Private-kNN: Practical Differential Privacy for Computer Vision

Yuqing Zhu[1,2] Xiang Yu[2] Manmohan Chandraker[2,3] Yu-Xiang Wang[1]
[1]University of California, Santa Barbara
[2]NEC Labs America
[3]University of California, San Diego

Abstract

*With increasing ethical and legal concerns on privacy for deep models in visual recognition, differential privacy has emerged as a mechanism to disguise membership of sensitive data in training datasets. Recent methods like Private Aggregation of Teacher Ensembles (PATE) leverage a large ensemble of teacher models trained on disjoint subsets of private data, to transfer knowledge to a student model with privacy guarantees. However, labeled vision data is often expensive and datasets when split into many disjoint training sets lead to significantly sub-optimal accuracy and thus hardly sustain good privacy bounds. We propose a practically data-efficient scheme based on private release of k-nearest neighbor (kNN) queries, which altogether avoids splitting the training dataset. Our approach allows the use of privacy-amplification by subsampling and iterative refinement of the kNN feature embedding. We rigorously analyze the theoretical properties of our method and demonstrate strong experimental performance on practical computer vision datasets for face attribute recognition and person re-identification. In particular, we achieve comparable or better accuracy than PATE while reducing more than **90%** of the privacy loss, thereby providing the "most practical method to-date" for private deep learning in computer vision.* [1]

1. Introduction

Recent studies have shown that many machine learning (ML) models trained on sensitive human subject data can be used to re-identify individual subjects [27] or reconstruct sensitive information such as social security and credit card numbers [6]. Moreover, recent legislative steps such as the General Data Protection Regulation (GDPR) have elevated privacy from a mere "risk" to a central "requirement" for institutions and governments across the world.

Differential privacy (DP) [11] is a quantifiable and composable definition of privacy that provides provable guarantees against identifications of individuals in a data set. As of today, DP has been widely adopted and has become the *de facto* standard for defining privacy. *Differentially private machine learning* [15, 7] is a burgeoning area of research that aims to train ML models with formal DP guarantees, which ensures no attacker (with arbitrary side information) may distinguish models trained with or without a specific training example, thus, addressing the aforementioned problems.

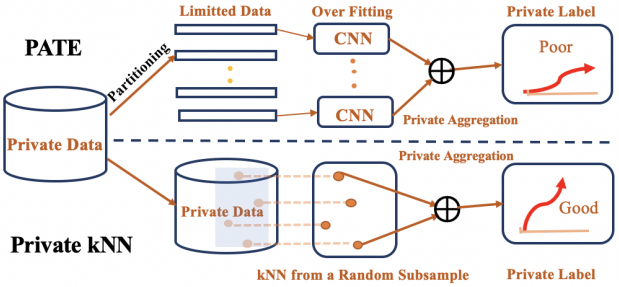

Figure 1. A comparison of PATE's framework and ours.

The key idea of differentially private machine learning is to appropriately *randomize* the training process (e.g. adding noise), so the fitted model parameters can be thought of as a sanitized "release" with individual information removed. Several existing approaches do not apply for deep learning [7, 10, 30, 26]. A notable exception – NoisySGD [28, 3, 1] – requires privately releasing the gradients for many iterations by adding noise proportional to $\sqrt{d}$ to every coordinate of the gradient in a model with d parameters, hence does not scale to large models with millions of parameters that are commonly used in computer vision.

A recent *model-agnostic* approach, termed "Private Aggregation of Teacher Ensembles" (PATE), introduces a model aggregation strategy and gains privacy by injecting randomness into the aggregation [24, 25]. It assumes a teacher-student knowledge transfer framework by leveraging an isolated private data and unrestricted public unlabeled data. The most critical parameter to choose in PATE is the number of disjoint teachers k. It largely determines the margin between the top two votes and is often as large as 250 for a meaningful privacy guarantee while ensuring the sufficiently accurate pseudo-labels. When the teacher model is a deep neural network, large-scale data is required for each model

[1]Code is available at https://github.com/jeremy43/Private_kNN

to achieve generalization for typical computer vision tasks. On the other hand, labeled visual data can be expensive to acquire and as such, using a large k for most public vision datasets would render them insufficient. For example, setting $k = 250$ for CIFAR-10 yields just 200 images for each disjoint teacher, leading to accuracy under 50%.

To address the problem, we propose a more data-efficient differentially private algorithm based on releasing pseudo-labels using the majority voting of the k-nearest neighbors (kNN). This approach avoids data-splitting since adding or removing an individual to the data can change at most one of any sample's k-nearest neighbor. This enables us to choose larger k without worrying about not having enough data to train teachers – kNN involves no training at all. Moreover, this allows leveraging recent advances in "privacy amplification by sampling" to pseudo-label orders-of-magnitude more public data at only a fraction of the privacy cost of PATE.

The careful reader may ask how this may present an advantage given modern deep networks. Despite the strong guarantee that kNN asymptotically achieves the Bayes rate [8], it is not known as a state-of-the-art classifier in finite-sample computer vision problems. Our novel solution to this problem is to make learning *iterative*. Specifically, we outsource representation-learning to the public domain, where the student trained with a deep neural network model shares the learned feature map with the teacher. So, the quality of kNN's pseudo-labels improves, which in return helps the student to learn a better representation as we iteratively conduct this. Though the student model is shared to extract features in the private domain, there is no model parameter update utilizing the private data, thus not violating the differential privacy setting. Within our framework, "privacy amplification by sampling" is the key component, which consumes the privacy budget more efficiently and thus enables releasing more pseudo-labels.

Our main contributions are summarized below:

- We propose Private k-Nearest Neighbor (Private-kNN), the *first* practical differentially private deep learning solution for large-scale computer vision that achieves theoretically meaningful DP guarantees ($\epsilon < 1$).
- We present a new Renyi-differential privacy analysis on the "noisy screening" mechanism proposed in [25]. This allows us to use it with the moments accountant for a tighter privacy accounting. Collectively, "subsampling" and "noisy screening" allow us to answer 10 times more queries with even less privacy budget compared to state-of-the-art PATE models. The data-dependent version of this "noisy screening" mechanism can be thought of as a post-hoc Gaussian-noise version of the well-known Sparse Vector Technique in differential privacy, which is of independent interest.
- We evaluate our approach on extensive vision tasks such as classification on MNIST, SVHN, CIFAR-10, as well as two realistic identity-relevant tasks of face attribute classification on Celeb-A and human body attribute classification on Market1501. Private-kNN achieves consistently better performance across privacy cost and accuracy for all the above, compared to other state-of-the-art methods for differentially-private learning.

2. Preliminaries

In this section, we review the necessary technical components that we build upon and we start by defining DP.

Definition 1 (Differential Privacy[11])**.** A randomized algorithm $\mathcal{M} : \mathcal{X} \to \Theta$ is (ϵ, δ)-DP (differentially private) if for every pair of neighboring datasets $X, X' \in \mathcal{X}$, and every possible (measurable) output set $E \subseteq \Theta$ the following inequality holds: $\Pr[\mathcal{M}(X) \in E] \leq e^{\epsilon} \Pr[\mathcal{M}(X') \in E] + \delta$.

