Offence and defence techniques of Federated Learning

Yishan Li 356li@uwaterloo.ca

Editor: Pascal Poupart

Abstract

Federated learning introduces a new decentralized training mechanism where all the data are trained across decentralized edge devices, which efficiently allows users' data to store and train locally, minimize the direct exposure in communication, and therefore securely protect user's privacy. However, after surveying some papers, I realized there are still some potential security issues that exist in this technology and improvements need to be done to optimize the model. In this paper, I will explore in detail the possible techniques of attacking the federated learning model and the possible defense solutions that could prevent the model from being hacked. Then demonstrate my observation and discuss the relative strengths and weaknesses of the approaches.

1. Introduction

As federated learning becomes increasingly popular nowadays. The design of federated learning has largely protected users' privacy and potentially eliminated the single-point-of-failure that the server-based architecture is very vulnerable to. However, federated learning does still introduce some new problems. Although the user data is not explicitly shared in the network, the frequent transmission of the updates is still possible to leak private information. Besides, the setting of federated learning allows the malevolent users to leverage more advanced technology to induce further security issues. Inspired by the weakness of this architecture, I investigated some existing research work that is associated with how potential loopholes could be leveraged by malicious people and how possible defense approaches could be applied to safeguard the security of the architecture.

2. Background

Federated learning(McMahan et al., 2017) is a distributed machine learning technique that enables a large corpus of private user data to reside on an edge device like a cellphone or tablet, which allows for smarter models, less power consumption, lower latency, and most importantly addresses the fundamental problems of the user privacy.

2.1 Non-IID property

Given the fact that data are trained locally on the edge device, it is common that the local model does not have the data from all classes. In another word, the assumption of i.i.d samples on the edge device in the traditional model setting does not hold for federated learning setting. The design of federated learning takes this into account and enables the

local model to train on the highly skewed data set. Besides, imbalanced user data set might result in the different performance of the training and validation process, which brings a few challenges to the research area.

2.2 Technology principle

Federated learning is a distributed machine learning setup that allows edge devices to keep the raw data local and the device itself is responsible for local data management, for example, storing data, expiring old data, and ensuring the data is encrypted properly, etc. Federated learning enables edge devices to collaboratively learn a shared prediction model locally. To build up a robust data model, the atomic set of the client-server interaction plays a key role in federated learning known as the federated learning round. In each round, the central server actively broadcasts the updated global parameters to all the participants' nodes. After downloading the model parameters, the edge nodes train the model locally for several predefined epochs and then transmit the local updates back to the servers. The global server collects updates emitted by the pre-selected participants and aggregates all the local updates by applying the federated model averaging, a generalized form of the federated stochastic gradient descent that takes the weighted average of all the updates to get better model parameters for the next round. This client-server communication will be operated iteratively to improve the global model. To make this communication phase more secure, the technique secure aggregation(Bonawitz et al., 2016) is proposed and treated as an optional extension to the client and server protocol that uses cryptographic techniques to enhances security guarantees, communication efficiency, and dropout tolerance. This protocol allows a group of mutually distrustful parties to collaboratively compute an aggregate value without revealing their private value to others, which helps to protect the privacy of each users' updates and prevent potential privacy leaks.

3. Inference Attack

Reverse engineering is defined as the process by which an artificial object is deconstructed to reveal its designs, architecture to extract knowledge from the object. Due to the property of federated learning that edge nodes periodically exchange updates with the global model, this paradigm gives a great opportunity for the adversary to exploit the user's privacy using the information-rich updates. The properties of the training data reside on edge devices could be reversely inferred by the adversary, for example, information like gender, race, etc. Researchers(Melis et al., 2016) from Cornell University proposed a way to infer the presence of data from updates and discuss how the property of the data could be reversely inferred from the updates. In detail, the adversary constantly captures the snapshots of the model parameters and obtain the aggregated updates from all the other users by taking the difference of model parameters in two consecutive snapshots. With the gradient features in hand, the adversary can train a classifier to conduct the property inference attack so that he can gain insights into users' properties from gradients. To better train this classifier, the researchers assume that the attackers have some auxiliary data, for example, the data with certain properties that the attacker is interested in and not interested in. The intuition of this attack is that the adversary can passively take advantage of the aggregated gradients

