The full version of this paper can be found at: https://arxiv.org/abs/1610.05755

Machine Learning with Privacy by Knowledge Aggregation and Transfer

Nicolas Papernot*
Pennsylvania State University
ngp5056@cse.psu.edu

Martín Abadi
Google Brain
abadi@google.com

Úlfar Erlingsson
Google
ulfar@google.com

Ian Goodfellow
OpenAI
ian@openai.com

Kunal Talwar
Google Brain
kunal@google.com

Abstract

Machine learning relies on the availability of high-quality training data and—whether by its inherent nature, or by accident—this data will sometimes contain private information. When the model is to be published or made publicly accessible and the training data is not, it is important that the details of the sensitive training data cannot be inadvertently revealed by the model.

This abstract presents a generally applicable approach to providing strong privacy guarantees for machine learning training data. The approach is based on combining, in a black-box fashion, multiple machine learning models trained with disjoint sensitive datasets, such as data for different users. Because they rely directly on sensitive data, these models are used only as "teachers" for a "student" machine learning model. However, when training the student, the teachers transfer only the labels upon which they all generally agree, via a noisy aggregation mechanism.

The student has privacy properties that can be understood both intuitively (since no single teacher dictates the student's training) and formally, in terms of differential privacy. These properties address "glass-box" attacks of the kind that arise if an adversary not only queries the student but also inspects its internal workings.

The approach imposes only weak assumptions on how the teachers are trained. It applies to powerful, deep models, possibly with many layers and parameters. Our experiments demonstrate that the approach applies to real-world machine learning tasks, at a reasonable cost in accuracy, privacy, and software complexity.

1 Introduction

In certain applications, the data used by learning algorithms is *sensitive*: e.g., medical data [2]. Users need to trust institutions and companies collecting and handling their personal data with the enforcement of their *privacy*. Machine learning is known to be vulnerable to attacks forcing trained models to reveal private data processed during their training phase [8]. This can be explained by the fact that models with sufficient capacity can "overfit" and encode training data in their parameters.

Ideally, the privacy of training data could be guaranteed, in a manner that prevented overfitting and protected data confidentiality. Several formalizations of privacy have been proposed [5], among which *differential privacy* [6, 7] is established as a rigorous standard. Informally, a differentially private algorithm guarantees that the inclusion or exclusion of any single training example has a bounded effect on the probability of any outcome of the learning process. Such guarantees can be provided by designing a variant of the algorithm involving carefully selected random noise [6, 11]. This introduces ambiguity, a requirement for non-trivial privacy guarantees [7].

*Work done while the author was at Google.

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

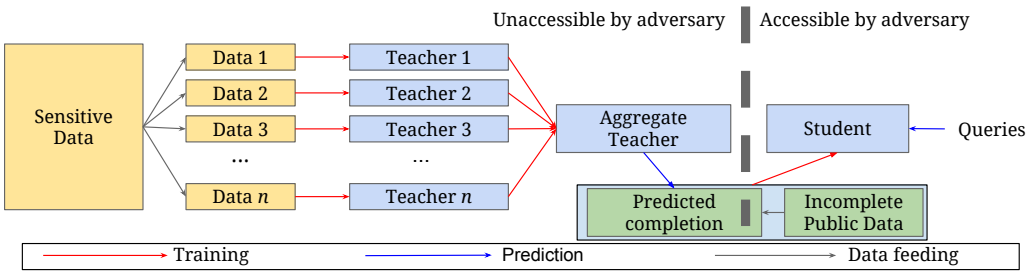

Figure 1: Overview of the approach: (1) an ensemble of teachers is trained on disjoint subsets of the sensitive data, (2) a student model is trained on public data labeled using the ensemble.

