NN)-based classifier. They also evaluated non-targeted adversarial attacks against convolutional NN-based device identification. To evaluate these attacks, the authors used combined indicators of logits to increase the perturbation levels and iterative steps, resulting in a high success rate of adversarial attacks. In [18], researchers used deep learning to recognize COVID-19 symptoms by training on medical data from user devices. They evaluated several adversarial attacks that aimed to falsify the data and symptom recognition. The study found that existing deep learning algorithms were vulnerable to these attacks, highlighting the need for advanced security measures.

In [15], an adversarial attack was developed to deactivate graph-based intrusion detection in a targeted wireless system. The attack began by building a shadow graph based on overheard data packets and features. A random walk algorithm was then used to evaluate each node in the attacker's graph, selecting the node with the largest weight to attack. The attack would perturb data features and alter classification labels. In [19], an adversarial attack was developed to utilize graph embedding and augmentation to misclassify system malware samples as benign. The graph-based attack aimed to embed a target malware sample into benign software. By combining the benign code sample and the target malware sample in the graph, the adversarial attack could learn complex features, resulting in a high misclassification rate at the user device.

In [20], a study was conducted on a Sybil-based data poisoning attack against deep reinforcement learning-based service placement in the Internet of Vehicles (IoV). The attack targeted the agent that is responsible for learning the service quality and deciding on service placement based on delay. A Sybil attacker, which is a malicious vehicle, used data poisoning techniques to masquerade as a legitimate vehicle by stealing or borrowing its identity. The attacker then maliciously sent false data to other vehicles.

Unfortunately, it is difficult for the attacker to formulate the adversarial data classification/label model in FL systems since the benign user devices can collaboratively conduct model training without sharing their private data.

B. Poisoning Attacks on FL

In order to corrupt the FL, the attacker can launch either a data poisoning or a model poisoning attack. In the data poisoning attack, the attacker injects fake data with manipulated features and flips labels into the benign user devices. In the model poisoning attack, the attacker submits malicious local models to the server. Both attacks aim to corrupt the FL by introducing false information.

In [11], the authors systematically categorized the existing threat models associated with poisoning attacks on FL, where practical boundaries of numerous parameters pertinent to FL robustness were delineated. An array of untargeted model and data poisoning attacks on FL was analyzed to encompass the existing attack strategies. A model poisoning attack was developed using gradient ascent to fine-tune the global model and increase its loss on benign data. The model poisoning attack adjusts the L_2-norm of the poisoned model update to circumvent the robustness criterion of the model aggregation.

In [21], an adversarial attack mitigation scheme based on clustering was studied. The scheme aimed to protect FL by using unsupervised weight training to split and merge weight clusters at the server to filter out malicious local models that were uploaded by the user devices without identity verification. In [22], malicious local models were derived from mislabeled data to manipulate the global model. The study found that this attack could result in a significant drop in classification accuracy, and that it was difficult to detect due to its negative impact on the target device and minimal impact on other benign devices.

In [23], an inference model was formulated to take local models as input and output the categories of data. A malicious local model based on a differential selection strategy was used to select two adjacent categories. To approximate the benign local model, a category inference attack was studied, in which the attacker learns the data features underlying benign local models.

The authors of [24] presented a backdoor attack against FL in mobile edge computing (MEC), which targeted the tail of the input data distribution at the local devices. The attack used projected gradient descent to maintain the distance between the malicious local model and the global model, to misclassify the targeted samples and bypass defense mechanisms.

In [25], generative adversarial networks (GANs) were utilized to construct data poisoning attacks against FL. The attacker trained the GAN to replicate the local data of the benign devices. Since the attacker had no information about the local data, the GAN-based data poisoning updated the global model to re-select the potential targeted devices. In [26], a GAN-based FL poisoning attack was studied, where the attacker posed as one of the benign devices and trained the GAN to mimic the dataset of the benign devices. The malicious data generated by the attacker were trained to compromise the global model. In [27], a malicious server deployed a GAN-based reconstruction attack against FL to tamper with the private data of the user devices. The malicious server discriminated the devices' identities and data representatives to supervise the training of GANs and generate malicious data for each specific device. In [28], the authors focused on a device-level privacy leakage attack launched by a malicious server. A GAN-based framework was presented to discriminate the data category and device's identity and recover the private data of the device. The attack could associate the data features from different devices to re-identify the local models.

Unfortunately, the existing data poisoning or model poisoning attacks have not exploited the implicit relationship

between local models [29], [30]. Moreover, the existing poisoning attacks generally require the attacker to have the knowledge of (part of) the datasets used for FL training.

III. System Model

In this section, we first describe an FL training process, e.g., for image classification. Next, we present the threat model, where malevolent devices can act as attackers. An attacker creates and uploads malicious local model updates to progressively contaminate the global model of the FL. At last, we describe an attacker detection model that the server can adopt to discern malicious local models by measuring the Euclidean distances between the models.

A. Federated Learning

We assume there are J benign user devices and an authorized (legitimate) but malicious user device (or an attacker) in the FL training process. A benign user device $j \in [1, J]$ has $D_j(\tau)$ amount of data at the τ-th iteration. Let x_j^i and y_j^i denote the input of the captured images and the output of the FL model at device j, respectively. $i \in [1, D_j(\tau)]$. A training loss function of device j, denoted by $f_i(\boldsymbol{\omega}_j(\tau); x_j^i, y_j^i)$, captures approximation errors over the input x_j^i and the output y_j^i. Here, $\boldsymbol{\omega}_j(\tau)$ is the weight parameter of the loss function in the model being trained by the FL. For instance, $f_i(\boldsymbol{\omega}_j(\tau); x_j^i, y_j^i)$ can be modeled by linear regression, i.e., $f_i(\boldsymbol{\omega}_j(\tau); x_j^i, y_j^i) = \frac{1}{2}(\boldsymbol{\omega}_j(\tau)^T x_j^i - y_j^i)^2$; or logistic regression, i.e., $f_i(\boldsymbol{\omega}_j(\tau); x_j^i, y_j^i) = y_j^i \log\Big(1 + \exp\big(-\boldsymbol{\omega}_j(\tau)^T x_j^i\big)\Big) - (1 - y_j^i)\log\Big(1 - \frac{1}{1+\exp\left(-\boldsymbol{\omega}_j(t)^T x_j^i\right)}\Big)$. Here, $(\cdot)^T$ denotes transpose. Given $D_j(\tau)$, the local loss function of the FL at device j for the τ-th iteration is

$$F_j(\boldsymbol{\omega}_j(\tau)) = \frac{1}{D_j(\tau)} \sum_{i=1}^{D_j(\tau)} f_i(\boldsymbol{\omega}_j(\tau); x_j^i, y_j^i) + \mu g(\boldsymbol{\omega}_j(\tau)), \quad (1)$$

where $g(\cdot)$ is a regularizer function that represents the effect of the local training noise, and $\mu \in [0, 1]$ is a coefficient [31].

The local model of user device j is updated by

$$\boldsymbol{\omega}_j(\tau + 1) = \boldsymbol{\omega}_j(\tau) - \eta \nabla F_j(\boldsymbol{\omega}_j(\tau)), \quad (2)$$

where η is the learning rate.

After every T_L local updates (or iterations), there is a communication round where the benign user devices upload their local models to a server. The server aggregates the local models to update the global model and broadcasts the global model to all user devices. While selecting the user devices with large training datasets can help improve the learning accuracy of FL, it often results in the fast depletion of the batteries at the user devices. On the other hand, selecting the user devices with small datasets can save the battery energy of the devices, but the accuracy of the global model could suffer. Existing resource allocation policies, such as those developed in [32] and [33], can be applied to balance the learning accuracy of FL and the energy consumption of the user devices.

B. Threat Model

We consider a new data-agnostic model poisoning attack, where malicious local models are generated solely based on the benign local models overheard and the correlation features of the benign local and global models. This attack could be particularly severe in FL systems under wireless settings, due to the broadcast nature of radio. As shown in Fig. 1, an attacker within the vicinity of benign user devices and equipped with radio transceivers can passively eavesdrop on the local models transmitted by some (if not all) of the benign user devices, extracting their features and generating malicious local models. A similar threat model has also been considered in the recent literature [34]–[36], where an attacker within proximity of benign user devices overhears the local and global models in an attempt to recover, at least partially, the private data of the benign user devices. Although cryptography can prevent eavesdropping to some extent, existing techniques, such as those developed in [37]–[39], have demonstrated the possibility of deciphering encrypted information with limited initial data.

The attacker creates and uploads a malicious local model, denoted by $\boldsymbol{\omega}^a(t)$, to contaminate the global model $\boldsymbol{\omega}_g(t)$, and subsequently the local models of the benign users, i.e., $\boldsymbol{\omega}_j(t)$, $\forall j \in [1, J]$, where t indicates the t-th communication round. $\boldsymbol{\omega}^a(t)$ is adversarially created based on the benign local model parameters overheard by the attacker in the t-th communication round.

Unaware of the ill-intentioned attacker, the server aggregates the local models of all user devices, including both the benign and malicious local models, and unintentionally creates a contaminated global model, denoted by $\boldsymbol{\omega}_g^a(t)$, at the t-th communication round. The total size of the local training data reported to the server is $D(t) = \sum_{j=1}^{J} D_j(t) + D_a(t)$, where $D_a(t)$ is the claimed data size of the attacker at the t-th communication round. Then, the contaminated global model is given by

$$\boldsymbol{\omega}_g^a(t) = \sum_{j=1}^{J} \frac{D_j(t)}{D(t)} \boldsymbol{\omega}_j(t) + \frac{D_a(t)}{D(t)} \boldsymbol{\omega}^a(t), \quad (3)$$

The server broadcasts $\boldsymbol{\omega}_g(t)$ to all user devices.

To this end, the FL training process in essence trains the global model based on the local datasets of all user devices, including the nonexistent dataset claimed by the attacker, by minimizing the following global loss function:

$$\min_{\boldsymbol{\omega}_g^a(t)} F(\boldsymbol{\omega}_g^a(t)) = \sum_{j=1}^{J} \frac{D_j(t)}{D(t)} F_j(\boldsymbol{\omega}_g^a(t)) + \frac{D_a(t)}{D(t)} F_a(\boldsymbol{\omega}_g^a(t)), \quad (4)$$

where $F_a(\cdot)$ is the claimed local loss function of the attacker, which is claimed to conform to (1).

To attack the FL training process, the attacker aims to maximize $F(\boldsymbol{\omega}_g^a(t))$, while keeping $\boldsymbol{\omega}^a(t)$ undetectable by the server that typically constantly assesses the similarities among all local models and rules out those substantially different from the rest, e.g., Krum or multi-Krum [40]. As a result, the attacked global model diverges in a direction opposite to the one intended in the absence of the attack.

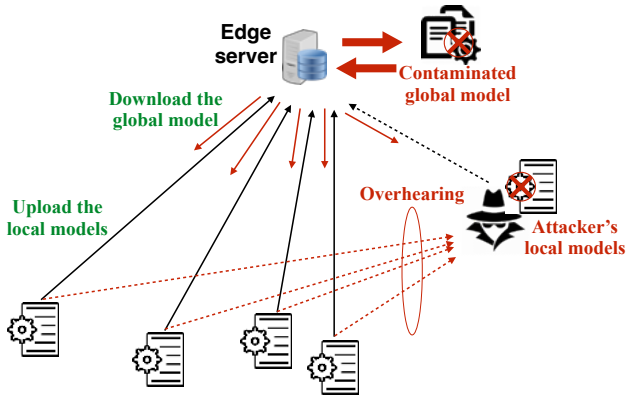


Fig. 1: The proposed data-agnostic model poisoning attack, where the attacker overhears the global model and the local models uploaded by the benign user devices. Next, the attacker generates a malicious local model to contaminate the global model and the benign local models.

