DP-Mix: Mixup-based Data Augmentation for Differentially Private Learning

Wenxuan Bao[1], Francesco Pittaluga[2], Vijay Kumar B G[2], Vincent Bindschaedler[1]

[1]University of Florida, [2]NEC Labs America

Paper website

Wenxuan Bao’s website

Motivation

- Mixup boosts model performance but is challenging to use for Differential Private Machine Learning (DPML) due to sensitivity issues.
- Diffusion processes generate high-quality images, but it is not clear that how to use them in DPML to enhance performance.
- We propose techniques: DP-MIX$_{self}$ and DP-MIX$_{diff}$ to apply Mixup and Diffusion in DPML and show it surpasses the prior SoTA **at no *extra* privacy cost.**

Background: Mixup Data Augmentation

$$\begin{aligned}\hat{x} &= \lambda x_i + (1-\lambda)x_j,\\ \hat{y} &= \lambda y_i + (1-\lambda)y_j,\end{aligned}$$

where $\lambda \in [0,1]$ is a random number

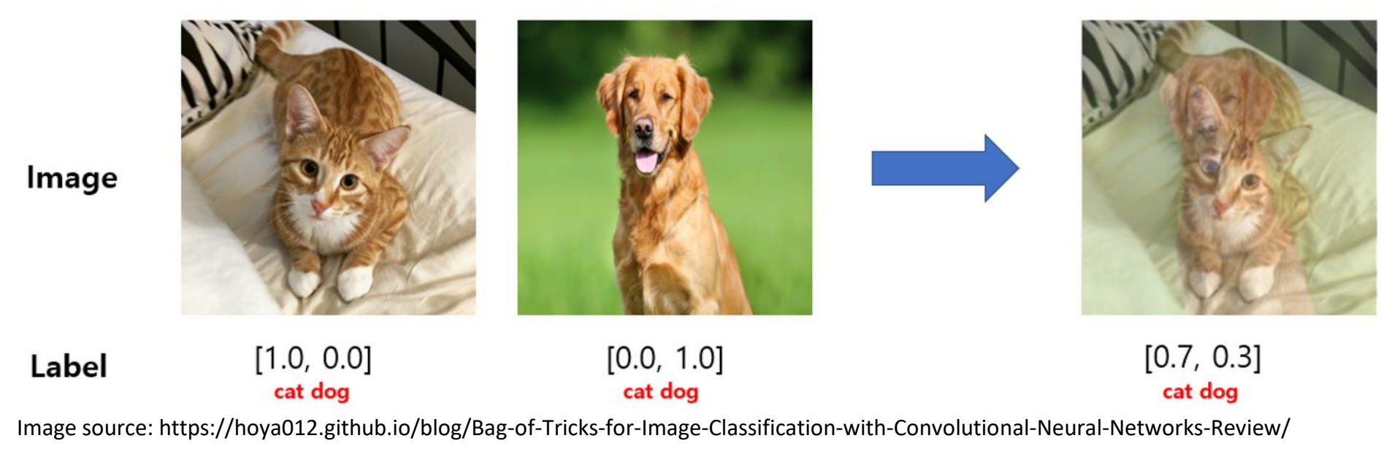
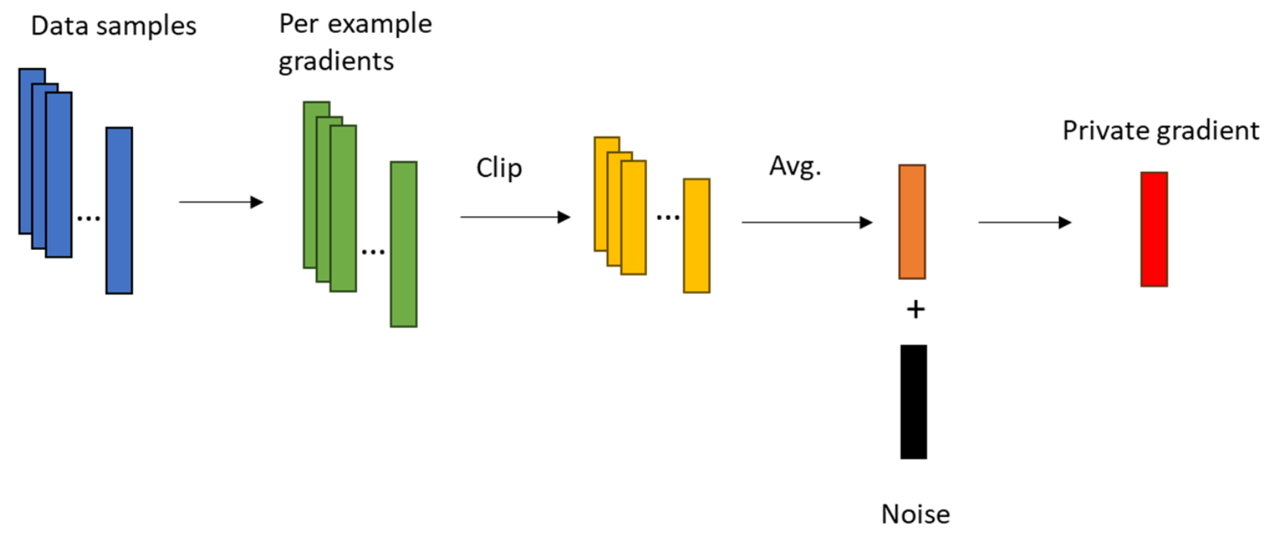


