Written Submission CSC588A

By: Payton Murdoch, V00904677

Due: August 20, 2024

1 Overview

The paper we selected is "Private, Efficient and Accurate: Protecting Models Trained by Multi-party Learning with Differential Privacy," referenced as [1]. To delve into the context of this paper, we first need to establish a definition for multi-party learning and Secure Multi-party Computation. Multi-party learning, or MPL for short, is a generalized framework that allows Machine learning models to be trained by numerous independent sources or parties holding data. Data privacy laws and regulations constrain all these parties training the model. Therefore, methods needed to be developed to allow this data to be transmitted to the model while maintaining privacy.[2] A standard related to these methods has been developed, denoted as Secure Multiparty Computation or SMPC, which employs cryptography on the distributed data so that no other party can see the unencrypted data stream.[3]

With SMPC employed within MPL, we can state that there is high security surrounding the private data in question. However, our paper notes that there exist different attack vectors, such as membership inference attacks, that can bypass the protections of MPL by directly querying the trained models to see if a piece of data they input exists as an output and, therefore, exists as a data point, which breaks the security of the framework.[1] Thus, [1] proposes employing differential privacy to increase protection against the attack mentioned above vectors. This comes with some caveats, as differential privacy adds noise to data and innately results in a loss of accuracy. In many studies, MPL lacks efficiency concerning the secure training of the models. As such, in consideration of this, [1] further recommends techniques that balance privacy with accuracy and efficiency.

Let us begin with Machine-learning-based techniques employed by [1] to maintain accuracy and efficiency before we discuss something as complex as the Differential Privacy methods proposed. There are two proposed optimization methods for training. First, each party will apply universal feature extraction methods to get high-level features that emphasize accuracy based on the input data, and these features will be utilized to train linear or neural networks with a few layers. As this feature extraction is universal, parties only need to add obfuscation through noise to allow for DP integration, emphasizing efficiency. The second method to preserve accuracy and efficiency is to enable the parties to aggregate locally trained models into a global initial model. By this, we mean that each party will be tasked with teaching one model based on a subset of data. Then, local model parameters can be averaged, resulting in the global model, or the model with the maximum accuracy could be selected as the initial global model.[1]

Next, we will discuss the algorithms proposed by [1], which integrate two different iterations of Differential Private Stochastic Gradient Descent, also called DPSGD. SGD aims to tune the model iteratively by updating model parameters based on the loss gradient denoted by a subset of data points. DPSGD expands upon this by incorporating noise to obfuscate based on the gradient so that the influence of a single data point cannot be determined. This is achieved by taking subsets of the data points, computing the loss function, clipping it to fall within a sensitivity bound and learning from that to derive subsequent iterations of the training model.[4] The first iteration is the standard DPSGD. However, the next iteration of SGD,

denoted as Secure DPSGD, is a custom-devised algorithm based on the inverse of the square root algorithm for the data points. While its functionality is essentially the same, it transforms the loss gradients into these inverse square roots to introduce further obfuscation on the data.[1]

Lastly, let us briefly review the entire Private, Efficient and Accurate framework proposed to better our understanding before delving into the results of [1]. To begin, each party involved will have a subset of unique data, and they will conduct feature extraction using a universally established method. Following the feature extraction, parties will initialize their iteration of a machine learning model, which will be trained with the data through the standard DPSGD protocol. Following this, the trained models are aggregated alongside the features extracted by all the parties. These features now make up a data private global dataset. Now, a global model is selected based on the average of parameters or the most accurate local model. The dataset and model chosen will be run through the Secure DPSGD protocol to iteratively train on all available data so that the resulting model can operate as needed by all parties as a DP-MPL Global Model.[1]

Finally, we will review the results for [1] now that we have covered the techniques involved. The paper proposes implementing PEA with two open-source MPL frameworks (TF-Encrypted and Queqiao) with the logistic regression classifier and using HOG, BERT and SimCLR feature extraction models on three datasets. In [1], the authors state that they maintained an accuracy of 88%, which matches that of CryptGPU, a state-of-the-art framework, in a fraction of the time. CryptGPU clocks in at 16 hours of computation time, with TF-Encrypted and Queqiao clocking in at 7 and 55 minutes. The results are further perpetuated by other framework comparisons concerning accuracy. Compared to CAPC and DD-Gauss federated learning DP-MPL frameworks, TF-Encrypted and Queqiao show an actual increase in accuracy of approximately 2-3%. Furthermore, there is the same margin of accuracy increase concerning global models trained in TF-Encrypted and Queqiao, compared to models trained simply in the local setting. Lastly, compared to plaintext MPL frameworks, we can see that accuracy for TF-Encrypted and Queqiao are only reduced by a total of 1-2%, showing high efficiency, marginal difference in accuracy and DP-privacy.[1]

2 Related Works

Concerning related works, the first work selected was "Secure Deep Neural Network Models Publishing Against Membership Inference Attacks Via Training Task Parallelism," referenced as [5] and cited in the presentation. While this paper does not address MPL frameworks directly, its significance comes from its aim to directly address the example attack vector denoted in [1]. As we know, Membership Inference Attacks, among other attack methods, were cited as the main motivation for developing more rigorous MPL frameworks incorporating Differential Privacy, as traditional security implementations of MPL overlook them. This paper takes this to the extent by which it employs threat modelling so that we can get definitive results for the reduction of accuracy for the attack following the implementation of its DP Machine Learning Publishing solutions. Of course, as we stated before, [5] differs from [1] because it does not address MPL frameworks. Instead, it works with a single party and a single deep neural network model, which is trained on multiple near-identical tasks, allowing for the grouping and averaging parameters that can be shared to reconstruct a model without knowledge of input data.[5]

The second related work is the article "Membership Inference Attack against Differentially Private Deep Learning Model," referenced as [6]. This paper has the utmost importance concerning [1]. As noted in the prior paragraph, a main concern with our chosen paper is that it does not definitively show us the results of the increased security of the models through threat modelling. [6] directly addresses this as it conducts white-box and black-box threat modelling specific to the Differential Privacy-based Stochastic Gradient Descent Deep learning model. White-box means that the attacker knows all information/parameters of the model, and black-box does not have this information. This is further perpetuated as it conducts this threat modelling using two of the three datasets interpreted within [1]. This being CIFAR-10 and MNIST. This better establishes the security of our DPSGD MPL framework regarding membership inference attacks.