At the t-th communication round, the attacker formulates a data-agnostic, model poisoning attack problem:

$$\max_{\boldsymbol{\omega}^a(t)} \quad F(\boldsymbol{\omega}_g^a(t)) \tag{5a}$$

$$\text{s.t.} \quad d(\boldsymbol{\omega}^a(t), \boldsymbol{\omega}_g^a(t)) \leq d_T, \tag{5b}$$

where $d(\boldsymbol{\omega}^a(t), \boldsymbol{\omega}_g^a(t))$ evaluates the Euclidean distance between $\boldsymbol{\omega}^a(t)$ and $\boldsymbol{\omega}_g^a(t)$, and d_T is a pre-specified threshold that ensures the generated malicious local model is close to the global model in the Euclidean space to escape the scrutiny of the server.

C. Defense Model for Attacker Detection

In response to the prevalent threat of model poisoning in FL, an attacker detection model residing on the server can be applied, which leverages the Euclidean distance metric to discern malicious local models, for instance, [8] and [41]. By measuring the straight-line distance between each incoming local model and the aggregated global model, this model aims to identify anomalous deviations indicative of malicious intent. The underlying rationale is that genuine local models from benign devices are expected to cluster within a certain proximity in the model space, while malicious local models, designed to sabotage the global model's integrity, would exhibit more pronounced deviations. By setting a distance threshold, local models that exceed this threshold can be flagged or discarded, effectively isolating and mitigating the impact of malicious local models on the global model's integrity. This server-side defense mechanism underscores the potential of geometric measures, like Euclidean distance, as powerful tools in safeguarding FL systems from adversarial attacks.

IV. Proposed Data-Agnostic Model Poisoning Attack on FL

In this section, we elaborate on the proposed data-agnostic model poisoning attack, where adversarial GAE is designed to extract the feature correlation among the local models of the benign user devices and reconstruct an adversarial adjacency matrix. With the adjacency matrix, the attacker trains the GAE to generate malicious local models without being detected by the server.

A. GAE Model for Data-Agnostic Model Poisoning

The arbitrary features of $\boldsymbol{\omega}^a(t)$ and those of the benign local models may have a low feature correlation, which can be potentially detected by the server. To address this, we develop a new GAE model for the novel, data-agnostic, model poisoning attack.

The optimization problem in (5) can be transformed using the Lagrangian method [42]. Let λ denote the dual variable. The Lagrange function is given by

$$L(\boldsymbol{\omega}^a(t), \lambda) = F(\boldsymbol{\omega}_g^a(t)) + \lambda(d_T - d(\boldsymbol{\omega}^a(t), \boldsymbol{\omega}_g^a(t))). \tag{6}$$

The Lagrange dual function is

$$\mathcal{D}(\lambda) = \max_{\boldsymbol{\omega}^a(t)} L(\boldsymbol{\omega}^a(t), \lambda). \tag{7}$$

The dual problem of the problem in (5) is given by

$$\min_{\lambda(t)} \mathcal{D}(\lambda). \tag{8}$$

At the t-th communication round, given $\lambda = \lambda(t)$, the primary variable $\boldsymbol{\omega}^a(t)$ of the data-agnostic model poisoning attack can be optimized by solving

$$\boldsymbol{\omega}^a(t)^* = \arg\max_{\boldsymbol{\omega}^a(t)} \{F(\boldsymbol{\omega}_g^a(t)) - \lambda(t) d(\boldsymbol{\omega}^a(t), \boldsymbol{\omega}_g^a(t))\}. \tag{9}$$

With obtained $\boldsymbol{\omega}^a(t)^*$, the sub-gradient descent method can be taken to update $\lambda(t)$ by solving the dual problem (8). Specifically, $\lambda(t)$ is updated by [43]

$$\lambda(t+1) = \left[\lambda(t) - \varepsilon \left(d(\boldsymbol{\omega}^a(t)^*, \boldsymbol{\omega}_g^a(t)) - d_T\right)\right]^+, \tag{10}$$

where ε is the step size, τ is the index to the iterations, and $[x]^+ = \max(0, x)$. At initialization, $\lambda(t)$ is non-negative, i.e., $\lambda(1) \geq 0$, to ensure (10) converges.

We propose to solve (9) by developing a new GAE model, followed by the sub-gradient descent to update (10). These two steps are performed in an alternating manner, as illustrated in Fig. 2. Specifically, we propose to decompose the local model parameters of the benign devices into a graph capturing the correlations (or similarity) between the benign local models, and the underlying spectral-domain data features that the local models capture. Then, we re-generate the graph with the GAE in a manipulative manner and subsequently compose malicious local models with the regenerated graph and the original, genuine data features. The rationale of this design is provided as follows.

- By regenerating the graph with the GAE, we retain and manipulate the correlations between the local models, and also deter the convergence of the global model, i.e., by maximizing (9). The decoder of the GAE reproduces the correlations while satisfying constraint (5b). This suppresses structural dissimilarity between the malicious and benign local models.

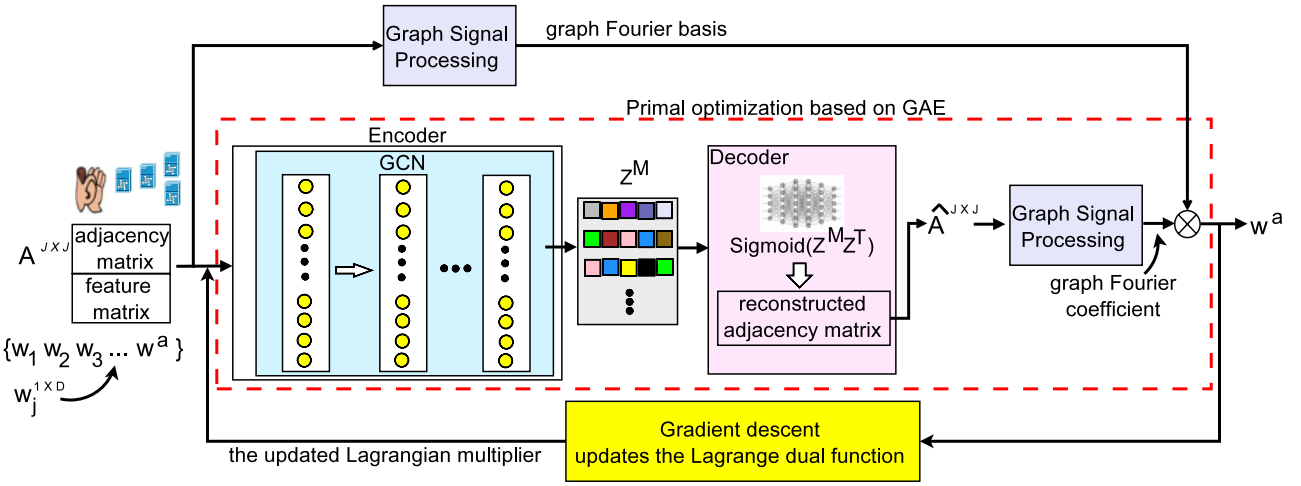


Fig. 2: The proposed GAE model for generating data-agnostic, malicious local models, where the attacker overhears $\boldsymbol{\omega}_g(t)$ and $\boldsymbol{\omega}_j(t)$, $\forall j$ and applies the GCN-based encoder to create $\mathcal{Z}^M$. The output of the encoder, i.e., the feature representations, is input to the decoder for feature reconstruction.

- By using the genuine underlying spectral-domain data features, the malicious local models are substantiated by the genuine data features. Hence, they are less likely to be detected by the server.

1) GAE for Malicious Model Generation: The attacker aims to construct $\boldsymbol{\omega}^a(t)$ without knowing any data of the benign devices. As illustrated in Fig. 2, a graph, denoted by $G(\mathcal{V}, E, \mathcal{F})$, is used to formulate the benign local models in FL, where $\mathcal{V}$, E, and $\mathcal{F}$ represent vertexes, edges, and the feature matrix of the graph, respectively.

Let $\mathcal{F} = [\boldsymbol{\omega}_1(t), \cdots, \boldsymbol{\omega}_j(t), \boldsymbol{\omega}^a(t)]$ collect all local models of both benign and malicious devices. $\boldsymbol{\omega}_j(t), \boldsymbol{\omega}^a(t) \in \mathbb{R}^{1\times D}$, $\forall j$. Also, let $\mathcal{A} \in \mathbb{R}^{J\times J}$ denote the adjacency matrix that describes the correlation among the local models of the user devices. At the t-th communication round of the FL, the (j, j')-th element of $\mathcal{A}$, denoted by $\overline{\omega}_{j,j'}$ $(j, j' \in [1, J])$, measures the inner product between $\boldsymbol{\omega}_j(t)$ and $\boldsymbol{\omega}_{j'}(t)$ [44], as given by

$$\overline{\omega}_{j,j'} = \frac{\boldsymbol{\omega}_j(t) \cdot \boldsymbol{\omega}_{j'}(t)}{\|\boldsymbol{\omega}_j(t)\| \cdot \|\boldsymbol{\omega}_{j'}(t)\|}. \tag{11}$$

According to $\mathcal{A}$, the topological structure of the graph $\mathcal{G}$ can be constructed.

The GAE consists of an encoder and a decoder, where the encoder encodes the graph data with the features and the decoder takes the encoder's output as the input to reconstruct $G(\mathcal{V}, E, \mathcal{F})$ [45].

• **Encoder:** The encoder in the proposed GAE is responsible for mapping $G(\mathcal{V}, E, \mathcal{F})$ to a lower-dimensional representation. We build the encoder based on an M-layer graph convolutional network (GCN) architecture, which learns a representation that captures the underlying features of $G(\mathcal{V}, E, \mathcal{F})$. The encoded representation is then used as input to the decoder, to reconstruct the original graph from the lower-dimensional representation to obtain the malicious local model $\boldsymbol{\omega}^a(t^*)$ in (9).

The encoder takes $\mathcal{A}$ as its input to its M-layer GCN. The output at the M-th layer is

$$\mathcal{Z}^M = f_G(\mathcal{Z}^{M-1}, \mathcal{A}|\mathbf{w}^M), \tag{12}$$

where $f_G(\cdot, \cdot|\cdot)$ is a spectral convolution function and $\mathbf{w}^M$ defines the weight matrix at the M-th layer of the GCN.

With the identity matrix $I \in \mathbb{R}^{J\times J}$, we define $\widetilde{\mathcal{A}} = \mathcal{A} + I$ and $\overline{\mathcal{A}}_{jj} = \sum_{j'} \widetilde{\mathcal{A}}_{jj'}$. To generate a feature representation of the graph, the encoder can be written as

$$f_G(\mathcal{Z}^{M-1}, \mathcal{A}|\mathbf{w}^M) = \Phi^M(\overline{\mathcal{A}}^{-\frac{1}{2}} \widetilde{\mathcal{A}} \overline{\mathcal{A}}^{-\frac{1}{2}} \mathcal{Z}^{M-1} \mathbf{w}^M), \tag{13}$$

where $\Phi^M(\cdot)$ represents a nonlinear activation function, e.g., $\tanh(\cdot)$ or ReLU$(\cdot)$; and $\overline{\mathcal{A}}^{-\frac{1}{2}} \widetilde{\mathcal{A}} \overline{\mathcal{A}}^{-\frac{1}{2}}$ is the symmetrically formulated adjacency matrix [44], [46].

• **Decoder:** The decoder is responsible for taking the lower-dimensional representation generated by the encoder, i.e., $\mathcal{Z}^M$ in (12), and mapping it back to the original $G(\mathcal{V}, E, \mathcal{F})$. This can be viewed as the inverse operation of the encoder. The decoder aims to generate the original graph from its reduced representation. The output of the decoder is compared with the original input graph to evaluate a loss. The encoder and decoder are trained together to minimize the loss.

