Research Statement: Scalable, Secure, and Efficient Approaches for Privacy-Preserving Machine Learning

Jinhyun So

1 Introduction

Machine learning (ML) applications can achieve significant performance gains by training on large volumes of data. In many applications, however, the training data is distributed across multiple data-owners, such as patient records at multiple medical institutions or personal images on mobile phones. Due to heterogeneity of computational and communication resource of devices, slow nodes, referred as stragglers or dropped users, create significant delays in ML training. Additionally, the training data often contains sensitive information (e.g., genetic information, financial transactions, and geolocation information). Such settings give rise to the following key problem that is the focus of my research: How can multiple data-owners jointly train a machine learning model while (1) enabling efficient and scalable training (2) keeping their individual datasets private from the other parties; and (3) providing robustness against malicious behavior of adversarial users.

This problem has attracted a surge of interests in both academia and industry in the recent past. In particular, in a centralized case, in which there is a server involved to orchestrate the training, the problem is commonly referred to as "federated learning (FL)" and is currently an active area of research. Also, in the server-less case some approaches based on multi-party computing (MPC) and homomorphic encryption (HE) have been proposed for secure model aggregation. For both cases, however, previous works fall short of addressing one of the critical problems that I am interested in addressing namely "privacy preserving machine learning (PPML) distributed training at scale". To that end, I propose to simultaneously address the critical requirements of scalability, security, and efficiency in distributed PPML.

2 Summary of Previous and Ongoing Works

2.1 Secure Aggregation Protocols for Federated Learning (Centralized PPML Frameworks)

FL is a promising new approach for improving ML models by training over a large volume of data residing in mobile devices. In FL, the training data is kept on the devices rather than sending it to a central server and users sends the locally trained ML model to the server, which protects the privacy of the individual users. Recent works, however, have demonstrated that the local model may reveal extensive information about the private datasets via model inversion attack. To prevent such information leakage, *secure aggregation* protocols ensure that the server can learn nothing beyond the aggregate of all local models. The overhead of the secure model aggregation, however, creates a major bottleneck in scaling federated learning to large number of users.

Turbo-Aggregate: scalability + privacy [1]. In fact, the overhead of state-of-the-art protocols for secure model aggregation in federated learning grows quadratically with the number of users. I proposed a new scheme, named Turbo-Aggregate, that in a network with N users achieves a secure aggregation overhead of $O(N \log N)$, as opposed to $O(N^2)$, while tolerating up to a user dropout rate of 50%. As shown in Figure 1, I demonstrated that Turbo-Aggregate achieves a total running time that grows almost linear in the number of users, and provides up to $40\times$ speedup over the state-of-the-art scheme with $N = 200$ users via experiments over Amazon EC2.

BREA: privacy + Byzantine-resilience [2]. While Turbo-Aggregate enables secure model aggregation with honest-but-curious users, it is not robust to adversarial attacks. The adversarial attacks (also referred as *Byzantine* attacks) may result either from an adversarial user who can manipulate the training data or the information exchanges during the protocol, or due to device malfunctioning. Notably, it has been shown that even a single Byzantine user can significantly alter the trained model in FL. The primary approach for defending against Byzantine attack is by comparing the local updates received from different users and removing the outliers at the server. Achieving the protection of model privacy and Byzantine-resilience simultaneously, however, presents a major