Image source: https://hoya012.github.io/blog/Bag-of-Tricks-for-Image-Classification-with-Convolutional-Neural-Networks-Review/

Background: Differential Privacy and DP-SGD[1]

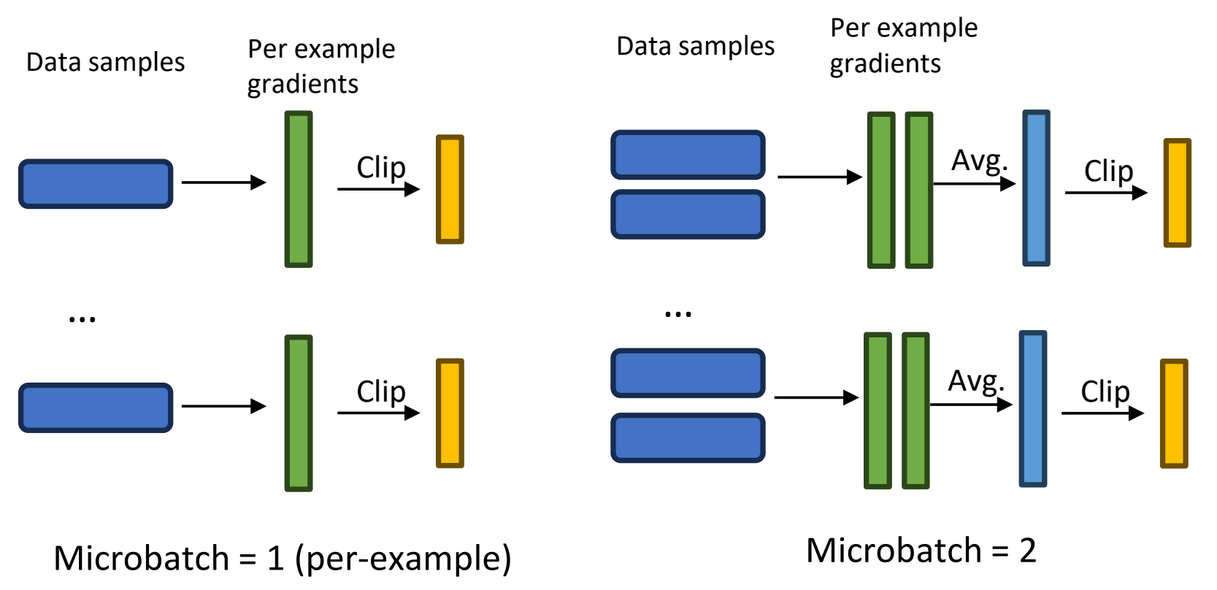
DP-SGD steps:

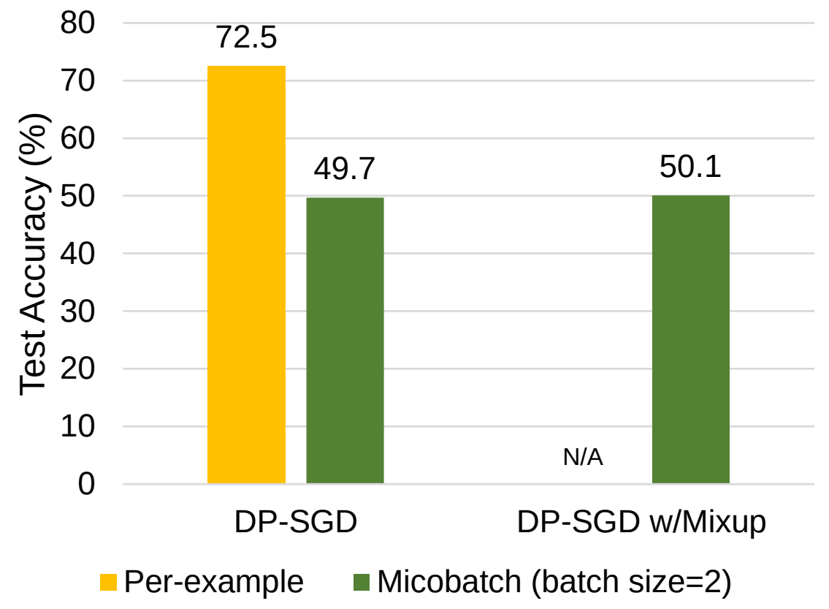
1. Compute per example gradients.
2. Clip them to norm ***C.***
3. Average clipped gradients
4. Add noise to average gradient.



Is Microbatching a Solution?