A reconstructed adjacency matrix is generated at the decoder, which is defined as

$$\widehat{\mathcal{A}} = \text{sigmoid}\left(\mathcal{Z}^M \left(\mathcal{Z}^M\right)^T\right). \tag{14}$$

where the Sigmoid function is defined as $\text{sigmoid}(x) = 1/(1 + \exp(-x))$. The larger the inner product $(\mathcal{Z}^M \left(\mathcal{Z}^M\right)^T)$, the more likely the vertexes j and j' are connected in the graph [47].

The output of the decoder is the reconstructed adjacency matrix $\widehat{\mathcal{A}}$. A reconstruction loss function that measures the difference between $\mathcal{V}$ and $\widehat{\mathcal{A}}$ can be formulated as [48]

$$\phi_{\text{loss}} = \mathbb{E}_{f_G(\mathcal{Z}^{M-1}, G|\mathbf{w}^M)}\Big[\log\ p(\ \widehat{\mathcal{A}}\ |\ \mathcal{Z}^M\ )\Big], \tag{15}$$

where $p(\ \widehat{\mathcal{A}} \mid \mathcal{Z}^M\ )$ at the decoder indicates the correlation among the embedding vertexes, and is given by

$$p(\ \widehat{\mathcal{A}} \mid \mathcal{Z}^M\ ) = \Pi_{j=1}^{J}\Pi_{j'=1}^{J} p(\ \widehat{\mathcal{A}}_{jj'} \mid \mathcal{Z}_j^M, \mathcal{Z}_{j'}^M\ ), \quad (16)$$

where

$$p(\widehat{\mathcal{A}}_{jj'} = 1 | \mathcal{Z}_j^M, \mathcal{Z}_{j'}^M\ ) = \text{sigmoid}\left(\mathcal{Z}_j^M \left(\mathcal{Z}_{j'}^M\right)^T\right). \quad (17)$$

• **Malicious Model Generation:** A graph signal processing module is designed to decompose the correlation features of the benign local models, and the data features substantiating the local models, as described earlier. A Laplacian matrix [49] is built based on the adjacency matrix of the benign models, i.e., $\mathcal{A}$, as given by

$$\mathcal{L} = diag(\mathcal{A}) - \mathcal{A}. \quad (18)$$

By applying singular value decomposition (SVD) [50] to $\mathcal{L}$, i.e., $\mathcal{L} = B\Sigma B^T$, we can obtain a complex unitary matrix $B \in \mathbb{R}^{J\times J}$, also known as graph Fourier transform (GFT) basis, that is used to transform graph data, e.g., $\mathcal{F}$, to its spectral-domain representation. Σ is a diagonal matrix with the eigenvalues of $\mathcal{L}$ along its main diagonal.

As a result, the attacker can obtain a matrix S that contains the spectral-domain data features of all benign local models, by removing the correlations among the models and subsequently focusing on the data features substantiating the local models. S is given by

$$S = B^{-1}\mathcal{F}. \quad (19)$$

Likewise, the attacker can use the graph signal processing module to produce a Laplacian matrix based on the output of the GAE, as given by

$$\widehat{\mathcal{L}} = diag(\widehat{\mathcal{A}}) - \widehat{\mathcal{A}}. \quad (20)$$

The corresponding GFT basis, denoted by $\widehat{B}$, can be obtained by applying SVD to $\widehat{L}$. With reference to (19), the malicious local model that follows $\mathcal{A}$ in the GAE can be determined by

$$\widehat{\mathcal{F}} = \widehat{B}S, \quad (21)$$

where $\widehat{\mathcal{F}} \in \mathbb{R}^{J\times D}$. The vector $\boldsymbol{\omega}^a(t)$ in $\widehat{\mathcal{F}}$ is selected as the malicious local model and uploaded by the attacker to the aggregator for global model aggregation in the t-th communication round.

Since the attacker aims to generate the malicious local models to disorient FL, the proposed GAE is constructed and trained to maximize $L(\boldsymbol{\omega}^a(t), \lambda(t)) - \phi_{\text{loss}}$. As a consequence, the malicious local model $\boldsymbol{\omega}^a(t)$ progressively and increasingly contaminates the FL training process with the increase in global model aggregations, i.e., $t = 1, 2, \cdots$.

B. Training Algorithm of the Proposed GAE Model

Algorithm 1 summarizes the training process of the proposed GAE-based, data-agnostic, model poisoning attack model, which operates along with the FL training of the benign devices and the server. Specifically, in every FL communication round, i.e., the t-th round, the server broadcasts $\boldsymbol{\omega}_g^a(t)$. The benign devices apply the Local-Training_start($\boldsymbol{\omega}_g^a(t)$) function to train their local models $\boldsymbol{\omega}_j(t)$, $\forall j = 1, \cdots, J$; see Steps 3 – 5 in Algorithm 1.

Algorithm 1 The proposed GAE-based, data-agnostic model poisoning attack against FL

1: **1. Initialize**: $G(\mathcal{V}, \mathcal{E}, \mathcal{F})$, T_L, J, d_T, $\boldsymbol{\omega}_g^a(t)$, $\boldsymbol{\omega}^a(t)$, and $\lambda(1) \geq 0$.
   % **Adversarial FL**:
2: **for** round t = 1, 2, 3, ... **do**
3: **for** Local iteration number $t_L = 1, \cdots, T_L$ **do**
4: All benign user devices train the benign local model $\boldsymbol{\omega}_j(t)$, $j = 1, \cdots, J$.
5: **end for**
6: All benign user devices upload their benign local models $\boldsymbol{\omega}_j(t)$, $j = 1, \cdots, J$ to the server, and the attacker overhears the benign local models.
7: The attacker carries out the proposed GAE, i.e., **GAE**($\boldsymbol{\omega}_j(t), \forall j, \mathcal{F}, \lambda(t)$), and obtains $\boldsymbol{\omega}^a(t)$, as follows.
8: · Calculate the adjacency matrix $\mathcal{A} = \{\overline{\omega}_{j,j'}\} \in \mathbb{R}^{J\times J}$ according to (11), and input $\mathcal{A}$ and $\mathcal{F}$ into the GAE.
9: · Train the GAE to maximize the reconstruction loss $L(\boldsymbol{\omega}^a(t), \lambda(t)) - \phi_{\text{loss}}$ to obtain $\widehat{\mathcal{A}}$.
10: · Obtain S based on (18) and (19), next obtain $\widehat{\mathcal{F}}$ based on (20) and (21), and then determine $\boldsymbol{\omega}^a(t)$ based on $\widehat{\mathcal{F}}$.
11: Update $\lambda(t)$, according to (10).
12: The attacker uploads the malicious local model $\boldsymbol{\omega}^a(t)$ to the server.
13: The server aggregates all the local models to obtain the global model under attack $\boldsymbol{\omega}_g^a(t)$ by (3), and broadcasts $\boldsymbol{\omega}_g^a(t)$.
14: All benign user devices update their local models with the global model, i.e., $\boldsymbol{\omega}_j(t) \leftarrow \boldsymbol{\omega}_g^a(t)$, $\forall j$.
15: **end for**

On the other hand, the attacker overhears the local model $\boldsymbol{\omega}_j(t)$, $\forall j$ from the benign devices at the t-th FL communication round, and recall the global model $\boldsymbol{\omega}_g^a(t-1)$ overheard from the server at the $(t-1)$-th round. The GAE is trained to maximize the data-agnostic model poisoning attack problem in (5) with $\mathcal{V}$ and $\mathcal{F}$. Specifically, the problem in (5) is transformed into a primal and a dual problem using the Lagrangian method. Given the dual variable $\lambda(t)$, $\boldsymbol{\omega}^a(t)$ is optimized using the GAE; see Steps 8 – 10 in Algorithm 1. With the obtained $\boldsymbol{\omega}^a(t)^*$, the sub-gradient descent method is taken to update $\lambda(t)$ by (10); see Step 11. At the output of the GAE, the attacker achieves the optimal malicious local model, i.e., $\boldsymbol{\omega}^a(t)$. Next, $\boldsymbol{\omega}^a(t)$ is uploaded to the server for the next round of the FL training. As $\boldsymbol{\omega}^a(t)$ is highly correlated with $\boldsymbol{\omega}^a(t)$ from the benign user devices, the server is unable to identify the attacker.

C. Convergence Analysis of FL under Attack

We derive the convergence upper bound for the FL under the proposed, data-agnostic, model poisoning attack. The following assumptions are made before the analysis, as typically considered in the literature [51]–[53].

Assumption 1: $\forall m \in \mathcal{M}$,

1) The gradient of $F_j(\boldsymbol{\omega}_j)$ is L-Lipschitz continuous [54], that is, $\|\nabla F_j(\boldsymbol{\omega}_j(t+1)) - \nabla F_j(\boldsymbol{\omega}_j(t))\| \leq L\|\boldsymbol{\omega}_j(t+1) - \boldsymbol{\omega}_j(t)\|, \forall \boldsymbol{\omega}_j(t+1), \boldsymbol{\omega}_j(t)$, with L being a constant depending on the loss function so that the gradient of the global loss function is also L-Lipschitz continuous;
2) $F_j(\boldsymbol{\omega}_j)$ is L_c-Lipschitz continuous; in other words, $\left|F_j(\boldsymbol{\omega}_j) - F_j(\boldsymbol{\omega}'_j)\right| \leq L_c \left\|\boldsymbol{\omega}_j - \boldsymbol{\omega}'_j\right\|, \forall \boldsymbol{\omega}_j, \boldsymbol{\omega}'_j$;
3) The learning rate is $\eta \leq \frac{1}{L}$;
4) At device j, the expected squared norm of the stochastic gradients is uniformly bounded by $\mathbb{E}\|\nabla F_j(\boldsymbol{\omega}_j(t))\|^2 \leq \kappa \|\nabla F(\boldsymbol{\omega}_g(t))\|^2, \forall j, \kappa \geq 0$;
5) With $\rho \geq 0$, $F_j(\boldsymbol{\omega})$ fulfills the Polyak-Lojasiewicz requirement [55], indicating that $F(\boldsymbol{\omega}_g) - F(\boldsymbol{\omega}_g^*) \leq \frac{1}{2\rho}\|\nabla F(\boldsymbol{\omega}_g)\|^2$, where $\boldsymbol{\omega}_g^* = \arg\min_{\boldsymbol{\omega}_g} F(\boldsymbol{\omega}_g)$;
6) $F(\boldsymbol{\omega}_g(0) - F(\boldsymbol{\omega}_g^*) = \Theta$, where Θ is a constant.

Under Assumption 1, we develop the following theorem that provides the convergence bound of the gap between $\boldsymbol{\omega}_g(t), \forall t$ and $\boldsymbol{\omega}_g^*$.

Theorem 1: At the t-th communication round, the convergence upper bound of the attacked FL is obtained as

$$F(\boldsymbol{\omega}_g^a(t)) - F(\boldsymbol{\omega}_g^*) \leq \Theta\zeta^t + \frac{1-\zeta^t}{1-\zeta} \cdot \frac{\rho\eta D D_a}{(D-D_a)^2} F^{\max}, \tag{22}$$

where $\zeta = 1 - \frac{\rho\eta D^2}{(D-D_a)^2}$, and $F^{\max}$ is a maximum value of $F(\boldsymbol{\omega}^a(t))$ due to the constraint in (5a) and (5b), i.e., $F(\boldsymbol{\omega}^a(t)) \leq F^{\max}$.

