Exploring Differential Privacy with Large-Scale Imbalanced Data

Naimu Deng, Wenjia Zhang

Course: Advanced Deep Learning
Carnegie Mellon University
School of Computer Science
March 2022

1 Introduction

Differential Privacy (DP) becomes state of the art privacy-preserving approach in machine learning, which replaces traditional methods such as K-anonymity. Research [4] [10] has found that differential privacy is adequate to prevent membership inference attacks and reconstruction attacks in deep learning tasks such as image recognition. However, training the model with DP-SGD will usually suffer in model accuracy, especially for complicated models and large datasets such as ImageNet. Existing research [11] shows that switching to a differentially private approach to train ResNet on ImageNet would result in nearly zero model accuracy. On the other hand, many other papers merely apply differentially private algorithms on small datasets such as MNIST and CIFAR-10, which does not generalize to large-scale computing in real-life tasks. In this project, we are based on the preliminary research [8] from researchers at Google and Columbia University to further explore the differentially private deep learning at the scale of training ImageNet size data. Due to the project's budget limit, we cannot implement the whole ImageNet dataset for training the model. Still, we will sample a marginally large dataset from the original ImageNet and explore the possible strategies to optimize the model with the marginally large dataset. It will provide meaningful guidance for future work with the complement dataset. Moreover, several pieces of research [12] [3] noticed that the results from the model trained with a balanced dataset might not be fully justified in real-life tasks, which more often deal with imbalanced data. Different loss functions or measuring scales are needed to evaluate the performance of a learning classification model; an imbalanced dataset and differentially private algorithm make the task even more complicated. In this project, our second goal is to evaluate how differentially privacy performs on the model trained with imbalanced datasets.

2 Literature Review

2.1 Reconstruction and Membership Inference Attack

A reconstruction attack is a kind of attack whose objective is to reconstruct a probabilistic version of the original dataset [7]. With the help of some auxiliary information and by analyzing the difference in the outputs from sequential queries, the attacker can learn the data labels. Research [9] shows that a linear number of queries ($N \log(N) + O(N)$) can fully reconstruct a dense dataset, which makes the method very promising for attacking large-scale databases. Machine learning models are essentially statistical aggregations of the training data [8], thus are also susceptible to reconstruction attacks. Another kind of attack acting on machine learning models is called membership inference attack. A membership inference attack is not primarily targeted at reconstructing the whole dataset but trying to figure out which record belongs to the original training set given the machine learning model black box. This kind of attack is specifically suitable for attacking generative deep neural networks such as GAN, from which the attacker can retrieve classified real training data from synthetic data generated by the model.

2.2 Differential Privacy

The success of reconstruction attacks to machine learning models largely relies on the model's sensitivity to changes in the data input. More specifically, how different will the output be with or without individual entries in the training dataset. The less sensitive the model is to the neighbor datasets, the less likely the model will give away information of an individual record. The most prevalent approach to limit the model's sensitivity and thus prevent the leakage of individual entries through the output is adding random noise to each data entry. Reconstruction attack and membership inference attack work by finding data points that make the observed model more likely, and those data points which make the model more probable are the datasets in the training set [8]. Therefore, making the model less sensitive to individual entry ensures that no data point can result in an enormous increase in the model likelihood is an effective approach to fight against those attacks.

More generally, any randomized algorithm is deemed differentially private if it can meet the following requirements. We hereby provide the formal definition for reference:

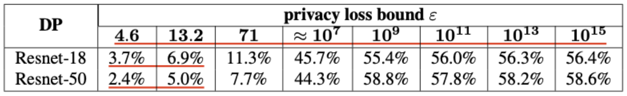
Definition 2.1 [8] [5] [6] *A randomized algorithm A is (ϵ, δ)- deferentially private if, for any pair of datasets D and D' differing in exactly one data point (called neighboring datasets) i.e., one data point is added or removed, and for all events S in the output range of A, we have*

$$\Pr[A(D) \in S] \leq e^{\epsilon} \cdot \Pr[A(D') \in S] + \delta$$

Notably, we are using the relaxed version of differential privacy, which adds an additive approximation factor δ to the multiplicative factor ϵ.

2.3 Training Large Scale Deep Learning Model with Differential Privacy

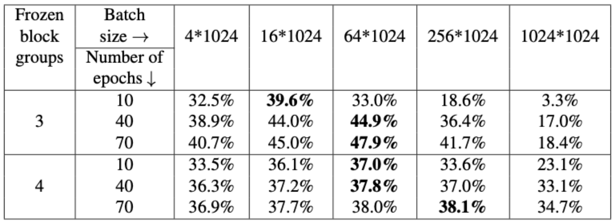
Although differential privacy seems to be a promising technique to preserve data privacy in training deep learning models, it achieves this by sacrificing model performance. Most of the recent research struggles to balance reasonable differentially private scale and high model performance. This situation is even worse when training on large datasets such as ImageNet. Research [2] [1] shows that when training ResNet-50 and ResNet-18 on ImageNet with standard naive differential private settings, the accuracy is nearly zero with a reasonable privacy scale ϵ. In the paper we refer to [8], we also find that: under standard DP algorithm, with a reasonable value of ϵ, the accuracy is typically below 10 percent, acceptable model accuracy can only be reached when the differential privacy scale is huge which makes the model non-private.