In our work, the training data contains the sensitive information that must be protected with differential privacy. Machine learning, as any statistical analysis tool, raises privacy concerns because predictions made could be harmful to the privacy of individual users—the presence or absence of an individual in a training corpus can change the output of a machine learning model despite the model being trained on a large corpus [3]. To understand why, consider a nearest neighbor classifier and how removing a single point from its training set is sufficient to change its predictions on certain inputs.

We present a generic strategy to learn privacy-preserving models, unlike some previous efforts [13, 1]. The strategy consists of an *ensemble* [4] of teacher models, trained on disjoint data, and a student model that learns to mimic the ensemble. When a quorum of teachers is reached, the corresponding prediction intuitively stems from generalization and not from overfitting to a specific training point—as all teachers were trained on disjoint subsets of the dataset. The student is allowed to see only the unified output of the ensemble, which doesn't convey the details of any single sensitive training point. The teacher ensemble returns noisy aggregated predictions to provide differential privacy.

Privacy is preserved better when the student requires few labeled examples and thus fewer queries to the teachers. The cost of deploying the student is independent from the number of predictions made because private information is only leaked during its training. We demonstrate that *semi-supervised learning* with *generative adversarial networks* [12] is an effective choice of student model.

An appealing aspect of our approach is that we can simultaneously provide intuitive and rigorous guarantees expressed in terms of differential privacy, making the solution compelling to both an expert and a non-expert audience. Our contributions are the following:

- We use semi-supervised learning together with the *moments accountant technique* [1] in our privacy analysis, allowing us to overcome limitations of previous teacher-student strategies. This allows us to improve the student's accuracy at no expense to the privacy cost.
- We evaluate our framework on MNIST and SVHN, allowing for a comparison of our results with previous differentially private learning efforts [13, 1]. Our classifiers are respectively $(2.04, 10^{-5})$ and $(8.19, 10^{-6})$ differentially private, with accuracies of 98.00% and 90.66%. In comparison, Abadi et al. obtained 97% accuracy with a $(8, 10^{-5})$ bound on MNIST [1]. In [13], Shokri et al. report about 92% accuracy on SVHN with $\epsilon > 2$ per model parameter and a model with over $300,000$ parameters. Naively, this corresponds to a total $\epsilon > 600,000$.

Our work improves upon [10] by (1) removing all assumptions made regarding the loss, and (2) providing a stricter privacy analysis to greatly reduce the numbers of teachers while achieving similar privacy/utility trade-offs. Hence, our approach is applicable to more classifiers and smaller datasets.

2 Private Learning with Ensembles of Teachers

Our approach, illustrated in Figure 1, consists of two stages: first an ensemble of teacher models is trained on disjoint subsets of the private data, and second a student model is trained to mimic this ensemble, using new unlabeled non-sensitive inputs that are labeled by querying the ensemble.

2.1 Training the Ensemble of Teachers

Instead of training a single model to solve the supervised task associated with a dataset (X, Y), we partition this dataset in n disjoint sets (X_n, Y_n) and train a model separately on each of these sets. Assuming that n is not too large with respect to the dataset size, we obtain n classifiers f_i called *teachers*. We then deploy them as an *ensemble* making predictions on unseen inputs x by querying each teacher for a prediction $f_i(x)$ and aggregating these into a single prediction.

The privacy guarantees of this ensemble stem from its aggregation. Let m be the number of classes in our task. The label count for a given class $j \in 1..m$ and an input $\vec{x}$ is the number of teachers that assigned class j to input $\vec{x}$: $n_j(\vec{x}) = |\{i : i \in 1..n, f_i(\vec{x}) = j\}|$. If we simply apply *plurality*—use the label with the largest count—the ensemble's decision may depend on a single teacher's vote. Indeed, when two labels have a vote count differing by at most one, there is a tie: the aggregated output changes if one teacher makes a different prediction. Therefore, we add random noise to the vote counts n_j to introduce ambiguity: $f(x) = \arg\max_j \left\{ n_j(\vec{x}) + Lap\left(\frac{1}{\varepsilon}\right) \right\}$. In this equation, the ε is a privacy parameter and $Lap(b)$ the Laplacian distribution with location 0 and scale b. Smaller values of ε provide stronger privacy guarantees but degrade the accuracy of the aggregation: the noise added can potentially change the respective order of label counts.

