Improvements and Analysis of Private Ensemble-Based Federated Learning

Roy Rinberg **Arjun Nichani**
Columbia University
{royrinberg, an3073}@columbia.edu

Abstract

In the last decade, there has been an outpouring of interest and research in both privacy preserving and federated machine learning. Ensemble-based federated learning entered the state-of-the-art around 2016 with the advent of Private Aggregation of Teacher Ensembles (PATE) [8], which trains federated "teacher" models to vote to label datapoints using differential-privacy to compute the argmax of the votes. Our primary investigation is into investigating improved voting mechanisms for ensemble-based federated learning models. We will also introduce a novel security analysis of ensemble-based federated learning systems like PATE and CaPC. We highlight a clear gap in the current state-of-the-art for federated learning, and suggest a simple, but new, addition to existing models to *complete the loop* in dramatically lowering the trust assumptions in ensemble-based federated learning systems. We provide empirical results for our ensemble-based learning mechanisms, along with a simple formula for their privacy guarantees; we also provide an an algorithm for secure, private, confidential computation.

1 Introduction

Federated learning (FL) is a method of collaborative training in which multiple entities train a model under the supervision of a central entity. This is vital in fields where data cannot be shared (e.g. medicine). With the development of federated learning, methods such as Federated Stochastic Gradient Descent (FedSGD)and Federated Averaging (FedAvg) emerged as effective ways of training models in federating learning settings. Despite not explicitly sharing data, FedAvg and FedSGD both fail to provide privacy guarantees, and information can still be learned about the data, through both white-box and black-box attacks [7] [3].

We discuss in further detail two ensemble-based federated learning algorithms Private Aggregation of Teacher Ensembles (PATE) [8] and Confidential and Private Collaborative Learning (CaPC) [3]. In this paper, we propose a novel adaptation to ensemble-based federated learning systems like PATE and CaPC: reputation based ensemble-based federated learning. We also provide a security and privacy analysis for existing ensemble-based federated learning systems.

Author Contributions: Arjun Nichani: Arjun explored Federated Averaging and Federated SGD algorithms and developed code to run these algorithms on an image classification problem (not included in this paper). Arjun did a literature review on existing FL methods, their applications, and their weaknesses. Arjun wrote the introduction and related works sections for FL and DP.
Roy Rinberg: Roy explored the landscape of ensemble-based federated learning algorithms, and directed the research. Roy developed the reputation-based voting model for PATE and CAPC. Roy wrote the code and conducted experiments for the PATE experiments and analysis, as well as the code to artificially sample datasets in unbalanced ways. Roy developed the security analysis for PATE and CaPC. Roy wrote the method/algorithm section as well as experimental results, and related works sections on PATE and CaPC.

Preprint. Under review.

2 Related Works

2.1 Federated Learning

The standard model of FL is exemplified through Federated SGD (FedSGD) and Federated Averaging (FedAvg). In both FedSGD and FedAvg, a central agent controls a model; the model is sent to edge-devices, where the model is trained for several batches, and the gradients are updated on-device, then the difference in gradients are sent back to the central agents. The gradient updates are averaged across the different models; the differences between FedAvg and FedSGD is primarily in how the gradient updates are averaged [12] [5]. Both of these methods allow for training on disjoint data sets without actually needing to observe the data; however, common to both do not offer any formal privacy guarantees.

2.2 Differential Privacy

Differential Privacy (DP) is a composable mathematical construct which adds noise into queries on a database (like during the training process of a model) in order to bound the maximum contribution of any individual user. A randomized algorithm A is (ϵ, δ)-differentially private if for any query and for all "adjacent" datasets D_0 and D_1. [1]

$$\Pr[M(D_0) \in S] \leq e^{\epsilon} \Pr[M(D_1) \in S] + \delta \tag{1}$$

2.3 Private Aggregation of Teacher Ensembles (PATE)

PATE is a general paradigm for training models with formal differential-privacy guarantees. The general framework works as follows:

