Computers & Security 130 (2023) 103270

Contents lists available at ScienceDirect

Computers & Security

journal homepage: www.elsevier.com/locate/cose

A Novel Data Poisoning Attack in Federated Learning based on Inverted Loss Function

Prajjwal Gupta[a], Krishna Yadav[b], Brij B. Gupta[g,h,i], Mamoun Alazab[c], Thippa Reddy Gadekallu[d,e,f,*]

[a] School of Computer Engineering, Vellore Institute of Technology, Vellore, India
[b] Department of Computer Engineering, National Institute of Technology Kurukshetra, India
[c] Faculty of Science and Technology, Charles Darwin University, NT, Australia
[d] Zhongda Group, Haiyan County, Jiaxing City, Zhejiang Province 314312, China
[e] Department of Electrical and Computer Engineering, Lebanese American University, Byblos, Lebanon
[f] School of Information Technology and Engineering, Vellore Institute of Technology, India
[g] International Center for AI and Cyber Security Research and Innovations, & Department of Computer Science and Information Engineering, Asia University, Taichung 413, Taiwan
[h] Symbiosis Centre for Information Technology (SCIT), Symbiosis International University, Pune, India, & Lebanese American University, Beirut, 1102, Lebanon
[i] School of Information Technology, Skyline University College, P.O. Box 1797, Sharjah, United Arab Emirates, & University of Petroleum and Energy Studies (UPES), Dehradun, Uttarakhand, India

ARTICLE INFO

Article history:
Received 28 November 2022
Revised 7 April 2023
Accepted 19 April 2023
Available online 24 April 2023

Keywords:
Data poisoning
Adversarial Machine Learning
Inverted gradients
Federated Learning

ABSTRACT

Data poisoning attack is one of the common attacks that decreases the performance of a model in edge machine learning. The mechanism used in most of the existing data poisoning attacks diverges the gradients to a minimal extent which prevents models from achieving minima. In our approach, we have come with a new data poisoning attack that inverts the loss function of a benign model. The inverted loss function is then used to create malicious gradients at every SGD iteration, which is almost opposite to that of minima. Such gradients are then used to generate poisoned labels and inject those labels into the dataset. We have tested our attack in three different datasets, i.e. MNIST, Fashion-MNIST, and CIFAR-10, along with some preexisting data poisoning attacks. We have measured the performance of a global model in terms of accuracy drop in federated machine learning settings. The observed result suggests that our attack can be 1.6 times stronger than the targeted attack and 3.2 times stronger than a random poisoning attack in certain cases.

1. Introduction

Industry 4.0 has led to a massive increase in the number of edge device based systems Gomathi et al. (2018) and has seen a growth in the number of sensitive data-driven industries. IoT devices are one of the widely used edge devices Chen et al. (2018). These edge devices are used for collecting different information from the surroundings and processing it Raj (2020). These days IoT devices have been equipped with sufficient resources such that the information collected can be processed with the help of machine learning algorithms Ghosh and Grolinger (2019). Federated machine learning is one such collaborative machine learning approach for decentralized edge devices Khan et al. (2020); Monitoring (2023); Supriya and Gadekallu (2023). In federated machine learning, the information from the edge nodes is not exported at the server-side for processing as in centralized machine learning. Due to its privacy-preserving nature, it has seen massive success. Industrial applications based on IoT and distributed systems constitute a cross-silo setting for federated learning (Cvitić and Perakovic, 2021). Applications can range from data sharing to predictive analysis (Li et al., 2023; Stergiou et al., 2021). In such a setting each client plays an significant role in the system and hence presence of malicious clients in such a setting can have equally catastrophic results. Industrial applications are heavily data driven, and this data in many cases can become an identifying feature for a particular node in the network of devices. In federated machine learning, all the edge nodes are connected with each other for federated training, where they exchange their local updates with other nodes to perform peer learning. Since the edge devices can be resource-constrained in nature, complex

* Corresponding author at: Zhongda Group, Haiyan County, Jiaxing City, Zhejiang Province 314312, China.
E-mail addresses: prajjwal.gupta2019@vitstudent.ac.in (P. Gupta), krishna.nitkkr1@gmail.com (K. Yadav), bbgupta@nitkkr.ac.in (B.B. Gupta), alazab.m@ieee.org (M. Alazab), thippa@zhongda.cn (T.R. Gadekallu).

https://doi.org/10.1016/j.cose.2023.103270
0167-4048/

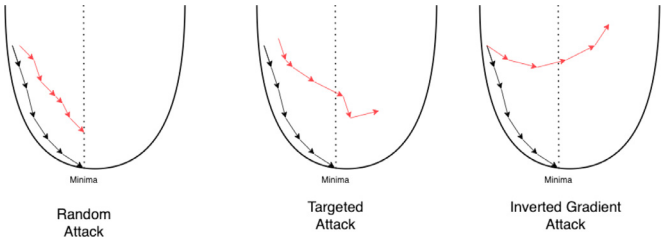

Fig. 1. Types of data poisoning attack.

security mechanisms cannot be employed (Li et al., 2022; Raj and Pani, 2022), and adversaries constantly target this vulnerability Dean and Agyeman (2018). They try to control the edge nodes and perform different malicious activities such as information theft and data poisoning. As federated training is very sensitive to the updates sent by the edge nodes, attacks like data poisoning do not only affect the local nodes, but also its peer nodes involved in federated training Yadav and Gupta (2021). Data poisoning attacks have been studied for a long time, and there are different attacks that exist in the literature. On the basis of freedom of poisoning, attacks can be classified into two categories, i.e., clean label poisoning and dirty label poisoning. In clean label poisoning, authors in Shafahi et al. (2018) discuss that the adversary cannot change the labels of the data as there exists a mechanism to certify the data belonging to a particular class/label. This generally involves poisoning the input data in a targeted or random manner depending upon the adversarial goal. In dirty label poisoning Chen et al. (2017), the adversary can introduce a number of samples that cause convergence satisfying the poisoning objective. These samples are mislabeled with respect to benign data, and this mislabeling can again follow a targeted or random approach.