The definition provides rigorous, information-theoretic guarantee against on an adversary's ability to infer whether one data is being used in the training process of randomized mechanism $\mathcal{M}$. $\epsilon, \delta \geq 0$ are privacy loss parameters, which quantify the strength of the privacy protection. In practice, we consider a privacy guarantee meaningful if $\epsilon \approx 1$ and $\delta = o(1/n)$ where n is the size of private dataset. One important property of DP is that it is *closed under post-processing*, which says that if we can privately label the public data, then the resulting model from training the public data enjoys the same privacy guarantee.

Rényi Differential Privacy and Moments Accountant. Rényi differential privacy (RDP)[19] is a generalization of $(\epsilon, 0)$-DP that uses Rényi-divergence as a distance metric.

Definition 2 (Rényi Differential Privacy [20])**.** We say that a mechanism $\mathcal{M}$ is (α, ϵ)-RDP with order $\alpha \in (1, \infty)$ if for all neighboring datasets X, X'

$$D_{\alpha}(\mathcal{M}(X) \| \mathcal{M}(X')) = \frac{1}{\alpha - 1} \log E_{\theta \sim \mathcal{M}(X')} \left[\left(\frac{p_{\mathcal{M}(X)}(\theta)}{p_{\mathcal{M}(X')}(\theta)} \right)^{\alpha} \right] \leq \epsilon.$$

As $\alpha \to \infty$, RDP converges to the standard $(\epsilon, 0)$-DP. More generally, we can convert RDP to standard (ϵ, δ)-DP for any $\delta > 0$ using:

Lemma 3 (From RDP to DP)**.** *If a mechanism $\mathcal{M}$ satisfies (α, ϵ)-RDP, then $\mathcal{M}$ also satisfies $(\epsilon + \frac{\log 1/\delta}{\alpha - 1}, \delta)$-DP for any $\delta \in (0, 1)$.*

There is a partial inverse, which says any $(\epsilon, 0)$-DP algorithm obeys $(\alpha, \alpha\epsilon^2/2)$-RDP [5].

It is often convenient to consider RDP in its function form. Hereafter, we denote $\epsilon_{\mathcal{M}}(\alpha)$ as the RDP ϵ of $\mathcal{M}$ at order α. The function $\epsilon_{\mathcal{M}}(\cdot)$ provides a more refined characterization of the privacy guarantee associated with $\mathcal{M}$. The Gaussian mechanism is such an example.

Lemma 4 (Gaussian Mechanism [5]). *Let $f : \mathcal{X} \to \mathcal{R}$ obeys that $\|f(X) - f(X')\|_2 \leq \Delta_2$ for any neighboring datasets X, X', the Gaussian mechanism $\mathcal{M}(X) = f(X) + \mathcal{N}(0, \sigma^2)$ obeys RDP with $\epsilon_{\mathcal{M}}(\alpha) = \frac{\alpha \Delta_2^2}{2\sigma^2}$.*

Another notable advantage of RDP over (ϵ, δ)-DP is that it composes very naturally.

Lemma 5 (Composition with Rényi Differential Privacy). *Let mechanism $\mathcal{M} = (\mathcal{M}_1, ..., \mathcal{M}_t)$ where $\mathcal{M}_i$ can potentially depend on the outputs of $\mathcal{M}_1, ..., \mathcal{M}_{i-1}$. Then $\mathcal{M}$ obeys RDP with $\epsilon_{\mathcal{M}}(\cdot) = \sum_{i=1}^{t} \epsilon_{\mathcal{M}_i}(\cdot)$.*

This allows to calculate the advanced composition [13] in the standard DP significantly more easily, and often tighter as well. An application of Lemma 3 and Lemma 5 largely benefits the moments accountant [1] technique — a data-structure that keeps track of RDP vector ϵ from a sequence of RDP mechanisms, through which it finds the smallest possible ϵ for any δ by searching over α. All privacy guarantee that we report in this paper is based on the analytical moments account [29] that keeps track of the entire RDP function in their analytical form and solve for ϵ given δ via a binary search.

Privacy Amplification by Subsampling. Subsampling is a widely used algorithmic tool in privacy, which deals with a composite mechanism that first randomly samples the data, and then applies a DP mechanism on the randomly selected subset. Intuitively, since the one person that differs between X and X' is often not selected in the subset, the overall privacy guarantee should be stronger. Loosely speaking, when we apply an (ϵ, δ)-DP mechanism to a random γ-proportion of the data, the whole procedure satisfies $(O(\gamma\epsilon), \gamma\delta)$-DP. The result of this style is also known as "subsampling lemma" or "secrecy of the samples" in the literature [2]. This is practically relevant as it is the reason why we can afford to run Noisy-SGD [28] for many iterations without blowing up the privacy cost. Recently, such as "subsampling lemma" was proven for the RDP. The benefits of the subsampling can be combined with the tight advanced composition of RDP [29, 33], which roughly says that under some restrictions on α:

$$\epsilon_{\mathcal{M} \circ \text{Sample}_\gamma}(\alpha) \leq O(\gamma^2 \epsilon_{\mathcal{M}}(\alpha)).$$

In this work, we apply a Poisson subsampled "RDP-amplification " bound from [33]. A more precise statement of this result is attached in the appendix. We emphasize that this is the main technical contribution leveraged in this work that simply cannot be done under the PATE approach.

Data-Dependent RDP and PATE The privacy analysis in PATE is straight-forward. It involves injecting Laplace noise [24] or Gaussian noise [25] to the teacher votes. For noise with standard deviation $O(k)$, a budget of ϵ, δ, roughly speaking, allows PATE to release $O(\frac{\epsilon^2 k^2}{\log(1/\delta)})$ pseudo-labels, which is insufficient for many cases.

A notion of data-dependent RDP is introduced to further take into account of the high margin that occurs when the teachers largely agree with each other, in which case the privacy cost is intuitively smaller.

Definition 6 (Data-dependent RDP [24]). A mechanism $\mathcal{M}$ is (α, ϵ)-data-dependent RDP with order $\alpha \in (1, \infty)$ if for all X' that is a adjacent to X

$$\max\{D_\alpha(\mathcal{M}(X) \| \mathcal{M}(X')), D_\alpha(\mathcal{M}(X') \| \mathcal{M}(X))\} \leq \epsilon.$$

In other words, the data-dependent *RDP function* ϵ is a joint function of X and α. There are a few other tricks proposed in [25] to reduce the total privacy loss. Notably, they designed a " noisy screen" step that first adds a larger Gaussian noise to $\max\{\text{votes}\}$, and then release a more confident version of votes only for those questions that passes the screening. This allows PATE to save privacy loss via data-dependent RDP in the second step with smaller noise. In this paper, we use the same "noisy screening" but provide a tighter analysis of this procedure that saves a constant fraction of the privacy budget.