1. Collect (potentially sensitive) labeled data $\mathcal{D}$, and split it into n datasets $\mathcal{D}_i$.
2. Train a model T_i on each dataset $\mathcal{D}_i$ (not allowing a model to see any data outside of its dataset). We call these models "Teachers". [2]
3. Collect a new unlabeled dataset, $\mathcal{D}'$
4. For each point x in $\mathcal{D}'$
   (a) Each teacher T_i makes prediction $T_i(x)$, and sends the prediction to a central aggregator.
   (b) Aggregate all the teacher votes, and add Laplacian noise (which is a common differential privacy technique) to the vote counts.
   (c) Generate a label for x by taking an argmax of the teacher votes. $l_i = \text{argmax}(\text{teachervotes}_i + \text{Lap}(\epsilon)$.
5. Train a new model S, called the "student", given the now labeled dataset, $\mathcal{D}'$, with a computable (ϵ, δ).
6. Release S, knowing that any query to S cannot reveal any more information than a (ϵ, δ) query.

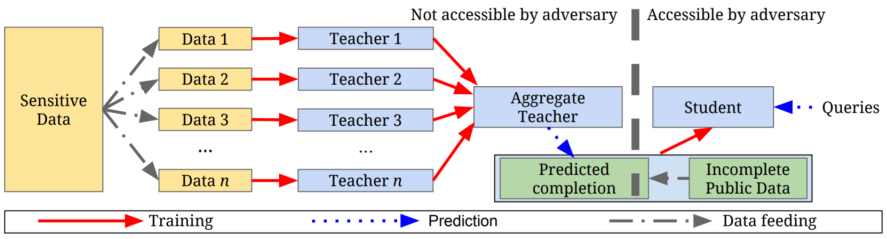

Figure 1: PATE diagram. [8]

[1] We assume the reader is familiar with DP, for more of an introduction, we recommend The Algorithmic Foundations of Differential Privacy by Aaron Roth and Cynthia Dwork

[2] This is a very natural design for Federated Learning, where a device or data center does not have access to other data.

PATE is model-agnostic, and makes no assumptions about the models used for teachers and for students (even a collection of humans would work). PATE can have very tight privacy guarantees, because the noise added only contributes when teachers are not in agreement, and a small number of teacher votes can sway the label. PATE is able to attain strong *data-dependent* privacy guarantees through two mechanisms. 1. Any user's data only contributes to a single model which only has a single vote out of many, so the contribution of a single user's data is limited. 2. DP is added to the argmax, obfuscating not only the contribution of a single model (typically trained on multiple users).

2.4 Confidential and Private Collaborative Learning (CaPC)

Prior to introducing CaPC, we briefly introduce two cryptographic concepts. **Homomorphic Encryption** (HE) is a form of encryption, which describes an encryption function E, decryption function D, and a set of functions $\{f_i\}$, such that: $E(f_i(x)) = f_i(E(x))$ and $D(E(x)) = x$. In words this means that if you apply a function on the a value and then encrypt it, this is equivalent to applying the function on the encrypted value. **Secure Multi-Party Computation** (MPC) is a form of encrypted computation that enables multiple parties to compute a joint function (like an average) without learning anything about the other parties' data.

CaPC Learning is nearly identical to PATE, except it removes the student that is learned at the end of the process. In CaPC, querying agents query all the ensemble with a datapoint, to label simply query the teachers directly, and take the noisy-argmax to label their datapoint[3]. CaPC uses MPC in order to calculate the argmax, so that no agent needs to be trusted. Additionally, CaPC uses HE in order to send each teacher datapoints to label, without the teachers learning what data they are labelling. [4] In addition, CaPC also describes a mechanism for teachers to update their models actively. Through the "*active learning paradigm*", a teacher T_i poses queries in the form of data samples x and all the other teachers T_j for $i \neq j$ provide predicted labels.

