Differentially Private Learning with Grouped Gradient Clipping

Haolin Liu
Institute of Information Engineering, CAS
School of Cyber Security, UCAS
Beijing, China
liuhaolin@iie.ac.cn

Chenyu Li
Institute of Information Engineering, CAS
School of Cyber Security, UCAS
Beijing, China
lichenyu@iie.ac.cn

Bochao Liu
Institute of Information Engineering, CAS
School of Cyber Security, UCAS
Beijing, China
liubochao@iie.ac.cn

Pengju Wang
Institute of Information Engineering, CAS
Beijing, China
wangpengju@iie.ac.cn

Shiming Ge*
Institute of Information Engineering, CAS
Beijing, China
geshiming@iie.ac.cn

Weiping Wang
Institute of Information Engineering, CAS
Beijing, China
wangweiping@iie.ac.cn

ABSTRACT

While deep learning has proved success in many critical tasks by training models from large-scale data, some private information within can be recovered from the released models, leading to the leakage of privacy. To address this problem, this paper presents a differentially private deep learning paradigm to train private models. In the approach, we propose and incorporate a simple operation termed grouped gradient clipping to modulate the gradient weights. We also incorporated the smooth sensitivity mechanism into differentially private deep learning paradigm, which bounds the adding Gaussian noise. In this way, the resulting model can simultaneously provide with strong privacy protection and avoid accuracy degradation, providing a good trade-off between privacy and performance. The theoretic advantages of grouped gradient clipping are well analyzed. Extensive evaluations on popular benchmarks and comparisons with 11 state-of-the-arts clearly demonstrate the effectiveness and genearalizability of our approach.

CCS CONCEPTS

• **Security and privacy** → *Domain-specific security and privacy architectures*; • **Computing methodologies** → *Neural networks.*

KEYWORDS

Deep learning, differential privacy, grouped gradient clipping.

ACM Reference Format:
Haolin Liu, Chenyu Li, Bochao Liu, Pengju Wang, Shiming Ge, and Weiping Wang. 2021. Differentially Private Learning with Grouped Gradient Clipping. In *ACM Multimedia Asia (MMAsia '21), December 1–3, 2021, Gold Coast, Australia.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3469877.3490594

1 INTRODUCTION

Deep learning has delivered impressive performance on many multimedia processing tasks like image classification [37], object recognition [41] and medical image analysis [43]. One of the keys of these advancements is the access of large-scale annotated data, that may contain sensitive or private information. Recent studies [13, 31, 32] have shown that the adversaries are able to extract model parameters and infer sensitive training data from deep learning services, even when they only have the access of the query APIs. It is necessary to explore effective solutions to learn high performance models that preserve the privacy at the same time.

Differential privacy mechanism [8, 9] provides a feasible way to train privacy-preserving models. It constitutes a strong standard for privacy guarantees for algorithms on aggregate databases. This mechanism can be achieved by adding noise at feature, loss or gradient levels. In [26, 38], the authors added noise at feature layers. In general, these approaches are very complex. Some approaches [3, 33] added the calibrated noise to the outputs of the loss functions, so that the adversary cannot distinguish whether the user's data within or without in training data. Recently, differentially private deep learning [1, 18] becomes popular in the field. This method trains private neural networks by differentially private stochastic gradient decent (DPSGD) algorithm [1]. DPSGD employs a per-sample gradient-decent optimization, where they perform gradient clipping operation to each sample's gradients with a predefined norm bound, aggregate a batch of clipped gradients and finally add Gaussian noise into the aggregated gradients. It can provide a tighter bound compared with advanced composition theorem [11], leading to higher model accuracy with the same privacy cost. The gradient clipping operation plays an important role in these approaches. In [7, 27, 29, 42], the "holistic" gradient clipping operation is analyzed theoretically, showing that the clipped gradient distribution is not symmetric to true gradient distribution due to clipping bias. Moreover, the noise addition operation has disparate impact [2]. The two issues will worsen the model accuracy.

In this work, we propose differentially private learning with grouped gradient clipping (DPL-GGC) to alleviate clipping bias and disparate impact. Towards this end, the gradients are divided into several groups and each group is clipped separately. In this way, gradient information loss is effectively reduced and the clipped

*Shiming Ge is the corresponding author (geshiming@iie.ac.cn).

MMAsia '21, December 1–3, 2021, Gold Coast, Australia

ACM ISBN 978-1-4503-8607-4/21/12…$15.00
https://doi.org/10.1145/3469877.3490594

gradients are closer to true gradient distribution. We further propose smooth noise calibration method to bound the noise scale by incorporating the smooth sensitivity [22] into Gaussian mechanism [10]. It can reduce the disparate impact and improve the model performance, while providing tight privacy guarantee. DPL-GGC is a general and efficient approach where the grouped gradient clipping operator functions as a plug-and-play module that can be easily integrated into other DPSGD-based learning frameworks to facilitate the utility and privacy of learned models.

The contributions of this work are three folds. First, we propose a differentially private model learning approach via grouped gradient clipping that can reduce the clipping bias evidently and further improve model utility. Second, we propose the smooth noise calibration method to reduce the injected Gaussian noise especially under strict privacy guarantees. Third, we conduct extensive experiments to show that our approach achieves the state-of-the-art performance in learning private models.