Finally, we note that the use of data-dependent RDP can be seen as controversial, as the resulting privacy loss ϵ is now a sensitive quantity that depends on the data. [25] provided a smooth-sensitivity based method [23] to privately release $\epsilon_{\mathcal{M},X}(\alpha)$ for a sequence of α, but that incurs additional privacy losses that are not reported in their main result. One major contribution of the current paper is to demonstrate that practical differential privacy can be achieved when training a deep networks under the "knowledge transfer" setting even without using data-dependent RDP.

3. Our Approach

We are now ready to describe our method: Private-kNN.

Notations and symbols. In this section and thereafter, we stick to the following notations. $x \in \mathcal{R}^d$ denotes the feature of both private and public data. Let $D_{private}$ be the private dataset of size n: $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ and $y_i \in [1, c]$ is the label, where c is the number of classes in $D_{private}$. Let m be the size of the unlabeled public data. γ is the sampling ratio used to sample a random subset D_γ from $D_{private}$. We define ϕ be the feature extractor for private kNN. $f_j(x)$ is the prediction of j^{th} neighbor on the public feature x and the total number of neighbors is k. In the noisy screening, we use σ_1 to denote the Gaussian noise scale, and T is the threshold for a screening check. σ_2 is the Gaussian noise scale for the noisy aggregation procedure. ϵ and δ are reserved for denoting privacy cost.

Setup. As defined in PATE, we have access to a private dataset and an unlabeled public dataset, and we seek to

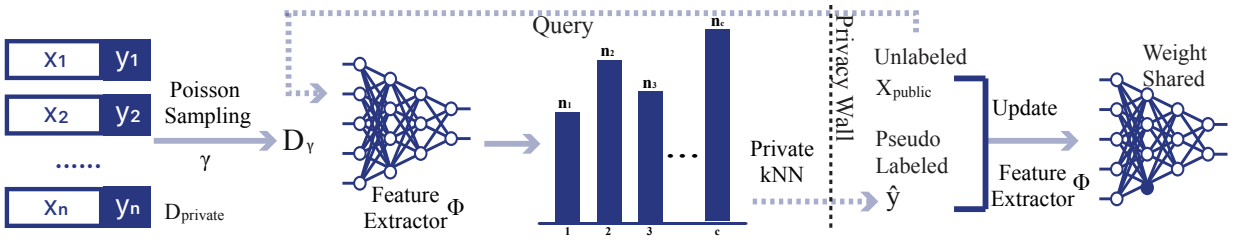

Figure 2. The overview of the proposed framework. Given the unlabeled public data X_{public}, we query through privacy wall for pseudo labels, where the private data and the queried public data are sent through feature extractor Φ and "Private-kNN" to assign pseudo labels. Combining the public data and the pseudo labels, the feature extractor Φ is further updated. This procedure can be iterated for rounds to achieve satisfied privacy-accuracy trade-off.

design an (ϵ, δ)-DP algorithm that outputs pseudo-labels for as much public data as possible. Then a student model is trained via semi-supervised learning using both pseudo labeled and unlabeled public data. Again, by the property of "closedness to postprocessing", the student model itself satisfies DP assumption.

Private-kNN. Our algorithm involves four simple steps.

1. PICK K-NEAREST NEIGHBORS WITH POISSON SAMPLING For each query x from the public domain, we use Poisson sampling[2] to get a random subset from the entire private dataset. Then we pick the k nearest neighbors from $\mathcal{D}_\gamma$ by measuring their Euclidean distance in feature space $\mathcal{R}^{d_\phi}$, where ϕ is a non-private feature extractor. The choice of Euclidean distance is general, whereas other distance metrics can also be applied. Our algorithm is designed into rounds of iterations. In the first iteration, ϕ is initialized with a Histogram of Oriented Gradient (HOG)[9] feature extractor, which is a popular descriptor used in the computer vision tasks. In the next iteration, we apply a deep neural network for the public student model (except for the last softmax layer) to update the feature extractor ϕ. In the experiment section, we show how this interactive scheme iteratively refines the feature embedding used by Private-kNN.

2. NOISY SCREENING. let $f_j(x)$ be the prediction of j^{th} neighbor on x, where $j \in [1, k]$. The label count of class $i \in [1, c]$ is

$$n_i(x) = |\{j : f_j(x) = i\}|$$

Answering all queries from public without selection leads to running out privacy budget instantly. To be more selective, we only answer those queries which have an overwhelming consensus in voting, and this screening process is implemented privately with Gaussian noise parameter σ_1, for the query not passing the noisy screening check, we return $\perp$, and ignore this data in re-training a student model.

$$\text{If } \max_i\{n_i(x)\} + \mathcal{N}(0, \sigma_1^2) \leq T \text{ then return } \perp$$

[2]Possion sampling includes each data point independently with probability γ. It can be efficiently implemented by first sample the size of the subset from a Binomial distribution then find a random subset.

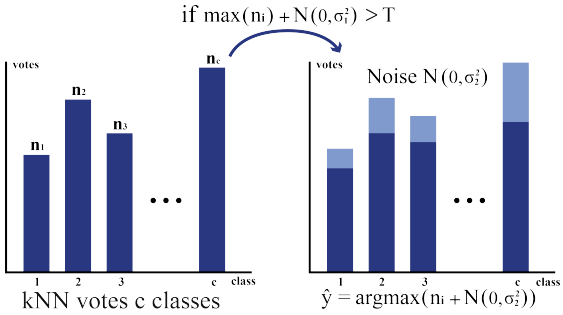

Figure 3. Illustration on the noisy screening and noisy aggregation procedure.

T here is the threshold parameter for screening, we set $T \approx 0.6 \times k$ in the hope of there is consensus among neighbors upon this query. Since we pay for private screening for every query, a larger σ_1 would be helpful for privacy concerns. As we mentioned before, the same screening procedure is used in PATE[25] and despite a larger noise, this is still the most costly part of PATE. PATE treated this screening procedure as a simple post-processing of the Gaussian mechanism. We note that the output is actually drawn from a discrete distribution of either $\top$ (Pass) or $\perp$ (Fail). In the next section we derive the RDP for this procedure, which allows to benefit from moments accountant.

3. NOISY AGGREGATION For those query x which pass the check, we release its label

$$f(x) = \arg\max_j\{n_j + \mathcal{N}(0, \sigma_2^2)\}$$

with a fresh random subsample of the data. The noisy screening process filters out about 50% query, which enables the noisy aggregation process to have a smaller σ_2 for better-aggregated accuracy.

4.TRAINING STUDENT MODEL Our model only answers a selected number of queries from the public. Otherwise the final privacy cost becomes meaningless. Taking the answered queries as pseudo labeled data, together with the unlabeled data, a student model is trained in the self-supervised manner. We consider two popular self-supervised methods: virtual adversarial training(VAT)[21] and unsupervised

Data Augmentation(UDA)[31]. VAT uses the virtual adversarial perturbation in the noisy process and UDA exploits advanced data augmentation instead of random augmentation. In our experiments, we find that UDA outperforms VAT in both SVHN and CIFAR-10 tasks. As shown in Figure 2, the student model is trained with the above mentioned self-supervised method. On the other hand, the student model is utilized to extract the updated feature in the private domain for private-kNN. This iterative feature distilling allows private-kNN to have similar capacity as ConvNet (replace the last softmax layer in ConvNet with kNN), and to further improve the accuracy of answering public queries. Besides, iterative training allows to exploit the benefits from unlabeled public data, which does not violate the DP assumption or incur any privacy cost, but is shown to enhance the utility of student model under the self-supervised training.