3 Method/Algorithm

3.1 Reputation-Based PATE

Our first contribution is to make a simple observation about distributions of data in the world, and in particular in federated learning settings: 1. Data follows a long-tailed distribution [2][6][13] 2. Data is not IID across devices [9]. Because of this very natural, and well-documented observations, we reason that the simple voting mechanism of PATE does not make sense, as it weights each model's contribution equally, regardless of the teacher's performance on that class. The CaPC model partially addresses this issue using "Active Learning", where a teacher will selectively query the other teachers for labels on specific data points, so as to improve its accuracy over time. [5]

We propose an alternative voting system to PATE and CaPC's "one-model one-vote" system; we investigate several methods for how to weight the teachers' votes. We first observe that the very nature of unlabeled data, is that a model does not know the data point's label; therefore, the weighting of models should not be data-dependent. After training, we propose an evaluation period, where each model gets assigned a weight, according to the model's performance on a validation dataset. After this, the teachers vote using their weight [3].We provide an algorithm of this in Security Analysis and Removing Trust in Weighted Ensemble-Based Federated Learning Systems.

3.1.1 Privacy Proof for Weighted PATE

While it seems like the calculation is different for weighted teacher voting, we observe that for any mechanism in DP ϵ is defined respect to sensitivity $||f||$, the maximal contribution of a single user to the result. Therefore, by changing the voting weights, we are change the sensitivity from 1 to $max - weight$, which, to preserve privacy, would require decreasing ϵ by the same amount. In short, to compute the data-dependent privacy budget bounds for weighted teacher voting, one can use the

[3] note: this means the privacy budget grows over time, unlike PATE's student

[4] "MP2ML: A Mixed-Protocol Machine Learning Framework for Private Inference" provides a comprehensive list of libraries that support pure HE or MPC protocols for secure inference of neural networks [4].

[5] One potential research direction is to study how teachers should select point to query the other teachers in the active learning paradigm.

same methodology as computing data-dependent privacy budget bounds for PATE and CaPC, except changing ϵ to $\epsilon' = \frac{\epsilon}{\text{max-weight}}$. More information is in the appendix section Privacy Proof Extras.

3.2 Security Analysis and Removing Trust in Weighted Ensemble-Based Federated Learning Systems

PATE has a central aggregator which is able to fully see all the labels from teachers. It has been shown that it is possible to generate inference attacks on a model, simply from observing (and minorly manipulating) api calls to a model [9] [11]. This implies a strong trust model in the central aggregator. CaPC removes that by using MPC to compute its votes, but it still requires having a "Privacy Guardian" and trusting that the Privacy Guardian does not collude with the querying agent; this is a moderate trust assumption. However, an important so-far-overlooked trust assumption is that of the teachers. Once the weights are decided, there is nothing preventing a teacher from maliciously attempting to alter the outcome of the voting aggregates, by sending incorrect labels. Because the data the teachers receive is encrypted, the worst the teachers can do is randomly guess an incorrect label; they cannot strategically collude. However, this can still have a large effect when teachers do not reach a clear consensus.

We now propose our trustless Ensemble-based FL algorithm.

3.2.1 Trustless Ensemble-based Federated Learning Algorithm

1. Split a dataset $\mathcal{D}$ across devices, creating datasets $\mathcal{D}_i$.
2. For each dataset, train a local model T_i on device.
3. Compute an evaluation of the model T_i on each of the other datasets $\mathcal{D}_i$.
   (a) Each teacher T_i sends a random sampling of their homomorphically encrypted data to the central agent.
   (b) The central agent collects all the encrypted data, and sends them to the other teacher T_j.
   (c) Teachers T_j send back the homomorphically encrypted labels, back to the central agent, which forwards them to the original teacher T_i.
   (d) T_i decrypts the labels, and sends back accuracies.
   Security Analysis: The central agent computes a weighting for each of the different agents, but does not ever see the model, or labels of the agents. No teacher sees another teacher's unencrypted data. No teacher sees another teacher's weighting. [6]
4. The central agent delivers each teacher T_i a homomorphically encrypted weight w_i by the central agent.
5. Using the same mechanism as described in CaPC, use MPC to enable trustless Voting. Each agent sends a "one-hot" vector to a privacy guardian, who forwards an aggregated version to the central party. The full MPC algorithm is explained in the appendix section Multi-Party Computation. The only difference is that the agent sends a one-hot vector multiplied by their homomorphically encrypted weight w_i.
6. Optional last step : to ensure that teachers vote with the same model that delivered them the accuracy-based weighting they received from the central aggregator, teachers can submit ZK-SNARKs alongside their votes to prove they used the same model to compute the new labels [7]