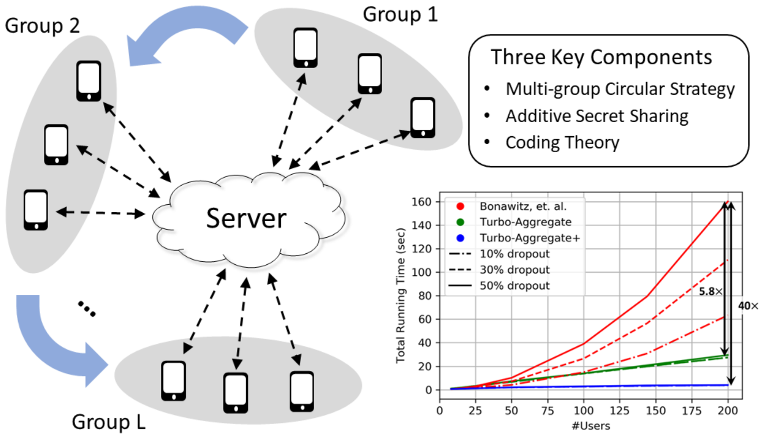


Figure 1: Architecture of Turbo-Aggregate [1] and total running time of Turbo-Aggregate versus the state-of-the-art (Bonawitz, et. al.) as the number of users increases, for various user dropout rates.

challenge as the local models are protected by random masks, which prevents the server from effectively detecting the Byzantine manipulations. I addressed this problem for the homogeneous data distribution, by presenting the first single-server Byzantine-resilient secure aggregation framework for federated learning (named BREA) which leverages quantization, verifiable secret sharing, and distance-based outlier detection.

2.2 Decentralized (or Serverless) PPML Frameworks

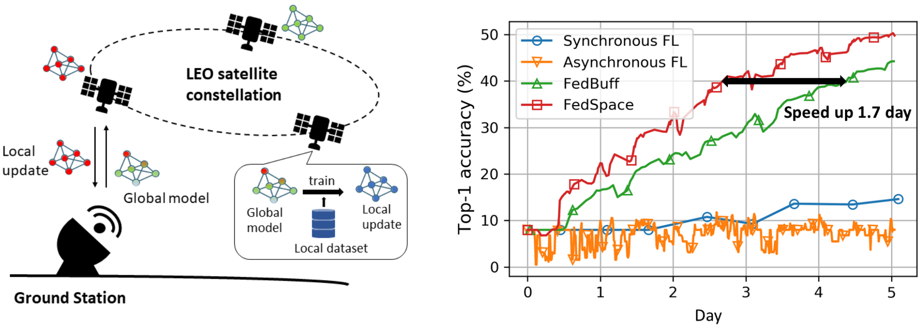
In centralized FL, the server orchestrates an iterative process to train the global model by exchanging the information with all participating users. In practice, however, the presence of a central server is not feasible when multiple competing entities may want to collaboratively learn, which motivates serverless PPML framework. To that end, some approaches based on MPC and HE have been proposed, but both approaches require extensive communication and computational overhead, which limits the participating users to 3-4 parties.

CodedPrivateML: task-offloading + scalability + privacy [3]. As a preliminary work, I study: How to train a machine learning model by offloading while keeping the data private and secure? We present CodedPrivateML, a fast and scalable approach to this critical problem. This work is a launching pad applying coding theory to a PPML framework to provide scalability and privacy simultaneously. We consider a privacy-preserving task-offloading problem where a single data-owner offloads a training task to multiple workers on the cloud due to the lack of computational resource. CodedPrivateML keeps both the data and the model information-theoretically private, while allowing efficient parallelization of training across distributed workers. We characterize CodedPrivateML's privacy threshold and prove its convergence for logistic (and linear) regression. Furthermore, via extensive experiments on Amazon EC2, we demonstrate that CodedPrivateML provides up to $5.2\times$ speedup over cryptographic approaches based on multi-party computing (MPC).

COPML: serverless + scalability + privacy [4]. I extended CodedPrivateML to a collaborative learning scenario in which multiple data-owners wish to jointly train a logistic regression model (without the server), while keeping their individual datasets private from the other parties. I proposed COPML, a fully-decentralized training framework that achieves scalability and privacy-protection simultaneously. The key idea of COPML is to securely encode the individual datasets to distribute the computation load effectively across many parties and to perform the training computations as well as the model updates in a distributed manner on the securely encoded data. I provided characterized COPML's privacy threshold and proved its convergence. Furthermore, via experiments over Amazon EC2, I demonstrated that COPML can provide an order of magnitude speedup (up to $16\times$) in the training time against the benchmark protocols.