Privacy analysis. We prove the DP guarantee in the following. Let $\mathcal{M}$ denote the mechanism of Private-kNN. Our method can be viewed as a composition of $(\mathcal{M}_s) \circ \text{Sample}_\gamma$ and $(\mathcal{M}_{\sigma_2}) \circ \text{Sample}_\gamma$. Based on composition theorem, the privacy cost can be traced by individually calculating the RDP of the two mechanisms and then add them up. For the latter, we can readily apply the tight bound of the subsampled Gaussian mechanism from [33]. Our main theoretical result is the following characterization of the noisy screening procedure via a tight RDP analysis.

Theorem 7 (RDP of "Noisy Screening")**.** *Let $\mathcal{M}_s$ be a randomized algorithm for noisy screening procedure with a predefined Gaussian noise scale σ_1 and the threshold T. Then $\mathcal{M}_s$ obeys RDP with*

$$\epsilon_{\mathcal{M}_s}(\alpha) = \max_{(p,q)\in\mathcal{S}} \frac{1}{\alpha-1} \log\left(p^\alpha q^{1-\alpha} + (1-p)^\alpha (1-q)^{1-\alpha}\right).$$

where $\mathcal{S}$ contains the following "pairs":

$$\left(\mathbb{P}[\mathcal{N}(t,\sigma_1^2) \geq T], \mathbb{P}[\mathcal{N}(t+1,\sigma_1^2)] \geq T]\right),$$
$$\left(\mathbb{P}[\mathcal{N}(t,\sigma_1^2) \geq T], \mathbb{P}[\mathcal{N}(t-1,\sigma_1^2)] \geq T]\right)$$

for all integer $\lceil k/c \rceil \leq t \leq k$.

We remark that the above bound can be calculated efficiently for any pairs of k, T in $O(k)$ time and can be evaluated by calculating the Gaussian cumulative density function using the efficient implementation of the error function erfc. A more detailed proof is provided in the appendix. Moreover, it is more numerically stable to directly represent the log of p and q above. By the information-processing inequality of Rényi-divergence, this bound is strictly better than that from the Gaussian mechanism for every α.

Finally, we estimate the overall privacy bound for the end-to-end method.

Theorem 8 (Asymptotic scaling)**.** *The total privacy bound of Private-kNN to label all m public data points with noise σ_1, σ_2 is (ϵ, δ)-DP, with any δ, and*

$$\epsilon = O(\gamma\sqrt{\log(1/\delta)}(\frac{\sqrt{m}}{\sigma_1} + \frac{\sqrt{m_{selected}}}{\sigma_2})).$$

The proof is in the appendix. Notice that this is only used for illustrating the amplification effect γ that is not present in PATE. The actually numerical calculation of ϵ is tighter using analytical moments accountant [29].

Table 1. Utility and privacy of semi-supervised student model

Dataset	Methods	#Queries	ϵ	Acc.	NP Acc.
MNIST	LNMAX	1000	8.03	98.1%	99.2%
	GNMAX	286	1.97	98.5%	
	Ours	735	**0.47**	98.8%	
SVHN	LNMAX	1000	8.19	90.1%	92.8%
	GNMAX	3098	4.96	91.6%	
	Ours	2939	**0.49**	91.6%	
CIFAR-10	GNMAX		$\leq 50\%$		80.5%
	Noisy SGD		4	70%	
	Ours	3877	**2.92**	70.8%	

Table 2. Ablative results of iterative training on SVHN dataset.

Iteration	kNN Acc.	retrain CNN	#Queries	ϵ
1	82.5%	86.6%	1022/3000	0.49
2	94.41%	91.6%	1917/3000	

4. Experiments

In this section, we demonstrate our Private-kNN for its data efficiency with character recognition tasks such as MNIST [17] and SVHN [22]. We show that our model achieves the same accuracy with only 10% of the privacy cost used in state-of-the-art (SOTA) methods such as PATE [24]. We also leverage the general vision tasks where data splitting for PATE is the bottleneck. CIFAR-10 [16] as a general object recognition task is investigated across the DP methods. More specifically, we focus on two realistic setting vision problems, namely face attribute classification on CelebA [18] and body attribute classification on Market1501 [32], which is the first to show that our method can facilitate to realistic multi-label classification tasks.

4.1. MNIST and SVHN Evaluation

MNIST and SVHN are two common datasets to measure the utility and privacy performance of differential private models [24, 25]. We evaluate Private-kNN using the same setup of private dataset and the model architecture as in PATE [24, 25]. On MNIST, the training set is reserved as the private dataset, half of the testing set acts as unlabeled student training data, and the remaining part is for real testing. For SVHN, the extended data, together with training data, are regarded as private data. Among the $26k$ testing set, $25k$

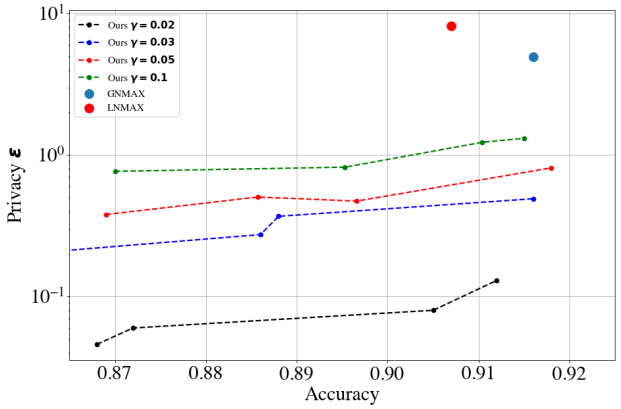

Figure 4. Tradeoff between utility and privacy for Private-kNN on SVHN. In this figure, different curve are generated with different sampling ratio γ. In each curve, we set different query number for student, and compute the total privacy and accuracy at test set. $\sigma_1 = 240, T = 480, \sigma_2 = 60, k = 800$. We also plot the results reported in PATE. It shows that the privacy cost of our model could achieve nearly two order of magnitude smaller privacy with better accuracy.

acts as publicly unlabeled student data for query and self-supervised training, where the remaining $1k$ is for testing. We defer the detailed information of model architectures in appendix and report their non-private baselines in Table 1.

As illustrated in the method, we conduct initial round kNN classification using a handcrafted feature — histogram of oriented gradients(HOG). Then we apply self-supervised training (e.g.[21, 31]) with the pseudo-labeled data from kNN for better feature representation learning.