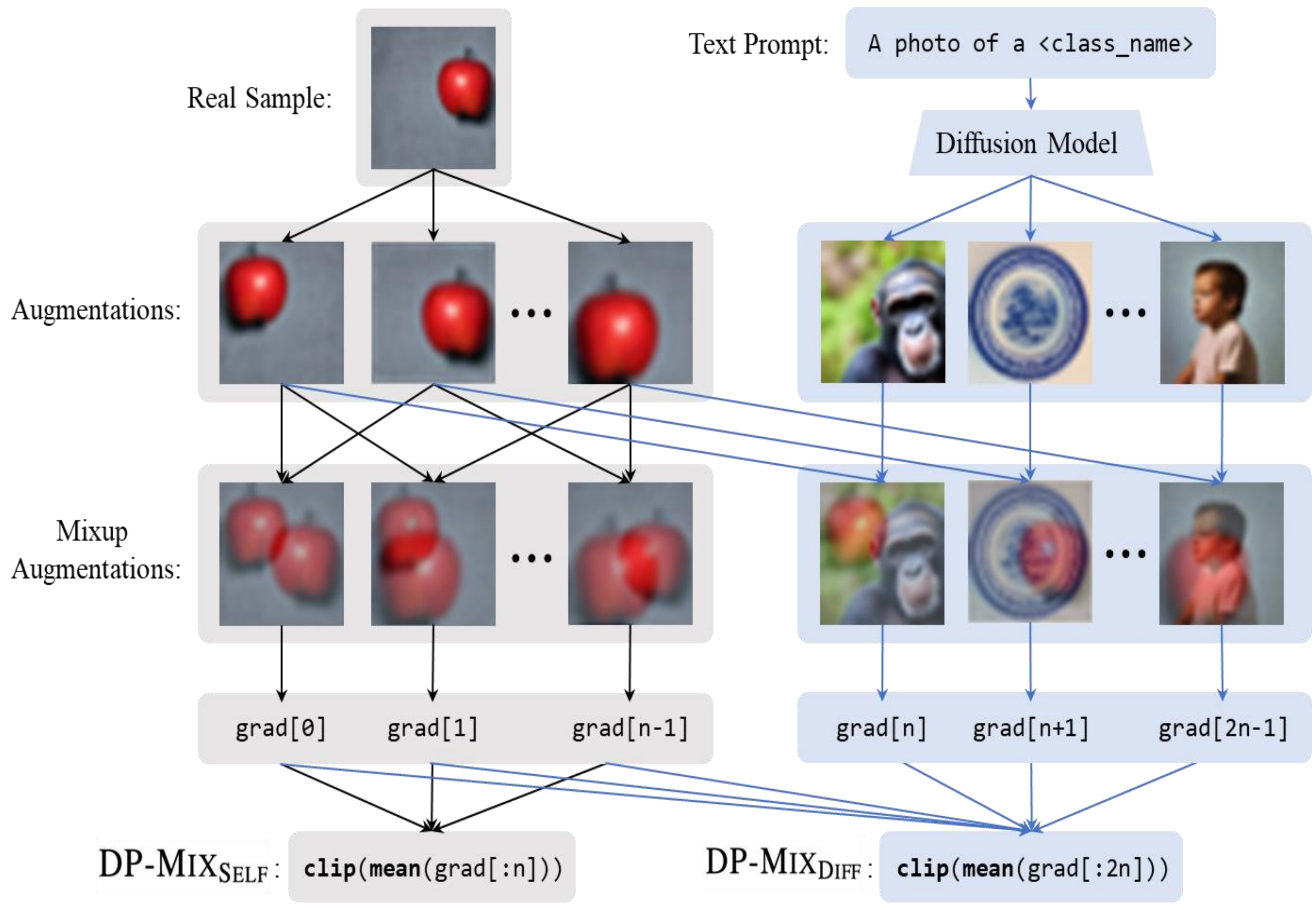
- Instead of per-example gradient clipping, we could use microbatch DP-SGD[2]. Most straightforward method to apply Mixup in DPML.
  - We clip the average of microbatch’s gradients.
- Drawback of microbatch DP-SGD is **increased sensitivity** (as pointed out by Ponomareva et al. [3]). This requires more noise for the same privacy budget.
- Experiments show that it fails to improve performance even with moderately large privacy budget (i.e., Ɛ=8).





Proposed Methods

- **DP-MIX$_{self}$** : Apply mixup to augmentations of real samples and then clip the average of those samples’ gradients.
- **DP-MIX$_{diff}$** : Pretrain a private model on a public dataset using a Diffusion model. Generate diffusion samples with text prompts like “A photo of a <class name>”. Mix these samples with augmented real samples and clip the average of their gradients