of data with certain property and data without the property respectively, which allows the adversary to train a binary classifier to classify the aggregated gradients. After training the model, the adversary can leverage this classifier to determine if the observed gradients have a certain property in the future.
Differential privacy is empirically proved to be the primary defense mechanism to the inference attack. Differential privacy works by adding a predetermined amount of randomness into the computation performed on a dataset to protect individual entries without significantly compromising the overall result, which to some extent guarantees that attacker can learn virtually nothing more about a particular individual than they would learn if that person's record were absent from the dataset. However there is a trade-off between privacy and accuracy when applying this technique, as a greater percentage of the updates are altered, the accuracy of the global model will degrade in turn. There is a few ongoing research related to the differential privacy proposed to defense against the inference attack. For instance, researchers (Chen et al., 2020) have effectively evaluated how differential privacy could help defend the membership inference attack on Lasso and CNN model on the genomic dataset. Another state of art privacy-preserving approach (Melis et al., 2020) based on differential privacy has also been proposed by researchers to perturb the parameter updates with minimal compromise of the overall accuracy in deep neural networks.

Discussion

The proposed inference attack has some limitations. To train a robust classifier, the paper assumes that the adversaries have enough specialized auxiliary data to train the classifier. The auxiliary data can be easily obtained if the adversary is interested in some simple properties, for example, the gender of the victim. However, if the adversary has a more complex interest where he could not easily obtain sufficiently large samples of auxiliary data, this could largely limit the attack. Another limitation of this attack is that the performance of the model degrades significantly if the number of participants increases. Based on the experiment result shown in the paper, the attack only works well in a small group of users. As the number of participants increases, even though the adversary can detect whether the particular property exists in the aggregated gradient, however, the adversary will have no way to figure out which users have that particular property which defeats his initial purpose. The introduction of differential privacy can effectively set extra barriers to the adversary to do inference attacks. Companies like Google, Apple are using differential privacy technology to help discover the usage patterns of a large number of users without compromising individual privacy. However, the limitation of the differential learning could still be profound, as it could be costly for the company to afford the degradation of global accuracy when using this technique.

4. GAN attack

Researchers from the Stevens Institute of Technology demonstrated that generative adversarial networks(Hitaj et al., 2017) can be trained to produce the prototypical samples from a private dataset on victim devices. GAN is composed of 2 parts in the setting, one is the generator, the other one is the discriminator. Generator learns to generate fake data with the same statistical characteristics as the training data and create more authentic

images to fool discriminator, whereas the discriminator needs to learn to distinguish the generator-producing fake data from real data. The adversary does not know anything about the target training set but intends to manipulate GAN to produce the data that is from the distribution of the target training set. To achieve the attack on federated learning using the GAN model, the attackers could extract the target victim's information by letting the generator and discriminator work collaboratively to generate the fake samples from the victims' dataset. This technique can potentially deceive victims to release more information on the class that the attackers have no knowledge of. A simple example could be given to describe how this attack works. Assume Alice is the victim and Bob is the attacker. Bob has labels [a,b] and Alice has labels[b,c] where a,b,c are Alice and bob's private data respectively. Bob wants to leverage the GAN attack model to infer information on Alice's data c that Bob does not know.

1. Alice trains the local model on her device.
    - Download the shared updates from the server and updates her local model.
    - Train the model locally on [b,c].
    - Share updates of her local model with the server.
2. The server aggregates the updates from a portion of users(including Alice in this scenario) and do federated averaging based on users' gradient.
3. Bob trains the model and deploys the attack secretly.
    - Download the updates from the server and updates his local model.
    - Train private GAN model to create fake images which are similar to the label c.
        - Feed in the replica of the parameters from server to the discriminator.
        - Targeting at data c, run generator based on discriminator feedback.
        - Get a pre-defined number of class c data produced by the generator.
        - Maliciously label those GAN-produced images to label a.
        - Train the model locally on [a,b] and upload the parameter (this will influence the global model eventually and thus corrupt the victim's model secretly)
    - This falls into a malicious loop that corrupted model will continuously leak more information about Alice private data c, in turn GAN will generate more realistic data(c) , which exacerbating the privacy breach.