2 PRELIMINARIES AND RELATED WORK

2.1 Differential Privacy

Differential privacy and its relaxation [5, 20] have been the de facto standard in the field of privacy protection. Differential privacy prevents adversaries from distinguishing two different distributions by adding random noise. We call two databases D and D' as adjacent, if they differ in one single entry, formulated as $\|D - D'\|_1 = 1$. The formal definition of differential privacy goes as follows.

Definition 1 (Differential Privacy [9]). *A randomized mechanism $\mathcal{A} : D^n \rightarrow \mathbb{R}^d$ provides (ε, δ)-differential privacy, if for any adjacent datasets $D, D' \in D^n$, and any set of possible output S of $\mathcal{A}$,*

$$Pr[\mathcal{A}(D) \in S] \leq \exp(\varepsilon) \times Pr\left[\mathcal{A}\left(D'\right) \in S\right] + \delta,$$

where ε represents the privacy budget of mechanism $\mathcal{A}$, a smaller ε enforces stronger privacy guarantee.

(ε, δ)-differential privacy can be achieved by adding noise to the query function. The scale of noise needed to provide enough protection is determined by the sensitivity of query function. There are several types of sensitivity including global sensitivity, local sensitivity and smooth sensitivity, which are defined as follows.

Given a pair of adjacent datasets D and D' and a query function f, the global sensitivity [9] measures the maximum change of outputs:

Definition 2 (Global Sensitivity). *For $D, D' \in D^n$ and $f : D^n \rightarrow \mathbb{R}^d$, the global sensitivity of f (with respect to the ℓ_2 metric) is*

$$GS_f = \max_{D,D':\|D-D'\|_1=1} \|f(D) - f(D')\|_2.$$

In DPSGD [1], the authors employed the Gaussian noise mechanism to achieve differential privacy, defined as:

$$\mathcal{A}_f(D) \triangleq f(D) + \mathcal{N}\left(0, {GS_f}^2\sigma^2\right), \tag{1}$$

where $\mathcal{N}\left(0, {GS_f}^2\sigma^2\right)$ is the Gaussian distribution with mean 0 and standard deviation $GS_f\sigma$. According to [10], the mechanism satisfies (ε, δ)-differential privacy if $\sigma > \sqrt{2\ln(1.25/\delta)}/\varepsilon$ and $\varepsilon < 1$.

In global sensitivity, the scale of Gaussian noise depends on GS_f and privacy budget ε, while not considering the individual inputs. [22] introduced local measure of sensitivity:

Definition 3 (Local Sensitivity). *For datasets $D, D' \in D^n$ and $f : D^n \rightarrow \mathbb{R}^d$, the local sensitivity of f at D (with respect to the ℓ_2 metric) is*

$$LS_f(D) = \max_{D':\|D-D'\|_1=1} \|f(D) - f(D')\|_2.$$

We can derive that $GS_f = \max_D LS_f(D)$, which means that local sensitivity calibrate less noises. However, it does not satisfy (ε, δ)-differential privacy. They further introduced smooth sensitivity[22]:

Definition 4 (Smooth Sensitivity). *For $f : D^n \rightarrow \mathbb{R}^d$, a dataset $D \in D^n$ and $\beta > 0$ the β-smooth sensitivity of f is*

$$S_{f,\beta}(D) = \max_{D' \in D^n} \left(LS_f\left(D'\right) \cdot e^{-\beta\|D-D'\|_1}\right).$$

The smooth sensitivity focuses on the instance-specific noise, and yields less noise than the worse case of GS_f/ε. It is trivial to compute the smooth sensitivity, when the query function is maximum or minimum.

2.2 Differentially Private Learning

There is a rich literature [1, 4, 14, 17, 21, 23–25, 27, 30, 35, 38, 40] about differentially private learning, which aims to incorporate differential privacy into deep learning. As a pioneering work, Abadi *et al.*[1] proposed a DPSGD method, which incorporates differential privacy into SGD optimization and proposed a moments accountant privacy analysis approach to bound the privacy loss. To solve the subsampling problem of DPSGD method, Wang *et al.*[34] proposed a replacement subsampling privacy analysis method to track the privacy bound. Koskela *et al.*[14] proposed a learning rate adaptation approach, which significantly improved private model utility.

However, existing differentially private learning methods are still inefficient. To alleviate this issue, Lee *et al.*[17] proposed a per-sample-gradient-clipping method to speed up differentially private training. Besides, DPSGD training needs tuning extensive parameters. To solve this problem, Thakkar *et al.*[30] proposed to estimate the distribution of clipping norm automatically to remove the burden of tuning clipping norm. To improve the privacy/utility trade-offs which is greatly affected by the noise addition mechanism, Nasr *et al.*[21] proposed to encode the gradients into small vector space to choice best noise distribution, which provides better privacy guarantee. Differently, ADLM [26] proposed to add adaptive Laplace noise to loss function and features to improve model utility. Subsequently, Xiang *et al.*[35] proposed a new optimized additive mechanism, which intentionally adds noise to model parameters depending on their impact on the outputs. In this way, the proposed mechanism provides a better trade-off on privacy/utility. Recently a private model network construction method was proposed in [25], which demonstrated that the tempered sigmoid activation function can evidently improve the private model performances.