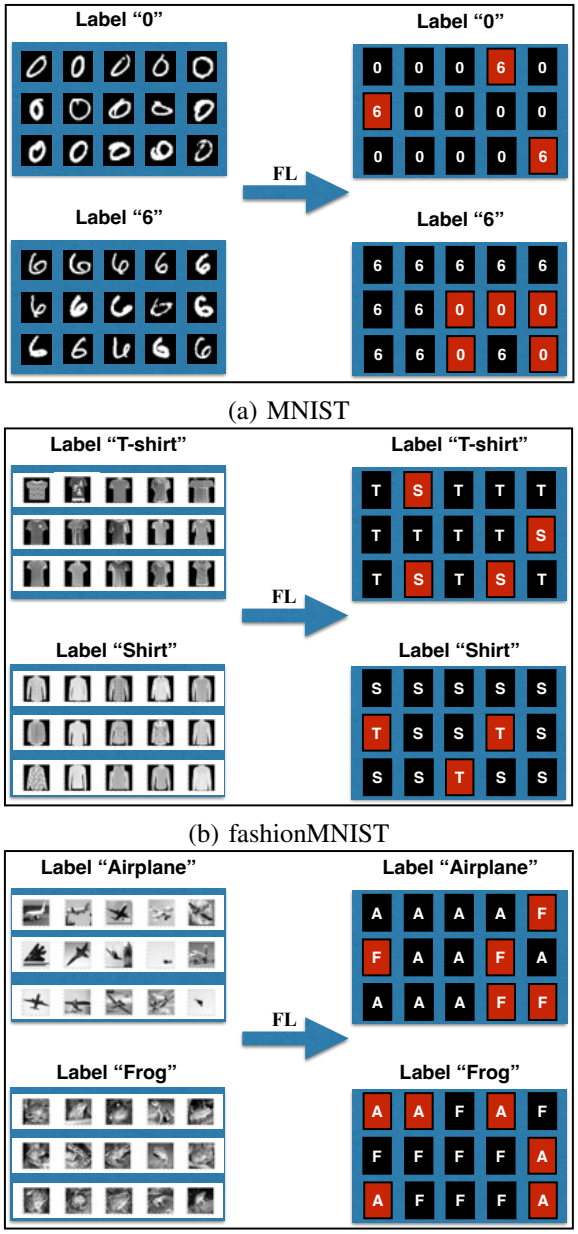
Proof: See Appendix A. ∎

As stated in Theorem 1, despite the attack launched by the attacker, the global model of FL can still converge, but to an inferior global model. Specifically, as $t \to \infty$, the optimality gap would stabilize at $\frac{2D_a L_c d_T}{D-D_a}$, which cannot be further reduced by training.

V. Performance Evaluation

In this section, we present the implementation of the proposed GAE-based, data-agnostic, model poisoning attack in PyTorch. We evaluate the testing accuracy of the local and global models of FL under attack, using the MNIST [56], fashionMNIST [57], and CIFAR-10 datasets. We also report the detection rate of the attack, where the detection is based on the Euclidean distances of the malicious local models and the benign local models to the global models, as typically done in the latest literature, e.g., [58], [59].

Moreover, we compare the proposed attack with the existent data-agnostic model poisoning (MP) attack that produces malicious local models by mimicking other benign devices' training samples to degrade the learning accuracy. As discussed earlier in Section IV, our GAE-based attack represents a novel type of attack, which only depends on the benign local models overheard and the global models, has no access to any of the training data, and attempts to compromise FL training processes. Few existing techniques can produce malicious models in such a way, i.e., based solely on the overheard benign models, as most existing techniques would require the knowledge of (part of) the dataset used in the FL training processes, e.g., for a different purpose, such as inserting a backdoor [58] or injecting malicious traffic into the benign training dataset [60]. The MP attack mechanism considered for comparisons with our proposed new attack has been implemented in several existing studies, e.g., [61] and [25], in which the attacker manipulates the training process by injecting a fake device



(a) MNIST

(b) fashionMNIST

(c) CIFAR-10

Fig. 3: An illustration of the local model of a user device trained to classify images.

and sending fake local models to the server.

A. Implementation with PyTorch

The number of benign devices is set to $J = 5, 10, 15, 20, 25$. The number of iterations per communication round is set to $T_L = 10$. The maximum number of communication rounds is $T_{FL} = 200$. By default, one attacker is considered unless otherwise specified.

We implement the proposed GAE-based attack against the FL on an SVM model using PyTorch 1.12.1, Python 3.9.12 on a Linux workstation with an Intel(R) Core(TM) i7-9700K CPU@3.60GHz (8 cores) and 16 GB of DDR4 memory@2400 MHz.

The experiments are conducted on three datasets:

- The standard MNIST dataset comprises 60,000 training examples and 10,000 testing examples, which are grayscale images of handwritten digits from 1 to 10;
- The fashionMNIST dataset, which contains Zalando's article images (i.e., 28×28 grayscale images) in ten classes, including 60,000 examples for training and 10,000 examples for testing;
- The CIFAR-10 dataset, which contains 60,000 images with the size of 32×32 in ten classes (6,000 per class), 50,000 for training and 10,000 for testing.

At each user device, we use a standard quadratic optimization algorithm to train the SVM models based on the three datasets, namely, the standard MNIST, fashionMNIST, and CIFAR-10. The loss function used for training the SVM models is $F_j(\boldsymbol{\omega}_j(t)) = \frac{1}{2}\left\|\boldsymbol{\omega}_j(t)\right\|_2^2 + \frac{1}{D_j}\sum_{i=1}^{D_j}\max\left\{0, 1 - y_j^i(\beta_j + \boldsymbol{\omega}_j^T(t)x_j^i)\right\}$, where β_j is a feature parameter based on $\boldsymbol{\omega}_j(t)$ [62]. The global model $\boldsymbol{\omega}_g^a(t)$, which is trained at the server according to (3), is broadcast to all user devices for the training of $\boldsymbol{\omega}_j(t+1)$ in the next, $(t+1)$-th communication round. In particular, the choice of model architecture, NNs or SVMs, for training at benign user devices, does not alter the underlying principle of the proposed data-agnostic model poisoning attack. This is because the proposed attack hinges on the creation of malevolent local models that, upon integration, deteriorate the performance of the global model by augmenting the FL training loss. Our attack model incorporates a new adversarial GAE designed to fabricate these deleterious local models by leveraging benign local models overheard. The adversarial GAE is adept at extracting and utilizing the correlation features inherent in both the benign local models (which could be based on either NN or SVM architectures) and the global model.

Fig. 3 illustrates an example of label classification with the three datasets. In the MNIST dataset, three images labeled as "0" are misclassified as "6" while five images labeled as "6" are misclassified as "0", resulting in an FL accuracy of 73.3%. Similarly, the FL accuracy with the fashionMNIST and CIFAR-10 datasets is 76.7% and 63.3%, respectively. The FL is designed to improve classification accuracy, while the proposed GAE-based attack aims to reduce accuracy and cause label misclassification. The GAE encoder is a two-layer GCN network (i.e., $M = 2$) with a dropout layer to prevent overfitting. The GAE decoder is an inner product. We use the Adam optimizer with a learning rate of 0.01 to optimize the network. For all datasets, we use the same encoder, decoder and SVM models.

B. Performance Analysis

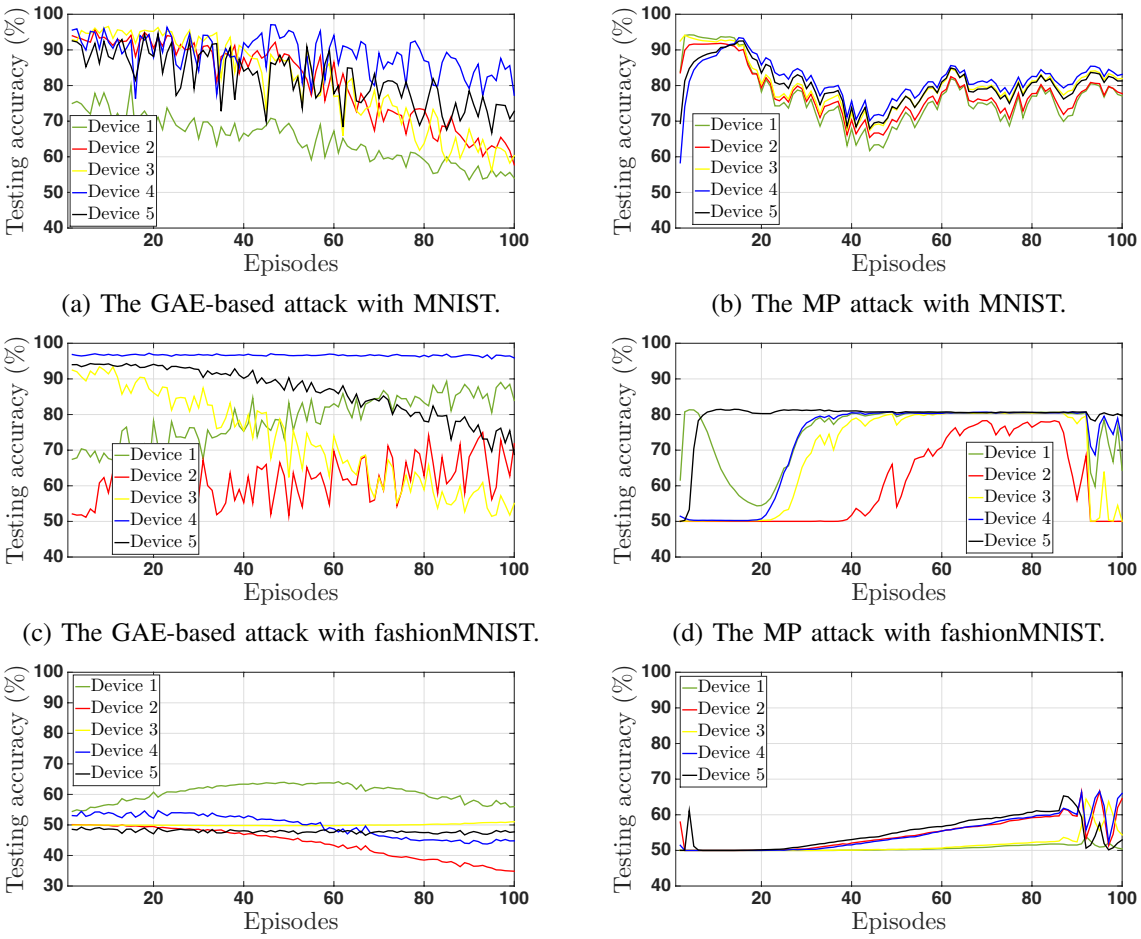
1) FL Accuracy under Attack: Fig. 4 plots the accuracy of the local models under the proposed GAE-based, data-agnostic, model poisoning attack on the MNIST, fashionMNIST, and CIFAR-10 datasets, where there are five benign devices (i.e., $J = 5$) and 100 communication rounds for the FL. The state-of-the-art model poisoning (MP) attack [60] is taken as the benchmark for our proposed attack, in which the attacker manipulates the training process by injecting a fake device and sending fake local models to the server. Since the MP attack in [60] shares the same objective as our proposed data-agnostic model poisoning approach, i.e., reducing the accuracy of FL, a comparison with this reference showcases the efficacy of our proposed method in the context of prevailing model poisoning attacks. For comparison purposes, Fig. 5 plots the accuracy of the benign local models without any attacks. In this scenario, the accuracy of the user device can be improved efficiently by FL and rapidly converge to 96%.

In Figs. 4(a) and 4(b), we show that when using the MNIST dataset, the accuracy of all five devices under the proposed GAE-based attack gradually decreases and fluctuates dramatically. The performance of devices 1 and 2 drops from 75% to 55% and from 92% to 59%, respectively. The accuracy of the model drops from 91% to 80% when exposed to the MP attack in which the performance of the five devices follows a similar pattern. This is because the new GAE-based attack reconstructs the adversarial adjacency matrix according to the individual features of the user devices. As a result, the attacker falsifies the local models to maximize the FL loss; see (9).

