Privacy and Security in Federated Learning

Anirudh Canumalla, anirudhc@uw.edu
Andrew Wei, nowei@cs.washington.edu

December 2019

Abstract

We survey privacy and security concerns in the context of Federated Learning. We will go through a series of papers we found on the subject and offer our interpretations and intuitions.

1 Introduction

What is Federated Learning?

Federated Learning is a setting in which we want to obtain some global model, but the data we want to learn from is unevenly distributed across a large number of nodes. This means that the data won't need to be brought into the data center setting, which is good from a data privacy standpoint; but makes optimization a little tricky because we don't have direct access to the information we want to learn. Additionally, federated learning models are less computationally intensive since much of the training is finely grained across the fleet of nodes.

Problem setup

The common set-up for this problem is that the data is non-IID, unbalanced, massively distributed, and there are limits to communications. Non-IID means that the data is not identically distributed among the nodes that contain data and data on a single user might not be representative of the overall distribution. Unbalanced means that some nodes may have more data than others. Massively distributed means that there are more nodes than the average number of things to train on in each node. The communication limits include things like devices being offline or needing to wait for the devices to not be in-use and be on a unmetered connection.

2 Motivations

Why is Federated Learning important?

As people generate more and more data that we want to use for training and creating more complex and personalized systems, there will be mounting concerns from users about the privacy of their data. Federated Learning is an attempt at addressing this concern. Federated Learning is able to utilize the data on user devices without seeing it, by only seeing the effects of it. This keeps the user's data for the most part private and also helps a lot in terms of access to compute, since the computations can be done client-side, so the main cost in this scheme is communication.

User privacy and security has gotten pretty big over the years due to companies' increased access to data through things like app privileges and their need for data to feed into data-driven pipelines like machine learning. There are probably many apps that ask for privileges that we, as users, agree to without fully understanding the implications of what we're allowing the apps and their creators access to through these privileges.

We predict that companies will have a hard time explaining schemes like Federated Learning to the public when it reaches the media, but that's pretty standard with most technological innovations that sound a little sketchy. Then there might also be companies that are less careful with user data and accidentally leak it, e.g. Facebook. This type of mistake makes massive headlines in the media and also has impacts on company stock/reputation, so it's pretty important.

Many users are unaware of what they allow companies access to, but governments have stepped up by enacting laws that protect user privacy, which makes it harder for companies to freely use the information they have access to. Still, it is important for users to think about the implications of what they're accepting when they allow this or that app to use their camera, microphone, etc...

Example 1

Google's Gboard actually uses Federated Learning and Differential Privacy for updates [1] on their LSTM-based next-word prediction model. The data was obtained from users using installed Google apps on their phones and the text being trained with was stored in logs. They limited their training to devices with at least 2GB of space, were charging, on wifi, and idle. Then they decided to use device caches because they had richer information. They ultimately concluded that federated learning performed better than current server-based training for neural language models. They achieved higher accuracies than a server-based training model using a smaller model.

Example 2

Hospitals generally aren't allowed to share their patient information with others under the Health Insurance Portability and Accountability Act (HIPAA) [2], but governments can circumvent that. If that's the case, then governments can use the patient information and then analyze the data using Federated Learning and draw some meaningful conclusion from that. This means that the government won't store everyone's information like some Big Brother dystopian setting and can still make advances in the health-care space using machine learning. This makes it easier to coordinate research efforts between hospitals and will hopefully help doctors and researchers find new treatment methods and cures.

In this paper

We go forward with a survey of methods we've seen for preserving privacy and security in the setting of Federated Learning. We review some techniques/attacks outlined in [3] and other papers in the field.

3 Survey of methods

Get more clients

Geyer et al. tried to hide client participation in the training of the Federated Learning model [4]. They do this by altering and approximating the average during the aggregation phase of the algorithm, i.e. subsampling the population and then using a Gaussian mechanism to add noise. They limit the privacy loss by using a moments accountant that we'll describe later, but the accountant essentially keeps track of the privacy loss and stops training once the privacy loss has passed a certain threshold. They performed cross-validation grid search experiments on a varying number of clients and found that when they had more clients, adding differential privacy doesn't impact performance as much.