To develop anti-GAN attacks in federated learning, researchers from the National University of Singapore introduced a mechanism that could potentially prevent the attackers from learning from the real distribution of the victims' data(Luo and Zhu, 2020). The main idea is that on the victims' side, before directly feeding the updates to the global model, it first applies the victim-side GAN model to manipulate the distribution of the original dataset to make GAN-attacks hard to infer the real distribution of victims' dataset and thus fail to reconstruct the victims' data. To prevent this technique degrades the global model performance, two tasks, preserving classification features and obfuscating visual features are applied in the research. To better complete these two tasks, researchers utilize WGAN

(Martin Arjovsky, 2017) introduced by Facebook AI research to be the defense GAN to ensure that defense GAN could improve the learning stability and avoid model collapse during constructing the dataset. Preserving classification features is achieved by introducing a special loss function for WGAN to ensure that the classification feature maps of generated images are similar to that of the original data, which could largely preserve the original data features and therefore avoid degrading the global model performance. Obfuscating visual features is achieved by leveraging a special objective function that involves the pixel variance of generated images for WGAN so that to make the images indistinguishable even the adversaries try to reconstruct the images using the GAN.

Discussion

The most powerful component in the GAN attack towards federated learning is that the GAN attack only relies on the global parameter downloaded from the server. So, as long as the target victim's device is still training the local model, has constant communication with the server, and maintains a relatively high test accuracy, the GAN attack can work well. Google had proposed a protocol called secure aggregation to protect sensitive data during communication to prevent the potential risk of a data breach. However, this security model only considers the case where the updates should be collected and decrypted by the legit server(in this case google server). Google failed to consider the case where "innocent" users could potentially take advantage of the global parameter to conduct the attack. Besides, the global model requires the edge device to constantly communicate with the server, which provides a great opportunity for adversaries to train the GAN to infer the distribution of the target dataset. Falsely labeling the data generated by GAN can force the victim's device to release more information about that data, which accelerates the adversaries to succeed in exploiting the data. This research has delivered a warning message to those companies who want to build applications using federated learning and give the future research direction to come up with approaches to tackle this problem.
The Anti-GAN approach delivers a state of the art solution that cleverly takes into consideration the principle behind GAN attack and takes advantage of the property of GAN to defeat the purpose of the adversary. Also, the introduction of WGAN, the special objective and loss function in WGAN can help the defender distort the distribution of the original dataset in a reasonable manner, which largely prevents the GAN attack to infer the real distribution from the victim's device without compromising the test accuracy of the global model. This technique largely safeguards the distribution of the users' data and mitigates the threat posed by GAN to federated learning. As mentioned in the paper, this architecture also seems to be a promising candidate for preventing the reconstruction attack since it can effectively distort the public statistics of the dataset that the attackers plan to reconstruct on.

5. Data Poisoning Attack

Another kind of attack is closely related to the Byzantine generals' problem. The Researchers from the University of Maryland are inspired by this problem and design an attack called data poisoning attack(Shafahi et al., 2018), a machine learning technique that attempts to fool models by supplying deceptive input and cause malfunction of a model.It

allows the attacker to inject the crafted poison instances in the training dataset which could poison the model, thus maliciously manipulate the test-time behavior of the model and induce it to make absurd mistakes. This poison attack shown in the paper is designed towards all the general deep learning models in machine learning. Unlike the GAN attack, the attacker targets the training data used for model training. Instead of looking for the weak and problematic correlation in the trained model, the data poisoning technique injects a problematic correlation into the model by manipulating the training data.
Researchers from the National engineering research center proposed a Blockchain-based Federated Learning framework(Li et al., 2020) to prevent the emergence of data poisoning attacks. The solution cleverly integrates the idea of Blockchain with federated learning, reducing the malicious attack by recording the nodes' performance on Blockchain, and applying mutual promotion mechanism among nodes. Challenges like consensus efficiency, model security, and framework scalability are all effectively resolved by this proposal. To understand how the model works, the framework can be depicted in 3 aspects.