As shown in Figs. 4(c) and 4(d), the accuracy of device 3 and device 5 drops significantly by 37% and 24%, respectively, when using the fashionMNIST dataset and the proposed GAE-based attack. However, while the accuracy of devices 1 and 2 may slightly increase, their convergence rates are greatly slowed in comparison to Fig. 5. Additionally, the accuracy under the MP attack varies between 50% and 80%, with a minimal decrease in accuracy observed.

In Fig. 4(e), it can be seen that the proposed GAE-based attack with the CIFAR-10 dataset greatly hinders the performance of FL, as the accuracy of all four user devices falls below 50%. In contrast, the accuracy of all five devices under the MP attack is above 50%, as shown in Fig. 4(f). Furthermore, it can be observed that the accuracy with the CIFAR-10 dataset is generally lower than the performance with the MNIST and fashionMNIST datasets. This is because the CIFAR-10 dataset contains a more diverse set of images, which makes it more challenging to differentiate and label, leading to lower overall accuracy.

Fig. 6 illustrates the accuracy of the global model at the model aggregator. It can be observed that the proposed



(a) The GAE-based attack with MNIST.

(b) The MP attack with MNIST.

(c) The GAE-based attack with fashionMNIST.

(d) The MP attack with fashionMNIST.

(e) The GAE-based attack with CIFAR-10.

(f) The MP attack with CIFAR-10.

Fig. 4: Given 100 FL communication rounds and five benign user devices, we compare the local model testing accuracy under the GAE-based attack and the existing MP attack on the MNIST, fashionMNIST, and CIFAR-10 datasets.

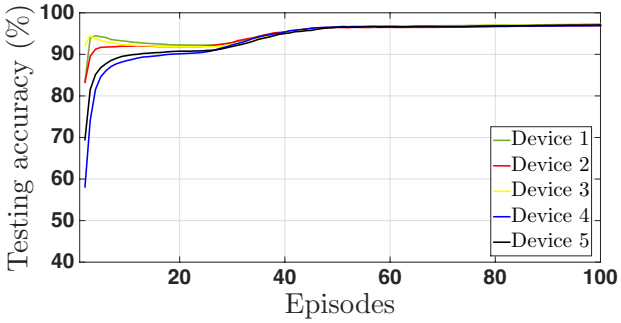


Fig. 5: The local model testing accuracy with no attack.

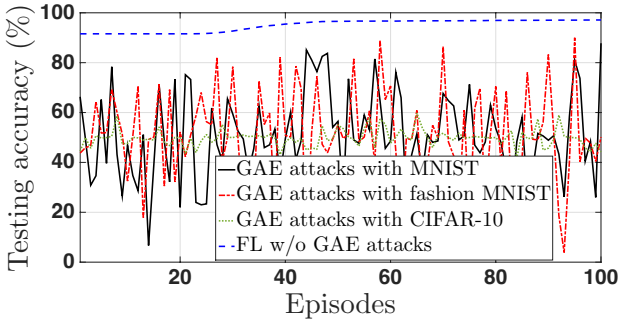
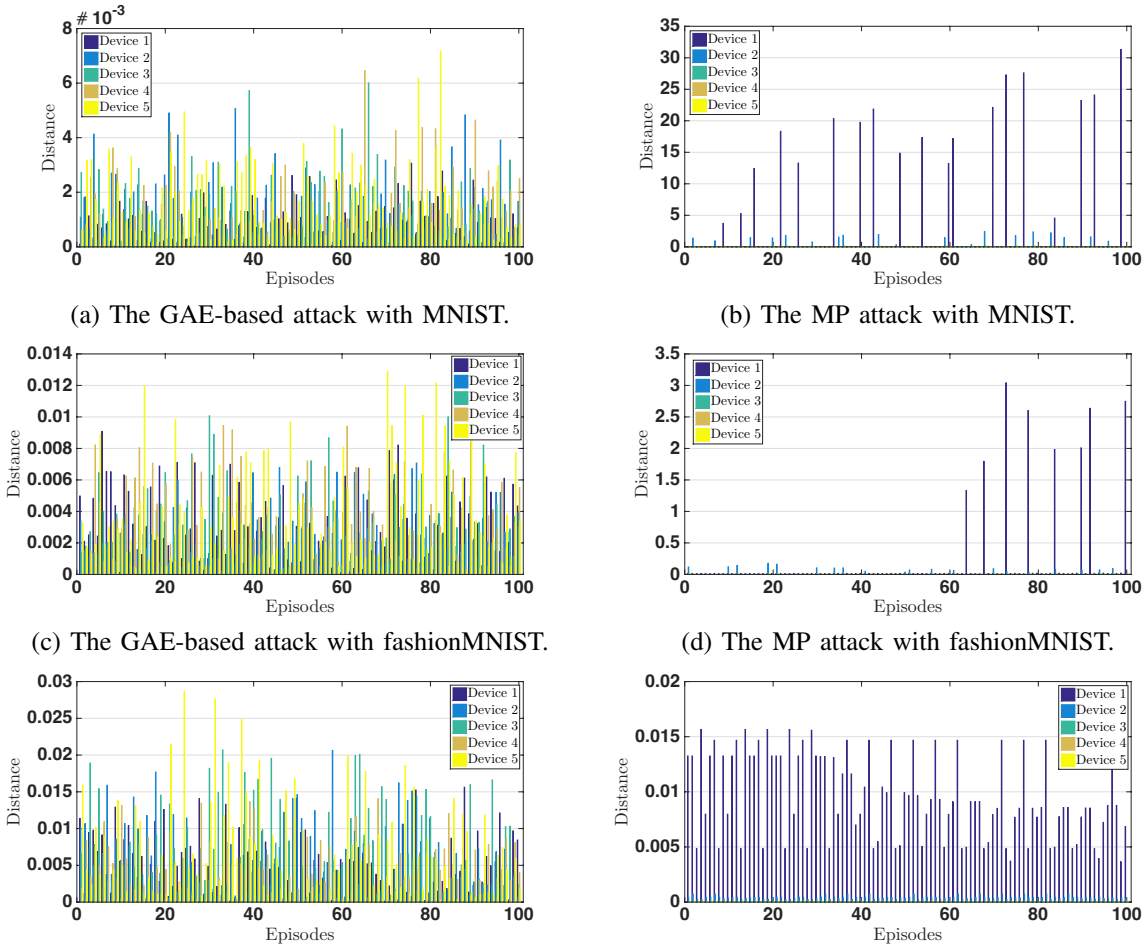


Fig. 6: The global model accuracy under the new attack.

GAE-based attack hinders the training convergence when compared to the performance without the attack. As a result of the infection of the local FL model, the accuracy with the MNIST, fashionMNIST, or CIFAR-10 dataset fluctuates around 82%, 81%, or 25%, respectively.

2) Detection of the Attack: Existing model poisoning attacks on FL aim to maximize the training loss of FL models. One way to detect these malicious attacks is to compare the distances (or differences) between the local models and the global model. A larger distance can be considered as an indication of a malicious local model, and the server can detect it accordingly. Both the Euclidean distance and cosine distance are commonly used metrics to assess the similarities between two vectors. Particularly, the Euclidean distance provides a straightforward, geometric measure of the absolute difference between vectors, which



(a) The GAE-based attack with MNIST.

(b) The MP attack with MNIST.

(c) The GAE-based attack with fashionMNIST.

(d) The MP attack with fashionMNIST.

(e) The GAE-based attack with CIFAR-10.

(f) The MP attack with CIFAR-10.

Fig. 7: Taking FL with five devices as an example, the Euclidean distances between the local models and the global model are presented, where device 1 is the attacker and launches the new GAE-based attack or the MP attack.

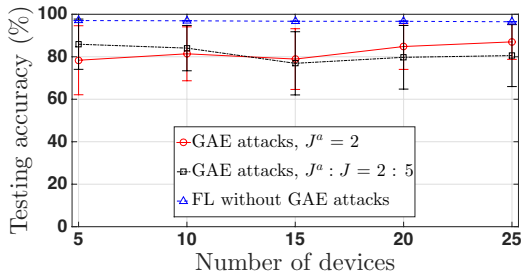
is useful in the considered scenarios since large deviations in the magnitude of model updates are often indicative of malicious model updates. For this reason, the Euclidean distance is considered in this paper, which is consistent with many recent studies, e.g., [8] and [41].

To evaluate the invisibility of the proposed GAE-based, data-agnostic, model poisoning attack Fig. 7 presents the Euclidean distance between the local models and the global model, with device 1 being the attacker. It can be observed that, in general, the local models with the MNIST dataset have the smallest distance compared to the other two datasets. This is expected as the handwritten digits in MNIST are relatively simple to recognize or falsify.

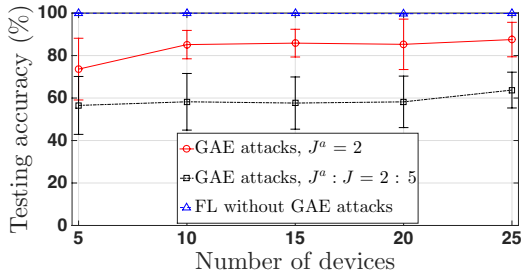
As shown in Figs. 7(a), 7(c), and 7(e), the Euclidean distances of the malicious local model (i.e., of device 1) generated by the GAE-based attack are below that of the benign local models. This makes it difficult for the aggregator to identify the attacker and defend against the attack. In contrast, the MP attack results in a significantly larger Euclidean distance between the malicious local model and the global model, making it easier to detect. This highlights the key advantage of the proposed GAE-based attack, which is designed to generate malicious local models based on the feature correlation between the benign local and global models, making the differences between the malicious local model and the benign local models indistinguishable.

3) Impact of Benign Local Model Number: Figs. 8(a), 8(b), and 8(c) show the average accuracy of the local models based on the MNIST, fashionMNIST, and CIFAR-10 datasets, respectively. The number of attackers is $J^a = 2$ by default. Otherwise, J^a increases proportionally with $J^a : J = 2 : 5$. It is observed that the new GAE-based attack reduces the average accuracy. As the number of devices increases, the average accuracy on MNIST, fashionMNIST, and CIFAR-10 drops by about 20%, 37%, and 12%, respectively, when $J = 15$.

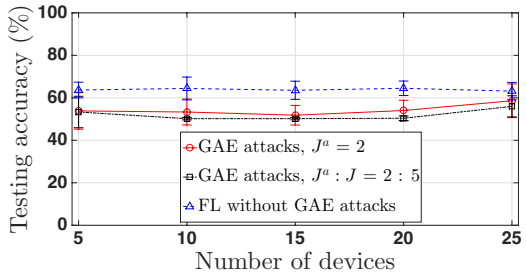
It is observed in Fig. 8 that on the three considered datasets, the average accuracy of FL under attack gradually increases as J grows from 5 to 25, while $J^a = 2$. This confirms that increasing the number of benign users improves the resistance of FL to the attacks. On the other hand, as the ratio of attackers to benign devices, i.e., $J^a : J$, increases, the proposed GAE-based attack can become increasingly effective and destructive.



(a) MNIST



(b) fashionMNIST



(c) CIFAR-10

Fig. 8: The average accuracy under the GAE-based attack based on the MNIST, fashionMNIST, or CIFAR-10 datasets, where the number of devices, i.e., J, ranges from 5 to 25. The number of attackers is $J^a = 2$ by default. Otherwise, J^a increases proportionally with J with $J^a : J = 2 : 5$.