2.3 Federated Learning Applications

LightSecAgg: secure aggregation for asynchronous FL + system-level optimization [5]. While Turbo-Aggregate significantly reduces the overhead of secure model aggregation, it incurs an additional round complexity and provides a weaker privacy guarantee than the state-of-the-art as Turbo-Aggregate guarantees model privacy in the



(a) FL framework at satellite networks

(b) Performance gain in training time to achieve a target accuracy

Figure 2: FL framework at satellite constellation and performance gain of FedSpace [6] over the state-of-the-art FL algorithm (FedBuff) to train an image classification ML model (DenseNet-161) on real-world satellite imagery dataset over real-world satellite networks (PlanetLabs).

average sense rather than in the worst-case scenario. In addition, both Turbo-Aggregation and the state-of-the-art secure aggregation scheme are not compatible with asynchronous FL setting. I propose a LightSecAgg, to overcome these challenges by changing the design principle of secure aggregation protocol from "random-seed reconstruction of the dropped users" to "one-shot aggregate-mask reconstruction". This is the first work to protect the privacy of the individual updates without relying on differential privacy or trusted execution environments in the asynchronous FL. I implemented an end-to-end training framework and conducted system-level optimization to demonstrated that LightSecAgg significantly reduces the total training time over the state-of-the-art scheme with diverse ML models and datasets.

FedSpace: FL Framework at Satellites Constellation [6]. Large-scale deployments of low Earth orbit (LEO) satellites collect massive amount of Earth imageries and sensor data, which can empower ML to address global challenges such as real-time disaster navigation. However, it is often infeasible to download all the high-resolution images and train the ML model on the ground station due to limited downlink bandwidth and sparse connectivity. To address these challenges, I leverage FL and show fundamental challenges in applying existing synchronous and asynchronous FL algorithms among satellites and grounds stations. I formulate an optimization problem which captures a unique trade-offs between staleness and idleness, and propose a FedSpace, which dynamically schedules model aggregation based on the deterministic and time-varying connectivity according to satellite orbits. We demonstrate the effectiveness of FedSpace with a real-world satellite imagery dataset and a satellite constellation by showing that FedSpace reduces the training time by 1.7 days (38.6%) over the state-of-the-art asynchronous FL algorithms.

FedML: a research FL library [7]. I have participated in implementing on open source research library and benchmark to facilitate FL algorithm development and fair performance comparison. Specifically, I have contributed to implement secure model aggregation frameworks for both synchronous and asynchronous FL.

3 Future Research Plan

Long-term privacy guarantee in secure aggregation framework. Conventional secure aggregation protocols including Turbo-Aggregate only ensure the privacy of the individual users in a *single training round*, and do not consider their privacy over multiple training rounds. To be specific, through gathering multiple rounds of collected data and participating information, individual model may be reconstructed from the aggregated models. In fact, I show that after a sufficient number of rounds, all local models can be recovered with a high accuracy if the server uniformly chooses a random subset of the users to participate at every round. As shown in Figure 3, performing model inversion attack with the recovered local models yields reconstructed images with a similar quality as the original images. This motivates an interesting future direction where I study long-term user privacy in FL. To that end, I define a new metric to capture long-term privacy guarantees in FL. This long-term privacy condition requires that the server cannot reconstruct any individual model using the aggregated models from any number of training rounds. Based on the newly defined long-term privacy, I have proposed a preliminary scheme, named MultiRoundSecAgg, a privacy-preserving structured user selection strategy that ensures the long-term privacy of the individual users over any number of training rounds. The main idea is to restrict certain sets of users, denoted

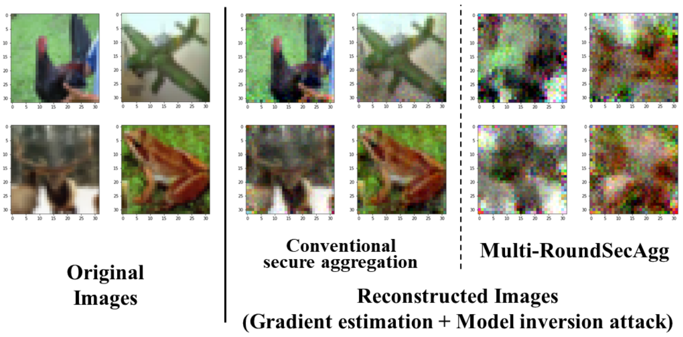

Original Images
Conventional secure aggregation
Multi-RoundSecAgg
Reconstructed Images (Gradient estimation + Model inversion attack)