4 Experimental Results

4.1 Implementation

In order to evaluate trade-offs, all algorithms were evaluated on an image classification task with the MNIST dataset. First all the teacher and student models were trained with 2-layer fully-connected

[6] It may be possible for the central agent to not even learn the weighting.

[7] A ZK-SNARK (Zero Knowledge Succinct Non-interactive Argument of Knowledge) is a short zero-knowledge proof that can verify the output of arbitrary functions, without revealing information about the computation itself [10].

networks (FCN), then with 2-layer CNNs. The CNNs seem to require more data to train well, and so some of the teachers did significantly worse; the effect of different voting methods was much more obvious for larger spread of teacher accuracies.

4.1.1 Weighted Ranking

We investigate 5 weighting algorithms for the relative voting. Let $s =$ min-weight, let $m =$ max-weight. In all cases, except 1-model 1-vote, we follow the following formula : $w_i = v_i * (s - m) + s$[8]

- 1-model-1-vote (ordinary voting): each model's vote is weighted by 1. $w_i = 1$
- Accuracy: Evaluate a model's accuracy a_i on a testset. $v_i = a_i$
- Percentile: Evaluate all the model's accuracies, compute the percentile p_i a model's accuracy a belongs to. $v_i = p_i$
- Z-score: Evaluate all the model's accuracies, compute z-score z_i for a given model. $v_i = z_i$
- Ranking: Given an ordered ranking of n model's accuracies, let r_i be a model's rank from $0...n$. $v_i = r_i$.

4.1.2 Unbalanced Data

In particular, we investigate the performance on different kinds of data distributions. We use two terms to describe our data **Balanced Dataset**, to signify that the total dataset is balanced across classes (i.e. all classes are equally well represented in the aggregated dataset $\mathcal{D}$). And **Sampled IID**, denoting how the individual teachers' datasets $\mathcal{D}_i$ are distributed. We consider 3 cases: 1. Balanced data with IID sampled teacher datasets, 2. Unbalanced dataset with IID sampled teacher datasets, 3. Unbalanced dataset with non-IID sampled teacher datasets. Unbalanced data follows a $\frac{1}{x}$ distribution (from 100% to 40% of the data class size).

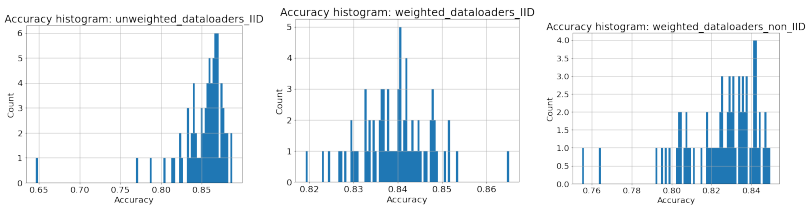

Figure 2: FCN - Accuracies of teacher models, given different data distributions.

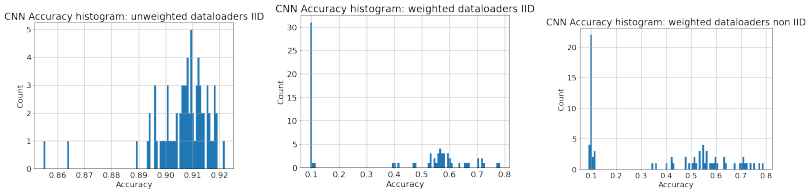

Figure 3: CNN - Accuracies of teacher models, given different data distributions.