4) Impact of Eavesdropped Local Models: Fig. 9 plots the average accuracy of the local models under the GAE-based attack based on the MNIST, fashionMNIST, or CIFAR-10 datasets, where the number of benign user devices that the attacker can eavesdrop on increases from 3 to 25. In general, the average accuracy of the local models falls with the growth of the eavesdropped benign user devices. The reason is that overhearing a greater number of benign local models results in capturing more correlation features of the models, leading to the generation of a malicious model for more effective poisoning. The average model accuracy drops substantially by 13.6%, 11.2% and 16.4% on the MNIST, fashionMNIST, and CIFAR-10 datasets, respectively.

5) Compared with Variational Autoencoder (VAE)-based Alternative: Fig. 10 illustrates the effects of the VAE-based attack on FL over 100 communication rounds, where five user devices and the MNIST dataset are considered. Fig. 10(a) reveals a consistent variation in local model testing accuracy under the VAE-based attack, with all five

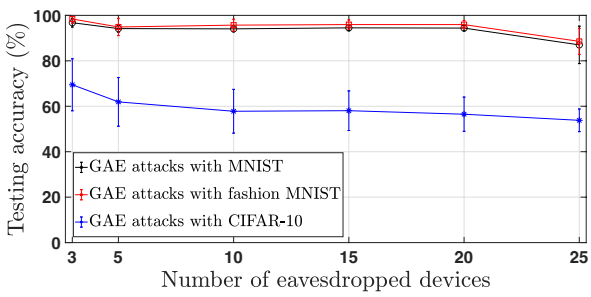
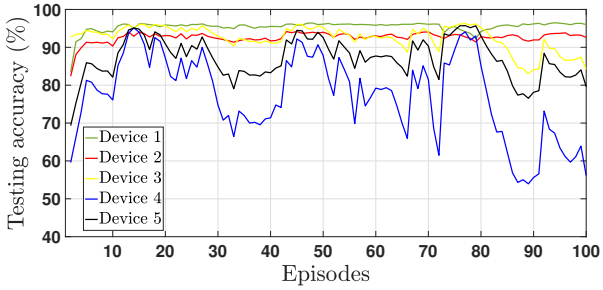
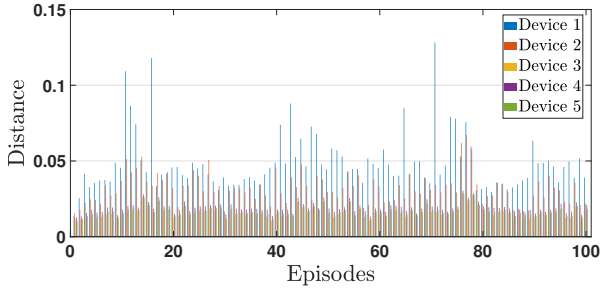


Fig. 9: The number of eavesdropped benign user devices' $\boldsymbol{\omega}_j(t)$ increases from 3 to 25, based on the MNIST, fashionMNIST, or CIFAR-10 datasets.



(a) The testing accuracy of local models.



(b) The Euclidean distances between the local models and the global model.

Fig. 10: The local model testing accuracy and the Euclidean distances under the VAE-based attack.

devices demonstrating analogous patterns. This is consistent with the observation made on the proposed GAE-based attack in Fig. 4. Fig. 10(b) shows the Euclidean distances between the local and global models under the VAE-based attack. A striking observation is that the malicious local model (from device 1) constructed via the VAE-based attack possesses a significantly larger Euclidean distance than the benign local models. This suggests that detecting the VAE-based attacks on the server side is feasible by assessing the Euclidean distance. The underlying reason is that the VAEs are general autoencoders and can handle high-dimensional and continuous data, such as images and audio. They do not capture graph structures inside data, as opposed to GAEs.

VI. Conclusion

In this paper, we have investigated a new, data-agnostic, model poisoning attack to FL, where the proposed ad-

versarial GAE gives rise to an infection of benign user devices and the FL training accuracy gradually drops. The adversarial GAE allows the attacker to extract the common underlying data features of the benign local models as well as their correlations to generate the malicious model with which the FL training loss is maximized. Since the malicious and benign local models are indistinguishable, it is difficult to identify the GAE-based attack at the server. We implemented the GAE-based attack against the FL on SVM models using PyTorch. Performances were evaluated based on the MNIST, fashionMNIST, and CIFAR-10 datasets.

APPENDIX A
PROOF OF THEOREM 1

Based on the Taylor expansion and the assumption that the gradient of the global loss function is L-Lipschitz continuous, it follows that

$$F(\boldsymbol{\omega}_g^a(t+1)) - F(\boldsymbol{\omega}_g^a(t)) \leq \nabla F\left(\boldsymbol{\omega}_g^a(t)\right)\left(\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\right) + \frac{L}{2}\left\|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\right\|^2. \tag{23}$$

By substituting (3) into (23), we obtain (24) on the top of the next page.

By substituting $\boldsymbol{\omega}_j(t+1) = \boldsymbol{\omega}_j(t) - \eta \nabla F_j(\boldsymbol{\omega}_j(t))$ and $\boldsymbol{\omega}_g^a(t) = \sum_{j=1}^{J} \frac{D_j}{D}\boldsymbol{\omega}_j(t) + \frac{D_a}{D}\left(\boldsymbol{\omega}^a(t+1) - \boldsymbol{\omega}^a(t)\right)$ into (24), the expectation of $F(\boldsymbol{\omega}_g^a(t+1))$ can be given by

$$\mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t+1))\right] \leq F(\boldsymbol{\omega}_g^a(t)) - \eta\left\|\nabla F\left(\boldsymbol{\omega}_g^a(t)\right)\right\|^2 + \frac{\eta^2 L}{2}\mathbb{E}\left[\left\|\sum_{j=1}^{J}\frac{D_j}{D}\nabla F_j(\boldsymbol{\omega}_j(t)) + \frac{D_a}{D}\nabla F_J(\boldsymbol{\omega}^a(t))\right\|^2\right].$$

It is generally assumed in FL that

$$\mathbb{E}\left[\sum_{j=1}^{J}\frac{D_j}{D}\nabla F_j(\boldsymbol{\omega}_j(t)) + \frac{D_a}{D}\nabla F_j(\boldsymbol{\omega}^a(t))\right] = \nabla F(\boldsymbol{\omega}_g^a(t)).$$

According to Jensen's inequality, $\mathbb{E}(x^2) \leq \mathbb{E}^2(x)$. Then,

$$\mathbb{E}\left[\left\|\sum_{j=1}^{J}\frac{D_j}{D}\nabla F_j(\boldsymbol{\omega}_j(t)) + \frac{D_a}{D}\nabla F_j(\boldsymbol{\omega}^a(t))\right\|^2\right] \leq \left\|\nabla F(\boldsymbol{\omega}_g^a(t))\right\|^2. \tag{26}$$

As a result, $\mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t+1))\right]$ is bounded by

$$\begin{aligned}
&\mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t+1))\right] \\
&\leq \mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t))\right] + \left(\frac{\eta^2 L}{2} - \eta\right)\left\|\nabla F\left(\boldsymbol{\omega}_g^a(t)\right)\right\|^2 && \text{(27a)}\\
&\overset{\eta=\frac{1}{L}}{=} \mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t))\right] - \frac{\eta}{2}\left\|\nabla F\left(\boldsymbol{\omega}_g^a(t)\right)\right\|^2. && \text{(27b)}
\end{aligned}$$

Next, we derive the relationship between $\nabla F(\boldsymbol{\omega}_g^a(t))$ and $\nabla F(\boldsymbol{\omega}_g(t))$. Since $d(\boldsymbol{\omega}_j^a, \boldsymbol{\omega}_g^a) = \|\boldsymbol{\omega}_j^a - \boldsymbol{\omega}_g^a\| \leq d_T$, we have

$$\begin{aligned}
&\|\boldsymbol{\omega}^a(t+1) - \boldsymbol{\omega}^a(t)\| \\
&\leq \|\boldsymbol{\omega}^a(t+1) - \boldsymbol{\omega}_g^a(t+1)\| + \|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}^a(t)\| \\
&= d_T + \|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}^a(t)\| \\
&\leq d_T + \|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\| + \|\boldsymbol{\omega}_g^a(t) - \boldsymbol{\omega}^a(t)\| && \text{(28a)}\\
&= 2d_T + \|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\|, && \text{(28b)}
\end{aligned}$$

where (28a) is based on the triangle inequality.

Likewise, we also have

$$\begin{aligned}
&\|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\| \\
&= \left\|\left(\boldsymbol{\omega}_g(t+1) + \frac{D_a}{D}\boldsymbol{\omega}^a(t+1)\right) - \left(\boldsymbol{\omega}_g(t) + \frac{D_a}{D}\boldsymbol{\omega}^a(t)\right)\right\| && \text{(29a)}\\
&\leq \|\boldsymbol{\omega}_g(t+1) - \boldsymbol{\omega}_g(t)\| + \frac{D_a}{D}\|\boldsymbol{\omega}^a(t+1) - \boldsymbol{\omega}^a(t)\| && \text{(29b)}\\
&\leq \|\boldsymbol{\omega}_g(t+1) - \boldsymbol{\omega}_g(t)\| \\
&\quad + \frac{D_a}{D}\left(2d_T + \|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\|\right). && \text{(29c)}
\end{aligned}$$

By reorganizing (29c), it follows that

$$\begin{aligned}
\|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\| &\leq \frac{D}{D - D_a} \times && \text{(30a)}\\
&\quad \left\|\boldsymbol{\omega}_g(t+1) - \boldsymbol{\omega}_g(t)\right\| + \frac{2D_a d_T}{D - D_a}. && \text{(30b)}
\end{aligned}$$

By substituting (30) into (28), it follows that

$$\begin{aligned}
\|\boldsymbol{\omega}^a(t+1) - \boldsymbol{\omega}^a(t)\| &\leq \frac{D}{D - D_a} \times && \text{(31a)}\\
&\quad \|\boldsymbol{\omega}_g(t+1) - \boldsymbol{\omega}_g(t)\| + \frac{2Dd_T}{D - D_a}. && \text{(31b)}
\end{aligned}$$

By taking expectation on both sides of (30), we have

$$\begin{aligned}
\mathbb{E}[\|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\|] &\leq \frac{D}{D - D_a} \times \\
&\quad \mathbb{E}[\|\boldsymbol{\omega}_g(t+1) - \boldsymbol{\omega}_g(t)\|] + \frac{2D_a d_T}{D - D_a}.
\end{aligned} \tag{32}$$

Note that $\mathbb{E}[\|\boldsymbol{\omega}_g^a(t+1) - \boldsymbol{\omega}_g^a(t)\|] = \eta\|\nabla F(\boldsymbol{\omega}_g^a(t))\|$ and $\mathbb{E}[\|\boldsymbol{\omega}_g(t+1) - \boldsymbol{\omega}_g(t)\|] = \eta\|\nabla F(\boldsymbol{\omega}_g(t))\|$. By substituting them into both sides of (32) and then reorganizing (32), we obtain

$$\|\nabla F(\boldsymbol{\omega}_g^a(t))\| \leq \frac{D}{D - D_a}\|\nabla F(\boldsymbol{\omega}_g(t))\| + \frac{2D_a d_T}{(D - D_a)\eta}. \tag{33}$$

By substituting (33) into (27), it follows

$$\begin{aligned}
&\mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t+1))\right] - \mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t))\right] \\
&\leq -\frac{\eta}{2}\left\|\nabla F\left(\boldsymbol{\omega}_g^a(t)\right)\right\|^2 && \text{(34a)}\\
&\leq -\frac{\eta}{2}\left(\frac{D}{D - D_a}\|\nabla F(\boldsymbol{\omega}_g(t))\| + \frac{2D_a d_T}{(D - D_a)\eta}\right)^2 && \text{(34b)}\\
&\leq -\frac{\eta D^2}{2(D - D_a)^2}\|\nabla F(\boldsymbol{\omega}_g(t))\|^2. && \text{(34c)}
\end{aligned}$$