In this paper, we have come up with the gradient inversion attack, which is the most powerful attack among all the attacks that exist in the literature to the best of our knowledge. This is because, unlike most of the attacks which targets at crafting poisonous labels in the dataset, our attack targets the direction of gradients during federated training, almost reverses it, and generates the poisoned label from reversed gradients. Fig. 1 provides a visual comparison of our attack from the rest of the attacks. We can see that the diversion of gradients is maximum in comparison to other attacks, which forces the model to diverge from the minima as training proceeds.

The key contributions of the paper are as follows:

1. This paper presents a practical comparison between three categories of data poisoning attacks namely Targeted Label Flipping, Random Label Flipping and Random Input Data Poisoning. These attacks are studied in a federated learning setting.
2. We propose a new error generic data poisoning attack which poisons the data using a poisoned model trained on a clean dataset using a modified training approach. We introduce and define "Inverted Loss Function" and "Anti Training" which are key to our attack.
3. We provide a proof of how anti training produces gradients which are in opposite direction to the gradients produced by normal training. We compare our attack with the existing attacks in a federated learning setting and show that it is the most powerful among them.
4. Finally we explain the trends observed in the attacks studied in this paper and reason out the varying attack effectiveness on the basis of properties of the gradients produced in each case.

The rest of the paper is organized as follows. Section 2 discusses the necessary background information. Section 3 discusses the threat methodology, Section 4 discusses theoretical analysis of the attack. Section 5 and 6 discuss the experimental setup and result. Finally, sections 7 and 8 discuss future work and conclude the paper.

2. Related Work

Federated learning has seen an increase in its adaptation in data driven industries as it allows to collaboratively learn a shared model without violating the privacy of edge device. Some applications include autonomous vehicles, drone ecosystems and medical applications. Several works have been proposed in order to enable safe and efficient federated learning for industrial applications. Authors in Lu et al. (2019) integrate federated learning with the consensus process of permission blockchain for privacy preserving data sharing between multiple parties.

Authors in Mei and Zhu (2015); Xiao et al. (2015) discuss the error-generic poisoning attack. In this kind of attack, the aim of the attacker is to achieve generic misclassification and not specific errors. The strengths of these attacks can be measured by the accuracy drop achieved. Random data perturbation by addition of random or sampled noise and random label flipping are some common attack models that form the base for several others. Authors in Suya et al. (2021) discuss targeted poisoning attacks where the aim of the attacker in these types of attacks is to cause specific errors or miss classifications. The strength of these attacks can be determined by the attack success rate, which is the fraction of the desired misclassification caused during model evaluation. Label flipping in accordance to a predefined mapping is a basic attack model for implementing these kinds of attacks. In a backdoor poisoning attack Bagdasaryan et al. (2020), poisoning is carried out by augmenting or altering a small part of training data with a specific pattern or perturbation and changing the label such that after the model is trained, the specific pattern or perturbation acts as a trigger for the desired misclassification.

Based on the threat model, the attacks can be classified into three broad categories Lyu et al. (2020): single or multi-adversarial Bhagoji et al. (2019), byzantine Blanchard et al. (2017) and sybil attacks. In single or multiadversarial attack, the malicious clients controlled by the adversary act in a specific or systematic manner governed by the adversarial goal or attack objective. The byzantine malicious clients act in an arbitrary manner and try to disguise themselves as benign clients by carefully manipulating outputs or updates to match the benign client output or updates while preserving the adversarial goal. In sybil attacks, the adversary simulates multiple dummy or counterfeit clients to compromise the system.

With the formulation of many such attacks through data poisoning and various other types of attacks such as gradient poisoning and model poisoning, many works formulating robust aggregation methods have been proposed. Some of the byzantine robust aggregation Guerraoui et al. (2018) methods include Krum, Bulyan, trimmed mean, and median.

Authors in Bhagoji et al. (2019) show how model poisoning attacks can be more effective in terms of achieving the misclassification target and also maintaining more stealth. They proposed an explicit boosting method for model poisoning, which has the ability to negate the benign clients updates and consequently requires less number of malicious clients to carry out the attacks. They also proposed the concept of stealth in their attacks as boosting methods that are easily detected in systems following byzantine resilient aggregation methods. In order to introduce stealth, they modified the adversary's objective to carry out stealthy model poisoning to avoid detection for the majority of rounds.

Authors in Zhang et al. (2019) used generative adversarial nets to mimic the prototypical samples of benign clients followed by

poisoning in accordance to the adversarial goal and finally sending the scaled updates to the server.

3. Background

To understand data poisoning attacks in federated learning systems, it's very important to know how data is collected from the nodes or clients and processed with the help of federated machine learning. In Section 3.1, we first discuss federated machine learning followed by a data poisoning attack in Section 3.2.

3.1. Federated Learning

In federated learning, let $N = \{n_1, n_2, n_3, ..., n_n\}$ be the number of nodes distributed across different geographical regions as shown in Fig. 1. If n_k is a local node, it has its corresponding local dataset d_k. The dataset d_k consists of labeled examples$\{x_i, y_i\}_{i=1}^{J}$. Each node n_k doesn't share its local dataset with other nodes, i.e., $\{N - n_k\}$ and with server S. At every training round, node n_k receives a global model consisting of weight W_{global}. Each node n_k trains the global model on d_k and updates the parameter of the global model such that $W_{new} \leftarrow W_{global}$. The objective of the local node is to minimize the loss function, which is given in Eq. (1).

$$min\ C(W_{new}, b) = \frac{1}{J}\sum_{i=1}^{J} E(y_i, \hat{y}_i) \tag{1}$$

The updated weight i.e., W_{new} is then sent to the server for averaging, and the resultant weight is W_{avg}. W_{avg} is sent again to the local nodes for the next round of training, and training continues until Eq. (2) is minimized. If there are N number of workers, the objective of federated learning becomes to minimize Eq. (2).

$$min\ C(W_{new}, b) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{J}\sum_{i=1}^{J} E(y_i, \hat{y}_i) \tag{2}$$

3.2. Data Poisoning Attack

Data poisoning attack is injecting either false feature vectors or a label to the dataset that is further used for training a model. Under adversarial federated learning, if the local node n_k have dataset d_k with y_{true} as a true label to feature vector x_i, then data poisoning attack tries to find the label, y_{false} such that the $E(y_{true}, y_{false})$ is maximized where E denotes the error between y_{true} and y_{false}. If E is maximized, then Eq. (1) is maximized, which ultimately maximizes the loss due to the federated training, which is given in Eq. (3).

$$max\ C(W_{new}, b) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{J}\sum_{i=1}^{J} E(y_i, \hat{y}_i) \tag{3}$$

There are several ways to maximize error function E, which is broadly categorized into two categories which are presented in Fig. 2.