Different from the methods above, there emerges some distributed differentially private learning framework. In [28], the authors trained a central private model by jointly optimizing multiple local models and central model. McMahan *et al.*[19] proposed a distributed differentially private recurrent language framework, which protects the user-lever privacy with negligible loss in model utility. Instead of perturbing the gradient, Papernot *et al.*[23, 24] proposed a general framework, which perturbs the votes of multiple

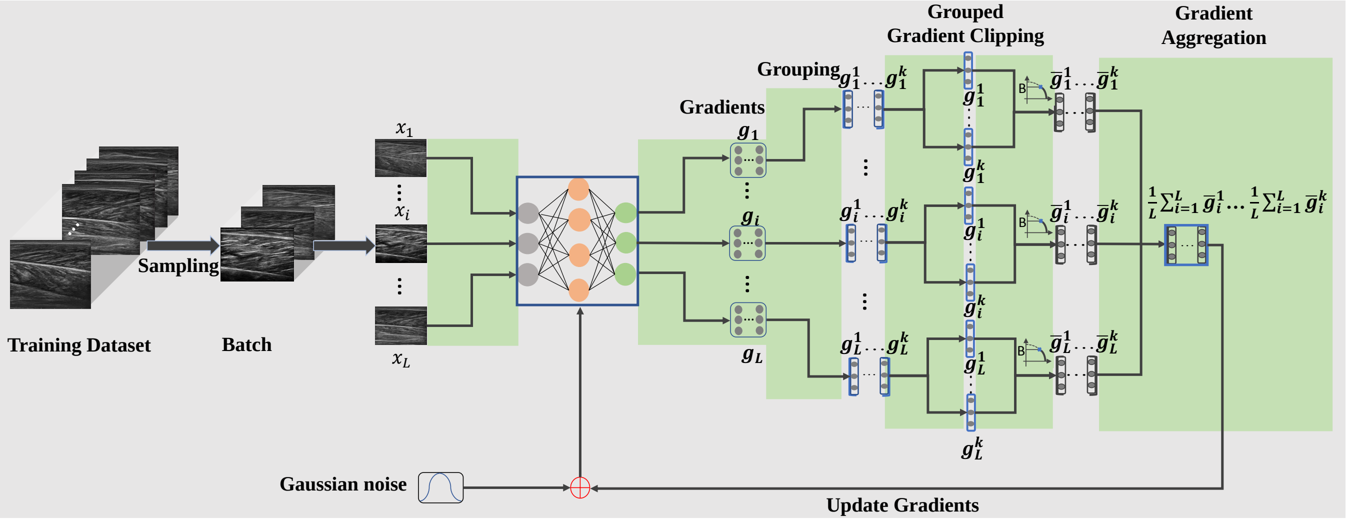


Figure 1: The framework of our DPL-GGC approach. In the framework, random examples are sampled from the dataset as the inputs of the deep model. To cope with the privacy leakage of sensitive data, the model gradients are firstly divided into k groups. And then the gradients of each group are clipped separately. Finally, all clipped gradients are aggregated and added with Gaussian noise before it is backpropagated to update model parameters.

local models. Recently, Zhu *et al.* [44] leverages the kNN feature embedding and privacy-amplification properties to release a private model. This method significantly improves model performance.

3 OUR APPROACH

3.1 Overall Framework

The framework of our proposed approach is shown in Fig 1. In each epoch, a batch of L examples are randomly sampled from the private dataset. Gradients for every example are extracted. To prevent the privacy leakage as well as preserve the model capacity, we propose to clip the gradient using our proposed grouped gradient clipping algorithm, before adding Gaussian noises. Besides, we propose to adopt smooth sensitivity to estimate the query function, so that we can inject less noise to the gradients.

3.2 Grouped Gradient Clipping

To train a model with parameters ω, one usually feed data in the training set $D = \{x_1, ..., x_m\}$ that contains m examples into the model, and minimize the empirical loss function $\mathcal{L}(\omega_{(j)}, x)$. At epoch t, the gradient $g_t(x_i)$ for sample x_i can be computed. Previous works [1] attempt to protect the privacy by clipping the gradient with a fixed bound B. The process can be formulated as:

$$\bar{g}_t(x_i) \leftarrow g_t(x_i)/\max(1, \frac{\|g_t(x_i)\|_2}{B}). \tag{2}$$

Here, our main insight is that the global gradient clipping and scaling without distinction could cause the distortion of gradient information. What's more, the private model would accumulate the loss in the back propagation progress, eventually resulting in terrible generalization performance. To solve the problems, we proposed the grouped gradient clipping method. As described in Algorithm 1, we divide the gradients into k groups, denoted as $\{g_i^1(x_i), ..., g_i^k(x_i)\}$. We then perform L_2 normalization for each group and the group norm can be formulated as $N = \{n_1, ..., n_k\}, n_k = \|g_t^k(x_i)\|_2$. Then each group is clipped by the norm bound B:

$$\bar{g}_t^k(x_i) \leftarrow g_t^k(x_i) * \min(1, \frac{B}{n_k}). \tag{3}$$

In the following, we state that the grouped gradient clipping algorithm always bring smaller gradient loss than the case in Eq. 2.

Theorem 1 (Group theorem). *For $k \geq 1$, the all gradients clipping for a sample x_i is the worse-case of the k grouped gradient clipping.*

When we set $k = 1$, the proposed grouped gradient clipping method degrades to the tradition clipping method, which clip the whole gradients with a norm bound. The proof of this theorem can be found in the *Supplementary Material*.