Considering the Polyak-Lojasiewicz condition, we have

$$\|\nabla F(\boldsymbol{\omega}_g(t)\|^2 \geq 2\rho\left(F(\boldsymbol{\omega}_g(t) - F(\boldsymbol{\omega}_g^*)\right). \tag{35}$$

Given the convex loss function of SVM models, it follows:

$$\begin{aligned}
F(\boldsymbol{\omega}_g^a(t)) &\leq F(\boldsymbol{\omega}_g(t)) + F\left(\frac{D_a}{D}\boldsymbol{\omega}^a(t)\right) \\
&\leq F(\boldsymbol{\omega}_g(t)) + \frac{D_a}{D}F(\boldsymbol{\omega}^a(t)).
\end{aligned} \tag{36}$$

By substituting (35) and (36) into (34), we have

$$\mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t+1))\right] - \mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t))\right]$$

$$F(\boldsymbol{\omega}_g^a(t+1)) - F(\boldsymbol{\omega}_g^a(t)) \leq \nabla F\left(\boldsymbol{\omega}_g^a(t)\right)\left[\frac{1}{D}\sum_{j=1}^{J} D_j(\boldsymbol{\omega}_j(t+1)-\boldsymbol{\omega}_j(t)) + \frac{D_a}{D}(\boldsymbol{\omega}^a(t+1)-\boldsymbol{\omega}^a(t))\right]$$
$$+ \frac{L}{2}\left\|\frac{1}{D}\sum_{j=1}^{J} D_j(\boldsymbol{\omega}_j(t+1)-\boldsymbol{\omega}_j(t)) + \frac{D_a}{D}(\boldsymbol{\omega}^a(t+1)-\boldsymbol{\omega}^a(t))\right\|^2 \tag{24a}$$
$$= \frac{\eta^2 L}{2}\left\|\frac{1}{D}\sum_{j=1}^{J} D_j(t)\nabla F_j(\boldsymbol{\omega}_j(t)) + \frac{D_a}{D}\nabla F_a(\boldsymbol{\omega}^a(t))\right\|^2$$
$$- \eta\nabla F\left(\boldsymbol{\omega}_g^a(t)\right)\left[\frac{1}{D(t)}\sum_{j=1}^{J} D_j(t)\nabla F_j(\boldsymbol{\omega}_j(t)) + \frac{D_a}{D}\nabla F_a(\boldsymbol{\omega}^a(t))\right]. \tag{24b}$$

$$\leq -\frac{\eta\rho D^2}{(D-D_a)^2}\left(F(\boldsymbol{\omega}_g(t) - F(\boldsymbol{\omega}_g^*)\right) \tag{37a}$$
$$\leq -\frac{\eta\rho D^2}{(D-D_a)^2}\left(F(\boldsymbol{\omega}_g^a(t) - \frac{D_a}{D}F(\boldsymbol{\omega}^a(t)) - F(\boldsymbol{\omega}_g^*)\right). \tag{37b}$$

By restructuring (37), we have

$$\begin{aligned}
&\mathbb{E}\left[F(\boldsymbol{\omega}_g^a\,(t+1))\right] - F(\boldsymbol{\omega}_g^*) \\
&\leq \left(1 - \frac{\rho\eta D^2}{(D-D_a)^2}\right)\left(\left[F(\boldsymbol{\omega}_g^a(t))\right] - F(\boldsymbol{\omega}_g^*)\right) \\
&\quad + \frac{\rho\eta D D_a}{(D-D_a)^2}F\left(\boldsymbol{\omega}^a(t)\right) \\
&\leq (1-\zeta)\left(\left[F(\boldsymbol{\omega}_g^a(t))\right] - F(\boldsymbol{\omega}_g^*)\right) + \frac{\rho\eta D D_a}{(D-D_a)^2}F^{\max},
\end{aligned} \tag{38}$$

where $\zeta = 1 - \frac{\rho\eta D^2}{(D-D_a)^2}$, and the second inequality is due to the constrained problem in (5a) and (5b) has a maximum value, i.e., $F(\boldsymbol{\omega}^a(t)) \leq F^{\max}$.

Finally, applying mathematical induction upon (38) gives

$$\begin{aligned}
&\mathbb{E}\left[F(\boldsymbol{\omega}_g^a(t))\right] - F(\boldsymbol{\omega}_g^*) \\
&\leq \left[\mathbb{E}\left[F(\boldsymbol{\omega}_g(0))\right] - F(\boldsymbol{\omega}_g^*)\right]\zeta^t + \frac{1-\zeta^t}{1-\zeta}\cdot\frac{\rho\eta D D_a}{(D-D_a)^2}F^{\max} \\
&= \Theta\zeta^t + \frac{1-\zeta^t}{1-\zeta}\cdot\frac{\rho\eta D D_a}{(D-D_a)^2}F^{\max},
\end{aligned} \tag{39}$$

which concludes this proof.

ACKNOWLEDGEMENTS

This work was supported by the CISTER Research Unit (UIDP/UIDB/04234/2020), project ADANET (PTDC/EEICOM/3362/2021) and project IBEX (PTDC/CCI-COM/4280/2021), financed by National Funds through FCT/MCTES (Portuguese Foundation for Science and Technology); and also supported in part by the AXA Research Fund (AXA Chair for Internet of Everything at Koç University), as well as the U.S National Science Foundation under Grants CNS-2128448 and ECCS-2335876.

REFERENCES

[1] Z. Zhang, L. Wu, C. Ma, J. Li, J. Wang, Q. Wang, and S. Yu, "LSFL: A lightweight and secure federated learning scheme for edge computing," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 365–379, 2022.

[2] K. Li, Y. Cui, W. Li, T. Lv, X. Yuan, S. Li, W. Ni, M. Simsek, and F. Dressler, "When internet of things meets metaverse: Convergence of physical and cyber worlds," *IEEE Internet Things J.*, vol. 10, no. 5, pp. 4148–4173, 2022.

[3] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for internet of things: Recent advances, taxonomy, and open challenges," *IEEE Comm. Surveys & Tutorials*, vol. 23, no. 3, pp. 1759–1799, 2021.

[4] H. Zhou, G. Yang, Y. Huang, H. Dai, and Y. Xiang, "Privacy-preserving and verifiable federated learning framework for edge computing," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 565–580, 2022.

[5] M. S. Jere, T. Farnan, and F. Koushanfar, "A taxonomy of attacks on federated learning," *IEEE Security Privacy*, vol. 19, no. 2, pp. 20–28, 2020.

[6] S. A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of things intrusion detection: Centralized, on-device, or federated learning?" *IEEE Network*, vol. 34, no. 6, pp. 310–317, 2020.

[7] Z. Tian, L. Cui, J. Liang, and S. Yu, "A comprehensive survey on poisoning attacks and countermeasures in machine learning," *ACM Computing Surveys*, vol. 55, no. 8, pp. 1–35, 2022.

[8] Z. Zhang, X. Cao, J. Jia, and N. Z. Gong, "FLDetector: Defending federated learning against model poisoning attacks via detecting malicious clients," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 2545–2555.

[9] M. Fang, X. Cao, J. Jia, and N. Gong, "Local model poisoning attacks to byzantine-robust federated learning," in *Proceedings of the USENIX Security Symposium*, 2020, pp. 1605–1622.

[10] L. Lyu, H. Yu, J. Zhao, and Q. Yang, "Threats to federated learning," in *Federated Learning*. Springer, 2020, pp. 3–16.

[11] V. Shejwalkar, A. Houmansadr, P. Kairouz, and D. Ramage, "Back to the drawing board: A critical evaluation of poisoning attacks on production federated learning," in *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE, 2022, pp. 1354–1371.

[12] H. Cai, V. W. Zheng, and K. C.-C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," *IEEE Trans. Knowledge Data Engineering*, vol. 30, no. 9, pp. 1616–1637, 2018.

[13] K. Li, X. Yuan, J. Zheng, W. Ni, and M. Guizani, "Exploring adversarial graph autoencoders to manipulate federated learning in the internet of things," in *Proceedings of the IEEE International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 2023, pp. 898–903.

[14] J. Zhao, H. Zhu, F. Wang, R. Lu, Z. Liu, and H. Li, "PVD-FL: A privacy-preserving and verifiable decentralized federated learning framework," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 2059–2073, 2022.

[15] X. Zhou, W. Liang, W. Li, K. Yan, S. Shimizu, I. Kevin, and K. Wang, "Hierarchical adversarial attacks against graph-neural-network-based IoT network intrusion detection system," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9310–9319, 2021.

[16] A. Singh and B. Sikdar, "Adversarial attack for deep learning based IoT appliance classification techniques," in *Proceedings of IEEE World Forum on Internet of Things*. IEEE, 2021, pp. 657–662.

[17] Z. Bao, Y. Lin, S. Zhang, Z. Li, and S. Mao, "Threat of adversarial attacks on DL-based IoT device identification," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 9012–9024, 2021.

[18] A. Rahman, M. S. Hossain, N. A. Alrajeh, and F. Alsolami, "Adversarial examples – security threats to COVID-19 deep learning systems in medical IoT devices," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 9603–9610, 2020.

[19] A. Abusnaina, A. Khormali, H. Alasmary, J. Park, A. Anwar, and A. Mohaisen, "Adversarial learning attacks on graph-based IoT malware detection systems," in *Proceedings of IEEE International Conference on Distributed Computing Systems*. IEEE, 2019, pp. 1296–1305.

[20] A. Talpur and M. Gurusamy, "Adversarial attacks against deep reinforcement learning framework in internet of vehicles," in *Proceedings of IEEE Globecom Workshops*. IEEE, 2021, pp. 1–6.

[21] Z. Chen, P. Tian, W. Liao, and W. Yu, "Zero knowledge clustering based adversarial mitigation in heterogeneous federated learning," *IEEE Trans. Network Science Engineering*, vol. 8, no. 2, pp. 1070–1083, 2020.

[22] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, "Data poisoning attacks against federated learning systems," in *Proceedings of the European Symposium on Research in Computer Security*. Springer, 2020, pp. 480–501.

[23] J. Gao, B. Hou, X. Guo, Z. Liu, Y. Zhang, K. Chen, and J. Li, "Secure aggregation is insecure: Category inference attack on federated learning," *IEEE Trans. Dependable Secure Computing*, vol. 20, no. 1, pp. 147–160, 2023.

[24] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee, and D. Papailiopoulos, "Attack of the tails: Yes, you really can backdoor federated learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 16 070–16 084, 2020.

[25] J. Zhang, B. Chen, X. Cheng, H. T. T. Binh, and S. Yu, "PoisonGAN: Generative poisoning attacks against federated learning in edge computing systems," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3310–3322, 2020.

[26] J. Zhang, J. Chen, D. Wu, B. Chen, and S. Yu, "Poisoning attack in federated learning using generative adversarial nets," in *Proceedings of the IEEE International Conference on Trust, Security and Privacy in Computing and Communications/IEEE International Conference on Big Data Science and Engineering*. IEEE, 2019, pp. 374–380.

[27] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, "Beyond inferring class representatives: User-level privacy leakage from federated learning," in *Proceedings of the IEEE Conference on Computer Communications*. IEEE, 2019, pp. 2512–2520.

[28] M. Song, Z. Wang, Z. Zhang, Y. Song, Q. Wang, J. Ren, and H. Qi, "Analyzing user-level privacy attack against federated learning," *IEEE J. Selected Areas Comm.*, vol. 38, no. 10, pp. 2430–2444, 2020.

[29] D. Caldarola, M. Mancini, F. Galasso, M. Ciccone, E. Rodolà, and B. Caputo, "Cluster-driven graph federated learning over multiple domains," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2749–2758.