In our experimentation, we have carried out two additional attack in addition to targeted attack: a random label flipping and a random input data poisoning attack. In random label flipping, the label y_i is changed to a label y_j such that y_j belongs to the set of classes in the dataset and y_i is not equal to y_j. If the adversary is following a discrete uniform probability distribution for flipping labels, then the probability of y_i being flipped to y_j can be given by

$$P(randomflip(c_i) = c_j | i \neq j \cap c_i, c_j \in C) = \frac{1}{n-1}$$

, where $C = \{c_1, c_2, c_3, ..., c_n\}$. Here, C is the set of classes/labels in the dataset. In a random whole data poisoning attack, which is a type of clean label poisoning attack, the input data or feature vectors $\{x_i\}_{i=1}^{J}$ are randomly perturbed, and the labels are not disturbed to produce the poisoned dataset.

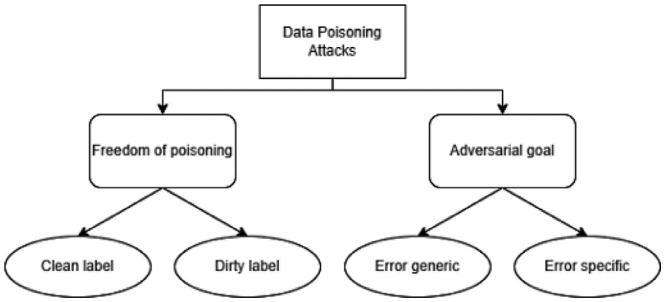

Fig. 2. Types of data poisoning attack.

4. Threat Methodology

4.1. Threat Model

The widely used threat model for data poisoning attacks involves the adversaries poisoning their local dataset by marking specific or random perturbations in order to achieve the adversarial goal at the server-side, which could be as simple as achieving an accuracy drop for aggregated model over a validation data set or the aggregated model producing specific miscalculations or misclassifications. We follow a similar concept of dataset perturbations in our attack.

We have made the following assumptions regarding the adversarial setting:

1. The adversary controls at least one participating node in the communication rounds. By control, we mean that the adversary has the freedom to choose and manipulate the dataset used by the client.
2. The adversary has access to a benign dataset that is representative of the local dataset distribution of most clients.
3. The adversary has the ability to train a copy of the model with different settings to carry out Anti-Training, which is discussed in Section 4.3

4.2. Adversarial Goal

The goal of the adversary is to carry out an error generic attack that results in the accuracy drop of the aggregated model at the server side with respect to some validation dataset D_v. Let m denotes the number of participating nodes, k denotes the number of adversarial clients, δ_s be the global update, δ_b be the update of benign clients and δ_p be the update of the adversarial client, then the averaged update calculated at the server from benign as well as adversarial client is given by

$$\delta_s^* = \frac{\sum_{i=1}^{k}\delta_{pi} + \sum_{j=1}^{m-k}\delta_{bj}}{m} \tag{4}$$

If δ_s^* is the aggregated update in the presence of adversaries and δ_s with only benign clients clients, then the change in aggregated update with the introduction of adversarial clients is given by

$$\Delta\delta_s = |\delta_s - \delta_s^*| \tag{5}$$

Now in a completely benign setting, the benign update δ_s an be given by

$$\delta_s = \frac{\sum_{i=1}^{m}\delta_{bi}}{m} \tag{6}$$

The change in the update can be obtained by subtracting Eq. (6) from Eq. (4), which will give us Eq. (7).

$$\Delta\delta_s = \frac{\sum_{i=1}^{k}\delta_{pi} + \sum_{j=1}^{m-k}\delta_{bj} - \sum_{l=1}^{m}\delta_{bl}}{m} \tag{7}$$

In our attack, the adversary uses the same poisoned dataset for all the compromised clients; hence the updates generated from these clients would be similar. Similarly, for IID data, the updates generated from the benign client would be similar to each other as well. For such a condition, we can approximate Eq. (7) as

$$\Delta\delta_s = \frac{\sum_{i=1}^{k}\delta_{pi} + \sum_{j=1}^{k}\delta_{bj}}{m}$$

The above equation gives

$$\delta_s = |\frac{k(\delta_b - \delta_p)}{m}| \tag{8}$$

In Eq. (8), δ_b and δ_p denote the generalized updates sent by a benign client and adversarial client. Then for the fraction of adversarial clients, i.e., $f = \frac{k}{m}$ participating in a federated training, the accuracy drop is given by

$$\Delta\delta_s = f|(\delta_b - \delta_p)| \tag{9}$$

In our approach, the adversarial goal is to produce an update δ_p such that the change in update $\Delta\delta_s$ is maximum to produce the desired accuracy drop using a poisoned dataset by controlling the fraction of clients participating in the communication rounds.

4.3. Inverted Gradient Attack

In our proposed attack, we leverage the power of reversing the direction of gradients to poison the dataset. This attack aims to find a mapping of labels for poisoning the data which would move the gradients of the clients participating in a federated training in an opposite direction.

The algorithm for the proposed inverted gradient attack is given in Algorithm 1. The algorithm follows the principle of anti-training.