3.3 Smooth Noise Calibration

To satisfy the differential privacy, the magnitude of added Gaussian noise depends on the privacy budget ε and the sensitivity of query function f. However if the magnitude of the noise we add is too large, the gradient distribution could be far way form the true one, resulting in severe performance loss. We derive the global sensitivity of the proposed grouped gradient clipping as:

Theorem 2. *For $k \geq 1$, norm bound $B > 0$ and the determined query function $f = \sum_i \bar{g}_t(x_i)$, if we perform grouped gradient clipping for $g_t(x_i)$, and the outputs is $\bar{g}_t(x_i)$, the global sensitivity of f (with respect to the l_2 metric) is*

$$GS_f = B. \tag{4}$$

As suggested by theorem 2, while the grouped gradient clipping operation can evidently alleviate the gradient distortion, it didn't reduce the sensitivity of the query function f, samely B.

Algorithm 1: Differentially Private Learning with Grouped Gradient Clipping

```
Input: Dataset $x_1, ..., x_m$, loss function $\mathcal{L}(\omega_{(j)}, x)$.
       Parameters: group size $L$, learning rate $\eta_r$, noise
       scale $\sigma$, gradient norm bound $B$, number of groups $k$.
Output: model parameters $\omega$, privacy cost $(\varepsilon, \delta)$
Initialize model parameters $\omega_{(0)}$ randomly.
for $t \le T$ do
    Data Sampling:
    Take a random samples $S$ with sampling probability $L/m$.
    Compute gradient:
    for each $x_i \in S$, $g_t(x_i) \leftarrow \nabla\mathcal{L}(\omega_{(t)}, x_i)$
    Grouped Gradient Clipping
    $\overline{g}_t(x_i) \leftarrow GroupedGradClip(g_t(x_i), B, k)$
    Add smooth noise
    $\overline{g}_t \leftarrow \frac{1}{L}(\sum_i \overline{g}_t(x_i) + \mathcal{N}(0, \sigma^2 S^2_{f,\beta}\mathbf{I}))$
    Descent
    $\omega_{(t+1)} \leftarrow \omega_{(t)} - \overline{g}_t\eta_t$
end
return $\omega_{(t)}$, $(\varepsilon, \delta)$ of the iteration.
Function GroupedGradClip($gradients, bound, groups$)
Gradients $g_t(x_i)$ are divided into $k$ groups
    $g_t(x_i) = [g_t^1(x_i), ..., g_t^k(x_i)]$.
$N = \{n_j = \|g_t^j(x_i)\|_2$ : for $g_t^j(x_i)$ in $g_t(x_i)\}$
for $n_j \in N, g_t^j(x_i) \in g_t(x_i)$ do
    $\overline{g}_t^j(x_i) = g_t^j(x_i) * \min(1, \frac{bound}{n_j})$
end
return $\overline{g}_t(x_i) = [\overline{g}_t^1(x_i), ..., \overline{g}_t^k(x_i)]$
```

Therefore, we introduce the smooth noise calibration mechanism. Thus we introduce the local sensitivity to the proposed grouped gradient clipping method, which is given in Lemma 3.

Lemma 3. *Given a database D, grouped gradient clipping with $k \geq 1$, norm bound $B \geq 0$ and a determined query function $f = \sum_i \overline{g}_t(x_i)$, the scale of local sensitivity of f (with respect to l_2 metric) is*

$$LS_f \leq B. \tag{5}$$

The Lemma 3 shows that the proposed grouped gradients clipping method has lower sensitivity than the global sensitivity, which also means lower magnitude of added noise. However, the local sensitivity cannot satisfy differential privacy, it will release the information of the gradient database. Therefore, we incorporate the smooth sensitivity into Gaussian mechanism and propose theorem 4, which focuses on the instance-specific noise.

Theorem 4. *For $k > 0$, $\beta > 0$, a given database D and a query function $f = \sum_i \overline{g}_t(x_i)$, if we perform grouped gradient clipping before, then the β-smooth sensitivity of f is*

$$\begin{aligned} S_{f,\beta}(D) &= \max_{D' \in D^n} \left(LS_f\left(D'\right) \cdot e^{-\beta\|D-D'\|_1} \right) \\ &\leq \max_{D' \in D^n} \left(LS_f\left(D'\right) \right) \leq B = GS_f. \end{aligned} \tag{6}$$

Different from the local sensitivity, the smooth sensitivity satisfies the differential privacy. Since the proposed grouped gradient clipping method divides the gradient into several groups, the smoothing sensitivity implies less noise than the global sensitivity.

Finally to make our algorithm differential private, we propose the theorem 5 for the grouped gradient clipping method. Specifically we select Gaussian noise as the calibration noise.