However, these guarantees do not necessarily hold if an adversary has access to the model itself. Indeed, as each teacher f_i was trained without taking into account privacy, it is conceivable that the teachers have sufficient capacity to retain details of the training data [8]. Furthermore, as more queries are made to the ensemble, the noise added must increase to continue providing strong privacy guarantees. The utility of the model is thus subject to strong degradation. To address these limitations, we then train a student model based on a fixed number of labels predicted by the teacher ensemble.

2.2 Semi-Supervised Transfer of the Knowledge from an Ensemble to a Student

We train a student on nonsensitive and unlabeled data that we label using the output of the aggregation mechanism. This student model is the one deployed to answer user queries. This fixes the privacy loss to a a value that is constant with respect to the number of *user* queries made to the *student* model. The privacy loss is a function of the number of queries made to the *teacher ensemble* during student training and does not increase as end-users query the deployed student model. Thus, the privacy of users who contributed to the original dataset used to train the ensemble of teachers is preserved even if the student's architecture and parameters are public or reverse-engineered by an adversary.

Learning the student in a semi-supervised fashion using generative adversarial networks [9] allows us to leverage the entire student training set while only labeling a subset of it. The generative adversarial network framework involves two machine learning models, a *generator* and a *discriminator*. They are trained in a competing fashion. The generator produces samples from the data distribution by transforming vectors sampled from a Gaussian distribution. The discriminator is trained to distinguish samples artificially produced by the generator from samples part of the real data distribution. The two models are trained via simultaneous gradient descent steps on both player's costs with the goal of converging to the Nash equilibrium. In our experiments, we follow the adaptation of generative adversarial networks to semi-supervised learning presented in [12]. The discriminator is extended from a binary classifier (data sample vs. generator sample) to a multi-class classifier (one of k classes of data samples, plus an additional class for generated samples). This classifier—discriminator—is then trained to classify real samples in one of the real classes, and the generated samples in the corresponding additional class. Although no formal results currently explain why yet, the technique was empirically demonstrated to greatly improve semi-supervised learning of classifiers on several datasets, especially when the classifier is trained with an additional *feature matching* loss [12].

2.3 Privacy Analysis of the Approach

We observe that the actual privacy cost of the noisy aggregation mechanism is very small when the margin between the best and the second-best label is large. So our analysis is data dependent, and to keep track of the privacy cost, we use the moments account introduced in [1]. Due to space constraints, we do not include our full privacy analysis, but we will do so in the poster and full paper.

Dataset	ε	δ	Queries	Baseline	Accuracy
MNIST	2.04	10^{-5}	100	99.18%	98.00%
MNIST	8.03	10^{-5}	1000	99.18%	98.10%
SVHN	5.04	10^{-6}	500	92.80%	82.72%
SVHN	8.19	10^{-6}	1000	92.80%	90.66%

Figure 2: Student test accuracy and (ε, δ) differential privacy bounds for varying number of queries.

3 Experimental Validation

Our evaluation is two-fold. We (1) train an ensemble of teachers that produces accurate and private aggregated labels, and (2) learn a student using as few queries to the teacher ensemble as possible to minimize the privacy budget. In our experiments, we use MNIST and the extended SVHN.

Our non-private MNIST model stacks two convolutional layers with max-pooling and one ReLU layer. It has a 99.18% test accuracy. For SVHN, we add two hidden layers [2] to achieve a 92.8% test accuracy, which is shy of the state-of-the-art. However, we are primarily interested in comparing the private student's accuracy with the one of a non-private model trained on the entire training dataset.