Algorithm 1: Inverted Gradient Attack.

```
Procedure Inverse_loss(target, prediction)
    loss=categorical_crossentropy(target)
    if loss<0.001 then
        loss = 0.001
    inv_loss=1/loss
return inv_loss

Procedure Inverted_poison(W_avg, X, Labels)
    model=instantiate_model
    set_weights(model, W_avg)
    train_model(model, Inverse_loss, X, Labels)
    new_labels = model.predict(X)
    mappings = find_mappings(labels, new_labels)
    poisoned_labels = flip_labels(labels, mappings)
return X, poisoned_labels
```

Definition 4.1 *(Anti-training) Anti-training can be defined as training a machine learning model using an inverted loss function such that at every iteration, instead of producing the gradients that converge towards the minima, the produced gradients diverge from the minima.*

In Algorithm 1, every node, i.e., n_k, initially, receives the averaged weight from the server, i.e., W_{avg}. The goal of an edge network deployed is to collect the true information, and we assume that every node that adversaries have control over has a non-poisoned dataset initially. If we see the function *inverse_loss*, then the adversary first calculates the loss value. If the loss function obtained by setting the weight of the model to W_{avg} tends toward zero, then we modify the loss according to Eq. (14). In the equation, t indicates the penalty factor, which ranges between $0 <= t <= 1$ for a categorical crossentropy loss. Higher t produces a higher loss. For example, in benign conditions, if $E = (y - \hat{y}) = 0.2$, then the presence of t, E will be $E = 1 - 0.2 = 0.8$, resulting in higher loss.

Alternatively the value of t can be set as the maximum error observed corresponding to a benign dataset and benign model. This threshold t will determine the degree of divergence per communication round. It can be an important parameter in evading defenses at the server side which eliminate the gradient updates based on anomaly detection algorithms. In our entire federated training, we kept $t = 1$ such that inverted loss is maximized at every iteration.

Once we get the inverted loss, if L_1 indicates the benign loss for global model M and L_2 indicates the inverted loss, we set $L_1 \leftarrow L_2$. Now for model M, the loss becomes L_2, and the local model is trained assuming inverted loss as its cost function. When we carry out such training, the model tries to reduce the inverted loss. When we try to predict the labels obtained from the model M that tries to minimize inverted loss, we move towards achieving a false label for the benign feature vectors. In this way, for a given feature vector $\{x_i, y_i\}_{i=1}^{l}$, we can obtain poisoned labels$\{y_i^*\}_{i=1}^{l}$. When a local node n_k carries out local training, it results in the weights that are poisoned that satisfies Eq. (4), which maximizes Eq. (9), and our adversarial goal of maximum accuracy drop can be achieved.

An argument could be made that since there are no constraints on what incorrect label the adversarial model should produce in case of anti-training, the attack might show characteristics similar to random label flipping. But this is not the case when the inverted loss is designed such that it has the model predictions and targets as factors, i.e the gradients are dependent on the predictions and targets. In such a case, the model will have an explicit constraint to produce predictions for which the adversarial loss is minimum and consequently the accuracy (of the model on benign dataset) is minimum, and the predictions would be specific to this minimum accuracy which was achieved.

4.4. Analysis of an Attack

In the previous section, we discussed an intuition behind the inverted loss function that diverges the gradients in federated learning. In this section, we provide a mathematical intuition behind it.

Let C be the cost function that a local node n_k tries to minimize in a benign condition is given by

$$min\ C(w, b) = \frac{1}{N}\sum_{i=1}^{N} E(y_i, \hat{y}_i) \tag{10}$$

In the above equation, E is the mean squared error, which is given by

$$E(y, \hat{y}_i) = (y - \hat{y})^2 \tag{11}$$

After each epoch, during a model training, the weight of a model is updated as follow

$$w = w - \alpha\frac{\partial}{\partial w}C(w, b) \tag{12}$$

Now, we substitute Eq. (10) in Eq. (12), and the resultant equation is given by,

$$w = w - \alpha\frac{\partial}{\partial w}\left(\frac{1}{N}E(y, \hat{y})\right)$$

Using mean squared error as error function in above equation, we get

$$w = w - \frac{\alpha}{N}\frac{\partial}{\partial w}\left(y - \hat{y}\right)^2$$

Calculating partial derivatives with respect to w we get

$$w = w - \frac{2 * \alpha}{N}\left(y - \hat{y}\right)\frac{\partial}{\partial w}\left(y - \hat{y}\right)$$

Finally solving above equation further we get

$$w = w + \frac{2 * \alpha}{N}\left(y - \hat{y}\right)\frac{\partial}{\partial w}\left(\hat{y}\right) \tag{13}$$

In the presence of adversaries, the adversarial mean squared error is given by

$$E^*(y, \hat{y}) = t - (y - \hat{y})^2 \tag{14}$$

Now, substituting the value of adversarial E, i.e, Eq. (14) in Eq. (12), and solving the equation as solved for benign nodes,

$$w = w - \alpha\frac{\partial}{\partial w}\left(\frac{1}{N}E^*(y, \hat{y})\right)$$

$$w = w - \frac{\alpha}{N}\frac{\partial}{\partial w}t - \left(y - \hat{y}\right)^2$$

$$w = w + \frac{2 * \alpha}{N}\left(y - \hat{y}\right)\frac{\partial}{\partial w}\left(y - \hat{y}\right)$$

$$w = w - \frac{2 * \alpha}{N}\left(y - \hat{y}\right)\frac{\partial}{\partial w}\left(\hat{y}\right) \tag{15}$$

From Eqs. (13) and (15), we can infer that the gradient produced in our proposed attack that uses anti-training is in the opposite direction to that seen in the benign case.