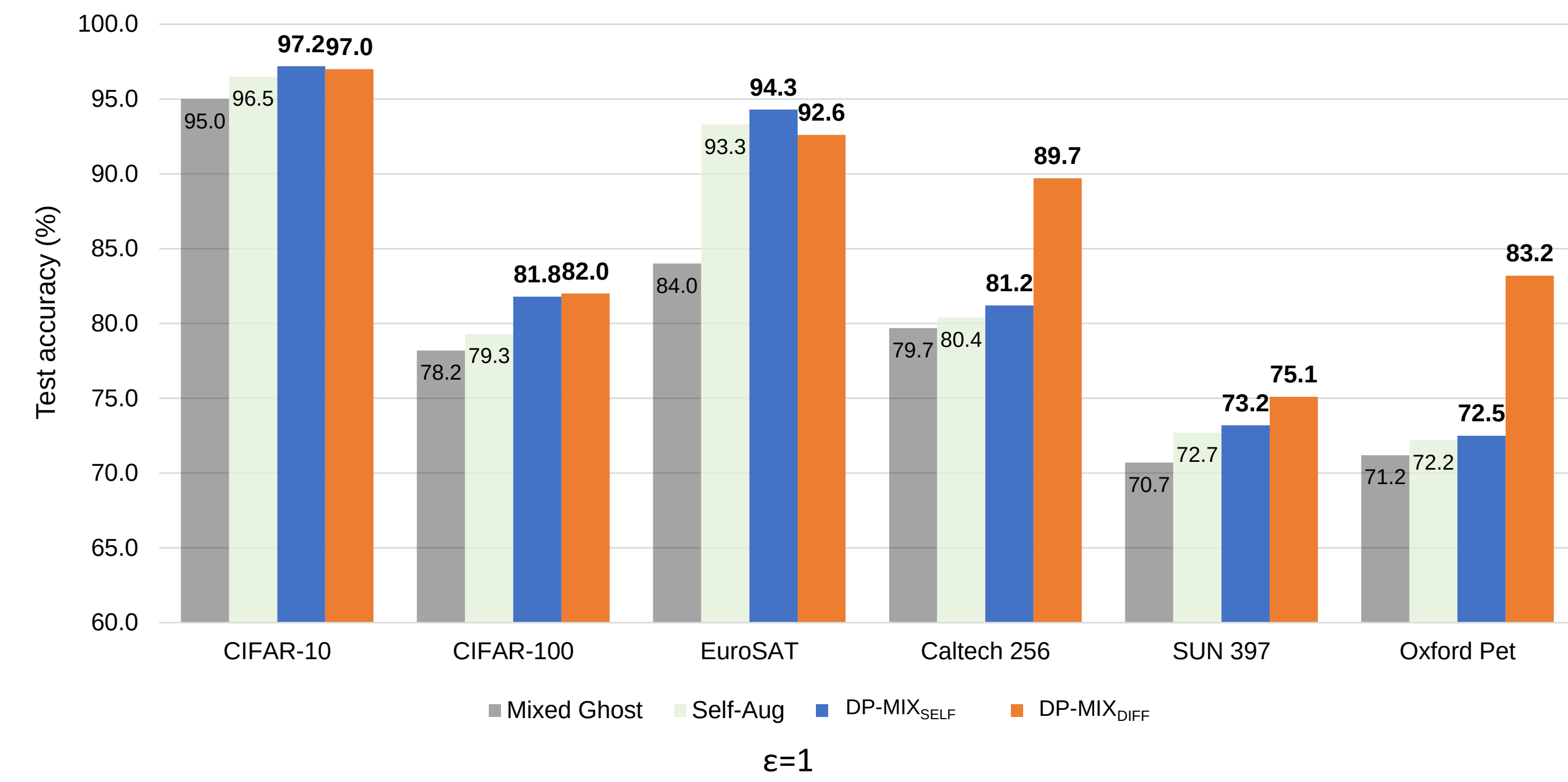
Our Contributions

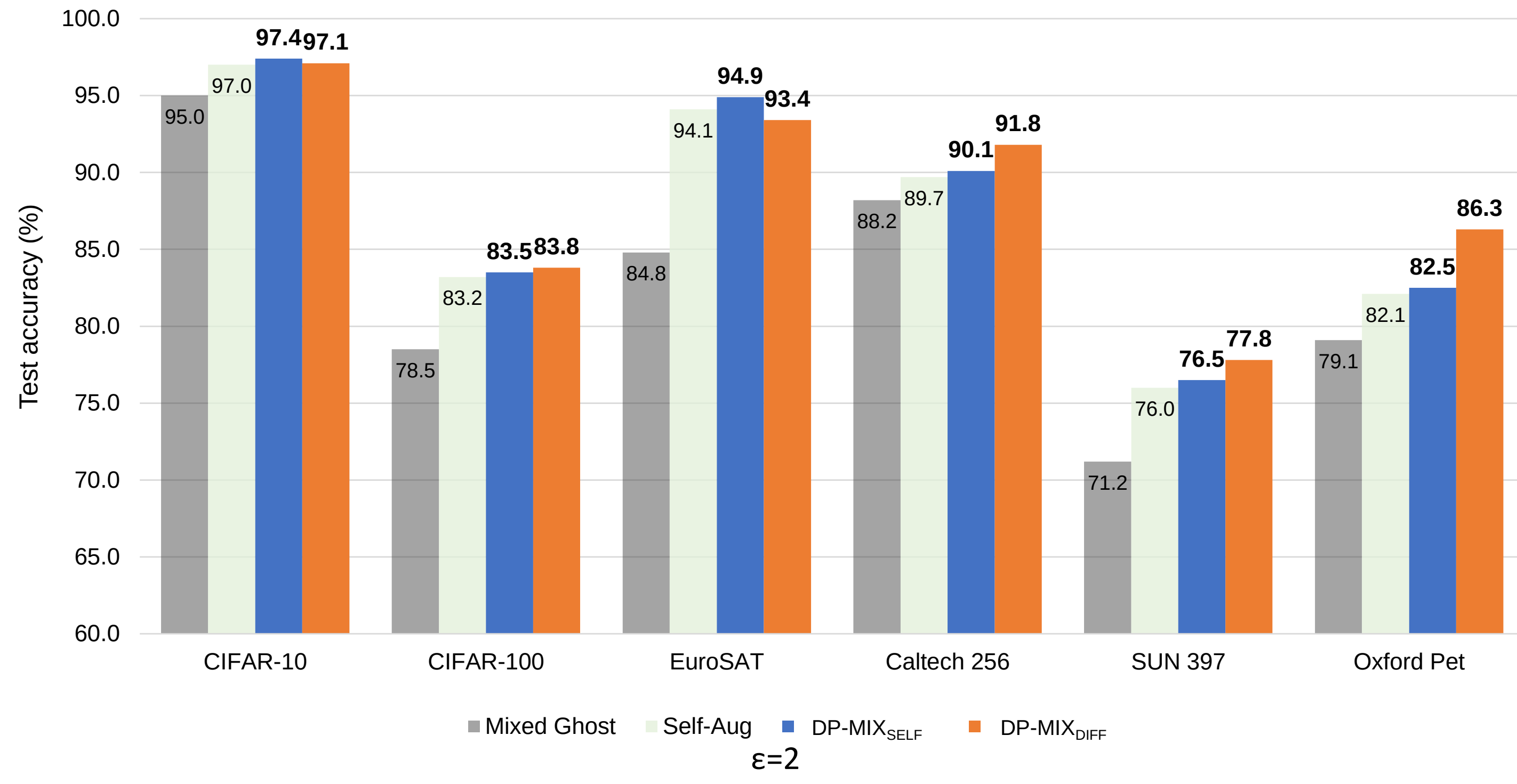
1. We show empirically the straightforward application of Mixup i.e., using Microbatch DP-SGD, fails to improve performance.
2. We propose a technique called **DP-MIXself** to apply Mixup in DP-SGD by using Mixup to self-augmentations of one training sample. This method achieves **SoTA** performance for training from scratch and finetuning pre-trained models.
3. We also propose a second technique called **DP-MIXdiff** to further enhance performance by using a text-to-image diffusion model to generate class-specific synthetic examples. We mixup those diffusion samples with real training sample to achieve new **SoTA** performance **with *no additional* privacy cost.**