We observe that the CNN models do particularly poorly on unbalanced datasets. Additionally, relative accuracies of the teacher models are more spread on the unbalanced dataset, and even more so on the non-IID sampled teacher datasets

4.2 Evaluation

We observe that the original PATE paper was able to achieve 98.00% accuracy using a 2 layer CNN and 250 teachers, with $\epsilon = 2.0$, on MNIST. Interesting, they found that simply the aggregated votes of the teachers (which is effectively what CaPC is), achieved accuracy of 93.18 %. Whereas they found the same model achieves 98.00% accuracy without differential privacy. [8]

Our results were not able to achieve the same accuracies, either in the teachers or in the students.

[8]We swept a range of values for the weights, we settled on a max-weight of 3, and a min-weight of 0.1.

4.3 Reputation Based Teachers

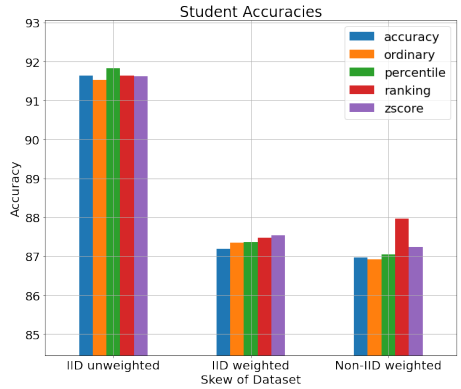

Figure 4: FCN models with $\epsilon = 2.0$ used for noisy argmax.

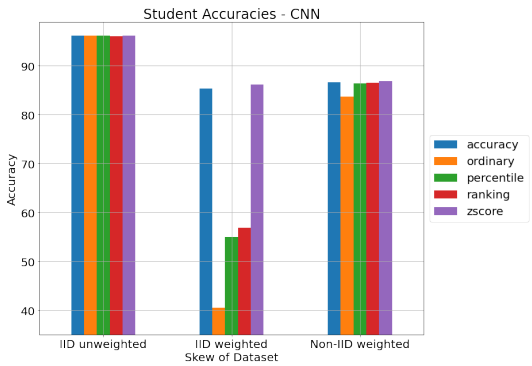

Figure 5: CNN models; $\epsilon = 0.75$ used for noisy argmax.

Figure 6: Test set accuracies of student models. Students trained for 25 Epochs. 75 teachers. *Note*: more clearly denoted plots for CNN available in appendix Student Accuracies

We compute the PATE-computed data-dependent privacy budget for our weighted dataset, and for a noisy-argmax voting with $\epsilon = 2.0$, the data-dependent privacy analysis, gets a privacy budget of 1.4 for the unweighted IID dataset, 6.4 for the weighted IID dataset, and 4.3 for the weighted non-IID dataset. We find that there is little difference in the privacy budget when using different epsilons for the noisy argmaxx (we trained with $\epsilon = 2.$, $\epsilon = 0.5$, and $\epsilon = 5.0$; data available upon request, and in the codebase). Thus we are impressed with the small effect on privacy budgets, relative to accuracy increases between changes in epsilon (see in appendix FCN Student Accuracies).

4.4 Experimental Discussion

We are not able to achieve the same accuracies that the original PATE paper achieved (98%, whereas our IID unweighted students only achieved 96%; however, this is likely a function of specific tuning of hyperparameters. We observe two very important details: that dataset skews have a dramatic effect on performance, and changing voting mechanisms has a much more profound impact when teacher accuracies have a large standard deviation. In the CNN model, the different voting mechanisms contributed to less that 0.07% difference; however in the non-IID weighted datasets, z-score weighting performed 3.5% better than one-model one vote. The effect was most evident in the IID weight datasets, where teacher performance was most skewed; one-model one-vote performed predictably poorly with bad teachers (40.5% accuracy), and z-score voting perform significantly better (86.2% accuracy). From this we conclude that while hyper parameter tuning could improve all performance of teachers and students; reputation-based ensemble-based federated learning performs more robustly in the presence of *bad teachers*. This is especially important, as we are most interested in the performance of the non-IID datasets, as there is substantial evidence that the world is filled with examples of long-tailed distributions [6][13].