Theorem 5. *Let the one-dimensional Gaussian distribution $N(0,1)$ be an (α, β)-admissible noise probability density function. For $\varepsilon, \delta \in (0,1)$, and a query function $f = \sum_i \overline{g}_t(x_i)$, we perform the grouped gradient clipping before the function f, let $S_{f,\beta}$ be the smooth sensitivity of function f, the algorithm $\mathcal{A}_f(D) \triangleq f(D) + \mathcal{N}\left(0, S^2_{f,\beta}\sigma^2\right)$ is (ε, δ)-differential privacy, where $\sigma = \frac{1}{\alpha}$ when*

$$\alpha \leq \frac{\varepsilon}{5\sqrt{2\ln(2/\delta)}}, \beta \leq \frac{\varepsilon}{4(1+\ln(2/\delta))}. \tag{7}$$

As implied by the theorem 5, the scale of Gaussian noise is proportional to $\frac{S_{f,\beta}}{\alpha}$. Also it is worth to note that, if we choose $\sigma \geq \frac{5\sqrt{2\ln(2/\delta)}}{\varepsilon}$ and a small δ, then $\beta < 1$, which allows us to inject less noise to the clipped gradients,

3.4 Privacy Analysis

The sum of clipped gradients are added with the careful calibrated gaussian noise in Algorithm 1. If the calibrated Gaussian noise followed the theorem 5, then every iteration of Algorithm 1 satisfied (ε, δ)-differential privacy. To track the accumulated privacy budget, we employ moments accountant [1] and have:

Lemma 6. *There exist constants c_1 and c_2 so that given the sampling probability $q = L/N$ and the number of steps T, for any $\varepsilon \in (0, min(1, c_1 q^2 T))$, $\delta \in (0,1)$, Algorithm 1 is (ε, δ)-differentially private for any $\delta > 0$, if we choose*

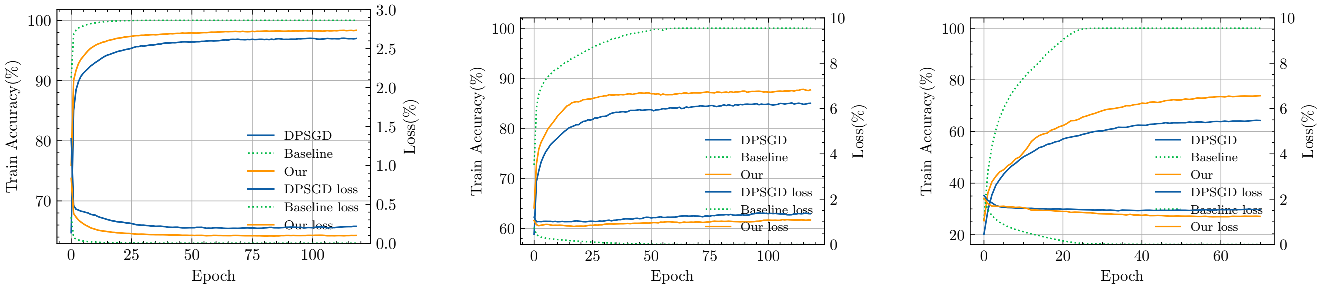
$$\sigma \geq c_2 \frac{q\sqrt{T\log(1/\delta)}}{\varepsilon}, \beta \leq \frac{\varepsilon}{4(1+\ln(2/\delta))}. \tag{8}$$

Theorem 7. *After running T step iteration in Algorithm 1, the learned model satisfied $(O(q\varepsilon\sqrt{T}), \delta)$-differential privacy.*

Compared with the traditional method, ours reduces the information loss of the gradient and the noise level added under the premise of ensuring the privacy protection ability. This is very important for the improvement of model performance. The proofs of all theorems and lemmas can be found in the *Supplementary Material*.

4 EXPERIMENTS

To verify the effectiveness of our approach, we conduct extensive experiments on different learning tasks to evaluate both the utility and the privacy of the trained model. We adopt 11 state-of-the-art approaches for comparison, including DPSGD [1], ADLM [26], DPBLSGD [6], DPOPT [35], zCDP [40], AdaCliP [27], GEDDP [21], ADADP [38], GDPSGD [4], FDPSGD [12], and TSADP [25].In the following, we first introduce the experimental settings including the benchmarking datasets, model and the implementation details. Next, we show the model convergence of the proposed method and the privacy trade-off by comparing with other state-of-the-art approaches. Finally, we evaluation its robustness on real scenarios.

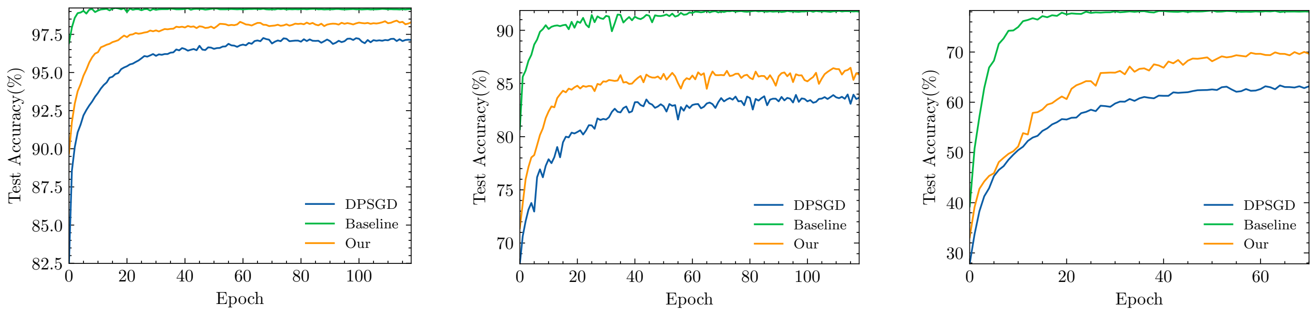


(a) Convergence results on MNIST

(b) Convergence results on Fashion-MNIST

(c) Convergence results on CIFAR10

Figure 2: Convergence results of the training accuracy and loss on various datasets.