[30] G. Mei, Z. Guo, S. Liu, and L. Pan, "Sgnn: A graph neural network based federated learning approach by hiding structure," in *Proceedings of the IEEE International Conference on Big Data*. IEEE, 2019, pp. 2560–2568.

[31] X. Guo, Z. Liu, J. Li, J. Gao, B. Hou, C. Dong, and T. Baker, "Verifl: Communication-efficient and fast verifiable aggregation for federated learning," *IEEE Trans. Info. Forensics Security*, vol. 16, pp. 1736–1751, 2020.

[32] J. Zheng, K. Li, N. Mhaisen, W. Ni, E. Tovar, and M. Guizani, "Federated learning for online resource allocation in mobile edge computing: A deep reinforcement learning approach," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2023, pp. 1–6.

[33] ——, "Exploring deep-reinforcement-learning-assisted federated learning for online resource allocation in privacy-preserving edgeiot," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 21 099–21 110, 2022.

[34] Y. Huang, S. Gupta, Z. Song, K. Li, and S. Arora, "Evaluating gradient inversion attacks and defenses in federated learning," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 34, pp. 7232–7241, 2021.

[35] T. Wang, Y. Li, Y. Wu, and T. Q. Quek, "Secrecy driven federated learning via cooperative jamming: An approach of latency minimization," *IEEE Trans. Emerg. Topics Comput.*, vol. 10, no. 4, pp. 1687–1703, 2022.

[36] Y.-A. Xie, J. Kang, D. Niyato, N. T. T. Van, N. C. Luong, Z. Liu, and H. Yu, "Securing federated learning: A covert communication-based approach," *IEEE Netw.*, 2022.

[37] N. Aviram, S. Schinzel, J. Somorovsky, N. Heninger, M. Dankel, J. Steube, L. Valenta, D. Adrian, J. A. Halderman, V. Dukhovni *et al.*, "{DROWN}: Breaking {TLS} using {SSLv2}," in *Proceedings of the USENIX Security Symposium*, 2016, pp. 689–706.

[38] S. Hebrok, S. Nachtigall, M. Maehren, N. Erinola, R. Merget, J. Somorovsky, and J. Schwenk, "We really need to talk about session tickets: A {Large-Scale} analysis of cryptographic dangers with {TLS} session tickets," in *Proceedings of the USENIX Security Symposium*, 2023, pp. 4877–4894.

[39] D. Diaz-Sanchez, A. Marín-Lopez, F. A. Mendoza, P. A. Cabarcos, and R. S. Sherratt, "TLS/PKI challenges and certificate pinning techniques for iot and m2m secure communications," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3502–3531, 2019.

[40] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[41] X. Li, Z. Qu, S. Zhao, B. Tang, Z. Lu, and Y. Liu, "Lomar: A local defense against poisoning attack on federated learning," *IEEE Trans. Dependable Secure Computing*, vol. 20, no. 1, pp. 437–450, 2023.

[42] C. Tran, F. Fioretto, and P. Van Hentenryck, "Differentially private and fair deep learning: A lagrangian dual approach," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, 2021, pp. 9932–9939.

[43] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[44] D. Zhu, Y. Ma, and Y. Liu, "Anomaly detection with deep graph autoencoders on attributed networks," in *Proceedings of the IEEE Symposium on Computers and Communications*. IEEE, 2020, pp. 1–6.

[45] Y. Wang, B. Xu, M. Kwak, and X. Zeng, "A simple training strategy for graph autoencoder," in *Proceedings of International Conference on Machine Learning Computing*, 2020, pp. 341–345.

[46] K. Li, W. Ni, X. Yuan, A. Noor, and A. Jamalipour, "Deep-graph-based reinforcement learning for joint cruise control and task offloading for aerial edge internet of things (edgeiot)," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 21 676–21 686, 2022.

[47] S. Pan, R. Hu, S.-f. Fung, G. Long, J. Jiang, and C. Zhang, "Learning graph embedding with adversarial training methods," *IEEE Trans. Cybernetics*, vol. 50, no. 6, pp. 2475–2487, 2020.

[48] C. Qiu, Z. Huang, W. Xu, and H. Li, "Fast community detection based on graph autoencoder reconstruction," in *Proceedings of the International Conference on Big Data Analytics*. IEEE, 2022, pp. 265–271.

[49] J. J. Molitierno, *Applications of Combinatorial Matrix Theory to Laplacian Matrices of Graphs*. CRC Press, 2016.

[50] K. Lange, "Singular value decomposition," in *Numerical Analysis for Statisticians*. Springer, 2010, pp. 129–142.

[51] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Trans. Info. Forensics Security*, vol. 15, pp. 3454–3469, 2020.

[52] S. Truex, L. Liu, K.-H. Chow, M. E. Gursoy, and W. Wei, "LDP-fed: Federated learning with local differential privacy," in *Proceedings of the ACM International Workshop on Edge Systems, Analytics and Networking*. ACM, 2020, pp. 61–66.

[53] Y. Zhao, J. Zhao, M. Yang, T. Wang, N. Wang, L. Lyu, D. Niyato, and K.-Y. Lam, "Local differential privacy-based federated learning for internet of things," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 8836–8853, 2020.

[54] M. O'Searcoid, *Metric Spaces*. Springer Science & Business Media, 2006.

[55] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition," in *Proceedings of Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2016, pp. 795–811.

[56] L. Deng, "The mnist database of handwritten digit images for machine learning research [best of the web]," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 141–142, 2012.

[57] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," 2017. [Online]. Available: arXiv preprint arXiv:1708.07747

[58] T. D. Nguyen, P. Rieger, H. Chen, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen, A. Mirhoseini, S. Zeitouni *et al.*, "FLAME: Taming backdoors in federated learning," in *Proceedings of the USENIX Security Symposium*, 2022, pp. 1415–1432.

[59] D. Cao, S. Chang, Z. Lin, G. Liu, and D. Sun, "Understanding distributed poisoning attack in federated learning," in *Proceedings of the IEEE International Conference on Parallel Distributed Systems*. IEEE, 2019, pp. 233–239.
[60] T. D. Nguyen, P. Rieger, M. Miettinen, and A.-R. Sadeghi, "Poisoning attacks on federated learning-based IoT intrusion detection system," in *Proceedings of the Workshop on Decentralized IoT Systems and Security*. NDSS, 2020, pp. 1–7.
[61] X. Cao and N. Z. Gong, "MPAF: Model poisoning attacks to federated learning based on fake clients," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2022, pp. 3396–3404.
[62] H. H. Yang, Z. Liu, T. Q. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 317–333, 2019.

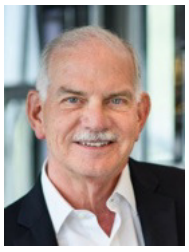
Kai Li (S'09–M'14–SM'20) received the B.E. degree from Shandong University, China, in 2009, the M.S. degree from The Hong Kong University of Science and Technology, Hong Kong, in 2010, and the Ph.D. degree in computer science from The University of New South Wales, Sydney, NSW, Australia, in 2014. Currently, he is a Visiting Research Scientist with the Division of Electrical Engineering, Department of Engineering, University of Cambridge, U.K., and a Senior Research Scientist with the CISTER Research Centre, Porto, Portugal. He is also a CMU-Portugal Research Fellow, jointly supported by Carnegie Mellon University, Pittsburgh, PA, USA, and the Foundation for Science and Technology (FCT), Lisbon, Portugal. In 2022, he was a Visiting Research Scholar with the CyLab Security and Privacy Institute, CMU. Prior to this, he was a Post-Doctoral Research Fellow with the SUTD-MIT International Design Centre, Singapore University of Technology and Design, Singapore, from 2014 to 2016. He has also held positions as a Visiting Research Assistant with the ICT Centre, CSIRO, Brisbane, QLD, Australia, from 2012 to 2013, and a Research Assistant with the Mobile Technologies Centre, The Chinese University of Hong Kong, Hong Kong, from 2010 to 2011. He has been an Associate Editor for the Nature Computer Science journal (Springer) since 2023, the Computer Communications journal (Elsevier) and Ad Hoc Networks journal (Elsevier) since 2021, and IEEE ACCESS journal from 2018 to 2024.

Jingjing Zheng (S'22) is currently near to completion of pursuing the Ph.D. degree in electrical and computer engineering with the University of Porto, Porto, Portugal. He is a Student Researcher with CISTER Research Center, Porto, Portugal. In 2022, he was a Visiting Research Scholar with the CyLab Security and Privacy Institute, CMU. His main research interests include federated learning, machine learning security, and edge computing.

Xin Yuan (M'19) received the B.E. degree in Communication Engineering from Taiyuan University of Technology, Shanxi, China, in 2013, and the dual Ph.D. degree from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, and the University of Technology Sydney (UTS), Sydney, Australia, in 2019 and 2020, respectively. She is currently a Senior Research Scientist at CSIRO, Sydney, NSW, Australia. She is also an Adjunct Senior Lecturer at the University of New South Wales (UNSW). Her research interests include machine learning and optimization, and their applications to the integrity, efficiency, and security of intelligent systems and networks. She has been an Editor for IEEE Transactions on Vehicular Technology since 2023.

Wei Ni (M'09–SM'15–F'24) received the B.E. and Ph.D. degrees in electronic engineering from Fudan University, Shanghai, China, in 2000 and 2005, respectively. He is currently a Principal Research Scientist with CSIRO, Sydney, Australia. He is also a Conjoint Professor with The University of New South Wales, an Adjunct Professor with the University of Technology Sydney, and a Honorary Professor with Macquarie University. He was a Post-Doctoral Research Fellow with Shanghai Jiaotong University from 2005 to 2008; a Deputy Project Manager with the Bell Laboratories, Alcatel/AlcatelLucent, from 2005 to 2008; and a Senior Researcher with Devices Research and Development, Nokia, from 2008 to 2009. He has authored nine book chapters, more than 300 journal articles, more than 100 conference papers, 26 patents, and ten standard proposals accepted by IEEE. His research interests include machine learning, online learning, stochastic optimization, and their applications to system efficiency and integrity. He served as the Secretary and the Vice-Chair/Chair for the IEEE VTS NSW Chapter from 2015 to 2022, the Track Chair for VTC-Spring 2017, the Track Co-Chair for IEEE VTC-Spring 2016, the Publication Chair for BodyNet 2015, and the Student Travel Grant Chair for WPMC 2014. He has been an Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS since 2018, an Editor of IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY since 2022, and an Editor of IEEE TRANSCTIONS ON INFORMATION FORENSICS AND SECURITY and IEEE COMMUNICATIONS SURVEYS AND TUTORIALS since 2024.

Ozgur B. Akan (F'16) received the PhD from the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, in 2004. He is currently the Head of Internet of Everything (IoE) Group, with the Department of Engineering, University of Cambridge, UK and the Director of Centre for neXt-generation Communications (CXC), Koç University, Turkey. His research interests include wireless, nano, and molecular communications and Internet of Everything.



H. Vincent Poor (S'72–M'77–SM'82–F'87) received the Ph.D. degree in EECS from Princeton University in 1977. From 1977 until 1990, he was on the faculty of the University of Illinois at Urbana-Champaign. Since 1990 he has been on the faculty at Princeton, where he is currently the Michael Henry Strater University Professor. During 2006 to 2016, he served as the dean of Princeton's School of Engineering and Applied Science. He has also held visiting appointments at several other universities, including most recently at Berkeley and Cambridge. His research interests are in the areas of information theory, machine learning and network science, and their applications in wireless networks, energy systems and related fields. Among his publications in these areas is the recent book *Machine Learning and Wireless Communications*. (Cambridge University Press, 2022). Dr. Poor is a member of the National Academy of Engineering and the National Academy of Sciences and is a foreign member of the Chinese Academy of Sciences, the Royal Society, and other national and international academies. He received the IEEE Alexander Graham Bell Medal in 2017.