5 Conclusion

We conclude briefly, recapping the highlights of the paper. We described PATE and CaPC, providing a algorithm to remove nearly all trust assumptions; which had a novel application of ZK-SNARKs, a technology so-far reserved primarily for blockchain technologies. We proposed a novel weighting algorithm for voting in ensemble-based federated learning systems, and ran our experiments. We were not able to achieve the original PATE paper's accuracy [8]; however, we did find significant evidence that reputation-based voting mechanisms, can make ensemble-based models significantly more robust to unbalanced data. This is especially significant due to their small effect on privacy budgets, relative to accuracy increases.

A Appendix

A.1 Privacy Proof Extras

This makes sense, as increasing the weight, increases the contribution of a single model, which decreases privacy, and to compensate one must decrease epsilon, which increases privacy.

For posterity, we include Lemma 3 from [8]. However, we observe that many codebases already provide PATE implementations, and importantly, PATE calculators. Notably, PySyft's 'syft.pate.perform_analysis' tool (though documentation only exists for version 0.2.9).

Lemma 1 *For $\gamma = \frac{\epsilon}{2}$, Let **n** be the label score vector for a database d with $n_{j*} > n_j$ for all j. Then*

$$Pr[M(d) \neq j^{\star}] \leq \sum_{j \neq j^{\star}} \frac{2 + \gamma(n_{j\star} - n_j)}{4exp(\gamma(n_{j\star} - n_j)} \tag{2}$$

A.2 Multi-Party Computation

1. The analyzer generates a private key "s" and corresponding public key "p", which they then share publicly.

2. Each contributor (company) converts their data into a number (which is trivial as data is stored in bits), then they generate an unbounded length random key mi and add it to their data d_i. Let $r_i = d_i + m_i$.

3. Each contributor sends r_i to the service provider.

4. Each contributor encrypts mi with the analyzer's public key, then sends the encrypted mi to the analyzer.

5. The service provider computes the sum $R = \sum r_i$, and publicly shares it.

6. The analyzer decrypts each mask mi , and computes the sum $M = \sum m_i$.

7. The analyzer computes $D = R - M$, which is equal to $\sum d_i$, through associativity (the rule that order of operations doesn't matter for addition: $(a + b) - c - d = (a - c) + (b - d)$). [1] [9]

[9] Note: This figure, and explanation of the algorithm was generated by Roy Rinberg for another course at Columbia, Policy for Privacy Technology, taught by Prof. Rachel Cummings.

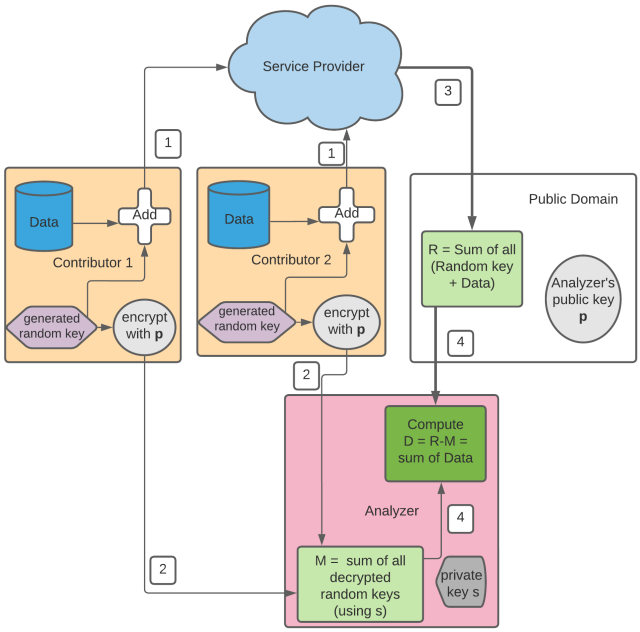

Figure 7: MPC for private averaging, which is trivially translated to private-argmax.