MNIST: In our method, the privacy cost is accumulated over 1000 queries of 2 iterations. We set the number of neighbors $k = 300$, $\sigma_1 = 75$ for screening, threshold $T = 180$ and $\sigma_2 = 25$ for aggregation, and fix the sub-sampling ratio $\gamma = 0.15$. In the initial iteration, the accuracy of the privately aggregated kNN model based on HOG feature is 92.1%. Then a student model is trained on the 735 answered queries with pseudo labels and VAT regularization, which achieves accuracy 98.8%. In Table 1, comparing to PATE of Laplace mechanism "LNMAX" and Gaussian mechanism "GNMAX", our method achieves significantly better accuracy-privacy trade-off. For instance, when we control the same number of queries between "GNMAX" and ours, Private-kNN achieves similar accuracy as 98.8% over 98.5%, but much better privacy cost as $\epsilon = 0.47$ compared to $\epsilon = 1.97$ of "GNMAX". More surprisingly, with a strict privacy cost of $\epsilon = 0.47$, our method shows only 0.4% deficit to the non-private model performance 99.2%.

SVHN: As shown in Table 2, we run our model for two iterations with hyper-parameters $k = 800, T = 480, \sigma_1 = 200, \sigma_2 = 60$ and $\gamma = 0.03$. In the first iteration, kNN with HoG feature provides 82.5% accuracy on 1022 answered queries. By retraining a CNN with the queried labels, it improves to 86.6%. In the second iteration, another 3000 queries are conducted via kNN, and 1917 queries are returned. KNN accuracy is evaluated on the selected queries, which passed the noisy screening check, whereas the retrain CNN is evaluated on the public testing set after self-supervised training and achieves 91.6% accuracy. These procedures can be iterated many times, where we empirically observe that two rounds can bring the converged performance. In total, we spend the privacy cost on 6000 samples for noisy screening and noisy aggregation with $2919(1022 + 1917)$ samples.

Table 1 shows the comparison to "GNMAX" and "LNMAX". Both "GNMAX" and ours achieve better privacy accuracy trade-off than "LNMAX". Though the number of queries in "LNMAX" is only 100, the privacy cost is as high as 8.19. This is mainly from the inefficiency of the Laplace mechanism compared to Gaussian mechanism, as Gaussian mechanism shows 30 times more queries with half of the privacy cost (4.96 over 8.19). Further comparing our method with "GNMAX", with the similar number of queries and exactly the same accuracy, we achieved 0.49 privacy cost, which is significantly smaller than 4.96 from "GNMAX". Notice that privacy cost below 1 indicates an excellent system which is ready for *practical* applications.

Figure 4 shows by varying sampling ratio γ, the privacy cost ϵ changes with respect to the number of queries. "GNMAX" and "LNMAX" are also compared. In the figure, all of our methods are advantageous, i.e. consistently lower privacy cost than those two spots of "GNMAX" and "LNMAX". Further exploring different levels of γ, we observe that all the curves are mostly flat, which indicates that when pushing accuracy high, the increase of privacy cost is marginal. Moreover, it shows that with different sampling ratio, our method can achieve different level of privacy cost. Across the large range of sampling ratio (0.02 to 0.1), we can push all the performance between 91% to 92%, which is at the same level of "GNMAX" and "LNMAX", while with an order of magnitude lower privacy cost.

4.2. CIFAR-10 Evaluation

CIFAR-10 is a general objection classification task, where the PATE model is hard to apply as the data partitioning results in limited training data for each teacher model. For instance, each teacher model is assigned only 200 data if we partition the training set into 250 teachers, which is far from sufficient to train a deep neural network. For our experimental setting, we split total $60k$ data into three parts: $30k$ is treated as private data, $29k$ is for unlabeled public data, and $1k$ for testing.

Regarding this dataset, a competitive method, termed Noisy-SGD [1], achieved accuracy 70% and $\epsilon = 4$ when $\delta = 10^{-5}$ as shown in Table 1 CIFAR-10. In the Noisy-SGD setting, CIFAR-100 is leveraged to pre-train a model. For

Table 3. Real sensitive dataset evaluation on CelebA [18] and Market1501 [32], we set $\tau = 10$ for both GNMAX and ours. T is the number of teachers in teacher ensemble model. We compare different methods under high privacy and low privacy regime. $\delta = 10^{-6}$ for CelebA and $\delta = 10^{-5}$ for Market.

Dataset	Methods	Parameter				#Queries	ϵ	Acc.	NP Acc.
		T	k	σ	γ				
CelebA	GNMAX	300	-	150	-	600	7.72	85.0%	89.5%
	GNMAX	800	-	300	-	500	3.31	84.4%	
	Ours	-	800	50	0.05	800	1.24	85.2%	
	Ours	-	800	100	0.10	800	1.20	84.9%	
Market1501	GNMAX	300	-	100	-	800	13.41	86.8%	92.1%
	GNMAX	300	-	250	-	80	1.41	85.6%	
	Ours	-	300	100	0.05	1200	0.67	88.8%	
	Ours	-	300	100	0.10	1200	1.38	89.2%	

fair comparison, we also use the CIFAR-100 model as a pre-trained model for each teacher in PATE [25] and extract the initial feature with it for the Private-kNN. The latter iterative updating of the student model remains the same. For PATE performance, we notice that after model aggregation, it is below 50% even after we set $\epsilon \gg 10$.

In our implementation, with the initial CIFAR-100 extracted feature, the Private-kNN aggregator answers 3877 over total 18000 queries from the public domain. We set neighbor $K = 300, T = 210, \sigma_1 = 85, \sigma_2 = 20$, sampling ratio $p = 0.2$ and adopt the same model architecture as [1]. The model architecture contains three convolutional layers with $32, 64, 128$ filters in each convolution layer. The non-private baseline of this model reaches 80.5% accuracy when trained with $30k$ private data, whereas SOTA models present over 10% higher accuracy. The reason of not leveraging the SOTA models in this experiment is because, for fair comparison to Noisy-SGD, we aim to emphasize the privacy-utility trade-off, but not the best utility. Our method achieves an accuracy of 70.8% with privacy cost $\epsilon = 2.92$, which thoroughly outperforms Noisy-SGD.

Notice that the privacy cost in Noisy-SGD is spent on every parameter of the network; Thus, their retraining only involves the fully connected layers. Another difference is, we assume there exists unlabeled auxiliary data in public domain while Noisy-SGD [1] directly train a private model with $50k$ private data. Comparing to Noisy-SGD, our Private-kNN is indeed model agnostic, no restriction on network structure or optimization methods for retraining a student model, whereas clipping gradient in Noisy SGD may result in unstable optimization.

4.3. Noisy Screening for Less Privacy Cost

Screening and private voting are the core components of privacy guarantee. The purpose of screening is to filter out queries where there is no consensus among the votes. The privacy cost on screening is the major expense as reported in PATE [24] since we need to pay privacy cost for each query.