1. Blockchain storage
   Two different types of blocks are placed on the Blockchain to store global model and local updates respectively. The model block is used to store the information of the global model, for example, block headers, number of rounds, and global learning model parameters, whereas the update block is used to store the information of verified updates, for example, block headers, number of rounds, local update gradient, uploader address,etc

2. Training process and committee consensus During the training phase, the edge nodes reside in the training nodes community accesses the most recently-updated model block to get the global parameter information to train the local model. The updates will be validated by the committee and only store the qualified local training updates to the update block. The committee then evaluates the updates on its own dataset(validation set) and assigns a score on each of the training nodes based on the validation accuracy. A new committee will be elected in the next round of training based on the scores assigned by the committee in the previous round. After a certain amount of verified updates being generated by the nodes, a global aggregation will be triggered. New model parameters will soon be stored in the model block for the next round of training. All the old model parameters in the block will be stored for backup and nodes' verification purposes and will be freed if running out of memory.

3. Nodes management mechanism To encourage the training nodes to contribute the updates to the global model, a node management mechanism is proposed called profit-sharing by contribution. This mechanism forces the training nodes to pay a permission fee for unlimited access to the latest model parameters and gain profit sharing after the successful aggregation. The more training nodes contribute to the updates, the more reward the node can gain during the training process. This mechanism encourages the nodes to engage in the training process, which largely enhances the global model performance.

Discussion

Data poison attacks can be easily adopted in federated learning. In the federated learning setting, the Byzantine army is the analog of edge devices and the castle is the analog of the global server. Suppose an edge device controlled by an attacker maliciously crafts poison instances into the training data, trying to send poisonous updates to the global server. The consequence could be catastrophic if a backdoor is left in the server or the model is induced to make mistakes. To combat the data poisoning attack, rather than using weighted averaging when aggregating, multiple aggregation rules are empirically established and proved to be working well even if some nodes exhibit the Byzantine failures. For example, Researchers claimed that distributed gradient decent algorithms based on trimmed mean and median are provably robust against Byzantine failures(Yin et al., 2018). However, most of the mathematical assumptions in the proof largely depend on user data is IID, which cannot address the problems radically.

The design of the defense framework effectively combines the Blockchain and federated learning to defend against malicious attacks and protect the global server from being manipulated by the malicious update. The so-called profit-sharing by contribution mechanism cleverly takes into consideration the fact that the edge device might not be available during training most of the time. This mechanism actively encourages the nodes to engage in the training process and contribute the updates once they are available. On a different aspect, instead of time-consuming mechanisms like broadcasting consensus to all the nodes, this framework enhances the efficiency of the consensus in the Blockchain setting by giving validation permission to the reliable committee. This form of training and validation is like K-fold cross-validation where the committee members do not participate in certain training rounds and the data on committee members' devices are treated as the validation set. So the multiple validation data with different data distribution could be used to validate the updates. The paper mentioned that this k-fold cross validation could potentially increase the generalization capability of the global model. However, this statement remains questioned and needs further explored. On one hand, this mechanism can potentially detect the maliciously crafted updates sent from the adversary when seeing a dramatic decrease in overall validation accuracy. However, the committee members do not have knowledge of each candidates' data distribution and the distribution of validation set reside on committee members' devices is different from the dataset residing on users' devices. Therefore, an update sent from an honest node might also degrade the overall validation accuracy.