5. Experimental Setup

We carried out and evaluated our attack on three widely used benchmark datasets, i.e., MNIST Digit dataset, Fashion MNIST, and the CIFAR-10. For the MNIST dataset, we used a three-layered artificial neural network(ANN) model with *tanh* activation function for the hidden layers and softmax activation layer for the output layer. A convolutional neural network was used for the Fashion MNIST and CIFAR-10 datasets. For the Fashion MNIST dataset, a model with three convolutional layers (with max pooling and dropout) and three dense layers (with dropout) were used. The model used for CIFAR-10 consisted of six convolutional layers (with max pooling, batch normalization, and dropout) and three dense layers with dropout.

To carry out federated training, we had ten local nodes. We carried out our federated training for one hundred communication rounds. We studied the attack in three different poisonous settings, i.e., under 10%, 20%, and 50%. A 10% poisoning means that among all the data points present inside the dataset of a local node, only 10% of the data was poisoned. The same approach is followed for 20% and 50% poisoning. Since targeted label flipping is an error-specific attack, we evaluated the attack in terms of a success rate as well. The success rate for an attack can be calculated with

$$success\_rate = \frac{S_a}{N},$$

where, S_a is the number of samples misclassified according to the adversarial goal by the aggregated model at the server-side on the validation set, and N is the total number of samples.

6. Result

Initially, we have carried out some of the most common data poisoning attacks available in the literature, a brief explanation of which can be found in Section 3.2. As expected, we see an increase in accuracy drop with increase in fraction of malicious clients or increase in the amount of data poisoned for all the three datasets. Targeted label flipping achieved a minimum accuracy drop of 0.13% ($f = 0.1$ and 10% of data poisoned) and a maximum accuracy drop of 50.38% ($f = 0.5$ and 100% of data poisoned) for MNIST Digit dataset, 0.2% ($f = 0.1$ and 10% of data poisoned) and 53.61% ($f = 0.5$ and 100% of data poisoned) for CIFAR-10 and finally 0.03% ($f = 0.1$ and 10% of data poisoned) and 30.67% ($f = 0.5$ and 100% of data poisoned) for Fashion MNIST dataset. Overall, we can conclude that the attack was most effective in the case of CIFAR-10 dataset. A similar conclusion can be drawn from the results obtained for success rate with highest recorded value of 52.9% ($f = 0.5$ and 100% of data poisoned) for CIFAR-10 dataset.

The random poisoning attacks demonstrated a significantly less accuracy drop for all test cases for MNIST Digit and Fashion MNIST datasets when compared to the accuracy drop achieved for CIFAR-10 dataset. The test case with $f = 0.5$ and 100% data poisoning remains the one with the highest accuracy drop for these attacks as well. For MNIST Digit dataset, targeted label flipping achieved a larger accuracy drop with $f = 0.2$ and 50% data poisoning than random label flipping with $f = 0.5$ and 100% data poisoning and random input data poisoning with $f = 0.5$ and 100% data poisoning. From the Tables 1, 2 and 3 we can conclude that the targeted label flipping attack requires a lower fraction of malicious clients and amount of data poisoning than the random attacks to produce a similar accuracy drop making it the most powerful among the three.

Among the traditional data poisoning attacks, the targeted label flipping attack achieved the highest accuracy drop followed by random label flipping and random input data poisoning. We take an example of the test case on CIFAR-10 (presented in Fig. 6) which

Table 1
Targeted label flipping attack.

Dataset	% of malicious clients	% of label flipped							
		10%		30%		50%		100%	
		Accuracy Drop	Success Rate	Accuracy Drop	Success Rate	Accuracy Drop	Success Rate	Accuracy Drop	Success Rate
MNIST Digit	10%	0.13	0.08	0.16	0.15	0.23	0.26	0.36	0.27
	20%	0.34	0.17	0.42	0.29	1.25	0.88	1.82	1.04
	50%	5.04	4.85	12.81	12.47	28.21	27.69	50.38	49.8
CIFAR-10	10%	0.2	0.26	0.27	0.57	1.33	0.9	1.37	1.39
	20%	0.93	0.73	1.3	1.41	1.32	1.74	2.27	2.33
	50%	3.16	3.37	5.64	5.65	23.04	22.58	53.61	52.9
Fashion MNIST	10%	0.03	0	0.13	0.01	0.17	0.02	0.19	0.96
	20%	0.52	0.02	0.58	0.02	0.67	0.11	0.81	1.36
	50%	5.34	4.7	12.3	11.16	24.56	20.94	30.67	30.79

Table 2
Random label flipping attack.

Dataset	%of malicious clients	% of labels flipped			
		10%	30%	50%	100%
MNIST Digit	10%	0.03	0.09	0.19	0.27
	20%	0.04	0.15	0.44	0.56
	50%	0.06	0.11	0.43	0.73
CIFAR-10	10%	0.12	0.35	1.12	1.22
	20%	1.21	1.35	1.53	1.98
	50%	4.57	10.39	13.51	20.55
Fashion MNIST	10%	0.05	0.08	0.12	0.23
	20%	0.14	0.43	0.65	0.75
	50%	0.41	0.7	1.12	1.54

Table 3
Random input data poisoning attack.

Dataset	%of malicious clients	% of data poisoned			
		10%	30%	50%	100%
MNIST Digit	10%	0.0	0.05	0.11	0.15
	20%	0.05	0.08	0.15	0.37
	50%	0.06	0.08	0.36	0.65
CIFAR-10	10%	0.21	0.4	0.51	1.04
	20%	0.26	0.45	1.25	1.36
	50%	0.37	1.74	3.03	8.69
Fashion MNIST	10%	0	0.03	0.08	0.11
	20%	0.02	0.05	0.09	0.13
	50%	0.08	0.12	0.44	.1.21

recorded the highest accuracy drop for all three attacks where they were carried out when $f = 0.5$. From Fig. 6, we can infer that random data poisoning produced an accuracy drop of 8.69%, random label flipping produced a drop of 20.55%, and targeted label flipping produced a drop of 52.9%. We also calculated the average success rate of misclassification for all classes in targeted label flipping, which was 52.9%. A similar training of maximum accuracy drop was observed in targeted label flipping, although the fraction of clients was reduced to 0.3 and 0.1. This suggests that targeted was the strongest attack among all.