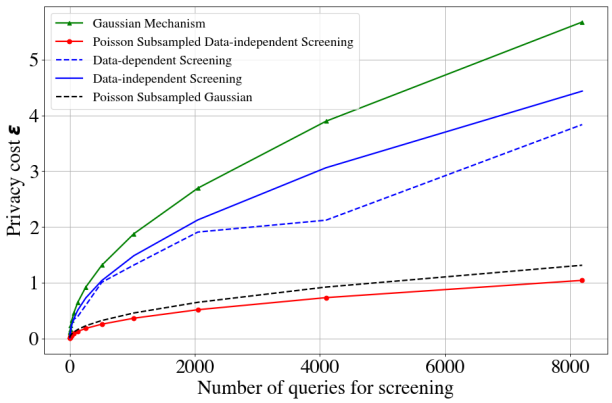

Figure 5. The privacy cost of answering 8192 queries with five randomized algorithms in the noisy screening process. The green line is the strong composition of Gaussian Mechanism used in PATE, the black dash line shows the privacy cost of Poisson subsampled Gaussian Mechanism after 8192 rounds' composition. The blue line is the ϵ of strong composition of data-independent screening and the blue-dash line is the strong composition of data-dependent screening. The red line is the Poisson subsampled data-independent screening $\mathcal{M}_s$. The sampling ratio $\gamma = 0.25, \sigma_1 = 85, k = 300$.

We investigate different private screening methods by exploring their privacy cost with respect to a different number of queries. In Figure 5, each screening algorithm is required to answer 8192 queries on the CIFAR-10 dataset, and the cumulative privacy cost is plotted along the y-axis. We use the HoG feature in the initial iteration for our Private-kNN and set the sampling ratio of $\gamma = 0.25$. The noisy scale $\sigma_1 = 85$, threshold $T = 210$ and $k = 300$ is used for all screening methods.

The green line describes the privacy cost of Gaussian mechanism applied by PATE[25], serving as our baseline. It achieves $\epsilon = 5.67$ after privately screening 8192 queries. The black dash line demonstrates the privacy amplification by Poisson sampling [33] with the same Gaussian mechanism. The privacy cost improves to 1.313. Even though, the

original data-splitting setting in PATE prevents it to benefit from sub-sampling. The red line shows our data-independent screening method composed of Poisson sampling achieves $\epsilon = 1.04$. The blue line and the blue dash line show the result of incorporating our data-independent and data-dependent analysis of screening into PATE. It improves ϵ from 5.67 to 4.43 with the data-independent screening and 3.83 with the data-dependent screening.

When compared to the black-dash line (sub-sampled Gaussian), our method saves 26% privacy budget with the same screening results. Our method allows to answer more queries from the public domain, which is of essential importance, especially when the training task itself is tough. For example, employing self-supervised training with CIFAR-10 requires at least 4000 ground-truth labeled data [31]. Then, the minimum number of queries demands at least 10000, since empirically, more than 50% data fails to pass the screening check. Our advantageous privacy cost 1.04 makes it a practical solution for private training with more difficult machine learning tasks.

4.4. Real Private Datasets Evaluation

We show that our Private-kNN is a practical framework that indeed can apply to real private datasets, i.e., face attribute classification from CelebA [18] and body attribute classification from Market1501 [32]. We aim to develop an attribute classification model, where the adversary is hard to detect whether one particular image has been used in training set with high probability. Both of the datasets target the human or face related tasks, where identity is crucial privacy to be preserved. Notice that they are multi-label classification tasks other than binary classification, which are more challenging. To reduce the privacy budget of multi-label tasks, we apply a τ approximation method where the basic idea is that, each neighbor could at most vote for τ attributes, or their total votes will be clipped to τ. The detailed definition and privacy guarantee can be found in Appendix. In our setting, we do not conduct noisy screening for multi-label classification because it is hard to guarantee all the labels within one query pass the screen.

CelebA is a large-scale face attribute dataset with more than $220k$ celebrity images, each with 40 attribute annotations. According to data splitting, we take the $160k$ training data as private data. From the $60k$ testing data, depending on the volume to be queried, i.e. 600 queries, the rest 59400 images are automatically regarded as testing. The non-private baseline is 89.5% trained via a Resnet50m structure. We apply PATE as another baseline. Since each image have 40 attributes, the global sensitivity grows as large as the dimension of attributes. We apply τ-approximation method to limit the range of global sensitivity and also consider the trade-off induced by the different τ. In Table 3, by choosing the parameters, when the privacy cost is smaller than "GNMAX", we achieve clear better accuracy of 85.18% compared to 84.4%. When the accuracy is at the same level around 85%, our method achieves significantly lower privacy cost 1.20 compared to 7.72 of "GNMAX".

Market1501 contains 1501 identities and 32668 images, where each image has 30 attributes. We split original training set for private data and validation set as unlabeled public data, performance is evaluated on original testing set. In this task, data-splitting is stressful. The total private data contains only 750 identities. For PATE, to guarantee the teacher models' independence, we need to partition the private data with respect to the identities. A meaningful privacy cost requires sufficient many teacher models, i.e., $K = 300$. With such many partitions of the private data, each teacher is trained with around 40 images from 2 identities, and the non-private accuracy of each teacher is only 71%.

Shown in Table 3, our method is able to answer 1200 queries compared to 80 in "GNMAX" where two methods achieve similar privacy cost 1.414 and 1.377. The significantly more queries lead to performance boost as 89.18% compared to GNMAX 85.61%. To push up the performance for "GNMAX"(i.e., from 85.6% to 86.8%), we tune the privacy-utility trade-off and the privacy cost goes high up to 13.41, which prevents the trade-off from improving performance further. We provide a relative close trade-off, accuracy 88.8%, and privacy $\epsilon = 0.67$, both of which are far better than the "GNMAX". The detailed utility and privacy trade-off can be found in the appendix, which demonstrates the consistent advantages of our method in real private tasks.

5. Conclusions

In this work, we propose a data-efficient privately releasing of k nearest neighbor framework, termed Private-kNN, to overcome the limited private data to train deep neural networks in vision applications. A new Rényi differential privacy analysis for noisy screening procedure is proposed, which allows our model to answer 10 times more queries compared to other DP models such as PATE. Extensive experiments are conducted across five vision benchmarks, showing that our method achieves comparable or better accuracy than PATE while saving more than **90%** privacy cost. Specifically, the two realistic identity related computer vision tasks demonstrate that our Private-kNN achieves high utility with practical DP guarantees.

6. Acknowledgement

We thank reviewers and meta-reviewers for their valuable feedback. YZ and YW were supported by the start-up grant of YW at UCSB Computer Science and generous gifts from Amazon Web Services and NEC Labs.