DP	**privacy loss bound** ε							
	4.6	**13.2**	**71**	$\approx \mathbf{10^7}$	$\mathbf{10^9}$	$\mathbf{10^{11}}$	$\mathbf{10^{13}}$	$\mathbf{10^{15}}$
Resnet-18	3.7%	6.9%	11.3%	45.7%	55.4%	56.0%	56.3%	56.4%
Resnet-50	2.4%	5.0%	7.7%	44.3%	58.8%	57.8%	58.2%	58.6%

Figure 1: Training ResNet with Naive DP Setting

Therefore, a more promising way introduced by the paper is to pre-train the model with public data before fitting the training set and freeze some layers according to the pre-training results. The transferred learning from public data will dramatically increase the model accuracy. One key point to keep in mind is that the embedding of the public dataset should be as much unrelated as possible with the training set to keep the training set private. The authors used the open-source dataset Places365 as public training data in training. Places365 dataset consists of purely landscape images, which are sufficiently dissimilar to ImageNet. From their results, we can see that higher accuracy is achieved with enough training epochs. However, the highest accuracy captured (47.9%) is still much lower than the state of art accuracy of 75% trained without DP set. The authors call for further investigation and optimization of the deferentially private training procedures and parameter refinement. Our work in this project is trying to explore the factors that hinder the model accuracy and hopefully find a more optimum solution.



Frozen block groups	Batch size → Number of epochs ↓	4*1024	16*1024	64*1024	256*1024	1024*1024
3	10	32.5%	**39.6%**	33.0%	18.6%	3.3%
	40	38.9%	44.0%	**44.9%**	36.4%	17.0%
	70	40.7%	45.0%	**47.9%**	41.7%	18.4%
4	10	33.5%	36.1%	**37.0%**	33.6%	23.1%
	40	36.3%	37.2%	**37.8%**	37.0%	33.1%
	70	36.9%	37.7%	38.0%	**38.1%**	34.7%

Figure 2: Training ResNet with pre-trained DP Setting of $\epsilon = 10$

3 Methods/Model

We have implemented ResNet-18 on CIFAR10 and MNIST with DP settings from two mainstream frameworks, TensorFlow-privacy and PyTorch Opacus. We also experimented with the automatically fast DP training algorithm JAX provided in the paper [8]. The results achieved on small data sets are set as benchmarks for further exploration when training deep learning models with DP at a large scale and on imbalanced datasets. We might not build models with all these three frameworks for further experiments, but now we tend to conduct broad research to obtain a general overview of model performance. Detailed baseline figures are demonstrated in the next section.

4 Preliminary Results

Below are the preliminary results we achieved from ResNet-18 models trained on MNIST and CIFAR-10 datasets. DP algorithms are implemented with a privacy loss bound ϵ=10, $\delta = 10^{-6}$, and random noise multiplier σ is set to be 1.0 by default. The preliminary results show that the model with a DP setting suffers from higher loss, which can be explained by the noises added during training. In addition, the Objax algorithm outperforms Opacus and TensorFlow-privacy by achieving a higher accuracy rate and ending up closer to the performance without DP settings. As a trade-off, Objax requires a much longer run time for training. We also notice that in most cases, models without DP settings run faster than those with DP settings due to the additional randomized computation. However, it is interesting that the Opacus model's run time per epoch is lower than the naive ResNet-18 without DP settings. This might be caused by a specific PyTorch Opacus PrivacyEngine parameter combination. We discovered during the experiment phase that DP models' performance is highly sensitive to parameter settings, and we will delve deeper into the relationship between model models' runtime and performance during the next period.