Ways to prevent malicious nodes from becoming committee members are the primary concern for this framework. To dive into the theoretical analysis, if there are X number of committee members, a malicious update will only be propagated to the Blockchain if more than X/2 malicious nodes join the committee. To have this possible, X/2 malicious nodes need to have top X scores in the previous round(X/2 malicious nodes need to reside in the previous-round committee).So, as long as we have more than X/2 honest nodes reside in the committee in the initial round, we can guarantee that the updates cannot be maliciously manipulated, since the cost of taking charge of more than 50% resources is expensive for adversaries to exploit. The design of the framework can effectively provide a defense mechanism for the federated learning setting. However, this paper doesn't discuss much about how to safely designate the initial round of committee. This might be worth exploring in

the future since only by ensuring the identity of the committee in the initial round can it guarantee the future attack-free environment. One possible solution is to let the global servers to step in deciding the members of the committee for the first round. More research needs to be done to come up with a final conclusion.
Storage overhead might be another primary concern in this setting. Not all the edge devices could meet the hardware requirement. A possible solution is to decrease the historical model blocks locally and keep the recent model blocks to ease the device memory. However, this might defeat the purpose of introducing Blockchain-based architecture, since removing the historical blocks will decrease the credibility of the community. It might be reliable to introduce trusted third-party cloud storage to ease the memory overhead on each of the devices. For example, Google could achieve this by encouraging users to store the blocks on google drive.

6. Lesson Learned

Compared with traditional centralized architecture, federated learning is a promising technology to alleviate privacy concerns. This survey helps me deeply dive into how federated learning works from end to end and enriches my knowledge of how the adversaries could take advantage of the properties of federated learning and machine learning models to exploit user privacy. Other than that, I also get a better understanding of how the state-of-art defense approaches are effectively established to actively defend the system from being exploited. In general, these researches and findings play essential roles in improving the system design in the federated learning framework and driving the advancement of federated learning in the future.

7. Furture Work

Federated learning is an active and ongoing area of research in machine learning. There is still a lot of critical open direction in federated learning awaiting to be explored. A more reliable and attack-free federated learning setting needs to be constructed to safeguard the privacy of the users and enhance the accountability of the system. In addition to the potential risks, there are still many other limitations remaining to be improved. One possible consideration is to enhance the communication efficiency of federated learning. Federated learning requires more significantly local device power and constant communication between servers and edge devices, which highly require a more reliable bandwidth to support communication. Insufficient bandwidth could potentially cause a bottleneck that increases communication latency and makes the overall learning process slower. Another observation is that a lot of promising research is conducted under the IID data setting, which is not empirically viable in the real world scenario. So, a more viable validation technique on non-IID should be actively explored in the future. It's good to see more and more application starts applying federated learning into practice, for example, Gboard (Yang et al., 2018), automobile cars, healthcare systems, etc. However, we should remain critical and carefully weigh the benefits and costs of new technology before putting it into practice. More extensive research work needs to be done to improve federated learning before it becomes fully mature in the industries.

References

Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for federated learning on user-held data. 2016.

Junjie Chen, Wendy Hui Wang, and Xinghua Shi. Differential privacy protection against membership inference attack on machine learning for genomic data. 2020.

Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. Deep Models Under the Gan: Information Leakage from Collaborative Deep Learning. 2017.

Yuzheng Li, Chuan Chen, Nan Liu, Huawei Huang, Zibin Zheng, and Qiang Yan. A blockchain-based decentralized federated learning framework with committee consensus. 2020.

Xinjian Luo and Xiangqi Zhu. Deep Models Under the Gan: Information Leakage from Collaborative Deep Learning. 2020.

Léon Bottou Martin Arjovsky, Soumith Chintala. Wasserstein gan. 2017.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient Learning of Deep Networks from Decentralized Data. 2017.

Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting Unintended Feature Leakage in Collaborative Learning. 2016.

Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Ldp-fed: Federated Learning with Local Differential Privacy. 2020.

Ali Shafahi, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison Frogs! Targeted Clean-label Poisoning Attacks on Neural Networks. 2018.

Timothy Yang, Galen Andrew, Hubert Eichner, Haicheng Sun, Wei Li, Nicholas Kong, Daniel Ramage, and Franc¸oise Beaufays. Applied federated learning: Improving google keyboard query suggestions. 2018.

Dong Yin, Yudong Chen, Kannan Ramchandran, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. 2018.