Low-information Updates

Konečný et al. try to reduce the communication costs in Federated Learning by limiting what information they communicate during each update [5]. They propose two types of methods for reducing communication costs, structured updates and sketched updates. Structured updates try to represent the space using a smaller number of variables, specifically using a low-rank approximation or using a random mask for updates. Then sketched updates are essentially compressed versions of the updates, specifically subsampling weights and quantizing weights based on some probability. For the quantization of weights, they had to multiply a random orthogonal matrix to reduce the quantization error.

They find that sketched updates performed worse than structured updates and justified this by saying that sketched updates throws away information gained from training. They find out that they can still reach pretty reasonable accuracies of around 85-90% with these methods on CIFAR data. This might not seem like a privacy/security paper and it generally isn't, but if we reduce the amount of information being transferred or specify the structure of the updates, it'll be harder to gain anything meaningful from this information if it's leaked.

Accounting

Abadi et al. proposes the idea of a moments accountant [6]. The idea of the privacy accountant comes from McSherry in [7], but it essentially keeps track of the overall privacy cost during training. Then the moments accountant is an alternative based on the idea that the strong composition theorem for differential privacy is loose due to not taking into account the type of distribution the noise is coming from. The moments accountant basically gives a tighter bound than the privacy accountant.

They try out the moments accountant in a differentially private implementation of stochastic gradient descent. They run experiments on MNIST and CIFAR and get accuracies for MNIST similar to a non-DP setup, but the accuracy on CIFAR dropped considerably more. Their experiments didn't use Federated Learning, but involves differential privacy schemes that can be applied to a differentially private Federated Learning setting.

Talk Less, Work More

The following paper is somewhat of a progenitor for the federated learning field. McMahon et al. proposes the idea for federated learning, where user's personal devices collaboratively learn shared prediction models without needing to move training data to a centralized location [8]. Instead, only the gradient updates across each device would be aggregated in order to update the collective model.

This paper outlines how in the federated learning setting, user data is finely distributed across millions of devices in a highly uneven fashion. These devices would include laptops, tablets, and especially mobile phones, and experience significantly higher-latency, lower-throughput connections. Furthermore, user devices are only intermittently available for training. In order to overcome these apparent shortcomings of the federated learning model, the authors propose the `FederatedAveraged` algorithm, which is shown to optimize deep neural networks 10-100x more efficiently than a naively implemented federated SGD algorithm. As part of the evaluation portion of the paper, they train and validate a variety of common neural network architectures in a federated learning setting on datasets like MNIST and CIFAR. The `FederatedAveraging` algorithm achieves over 99% accuracy on MNIST and roughly 83% on CIFAR-10.

While federated learning provides strong practical advancements for data privacy (by avoiding the large scale aggregation of sensitive information), the authors hypothesize that federated learning might provide stronger theoretical guarantees if it were united with differential privacy, which can be naturally applied to synchronous algorithms like `FederatedAveraging`.

GANs

Triastycn et al. propose a federated learning framework `FedGP` that uses generative adversarial networks (GANs) to synthesize artificial samples for data release in a manner that preserves the privacy of the samples [9]. This paper addresses the inflexibility of federated learning models after training and potential discrepancy issues with having to label data at the source by incorporating generative models into the training process.

FedGP follows a mathematically relaxed differentially private notion for a user, called empirical DP. The choice of training on the data from the federated GANs for the generative model dramatically increased the accuracy of `FedGP` framework on baseline tasks (increase from 79% to 97% on MNIST).

An exciting result achieved for facial recognition privacy was that the federated GAN was surprisingly robust against model inversion attacks. FedGP framework reduced the accuracy of face detection in recovered images from 25.5% to 1.2% accuracy on the CelebA dataset. The main drawback to the empirical DP approach used in FedGP is that the privacy guarantee is not as strong as traditional differential privacy, and this is proposed as an area of further research in the conclusion.