In random data poisoning, for every label, at the initial round of training, a random poisoned label is produced but in a targeted attack, at the initial round of training, unlike random poisoning, a specific poisoned label is produced. Thus, error generic poisoning results in a randomized dataset in terms of patterns observed, and the correlation between dependent and independent variables would be lesser than that observed in benign and targeted poisoned dataset (which are more deterministic datasets) if not close to 0. Training on such a dataset will create confusion for the model. The minimum possible error z would be much higher than that observed in the deterministic ones. The usage of a uniform probability distribution can be associated with the presence of a large number of minimas in the loss plateau. The poisoned gradients produced in this case would have a random direction with each communication round. If the adversary is controlling more than one client, then the adversarial updates generated wouldn't find much support from each other in terms of directionality because of their random nature and the overall effect of adversarial presence is lesser than that observed in homogeneous updates produced from deterministically poisoned datasets.

Once we infer that the targeted attack was strongest, we have used it as a base to compare with our proposed data poisoning attack in further results. Figs. 3, 4, 5 presents the result obtained by carrying out targeted and inverted gradient attacks on MNIST, Fashion MNIST, and CIFAR-10. All the attacks were carried out at *f=0.5, 0.2, 0.1*. Generally, whenever any node registers itself for federated training, it is not malicious. After a certain instance of time, adversaries gain control over it Park et al. (2021) and make nodes malicious. To simulate such a scenario, we introduced the malicious nodes after the 10th communication round, which can be clearly seen with the introduction of accuracy drop after the 10th training round in Fig. 3, 4, and 5. In the given figure, if we take the result when *f=0.5*, then the accuracy drop for the targeted attack is 35%, 29%, and 60% for CIFAR-10, Fashion MNIST, and MNIST Digit, whereas for targeted attacks, the accuracy drop is 58%, 62%, and 67%. When we calculated the average accuracy drop across both

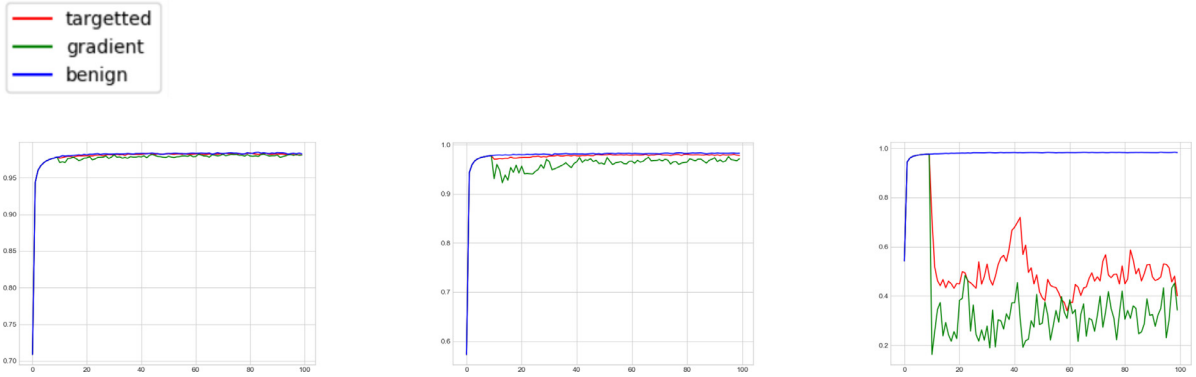

Fig. 3. Accuracy drop on MNIST Digit dataset(a) *f=0.1. (b) f=0.2 (c) f=0.5* (X-axis:Communication round, Y-axis:Accuracy).

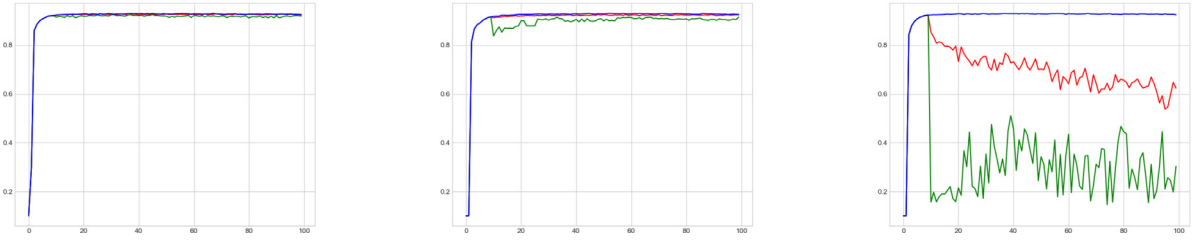

Fig. 4. Accuracy drop on Fashion MNIST dataset(a) *f=0.1. (b) f=0.2 (c) f=0.5*(X-axis:Communication round, Y-axis:Accuracy).

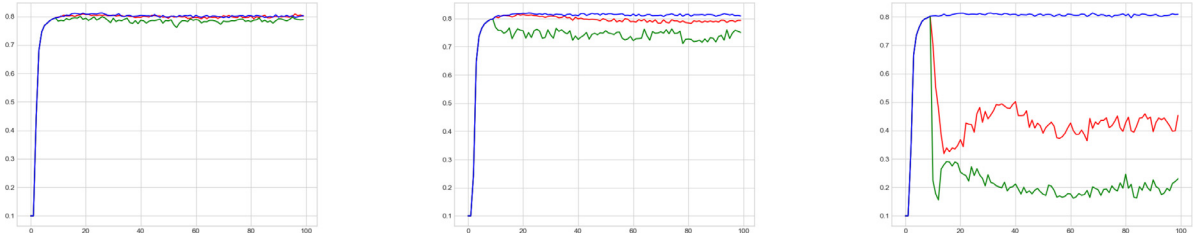

Fig. 5. Accuracy drop on CIFAR-10 dataset(a) *f=0.1. (b) f=0.2. (c) f=0.5* (X-axis:Communication round, Y-axis:Accuracy).