Similar to the previously related paper, this one does not incorporate the multifaceted aspect of the MPL framework, where a global model is trained off of multiple iterations of DPSGD. Additionally, the models utilize universal feature extraction methods in [1] and [6], and the models are tuned to perform optimally with the individual dataset they are using directly.

3 Future Direction

I want to address my main grievance by expanding upon the paper and proposing a future project plan. My comment, of course, concerns the validity of the security of the MPL framework. As depicted in the motivation for [1], a main aspect of creating the DPSGD framework is to counteract membership inference attacks. However, the paper does not measure or conduct any threat modelling to prove its effectiveness. I sought out [5] and [6] and noted their significance. Of course, you can make inferences based on other works utilizing similar methods to determine its security. However, directly modelling the attack would be more effective. Therefore, we will consider this, and our project plan would be to conduct membership inference attacks on the proposed MPL framework. The protocol to implement this is dependent on which version of the threat we hope to model, as depicted in [7], we can either consider a black-box model, where the attacker does not know the specific training algorithm or model structure or a white-box model, where they have an understanding of such. As this is a custom algorithm, it would be difficult to conduct a black-box attack unless the MPL DPSGD algorithm is publicly published, so a white box would be the easier threat to model. To create a successful threat model, the attackers construct a series of shadow models trained similarly on similar data records so that they behave similarly. The shadow models publish records of labelled denoted "in" or "out," based on these records, they examine the outputs of the true model and attempt to infer if the data point is a part of it.[7]

Based on this extension, we need to break this down into a series of major milestones to propose a weekly schedule for a seven-week project. Milestone 1 would replicate the construction of the initial MPL frameworks utilizing DPSGD algorithms and the two open-source frameworks, Queqiao and TF-Encrypted, as described in [1]. Milestone 2 would construct the denoted shadow models for each model, which would similarly interpret the data and thus only feature minor changes from Milestone 1 as described in [7]. Milestone 3 would require trial and error as we would compute the n number of shadow models to establish sufficient records for the comparison. In Milestone 4, we would develop code to aggregate the results from the Global Secure DPSGD model and the records from our shadow models to infer membership. In milestone 5, we will run the tests multiple times and aggregate the results to determine the accuracy of the attack. As we see, there are five milestones and seven weeks. This is because Milestone 1 would be a three-week-long component, as construction of the initial MPL framework is the most time-intensive and complex task.

Of course, this is not without its challenges. Even though we are utilizing open-sourced frameworks, the complexity of the framework may be above our level of understanding, thus resulting in the scope of this project becoming increasingly outlandish during the 7-week time frame. Furthermore, let us consider system and runtime requirements. As stated in [1], they utilized three Linux systems with identical clock speeds and RAM; we may not have access to those resources. Even utilizing virtual machines may prove challenging for this task. By extension, we run the entire MPL framework for each model when considering the shadow models we need to create. Therefore, we need to run the same configuration of $3n$ to generate and save all shadow models. These are intensive tasks and possibly unfeasible with our system limitations.[7]

References

[1] W. Ruan, M. Xu, W. Fang, L. Wang, L. Wang, and W. Han, "Private, Efficient, and Accurate: Protecting Models Trained by Multi-party Learning with Differential Privacy," *2023 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA, 2023, pp. 1926-1943, doi: 10.1109/SP46215.2023.10179422.

[2] M. Gong, Y. Gao, Y. Wu, Y. Zhang, A. K. Qin, and Y.-S. Ong, "Heterogeneous Multi-Party Learning With Data-Driven Network Sampling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 11, pp. 13328-13343, 1 Nov. 2023, doi: 10.1109/TPAMI.2023.3290213.

[3] N. Khalid, A. Qayyum, M. Bilal, A. Al-Fuqaha, and J. Qadir, "Privacy-preserving artificial intelligence in Healthcare: Techniques and Applications," *Computers in Biology and Medicine*, vol. 158, p. 106848, May 2023, doi: 10.1016/j.compbiomed.2023.106848.

[4] M. Abadi et al., "Deep learning with differential privacy," *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, Oct. 2016, doi: 10.1145/2976749.2978318.

[5] Y. Mao, W. Hong, B. Zhu, Z. Zhu, Y. Zhang, and S. Zhong, "Secure Deep Neural Network Models Publishing Against Membership Inference Attacks Via Training Task Parallelism," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 3079-3091, 1 Nov. 2022, doi: 10.1109/TPDS.2021.3129612.

[6] A. Rahman, T. Rahman, R. Laganiere, N. Mohammed, and Y. Wang, "Membership Inference Attack against Differentially Private Deep Learning Model," *Transactions on Data Privacy*, no. 11, pp. 61–79, 2018.

[7] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership Inference Attacks Against Machine Learning Models," *2017 IEEE Symposium on Security and Privacy (SP)*, San Jose, CA, USA, 2017, pp. 3-18, doi: 10.1109/SP.2017.41.