A.3 Student Accuracies

A.3.1 CNN Student Accuracies

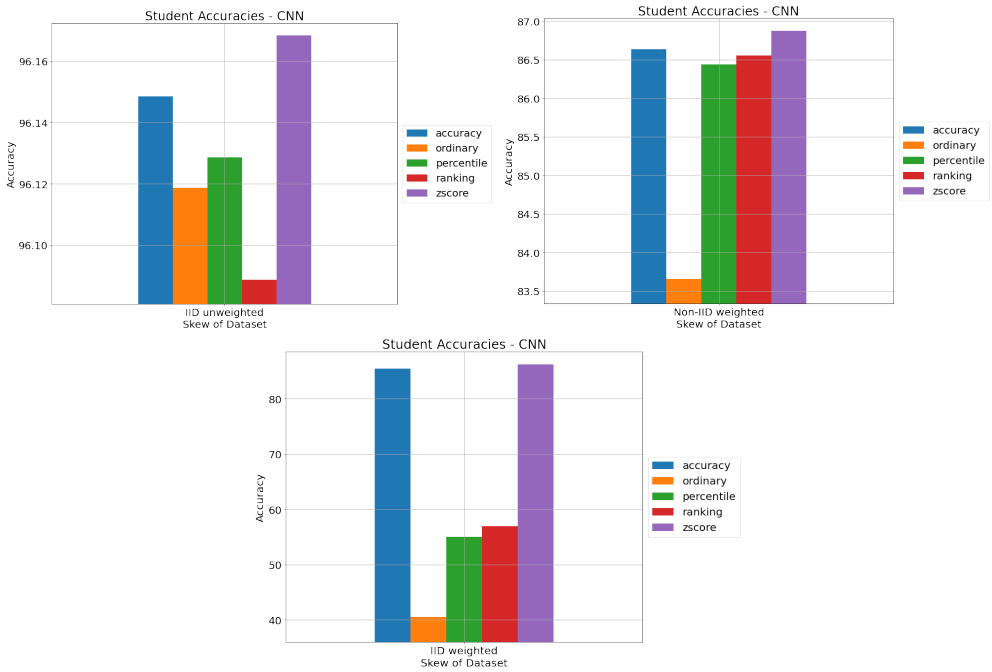

Figure 8: CNN Student Accuracies - $\epsilon = 0.75$

A.3.2 FCN Student Accuracies

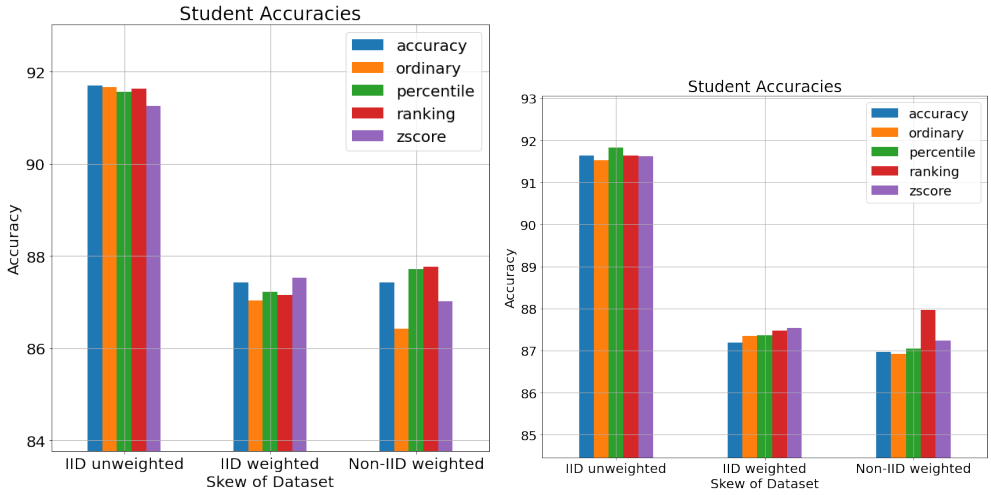

Figure 9: FCN Student Accuracies. Left: $\epsilon = 0.5$; Right: $\epsilon = 2.0$

We compute the PATE-computed data-dependent privacy budget for our weighted dataset, and find that there is little difference in the privacy budget when using a noisy argmax, when using different epsilons but there is a large effect on the skew of the data (we trained with $\epsilon = 2.$, $\epsilon = 0.5$, and $\epsilon = 5.0$; data available upon request, and in the codebase). For a noisy-argmax voting with $\epsilon = 2.0$, the data-dependent privacy analysis, gets a privacy budget of 1.4 for the unweighted IID dataset, 6.4 for the weighted IID dataset, and 4.3 for the weighted non-IID dataset.

References

[1] A. Bestavros F.Jansen A. Lapets, N. Volgushev and M. Varia. Secure multi-party computation for analytics deployed as a lightweight web application. https://www.semanticscholar.org/paper/Secure-multi-party-computation-for-analytics-as-a-Lapets-Volgushev/f69d9844bcc6dfb59607ab966fa33db02ce80bd4, 2016.

[2] R. Babbar and B. Scholkopf. Data scarcity, robustness and extreme multi-label classification. https://link.springer.com/content/pdf/10.1007/s10994-019-05791-5.pdf, 2019.

[3] A. Dziedzic Y. Zhang S. Jha N. Papernot C.A. Choquette-Choo, N. Dullerud and X. Wang. Capc learning: Confidential and private collaborative learning. https://arxiv.org/pdf/2102.05188.pdf, 2021.