Results

Main results

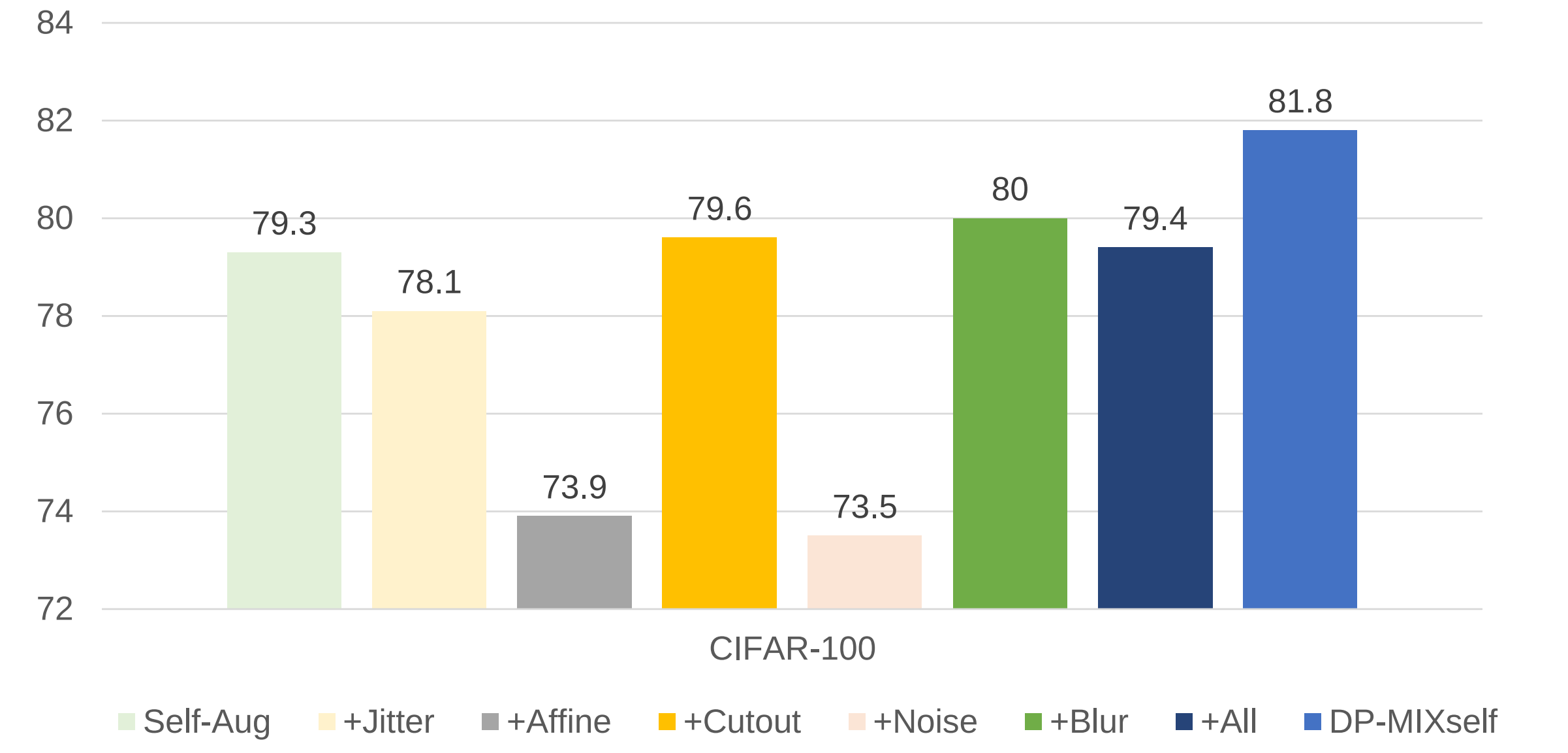
- Our proposed techniques **DP-MIX$_{self}$** and **DP-MIX$_{diff}$** achieve better results than prior SoTA. For Caltech 256 and Oxford Pet, we achieve about *9%* test accuracy boost for ε=1.
- For larger privacy budgets, performance boosts decrease but still exists.