(a) Accuracy on MNIST dataset

(b) Accuracy on Fashion-MNIST dataset

(c) Accuracy on CIFAR10 dataset

Figure 3: Model test accuracy on various datasets

4.1 Experimental Settings

Datasets. We evaluate the proposed approaches on three public datasets: MNIST [16], FashionMNIST [36], and CIFAR10 [15]. We also adopt a realistic medical dataset MedMNIST [39] to validate the generalization on the real-world scenario. The MNIST dataset is a database of handwritten digits containing a train set with $60k$ examples and a test set with $10k$ examples. Each example is a 28×28 grayscale image, with a label from 10 classes. The FashionMNIST dataset also contains $60k$ training examples and a test set with $10k$ examples. It is designed for a more complex classification task compared with MNIST. The CIFAR10 dataset consists of $60k$ 32×32 colour images in 10 classes. There are $50k$ training images and $10k$ test images. The MedMNIST dataset is a collection of 10 medical open datasets, standarized into 28×28 colour medical images. In this work, we employ the PathMNIST and three OrganMNIST datasets (Axial, Coronal, Sagittal) for multi-class classification.

Models. In the experiments on the MNIST and FashionMNIST datasets, we employ a model that contains two convolution layers and two linear layers, and finally a softmax layer on the top. For the experiments on the CIFAR10 dataset, we employ a model that consists of six convolution layers and two linear layers. In the experiments on the MedMNIST dataset, the model used consists of two convolutional layers and two linear layers.

Implementation Details. In all the experiments, we randomly shuffle the datasets and split them into training sets and test sets, to train and evaluate the private model, respectively. And we set $\beta = 0.95$. In experiments on MNIST and FashionMNIST, we set the batch size $S = 256$, the norm bound $B = 1$, the Gaussian noise $\sigma = 1.1$, the learning rate $lr = 0.05$ the group size $k = 8$ and fix $\delta = 10^{-5}$. For the experiments on CIFAR10 dataset, we set $S = 256, B = 0.1, \sigma = 1.54, lr = 1, k = 16, \delta = 10^{-6}$. In experiments on MedMNIST, we set $S = 512, B = 1, \sigma = 1.1, lr = 0.05, k = 8, \delta = 10^{-5}$. Our implementation is based on Pytorch platform [1].

4.2 Model Convergence and Accuracy

Model Convergence. We compare our proposed method with the baseline and the DPSGD method on three datasets in terms of the model convergence. The baseline is the unperturbed case. Fig. 2(a)-2(c) give the train accuracy and loss. The results show that the proposed method converge faster than DPSGD and has a closer performance to the baseline, especially on the simple tasks.

Accuracy. Fig. 3(a)-(c) show the results on three datasets. The accuracy of our method ramps up quickly in the first few iterations, and calms down eventually. Our method always performs better than DPSGD. It improves the accuracy by 1.2% on average (to 98.4%). Especially, on the MNIST dataset, our model achieves an accuracy boost of 2.47%. And on CIFAR10, our method improves the DPSGD method by 6.98%. In sum, our method can better preserve the model's task-related capacity while ensuring privacy.

[1] https://pytorch.org/

Table 1: Comparisons with the state-of-the-art approaches on MNIST Dataset. Here, $\delta = 10^{-5}$.

Approach	ε	Accuracy (%)
DPSGD [1]	2.00	95.00
ADLM [26]	2.00	93.66
DP-BLSGD [6]	2.50	90.00
DPOPT [35]	1.00	94.69
zCDP [40]	6.78	93.20
AdaCliP [27]	4.00	96.31
GEDDP [21]	3.20	96.10
ADADP [38]	1.40	96.00
GDPSGD [4]	2.32	96.60
FDPSGD [12]	1.20	96.60
TSADP [25]	2.93	98.10
Our DPL-GGC	**2.85**	**98.23**
	0.94	**97.05**

Table 2: Comparisons with the state-of-the-art approaches on Fashion-MNIST Dataset. Here, $\delta = 10^{-5}$.

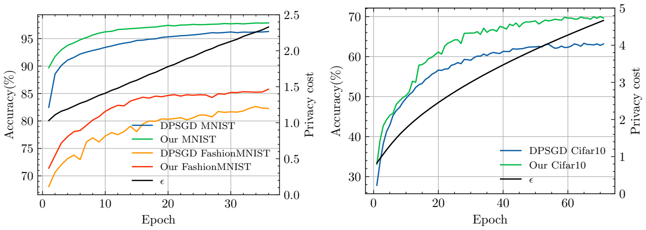
Approach	ε	Accuracy (%)
DPSGD [1]	2.70	81.90
DP-BLSGD [6]	3.00	82.30
TSADP [25]	2.7	86.10
Our DPL-GGC	**2.60**	**86.28**
	1.81	**84.76**

Table 3: Comparisons with the state-of-the-art approaches on CIFAR10 Dataset. Here, $\delta = 10^{-5}$.