[4] D. Demmler T. Schneider F. Boemer, R. Cammarota and H. Yalam. Mp2ml: A mixed-protocol machine learning framework for private inference. https://eprint.iacr.org/2020/721, 2020.

[5] D. Ramage S. Hampson H. McMahan, E. Moore and B. Arcas. Communication-efficient learning of deep networks from decentralized data. https://arxiv.org/pdf/1602.05629.pdf, 2016.

[6] G. Van Horn and P. Perona. The devil is in the tails: Fine-grained classification in the wild. https://arxiv.org/abs/1709.01450, 2017.

[7] R. Shokri M. Nasr and A. Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning from decentralized data. https://arxiv.org/pdf/1812.00910.pdf, 2018.

[8] U. Erlingsson I. Goodfellow N. Papernot, M. Abadi and K. Talwar. Semi-supervised knowledge transfer for deep learning from private training data. https://arxiv.org/pdf/1610.05755.pdf, 2017.

[9] et al. P. Kairouz. Advances and open problems in federated learning. https://arxiv.org/pdf/1912.04977.pdf, 2021.

[10] M. Petkus. Why and how zk-snark works. https://arxiv.org/pdf/1906.07221.pdf, 2019.

[11] C. Song R. Shokri, M. Stronati and V. Shmatikov. Membership inference attacks against machine learning models. https://www.cs.cornell.edu/~shmat/shmat_oak17.pdf, 2017.

[12] R. Shokri. and V. Shmatikov. Privacy preserving deep learning. https://www.cs.cornell.edu/~shmat/shmat_ccs15.pdf, 2015.

[13] D. Anguelov X. Zhu and D. Ramanan. Capturing long-tail distributions of object subcategories. https://openaccess.thecvf.com/content_cvpr_2014/html/Zhu_Capturing_Long-tail_Distributions_2014_CVPR_paper.html, 2014.