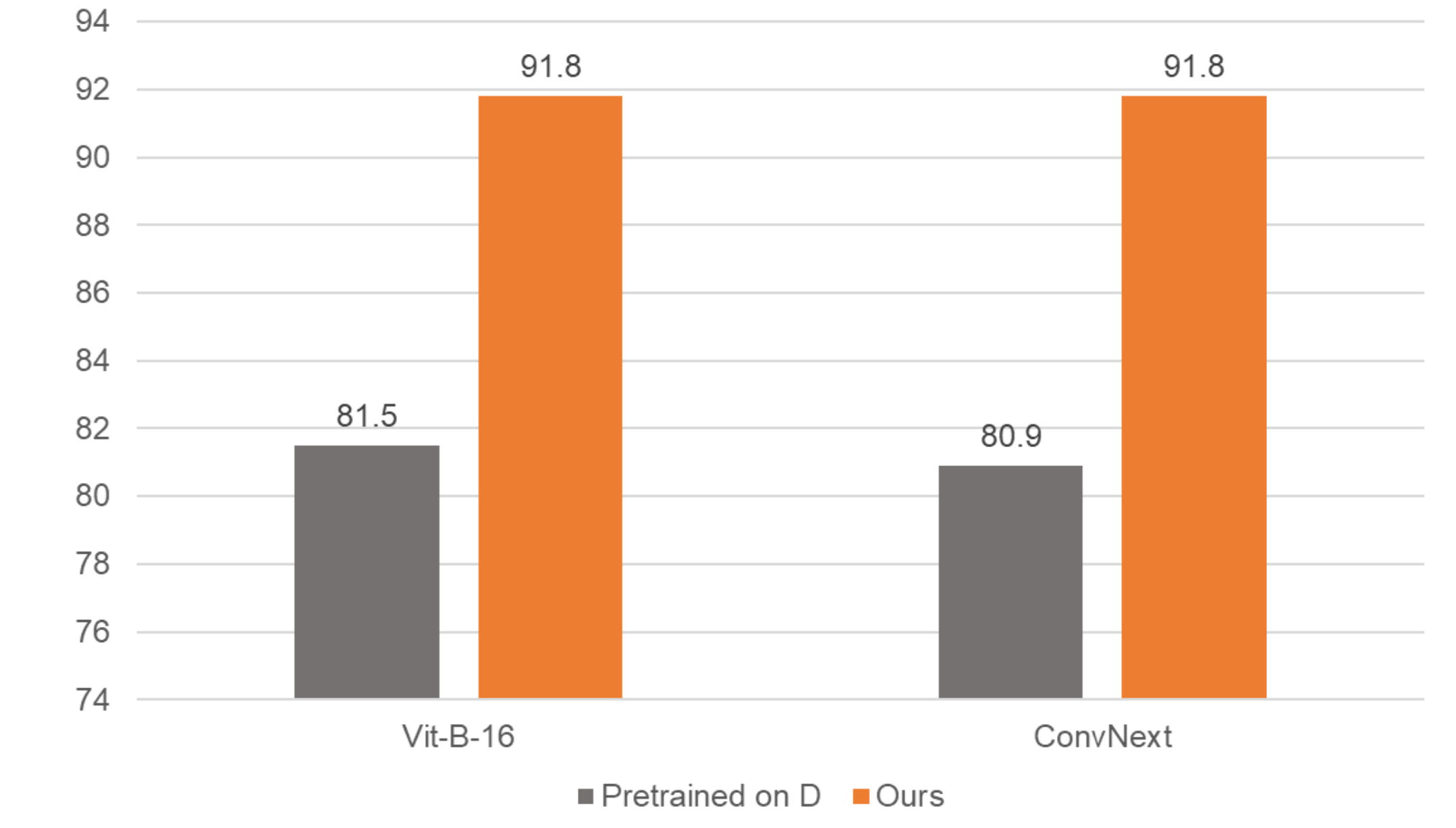
Ablation study | Other augmentations

- Other single-sample augmentations do **not** provide anywhere near as much of an improvement as **mixup**



Ablation study | DP-Mix$_{diff}$ vs Pretraining with Diffusion Data

- Pre-training on sample diffusion models does **not** improve performance, but mixing up training samples with them does.



Caltech-256

Take aways and Future works

- We show how to apply **mixup** for DP training of ML models and demonstrate it surpasses the prior SoTA ***at no extra privacy cost***.
- Other multi-sample data augmentations methods could be applied to DPML. We use public data to pretrain the diffusion model and the private model, but how to use public data in the most efficient way for DPML remains an open research question.

Acknowledgments

This work was supported in part by the National Science Foundation under CNS-2055123. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Abadi, Martin, et al. "Deep learning with differential privacy." Proceedings of the 2016 ACM SIGSAC conference on computer and communications security. 2016.

[2] McMahan, H. Brendan, et al. "A general approach to adding differential privacy to iterative training procedures." arXiv preprint arXiv:1812.06210 (2018).

[3] Ponomareva, Natalia, et al. "How to dp-fy ml: A practical guide to machine learning with differential privacy." Journal of Artificial Intelligence Research 77 (2023): 1113-1201.