Approach	ε	Accuracy (%)
DPSGD [1]	3.19	60.65
zCDP [40]	6.78	44.30
GEDDP [21]	3.00	55.00
DP-BLSGD [6]	8.00	53.00
TSADP [25]	7.53	66.20
Our DPL-GGC	**3.19**	**67.11**
	3.00	**65.91**

4.3 Trade-Off between Privacy and Accuracy

In addition, we compare our proposed method with the state-of-the-art differential private algorithms. To bound the privacy loss of our proposed method, we employ the moments accountant method to compute the iteration's privacy cost. As shown in Tab.1, our method achieves 98.19% accuracy with $(2.39, 10^{-5})$-differential privacy on the MNIST dataset. Compared with the TSADP method, which achieves 98.28% accuracy with $(2.85, 10^{-5})$, we have a comparable accuracy with a smaller value of ε, which means stronger privacy guarantees. It is worth noting that the performance of TSADP method is based on the specific model structure(ie., activation function). However, for fair comparison with other methods, we adopt the same naive model structure as DPSGD method and gain a better privacy trade-off. Tab. 2 and Tab. 3 show that our method also surpass the prior works. To illustrate the trade-off between privacy cost and accuracy, we show the curve of model accuracy varies with the privacy cost in Fig. 4(a)-4(b), we set a medium privacy level for MNIST and Fashion-MNIST datasets and high privacy level for CIFAR10 dataset, the results show that our proposed approach has a better trade-off to balance the privacy and the model accuracy.



(a) Medium privacy level on MNIST and Fashion-MNIST **(b) High privacy level on CIFAR10**

Figure 4: Accuracy under different privacy levels.

Table 4: The trade-off comparisons between privacy and accuracy (%) on four MedMNIST datasets. Here, $\delta = 10^{-5}$.

Dataset	ε	DPSGD	Our DPL-GGC
PathMNIST	4.27	52.78	**59.10**
OrganMNIST (Axial)	7.65	39.88	**48.22**
OrganMNIST (Coronal)	10.00	33.11	**36.91**
OrganMNIST (Sagittal)	12.50	29.51	**35.93**

4.4 Evaluation on Real-World Datasets

To evaluate the robustness of the proposed approach in real-world scenarios, we test our method on four MedMNIST datasets, which contain the primary data modalities of medical image analysis. As shown in Tab. 4, compared with the DPSGD, our approach performs higher accuracy under the same privacy levels. The accuracy exceeds that of DPSGD by 6.22% in average. We also notice that the performance is not satisfactory in general. For example, the highest accuracy is only 59.10%. This may be because we use a naive model and restrict the data scale. This is an open problem in future works.

5 CONCLUSION

The released deep learning models bring widespread privacy concerns. In this paper, we propose an effective differentially private learning approach to facilitate model accuracy while preserving privacy. The approach uses a simple grouped gradients clipping operation to reduce gradient deviation, and combines smooth sensitivity with Gaussian mechanism to bound the noise addition. Extensive experiments demonstrate that the proposed approach can achieve a better trade-off between privacy and utility. Our further work is to extend the approach on more real-world applications.

Acknowledgement. This research is supported in part by grants from the National Natural Science Foundation of China (61772513), Beijing Natural Science Foundation(L192040) and the National Key Research and Development Program of China (2020AAA0140001). Shiming Ge is also supported by the Youth Innovation Promotion Association, Chinese Academy of Sciences.

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep Learning with Differential Privacy. In *ACM SIGSAC Conference on Computer and Communications Security*. 308—-318.

[2] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. 2019. Differential Privacy Has Disparate Impact on Model Accuracy. In *Advances in Neural Information Processing Systems*, Vol. 32. 15479–15488.

[3] Raef Bassily, Adam Smith, and Abhradeep Thakurta. 2014. Private Empirical Risk Minimization: Efficient Algorithms and Tight Error Bounds. In *IEEE 55th Annual Symposium on Foundations of Computer Science*. 464–473.

[4] Zhiqi Bu, Jinshuo Dong, Qi Long, and Su Weijie. 2020. Deep learning with Gaussian differential privacy. *Harvard data science review* 23 (2020), 1–48.

[5] Mark Bun and Thomas Steinke. 2016. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*. 635–658.

[6] Chen Chen and Jaewoo Lee. 2020. Stochastic Adaptive Line Search for Differentially Private Optimization. In *IEEE International Conference on Big Data (Big Data)*. 1011–1020.

[7] Xiangyi Chen, Steven Z. Wu, and Mingyi Hong. 2020. Understanding Gradient Clipping in Private SGD: A Geometric Perspective. In *Advances in Neural Information Processing Systems*, Vol. 33. 13773–13782.

[8] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. 486–503.

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

[10] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Foundations Trends in Theoretical Computer Science*. 9, 3-4 (2014), 211–407.

[11] Cynthia Dwork, Guy N. Rothblum, and Salil Vadhan. 2010. Boosting and Differential Privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science (FOCS)*. 51–60.

[12] Vitaly Feldman and Tijana Zrnic. 2020. Individual Privacy Accounting via a Renyi Filter. *arXiv* (2020). https://arxiv.org/abs/2008.11193

[13] Nikhil Joshi and Rewanth Tammana. 2019. GDALR: an efficient model duplication attack on black box machine learning models. In *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*. 1–6.

[14] Antti Koskela and Antti Honkela. 2020. Learning Rate Adaptation for Differentially Private Learning. In *International Conference on Artificial Intelligence and Statistics*. 2465–2475.

[15] Alex Krizhevsky. 2009. *Learning multiple layers of features from tiny images*. Technical Report.

[16] Yann LeCun and Corinna Cortes. 2010. MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/. (2010). http://yann.lecun.com/exdb/mnist/

[17] Jaewoo Lee and Daniel Kifer. 2021. Scaling up Differentially Private Deep Learning with Fast Per-Example Gradient Clipping. *Proceedings on Privacy Enhancing Technologies* 2021, 1, 128–144.