Figure 3: A qualitative comparison of the reconstructed images in two settings is shown. The first setting corresponds to the case that model privacy with random user selection (e.g., `FedAvg`) is protected by conventional secure aggregation schemes including Turbo-Aggregate [1] at each round. The second setting corresponds to my preliminary result, named Multi-RoundSecAgg, which ensures the long-term privacy of individual models over any number of rounds, and hence model inversion attack cannot work well.

as batches, to either participate together or not participate at all. While this simple scheme ensures the long-term privacy as shown in Figure 3, it incurs vulnerable to user dropouts. I plan to elaborate the preliminary result to provide robustness against user dropouts and further optimize the batch structure to increase the convergence rate.

Real-world demonstration of FedSpace. While extensive numerical evaluations of FedSpace over the real-world imagery dataset and the real-world satellite constellation demonstrated its effectiveness, there are couple of challenges for real-world implementation. First, we assume that multiple ground stations have unlimited backhaul bandwidth such that they can be viewed as a single parameter server to maintain a single global model. In the real-world setting, however, this is not the case. I plan to model the multiple ground stations as multiple nodes in network topology and propose a multiple dynamic aggregation strategy based on the time-varying and deterministic network topology. Second, for simplicity, I assume that the ground station uses the same dataset (captured by the satellites) to train a regression model to estimate the utility function which is used to solve the optimization problem. In the real-world setting, it is infeasible for ground station to download all images to train the regression model due to limited bandwidth and regularization on the imagery resolution. I plan to leverage transfer learning or to utilize another FL framework to train the regression model.

Secure aggregation framework for satellite constellation. In FedSpace, I have not considered the model privacy even though the parameter server (or ground stations) can reveal the extensive information about the satellite imageries via model inversion attack, which may break the regulation restrictions and privacy concerns of high-resolution images captured by the satellites. Investigating a secure aggregation protocol to protect the model privacy would be very interesting and impactful future direction as it can be applied to many applications of asynchronous FL with sequential connectivity.

References

[1] J. So, B. Güler, and A. S. Avestimehr, "Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning," *IEEE Journal on Selected Areas in Information Theory*, vol. 2, no. 1, pp. 479–489, 2021.

[2] J. So, B. Güler, and A. S. Avestimehr, "Byzantine-resilient secure federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 7, 2020.

[3] J. So, B. Güler, and A. S. Avestimehr, "Codedprivateml: A fast and privacy-preserving framework for distributed machine learning," *IEEE Journal on Selected Areas in Information Theory*, vol. 2, no. 1, pp. 441–451, 2021.

[4] J. So, B. Guler, and A. S. Avestimehr, "A scalable approach for privacy-preserving collaborative machine learning," in *Advances in Neural Information Processing Systems*, 2020.

[5] J. So, C. He, C.-S. Yang, S. Li, Q. Yu, R. E. Ali, B. Guler, and S. Avestimehr, "Lightsecagg: a lightweight and versatile design for secure aggregation in federated learning," *accepted to 5th Conference on Machine Learning and Systems (MLSys)*, 2022.

[6] J. So, K. Hsieh, B. Arzani, S. Noghabi, S. Avestimehr, and R. Chandra, "Fedspace: An efficient federated learning framework at satellites and ground stations," *arXiv preprint arXiv:2202.01267*, 2022.

[7] C. He, S. Li, J. So, M. Zhang, H. Wang, X. Wang, P. Vepakomma, A. Singh, H. Qiu, L. Shen, *et al.*, "Fedml: A research library and benchmark for federated machine learning," *arXiv preprint arXiv:2007.13518*, 2020.