3.1 Training of the Ensemble of Teachers

For the MNIST and SVHN datasets we are able to train ensembles of $n = 250$ teachers making sound predictions despite the injection of large amounts of random noise to ensure privacy. This is due to the large gap between the number of votes received by the first and second most frequent labels. The gap increases with n. It must be sufficiently large to ensure the aggregation mechanism is robust to the Laplacian noise added to perturb the label vote counts. For instance, if the Laplacian noise introduced has scale 20, the aggregation mechanism has an accuracy of 93.18% for MNIST and 87.79% for SVHN, while each query has a low privacy budget of $\varepsilon = 0.05$.

The large Laplacian scale is explained by the sensitivity of each machine learning model, which is considered to be 1 in the privacy analysis: in other words it is assumed that changing one point in the training set of a single teacher can suffice to change the predictions made by the teacher, and thus two of the vote counts by 1. As a consequence, a large amount of data is potentially required to be able to train large number of teachers, depending on the classification task complexity.

3.2 Semi-Supervised Student Training

We apply semi-supervised learning with generative adversarial networks as follows. For MNIST, we use $9,000$ samples from the traditional test set as our source of public training data for the student network. We use the remaining $1,000$ samples to evaluate the student's generalization error. Of the $9,000$ training samples, most are used as unlabeled examples for semi-supervised learning, and a class-balanced random subset of either 100 or $1,000$ are labeled using the aggregation mechanism. For SVHN, the student has access to $10,000$ training inputs, among which it labels 500 or $1,000$ samples using the teacher ensemble. Its performance is evaluated on the remaining $16,032$ samples.

For both datasets, we use 250 teachers and a Laplacian scale of 20. This parameter choice is motivated by results from Section 3.1 and corresponds to a privacy bound of $\varepsilon = 0.05$ per query. In Figure 2, we report values of the (ε, δ) differential privacy guarantees and student accuracy for varying number of queries made to the aggregation mechanism. The MNIST student is able to learn a 98% accurate model, which is shy of 1% when compared to the accuracy of a model learned with the entire training set, with only 100 label queries. This results in a strict differentially private bound of $\varepsilon = 2.04$ with a failure probability of 10^{-5}. The SVHN student has an accuracy of 90.66%, compared to 92.80% for a model learned with the entire dataset. The privacy bound $\varepsilon = 8.19$ with $\delta = 10^{-6}$ is looser. These bounds are themselves sensitive and should not be released publicly. When working with real private data, one should publish values of ε perturbed with noise proportional to its smooth sensitivity.

[2] The model is adapted from: `https://www.tensorflow.org/tutorials/deep_cnn`

4 Conclusions

This abstract demonstrated how semi-supervised learning, together with a strict privacy analysis based on the moments accountant technique, can increase the accuracy of private models while lowering the differentially private bounds that are guaranteed. Namely, we showed in our experiments on the MNIST and SVHN datasets that the utility lost to provide privacy guarantees is limited: our $(2.04, 10^{-5})$ and $(8.19, 10^{-6})$ differentially private MNIST and SVHN students are only respectively 1% and 3% less accurate than our model trained on the entire corresponding dataset.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *CCS*, 2016. arXiv preprint arXiv:1607.00133.

[2] Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, 2015.

[3] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar):1069–1109, 2011.

[4] Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[5] Cynthia Dwork. A firm foundation for private data analysis. *Communications of the ACM*, 54(1):86–95, 2011.

[6] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography*, pages 265–284. Springer, 2006.

[7] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[8] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333. ACM, 2015.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[10] Jihun Hamm, Paul Cao, and Mikhail Belkin. Learning privately from multiparty data. *arXiv preprint arXiv:1602.03552*, 2016.

[11] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*, pages 94–103. IEEE, 2007.

[12] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. *arXiv preprint arXiv:1606.03498*, 2016.

[13] Reza Shokri and Vitaly Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1310–1321. ACM, 2015.