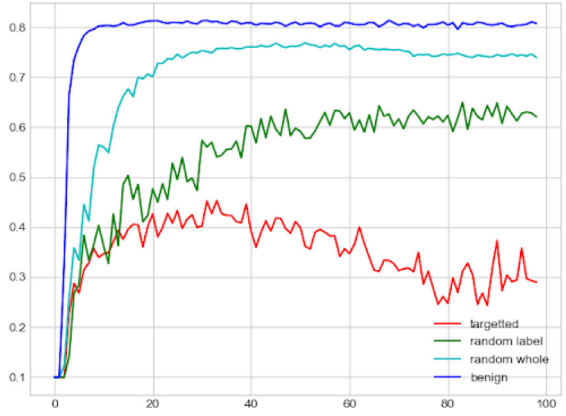

Fig. 6. Accuracy drop of most common attacks on CIFAR 10 dataset.(X-axis:Communication round, Y-axis:Accuracy).

the attacks, then our proposed attack resulted as much as 1.63 times stronger than the targeted attack, which is itself 3.2 times stronger than random label flipping attack and 11.6 times than random input data poisoning attack under the same setting.

7. Future Work

The following could be potential weaknesses of the attack:

1. Dependency on f: In a cross device setting (large number of clients), the effect of presence of adversaries is diluted with increasing number of clients, i.e the adversary would have to control a larger number of clients with increasing number of participating devices to maintain the value of f.
2. Due to the opposite nature of gradients produced by the adversarial nodes, the attack is prone to several detection mechanisms.

The proposed attack can be improved by addition of stealth mechanisms by enabling gradient masking or even reducing the intensity and changing the direction again with respect to benign gradients to evade any detection mechanisms. Another extension of the attack could be to produce gradients to enable convergence towards a specific point which is not necessarily the global or local minima. This could be to enable targeted misclassifications. Adversarially robust optimization algorithms can be designed which cluster gradients on the basis of magnitude and direction, and the aggregated update is calculated based on an optimization procedure weighing each cluster differently according to its density.

8. Conclusion

In this paper, we have proposed a new data poisoning algorithm based on gradient divergence. At every iteration, an inverted loss is calculated, which diverges the gradients from the minima. With the help of inverted gradients, we further calculate poisoned labels and poison the dataset of a local node that is involved in federated training. In terms of the accuracy drop of a global model, our proposed attack is the strongest among targeted and random poisoning attacks. In the coming future, we will test our proposed attack with the existing data poisoning detection method and see the efficiency of our attack in evading those algorithms. Moreover, we would like to prepare our own data poisoning detection algorithm to detect novel attacks like gradient divergence.

Declaration of Competing Interest

The authors declare that they don't have any conflicts of interests to disclose.

CRediT authorship contribution statement

Prajjwal Gupta: Conceptualization, Writing – original draft, Software. **Krishna Yadav:** Conceptualization, Writing – original draft, Software. **Brij B. Gupta:** Validation, Writing – review & editing, Data curation. **Mamoun Alazab:** Resources, Writing – review & editing, Supervision, Validation. **Thippa Reddy Gadekallu:** Resources, Writing – review & editing, Supervision, Validation.

Data availability

Data will be made available on request.

References

Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., Shmatikov, V., 2020. How to backdoor federated learning. In: International Conference on Artificial Intelligence and Statistics. PMLR, pp. 2938–2948.
Bhagoji, A.N., Chakraborty, S., Mittal, P., Calo, S., 2019. Analyzing federated learning through an adversarial lens. In: International Conference on Machine Learning. PMLR, pp. 634–643.
Blanchard, P., El Mhamdi, E.M., Guerraoui, R., Stainer, J., 2017. Machine learning with adversaries: Byzantine tolerant gradient descent. Advances in neural information processing systems 30.
Chen, B., Wan, J., Celesti, A., Li, D., Abbas, H., Zhang, Q., 2018. Edge computing in iot-based manufacturing. IEEE Communications Magazine 56 (9), 103–109.
Chen, X., Liu, C., Li, B., Lu, K., Song, D., 2017. Targeted backdoor attacks on deep learning systems using data poisoning. arXiv preprint arXiv:1712.05526.
Cvitić, I., Perakovic, D., et al., 2021. Boosting-based DDoS detection in internet of things systems. IEEE Internet Things J. 9 (3), 2109–2123.
Dean, A., Agyeman, M.O., 2018. A study of the advances in iot security. In: Proceedings of the 2nd international symposium on computer science and intelligent control, pp. 1–5.
Ghosh, A.M., Grolinger, K., 2019. Deep learning: Edge-cloud data analytics for iot. In: 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE). IEEE, pp. 1–7.
Gomathi, R., Krishna, G.H.S., Brumancia, E., Dhas, Y.M., 2018. A survey on iot technologies, evolution and architecture. In: 2018 International Conference on Computer, Communication, and Signal Processing (ICCCSP). IEEE, pp. 1–5.
Guerraoui, R., Rouault, S., et al., 2018. The hidden vulnerability of distributed learning in byzantium. In: International Conference on Machine Learning. PMLR, pp. 3521–3530.
Khan, L.U., Pandey, S.R., Tran, N.H., Saad, W., Han, Z., Nguyen, M.N., Hong, C.S., 2020. Federated learning for edge networks: Resource optimization and incentive mechanism. IEEE Communications Magazine 58 (10), 88–93.
Li, D., Lai, J., Wang, R., Li, X., Vijayakumar, P., et al, 2023. Ubiquitous intelligent federated learning privacy-preserving scheme under edge computing. Fut. Gen. Comput. Syst. 144, 205–218.