Sybil (Data Poisoning)

Most papers we have seen so far care more about securing the privacy of user data in Federated Learning, but we've seen less about attacks that can be made on the models themselves. Fung et al. evaluate the vulnerability of federated learning to sybil-based poisoning attacks [10]. This paper was the first among those we read which outlines a dangerous design trade-off of federated learning: while clients are merely passive data providers in the traditional, non-federated settings, now directly observe the intermediate model state.

If malicious actors so desired, they could work backwards from the model state or simply provide arbitrary updates to disrupt the decentralized training process. In a traditional setting, the main kinds of data poisoning are label flipping or backdoor attacks. In the case of federated learning, malicious clients can increase the effectiveness of their attacks by creating a large number of false identities and subverting the reputation of the entire distributed training process for the entire network of users (sybil poisoning).

As a solution, the researchers propose the Fool's Gold algorithm, which maintains the performance of FederatedAveraging [8] in the absence of attacks, and is robust to large numbers of sybils (bad actors) in a poisoning attack. Their results show that Fool's Gold is fairly effective in mitigating poisoning even when the number of malicious clients outnumbers good actors. Although Fool's Gold significantly outperforms previous work, bad actors can still destroy a federated learning model's integrity if they were to consider advanced data attacks, which mix poisoned and honest data, and adaptively limit the percentage of poisioned updates to fool Fool's Gold.

Poison damage (Model Poisoning)

Bhagoji et al. showcase Model Poisoning by performing targeted attacks on the model that would make the model misclassify with high confidence [11]. They also show that this attack works for Byzantine-resilient aggregation, which is generally thought of as a defense to workers sending adversarial updates. On a high level, they introduce a back door to the model in a way that allows the model to still converge.

They achieve this by training on the adversarial agent and boosting the contributions to overcome the normalizing effects of updates from other agents. They note that checking the statistics of the updates would catch this adversary, so they add loss terms with respect to these statistics and keep tacking on terms to make it close to the updates from other nodes so it's harder to identify.

The paper also shows that one current scheme for defense against Byzantine attacks (Krum) fails to defend against this sort of targeted attack, which is reasonable due to the attack not impeding model convergence. The attack also doesn't achieve its goal of reaching a model that converges, but this is mostly due to the defense only updating weights one at a time. Then another defense scheme (Coordinate-wise median) converges and can be successfully attacked using this scheme.

The paper ends by noting that model poisoning attacks are more effective than data poisoning attacks in a federated learning setting due to boosting and that if the data poisoning attacks were scaled up, they would more heavily impact the global model than the model poisoning attack.

4 Conclusion/Takeaways

As we become a more data-centric society that leverages data to do cool things like machine learning and artificial intelligence, user privacy will become an even hotter topic. Some interesting questions may arise, like: How will companies balance user privacy with their goals for technology innovation? How will schemes like Federated Learning be received by the general public? What are the long-term effects of Federated Learning? On user hardware? On company culture/transparency? These are just some of the questions we're hoping to see evolve and/or answered in the next few years.

We hope to have introduced some of the literature and gave some insight into aspects of privacy and security that have been explored in the space of Federated Learning. We believe that it will become a booming field in machine learning within the next few years and we will likely see more use as companies are better able to actualize the potential of user data.

References

[1] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile

keyboard prediction," 2018.

[2] "The health insurance portability and accountability act of 1996.," 1996.

[3] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," 2019.

[4] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," 2017.

[5] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016.

[6] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS'16*, 2016.

[7] F. D. McSherry, "Privacy integrated queries: An extensible platform for privacy-preserving data analysis.," *SIGMOD*, 2009.

[8] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," 2016.

[9] A. Triastcyn and B. Faltings, "Federated generative privacy," 2019.

[10] C. Fung, C. J. M. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," 2018.

[11] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, "Analyzing federated learning through an adversarial lens," 2018.