References

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318. ACM, 2016. 1, 3, 6, 7

[2] B. Balle, G. Barthe, and M. Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. In *Preprint*, 2018. 3

[3] R. Bassily, A. Smith, and A. Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *Foundations of Computer Science (FOCS-14)*, pages 464–473. IEEE, 2014. 1

[4] M. Bun, C. Dwork, G. N. Rothblum, and T. Steinke. Composable and versatile privacy via truncated cdp. In *STOC-18*, 2018. 11

[5] M. Bun and T. Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*, pages 635–658. Springer, 2016. 2, 3

[6] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, and D. Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 267–284, Santa Clara, CA, Aug. 2019. USENIX Association. 1

[7] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate. Differentially private empirical risk minimization. *The Journal of Machine Learning Research*, 12:1069–1109, 2011. 1

[8] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967. 2

[9] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. 2005. 4

[10] C. Dimitrakakis, B. Nelson, A. Mitrokotsa, and B. I. Rubinstein. Robust and private bayesian inference. In *Algorithmic Learning Theory*, pages 291–305. Springer, 2014. 1

[11] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography*, pages 265–284. Springer, 2006. 1, 2

[12] C. Dwork, M. Naor, O. Reingold, G. N. Rothblum, and S. Vadhan. On the complexity of differentially private data release: efficient algorithms and hardness results. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 381–390. ACM, 2009. 13

[13] C. Dwork, G. N. Rothblum, and S. Vadhan. Boosting and differential privacy. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 51–60. IEEE, 2010. 3

[14] M. Hardt and G. N. Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 61–70. IEEE, 2010. 13

[15] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011. 1

[16] A. Krizhevsky. Learning multiple layers of features from tiny images. In *Tech Report*, 2009. 5

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, 1998. 5

[18] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 5, 7, 8

[19] I. Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017. 2

[20] I. Mironov. Rényi differential privacy. In *Computer Security Foundations Symposium (CSF), 2017 IEEE 30th*, pages 263–275. IEEE, 2017. 2

[21] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018. 4, 6

[22] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop*, 2011. 5

[23] K. Nissim, S. Raskhodnikova, and A. Smith. Smooth sensitivity and sampling in private data analysis. In *ACM symposium on Theory of computing (STOC-07)*, pages 75–84. ACM, 2007. 3, 13

[24] N. Papernot, M. Abadi, U. Erlingsson, I. Goodfellow, and K. Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *ICLR*, 2017. 1, 3, 5, 7

[25] N. Papernot, S. Song, I. Mironov, A. Raghunathan, K. Talwar, and Ú. Erlingsson. Scalable private learning with pate. *arXiv preprint arXiv:1802.08908*, 2018. 1, 2, 3, 4, 5, 7, 12, 13

[26] M. Park, J. Foulds, K. Chaudhuri, and M. Welling. Variational bayes in private settings (vips). *arXiv preprint arXiv:1611.00340*, 2016. 1

[27] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. In *Security and Privacy (SP), 2017 IEEE Symposium on*, pages 3–18. IEEE, 2017. 1

[28] S. Song, K. Chaudhuri, and A. D. Sarwate. Stochastic gradient descent with differentially private updates. In *Conference on Signal and Information Processing*, 2013. 1, 3

[29] Y.-X. Wang, B. Balle, and S. Kasiviswanathan. Subsampled Rényi Differential Privacy and Analytical Moments Accountant . In *AISTATS'19, to appear.*, 2018. 3, 5

[30] Y.-X. Wang, S. Fienberg, and A. Smola. Privacy for free: Posterior sampling and stochastic gradient monte carlo. In *International Conference on Machine Learning (ICML-15)*, pages 2493–2502, 2015. 1

[31] Q. Xie, Z. Dai, E. Hovy, M.-T. Luong, and Q. V. Le. Unsupervised data augmentation. *arXiv preprint arXiv:1904.12848*, 2019. 5, 6, 8

[32] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. 5, 7, 8

[33] Y. Zhu and Y.-X. Wang. Poission subsampled rényi differential privacy. In *International Conference on Machine Learning*, pages 7634–7642, 2019. 3, 5, 7, 11

Appendices

A. Appendix

In this supplementary, we provide the proofs of Theorem 7 and Theorem 8. Moreover, we present a discussion of utility and privacy trade-off in Market1501 dataset. Later, we describe the τ-approximation approach to reduce the global sensitivity in multi-label tasks.

B. Proofs of Theorem 7 and 8

Theorem 9 (RDP of "Noisy Screening", Restatement of Theorem 7)**.** *Let $\mathcal{M}_s$ be a randomized algorithm for noisy screening procedure with a predefined Gaussian noise scale σ_1 and the threshold T. Then $\mathcal{M}_s$ obeys RDP with*

$$\epsilon_{\mathcal{M}_s}(\alpha) = \max_{(p,q)\in\mathcal{S}} \frac{1}{\alpha-1}\log\left(p^\alpha q^{1-\alpha} + (1-p)^\alpha(1-q)^{1-\alpha}\right).$$

where $\mathcal{S}$ contains the following "pairs":

$$\left(\mathbb{P}[\mathcal{N}(t,\sigma_1^2)\geq T], \mathbb{P}[\mathcal{N}(t+1,\sigma_1^2)]\geq T]\right),$$
$$\left(\mathbb{P}[\mathcal{N}(t,\sigma_1^2)\geq T], \mathbb{P}[\mathcal{N}(t-1,\sigma_1^2)]\geq T]\right)$$

for all integer $\lceil k/c\rceil \leq t \leq k$. The bound can be computed in time time $O(k)$.

Proof. For a given query x, set $n^*(x)$ be the vote count of the plurality and p, q denote the probability of x passes the noisy screening procedure with neighboring private datasets X, X' respectively. The output space of both $\mathcal{M}_s(X)$ and $\mathcal{M}_s(X')$ is $\{\top, \perp\}$, where $\top$ indicates x passes noisy screening process, and vice versa. Then $\mathcal{M}_s(X)$ and $\mathcal{M}_s(X')$ satisfy the Bernoulli distribution with the parameter p, q respectively.

By definition of Renyi Differential privacy and the Renyi Divergence of two Bernoulli distributions:

$$\epsilon_{\mathcal{M}}(\alpha) = \sup_{X,X' \text{are neighbors}} \frac{1}{\alpha-1}\log E_q(\frac{p}{q})^\alpha$$
$$= \sup_{X,X' \text{are neighbors}} \frac{1}{\alpha-1}\log\left(p^\alpha q^{1-\alpha} + (1-p)^\alpha(1-q)^{1-\alpha}\right)$$

The key of deriving RDP is to maximize over the two neighboring datasets. We make two observations. First, the notion of datasets X, X' are completely captured by their max votes t, t'. By the fact that the two datasets differ by at most one individual, $|t-t'| \leq 1$. In other word, to enumerate all neighboring datasets, it suffices to consider integer t, t' from $\lceil k/c\rceil$ to k such that $t' \in \{t-1, t+1\}$. Second, p, q can be directly calculated from t and t' respectively: $p = 1 - \text{cdf}(\frac{T-t}{\sigma})$ and $q = 1 - \text{cdf}(\frac{T-t'}{\sigma^1})$. Where cdf denotes the CDF of a standard normal random variable. Note that p monotonically increases as t increases.

These two observations ensure that we can calculate the RDP $\epsilon_{\mathcal{M}}(\alpha)$ for any fixed α in time $O(k)$. □

Where is t^* in practice? In practice, the worst pair of neighboring datasets occur either around $\max\{\text{votes}\} = T$ or around the boundaries when $\max\{\text{votes}\} = k$ (the largest possible) or $\max\{\text{votes}\} = \lceil k/c\rceil$ (the smallest possible due to pigeon hole principle).