Li, S., Qin, D., Wu, X., Li, J., Li, B., Han, W., 2022. False alert detection based on deep learning and machine learning. Int. J. Semant. Web Inf. Syst. 18 (1), 1–21.
Lu, Y., Huang, X., Dai, Y., Maharjan, S., Zhang, Y., 2019. Blockchain and federated learning for privacy-preserved data sharing in industrial iot. IEEE Transactions on Industrial Informatics 16 (6), 4177–4186.
Lyu, L., Yu, H., Yang, Q., 2020. Threats to federated learning: A survey. arXiv preprint arXiv:2003.02133.
Mei, S., Zhu, X., 2015. Using machine teaching to identify optimal training-set attacks on machine learners. In: Proceedings of the aaai conference on artificial intelligence, Vol. 29.
Monitoring, P., 2023. Guest editorial federated learning for privacy preservation of healthcare data in internet of medical things and patient monitoring. IEEE Journal of Biomedical and Health Informatics 27 (2).
Park, J., Han, D.-J., Choi, M., Moon, J., 2021. Sageflow: Robust federated learning against both stragglers and adversaries. Advances in neural information processing systems 34, 840–851.
Raj, J.S., 2020. A novel information processing in iot based real time health care monitoring system. Journal of Electronics 2 (03), 188–196.
Raj, M.G., Pani, S.K., 2022. Chaotic whale crow optimization algorithm for secure routing in the IoT environment. Int. J. Semant. Web Inf. Syst. 18 (1), 1–25.
Shafahi, A., Huang, W.R., Najibi, M., Suciu, O., Studer, C., Dumitras, T., Goldstein, T., 2018. Poison frogs! targeted clean-label poisoning attacks on neural networks. Advances in neural information processing systems 31.
Stergiou, C. L., Psannis, K. E., et al, 2021. InFeMo: flexible big data management through a federated cloud system. ACM Trans. Internet Technol. (TOIT) 22 (2), 1–22.
Supriya, Y., Gadekallu, T.R., 2023. A survey on soft computing techniques for federated learning-applications, challenges and future directions. ACM Journal of Data and Information Quality.
Suya, F., Mahloujifar, S., Suri, A., Evans, D., Tian, Y., 2021. Model-targeted poisoning attacks with provable convergence. In: International Conference on Machine Learning. PMLR, pp. 10000–10010.
Xiao, H., Biggio, B., Brown, G., Fumera, G., Eckert, C., Roli, F., 2015. Is feature selection secure against training data poisoning? In: international conference on machine learning. PMLR, pp. 1689–1698.
Yadav, K., Gupta, B., 2021. Clustering algorithm to detect adversaries in federated learning. arXiv preprint arXiv:2102.10799.
Zhang, J., Chen, J., Wu, D., Chen, B., Yu, S., 2019. Poisoning attack in federated learning using generative adversarial nets. In: 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). IEEE, pp. 374–380.

Prajjwal Gupta is a student of Bachelors in Computer Science from Vellore Institute of Technology, Tamil nadu, India. His areas of interests include federated learning, security and privacy.

Krishna Yadav is a research scholar from the department of computer engineering, National Institute of Technology Kurukshetra, India. His areas of interests include federated learning, security and privacy.

Brij B. Gupta is working as Director of International Center for AI and Cyber Security Research and Innovations, and Full Professor with the Department of Computer Science and Information Engineering (CSIE), Asia University, Taiwan. In more than 17 years of his professional experience, he published over 500 papers in journals/conferences including 35 books and 11 Patents with over 20,000 citations. He has received numerous national and international awards including Canadian Commonwealth Scholarship (2009), Faculty Research Fellowship Award (2017), MeitY, GoI, IEEE GCCE outstanding and WIE paper awards and Best Faculty Award (2018 & 2019), NIT KKR, respectively. Prof. Gupta was recently selected for 2022 Clarivate Web of Science Highly Cited Researchers in Computer Science. He was also selected in the 2022, 2021 and 2020 Stanford University's ranking of the world's top 2% scientists. He is also a visiting/adjunct professor with several universities worldwide. He is also an IEEE Senior Member (2017) and also selected as 2021 Distinguished Lecturer in IEEE CTSoc. Dr Gupta is also serving as Member-in-Large, Board of Governors, IEEE Consumer Technology Society (2022-2024). Prof Gupta is also leading IJSWIS, STE and IJCAC as Editor-in-Chief. Moreover, he is also serving as lead-editor of a Book Series with CRC and IET press. He also served as TPC members in more than 150 international conferences also serving as Associate/Guest Editor of various journals and transactions. His research interests include information security, Cyber physical systems, cloud computing, blockchain technologies, intrusion detection, AI, social media and networking.

Mamoun Alazab is an Associate Professor at the College of Engineering, IT and Environment, at Charles Darwin University, Australia. He received his PhD degree in Computer Science from the Federation University of Australia, School of Science, Information Technology and Engineering. He worked previously as a Senior Lecturer (Australian National University), Lecturer (Macquarie University), and Post Doc Fellow (Japan Society for the Promotion of Science (JSPS) through the Australian Academy of Science. He is a cyber security researcher and practitioner with industry and academic experience. Associate Prof Alazab's research is multidisciplinary that focuses on cyber security and digital forensics of computer systems including current and emerging issues in the cyber environment like cyber-physical systems and internet of things, with a focus on cybercrime detection and prevention.

Thippa Reddy Gadekallu is currently working as a chief Engineer in Zhongda Group, Haiyan County, Jiaxing City, Zhejiang Province, China as well as an Associate Professor in the School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India. He obtained his Bachelors in Computer Science and Engineering from Nagarjuna University, India, in the year 2003, Masters in Computer Science and Engineering from Anna University, Chennai, Tamil Nadu, India in the year 2011 and his Ph.D in Vellore Institute of Technology, Vellore, Tamil Nadu, India in the year 2017. He has more than 14 years of experience in teaching. He has than 150 international/national publications in reputed journals and conferences. Currently, his areas of research include Machine Learning, Internet of Things, Deep Neural Networks, Blockchain, Computer Vision. He is an editor in several publishers like Springer, Hindawi, Plosone, Scientific Reports (Nature), Wiley. He also acted as a guest editor in several reputed publishers like IEEE, Elsevier, Springer, Hindawi, MDPI. He is recently recognized as one among the top 2% scientists in the world as per the survey conducted by Elsevier in the year 2021.