You Can Have Your Cake and Eat It Too: Ensuring Practical Robustness and Privacy in Federated Learning

Nojan Sheybani, Farinaz Koushanfar

University of California, San Diego
{nsheyban, fkoushanfar}ucsd.edu

Federated learning (FL) has gained popularity as a collaborative method for training a centralized model using data distributed across multiple parties. This approach involves transmitting model updates without necessitating the sharing of the actual data among the parties. Recently, privacy-preserving FL has emerged as a significant topic of research in the field of collaborative AI. In standard FL schemes, clients send their updates in plaintext to the central model. While protecting the raw data, these updates can sometimes leak sensitive information, such as the users' private training data through gradient inversion attacks. Privacy-preserving FL aims to provide robustness to this by ensuring that the central model never sees any users' individuals updates. While there have been a myriad of techniques that have been employed to provide these privacy guarantees, the most prominent solution utilizes secure multi-party computation (MPC). MPC is a widely used technique that allows n parties to jointly compute a function $f(x_1, ..., x_n)$ on private inputs from each party, without leaking any information or revealing the inputs to each other.

Using MPC as the core privacy-preserving technique, (Bonawitz et al. 2017) proposed the concept of secure aggregation, in which the main idea is to allow the clients to jointly compute the aggregate updates of their updates that can be shared to the central model. Secure aggregation only reveals the final aggregation result to the central model, ensuring users' private training data remains secure. Unfortunately, hiding individual updates poses a large threat to the central model, as malicious users can send invalid updates that compromise the integrity of the FL training procedure. These attacks are known as *Byzantine attacks*, and they are done by malicious clients who modify their local updates to degrade central model accuracy. While there have been proposed defenses to these attacks using popular privacy-preserving primitives, such as fully homomorphic encryption, they often face trouble balancing an all-important question that is present in all privacy-preserving systems: How much utility and practicality am I willing to give up to ensure privacy and robustness?

Inherently, FL robustness is very challenging to guarantee, especially when trying to maintain privacy. Compared to standard ML settings, FL's open training process allows for malicious clients to easily go under the radar. Alongside this, malicious clients can easily collude to attack the training process continuously, and without detection. FL models are also still susceptible to attacks on standard ML training procedures. This massive attack surface makes balancing the tradeoff between utility, practicality, robustness, and privacy extremely challenging.

Naively, the best way to ensure robustness against Byzantine attacks is for the central model holder to check each user's update and ensure it is reasonable. Obviously, this does not provide privacy, but it leads us to the following question: Can we check datapoints without learning anything about the datapoints? Zero-knowledge proofs (ZKPs) are a powerful cryptographic primitive that solve this problem. ZKPs allow a prover $\mathcal{P}$ to prove to a verifier $\mathcal{V}$ that they know a secret value w, without revealing anything about w. ZKPs can be powerful in providing robustness while maintaining privacy, as a prover can prove that their data is compliant to a verifier's rules, without revealing anything about their private data. In FL settings, this can be extended to user's proving that their local updates are within a certain acceptable range to the central model holder, without revealing anything about their local updates. While computationally intensive, ZKPs are an extremely powerful primitive to ensure robustness against attacks, while maintaining privacy.

Our recent work, `zPROBE` (Ghodsi et al. 2023), presents a novel framework for low overhead, scalable, private, and robust FL in the malicious client setting. With a combination of interactive ZKPs and MPC techniques, `zPROBE` provides robustness guarantees without compromising client privacy. This work can operate in practical runtimes, with little to no effect on utility.

References

Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H. B.; Patel, S.; Ramage, D.; Segal, A.; and Seth, K. 2017. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 1175–1191.

Ghodsi, Z.; Javaheripi, M.; Sheybani, N.; Zhang, X.; Huang, K.; and Koushanfar, F. 2023. zPROBE: Zero peek robustness checks for federated learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4860–4870.