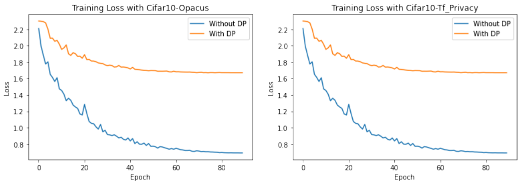


Figure 3: Training Loss on CIFAR-10

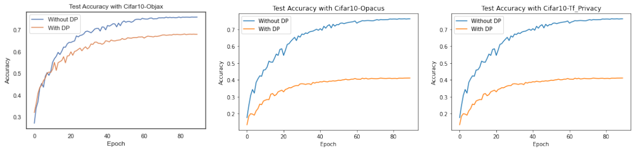


Figure 4: Test Accuracy on CIFAR-10

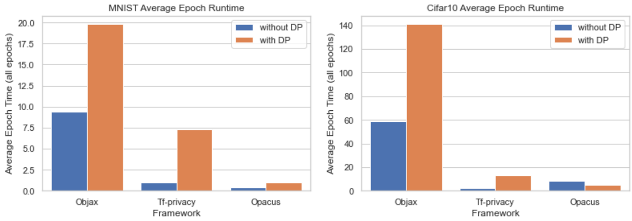


Figure 5: Runtime on MNIST and CIFAR-10

5 Evaluation of Preliminary Work

From Figure 4, we can see that, using the Objax framework, we can achieve 67.4% accuracy for training our model on the CIFAR-10 dataset, which is just 10 percent below the baseline accuracy reached without DP (75.6%). However, the difference between the accuracy with/without DP algorithm is much more significant when using Opacus and TensorFlow frameworks. Contrary to the result from [8], we found that the runtime per epoch for Objax is higher than that for Opacus and TensorFlow. We will re-examine the mechanism and implementation and analyze the cause of the different results.

On the other hand, it is not surprising that we achieve better performance on the CIFAR-10 dataset than the baseline in the paper using the ImageNet dataset, as CIFAR-10 has a much lower resolution (32X32) than ImageNet (256X256), which makes the classification model more complicated. Therefore, considering the project budget, we will treat 67.4% as our baseline performance on the CIFAR-10 dataset with a reasonable privacy parameter constraint ($\epsilon = 10$, $\delta = 10^{-6}$ in this case) and 47.9% from [8] as a reference, if we have enough computation and time budget for running the model on the entire ImageNet dataset.

6 Future Work

In the previous session, we described that, in our preliminary work, the accuracy of training the model with Objax is much higher than other frameworks such as TensorFlow and Opacus. The improvement in performance by using Objax is not reported in the original paper [8], and we will further investigate the validity of this phenomenon and explore the underlying mechanism or causes. Also, we found that the runtime for Objax is significantly slower than the other two frameworks, which is contradictory to the original finding in [8], we will check whether this is due to the different sizes of the datasets used (CIFAR-10, MNIST as opposed to ImageNet) or an erroneous implementation of the Objax framework.

Notably, the original research topic is to examine the performance of implementing differentially private algorithms at scale, scalability is an important factor to consider in our project. To avoid losing the generalizability of our results to large-scale computing tasks and accomplish our project within the budget, we plan to obtain a marginally large dataset by randomly sampling data from the original ImageNet. Moreover, in real life, we have to deal with imbalanced samples much more frequently than balanced ones, and the concept of differential privacy was initially brought up to protect user email records from leakage by spam detection models, which also takes imbalanced data as input. Therefore, it would be meaningful and exciting if we analyze the DP model performance on imbalanced datasets. We plan to obtain the imbalanced dataset by extracting data from ImageNet with pre-designed weights as in [12]. In general, our work aims to mimic the real-life scenario of large-scale computing with imbalanced data and examine the performance of DP under such situations.

7 Teammates and Work Division

Work on this project is evenly distributed between the two members. Naimu will lead and summarize the observations in the next phase, and Wenjia will conduct experiments with different data and parameter settings. Experiment observations and periodical progress will be analyzed and discussed via regular meetings.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[2] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 464–473. IEEE, 2014.

[3] Mohamed Bekkar, Hassiba Kheliouane Djemaa, and Taklit Akrouf Alitouche. Evaluation measures for models assessment over imbalanced data sets. *J Inf Eng Appl*, 3(10), 2013.

[4] Mahawaga Arachchige Pathum Chamikara, Peter Bertók, Ibrahim Khalil, Dongxi Liu, and Seyit Camtepe. Privacy preserving face recognition utilizing differential privacy. *Computers & Security*, 97:101951, 2020.

[5] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 486–503. Springer, 2006.

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

[7] Sébastien Gambs, Ahmed Gmati, and Michel Hurfin. Reconstruction attack through classifier analysis. In *IFIP Annual Conference on Data and Applications Security and Privacy*, pages 274–281. Springer, 2012.

[8] Alexey Kurakin, Steve Chien, Shuang Song, Roxana Geambasu, Andreas Terzis, and Abhradeep Thakurta. Toward training at imagenet scale with differential privacy. *arXiv preprint arXiv:2201.12328*, 2022.

[9] Marie-Sarah Lacharité, Brice Minaud, and Kenneth G Paterson. Improved reconstruction attacks on encrypted data using range query leakage. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 297–314. IEEE, 2018.

[10] Md Atiqur Rahman, Tanzila Rahman, Robert Laganière, Noman Mohammed, and Yang Wang. Membership inference attack against differentially private deep learning model. *Trans. Data Priv.*, 11(1):61–79, 2018.

[11] Shuang Song, Kamalika Chaudhuri, and Anand D Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 245–248. IEEE, 2013.

[12] Shoujin Wang, Wei Liu, Jia Wu, Longbing Cao, Qinxue Meng, and Paul J Kennedy. Training deep neural networks on imbalanced data sets. In *2016 international joint conference on neural networks (IJCNN)*, pages 4368–4374. IEEE, 2016.