[18] H Brendan McMahan, Galen Andrew, Ulfar Erlingsson, Steve Chien, Ilya Mironov, Nicolas Papernot, and Peter Kairouz. 2018. A general approach to adding differential privacy to iterative training procedures. In *Advances in Neural Information Processing Systems (NeurIPS) Workshop on Privacy Preserving Machine Learning*.

[19] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning Differentially Private Recurrent Language Models. In *International Conference on Learning Representations*. 1–14.

[20] Ilya Mironov. 2017. Rényi Differential Privacy. In *IEEE 30th Computer Security Foundations Symposium (CSF)*. 263–275.

[21] Milad Nasr, Reza Shokri, et al. 2020. Improving Deep Learning with Differential Privacy using Gradient Encoding and Denoising. *arXiv* (2020). https://arxiv.org/abs/2007.11524

[22] Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2007. Smooth Sensitivity and Sampling in Private Data Analysis. In *The Thirty-Ninth Annual ACM Symposium on Theory of Computing*. 75–84.

[23] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. 2017. Semi-supervised knowledge transfer for deep learning from private training data. In *International Conference on Learning Representations*. 1–16.

[24] Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Úlfar Erlingsson. 2018. Scalable private learning with pate. In *International Conference on Learning Representations*. 1–34.

[25] Nicolas Papernot, Abhradeep Thakurta, Shuang Song, Steve Chien, and Úlfar Erlingsson. 2021. Tempered Sigmoid Activations for Deep Learning with Differential Privacy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 9312–9321.

[26] NhatHai Phan, Xintao Wu, Han Hu, and Dejing Dou. 2017. Adaptive Laplace Mechanism: Differential Privacy Preservation in Deep Learning. In *IEEE International Conference on Data Mining (ICDM)*. 385–394.

[27] Venkatadheeraj Pichapati, Ananda Theertha Suresh, Felix X Yu, Sashank J Reddi, and Sanjiv Kumar. 2019. AdaCliP: Adaptive clipping for private SGD. *arXiv* (2019). https://arxiv.org/abs/1908.07643

[28] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-Preserving Deep Learning. In *ACM SIGSAC Conference on Computer and Communications Security*. 1310–1321.

[29] Shuang Song, Thomas Steinke, Om Thakkar, and Abhradeep Thakurta. 2021. Evading the curse of dimensionality in unconstrained private glms. In *International Conference on Artificial Intelligence and Statistics*. 2638–2646.

[30] Om Thakkar, Galen Andrew, and H Brendan McMahan. 2019. Differentially private learning with adaptive clipping. *arXiv* (2019). https://arxiv.org/abs/1905.03871

[31] Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. 2016. Stealing machine learning models via prediction apis. In *{USENIX} Security Symposium ({USENIX} Security)*. 601–618.

[32] Binghui Wang and Neil Zhenqiang Gong. 2018. Stealing Hyperparameters in Machine Learning. In *IEEE Symposium on Security and Privacy (SP)*. 36–52.

[33] Di Wang and Jinhui Xu. 2019. Differentially private empirical risk minimization with smooth non-convex loss functions: A non-stationary view. In *AAAI Conference on Artificial Intelligence*, Vol. 33. 1182–1189.

[34] Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. 2019. Subsampled Rényi differential privacy and analytical moments accountant. In *The 22nd International Conference on Artificial Intelligence and Statistics*. 1226–1235.

[35] Liyao Xiang, Jingbo Yang, and Baochun Li. 2019. Differentially-Private Deep Learning from an optimization Perspective. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. 559–567.

[36] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747* (2017).

[37] Jie Xie, Nanjun He, Leyuan Fang, and Pedram Ghamisi. 2020. Multiscale Densely-Connected Fusion Networks for Hyperspectral Images Classification. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 1 (2020), 246–259.

[38] Zhiying Xu, Shuyu Shi, Alex X Liu, Jun Zhao, and Lin Chen. 2020. An adaptive and fast convergent approach to differentially private deep learning. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. 1867–1876.

[39] Jiancheng Yang, Rui Shi, and Bingbing Ni. 2021. MedMNIST Classification Decathlon: A Lightweight AutoML Benchmark for Medical Image Analysis. In *IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. 191–195.

[40] Lei Yu, Ling Liu, Calton Pu, Mehmet Emre Gursoy, and Stacey Truex. 2019. Differentially Private Model Publishing for Deep Learning. In *IEEE Symposium on Security and Privacy (SP)*. 332–349.

[41] Ting Yu, Jun Yu, Zhou Yu, Qingming Huang, and Qi Tian. 2020. Long-Term Video Question Answering via Multimodal Hierarchical Memory Attentive Networks. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 3 (2020), 931–944.

[42] Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. 2019. Why Gradient Clipping Accelerates Training: A Theoretical Justification for Adaptivity. In *International Conference on Learning Representations*.

[43] Ke Zhang, Yuanqing Li, Jingyu Wang, Erik Cambria, and Xuelong Li. 2021. Real-Time Video Emotion Recognition based on Reinforcement Learning and Domain Knowledge. *IEEE Transactions on Circuits and Systems for Video Technology* (2021), 1–14.

[44] Yuqing Zhu, Xiang Yu, Manmohan Chandraker, and Yu-Xiang Wang. 2020. Private-kNN: Practical Differential Privacy for Computer Vision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 11854–11862.