In Figure 6, we plot the data-independent RDP of "Noisy screening" of all possible plurality. The plurality n^* ranges from $\lceil k/c\rceil$ to k and we set $k = 300$, threshold $T = 210, \sigma_1 = 85$. The x-axis is the RDP order α ranges from 1 to 50, the y-axis is the range of possible n^*, and we plot the corresponding RDP $\epsilon(\alpha)$ with the fixed α, n^*. The red curve shows the $\epsilon(\alpha)$ when $\max\{\text{votes}\} = T$, and we plot the red-dash line to view its exact RDP value more clearly. This figure shows that when α is small (below 50), the worst case of data-independent RDP is when $\max\{\text{votes}\} = T$. In Figure 7, we pick 5 curves from Figure 6 to further compare the RDP under the different choices of n^*. It shows that when $\alpha \leq 80$, the maximum data-independent $\epsilon(\alpha)$ is achieved when $n^* \approx T$, and when $\alpha \geq 80$ the $\epsilon(\alpha)$ is maximized when $n^* = k$. So for the upper bound of RDP of noisy screening, we only need to evaluate $\mathcal{M}_s$ for several neighboring datasets. In Figure 8, we plot the privacy cost of answering 8192 queries with 5 different data-independent analysis (from Figure 7 in the noisy screening procedure. The red line shows the privacy cost when $n^* = T$, and it's on the top the five curves which verifies our conjecture: the worst-case appears around $n^* = T$ or $n^* = k$. In the first 10 iterations, $n^* = k$ achieves the maximum. From Lemma 3, we know $\epsilon = \min_\alpha \epsilon(\alpha) + \frac{\log 1/\delta}{\alpha-1}$. When the number of iteration is small, the total privacy cost ϵ is minimized when α is large. As the number of iterations keeps increasing, ϵ is minimized when α is small. This phenomenon explains that the maximum data-independent privacy cost could be caused by several choices of n^*, which maximizes $\epsilon(\alpha)$ in the different range of α. However, $\epsilon_{\mathcal{M}}(\alpha)$ is not always maximized when $\max$ votes $= k$or$T$. For a larger α,the max $\epsilon_{\mathcal{M}}(\alpha)$ is attained when $n^* = k$. Check these two cases can give us a fast approximation of $\epsilon_{\mathcal{M}}(\alpha)$.

Theorem 10 (Asymptotic scaling, formal version of Theorem 8)**.** *Assume parameter $\gamma, \sigma_1, \sigma_2, \delta$ are chosen such that $\gamma < 0.1$, $\sigma_1 \geq \sqrt{5}$, $\sigma_2 \geq 2\sqrt{5}$, and moreover $\frac{4\log(1/\delta)\sigma_1^2}{\gamma^2\left(\min\{\sigma_1^2,\sigma_2^2\}\log^2(1/\gamma)-2\right)} \leq m \leq \frac{\sigma_1^2\log(1/\delta)}{3\gamma^2}$, $m_{select} \leq \frac{\sigma_2^2\log(1/\delta)}{6\gamma^2}$. Then, the end-to-end Private-KNN algorithm that processes all m public data points using with noise σ_1, σ_2 and sampling ratio γ obeys (ϵ, δ)-DP, with*

$$\epsilon = 20\gamma\sqrt{\log(1/\delta)}\left(\frac{\sqrt{m}}{\sigma_1} + \frac{\sqrt{m_{selected}}}{\sigma_2}\right).$$

Proof. The algorithm that process all m data points is an adaptive composition of two steps. In the first step, we release the $\{\top, \perp\}$ with the "noisy screening". In the second

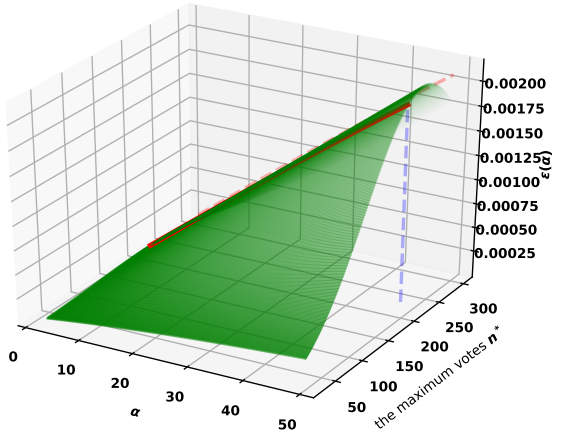

Figure 6. Searching the worst case for data-independent RDP of "Noisy Screening".

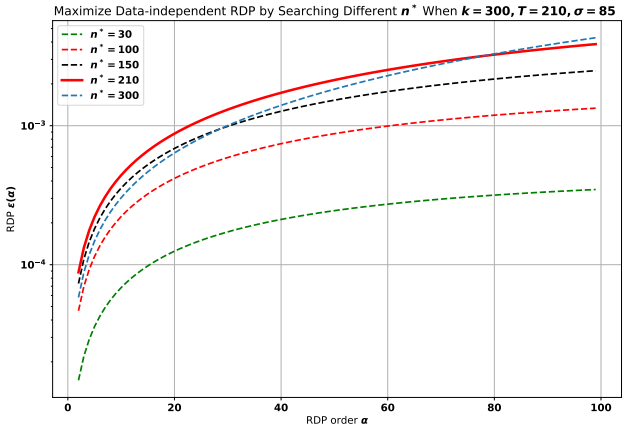

Figure 7. An example for data-independent RDP of " Noisy Screening" with different plurality.

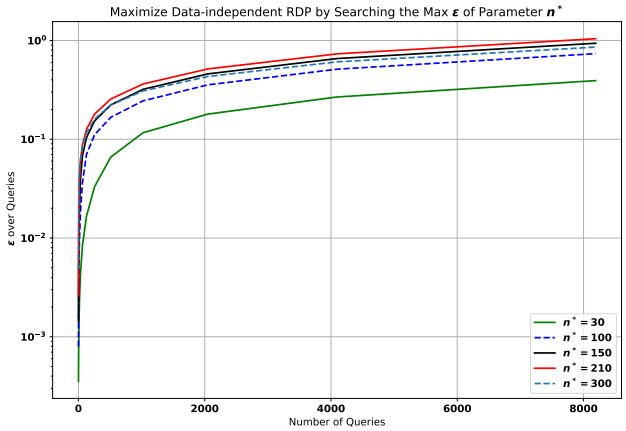

Figure 8. Privacy cost of answering 8192 queries with different data-independent RDP of "Noisy Screening". The sampling ratio $\gamma = 0,25, \sigma_1 = 85, k = 300$. n^* is the fixed max votes.

step, we release the "noisy max" for those that passes the screening rule. In both steps, the randomized procedure is amplified by Poisson subsampling. As a result, both has an RDP that is upper-bounded by the Poisson subsampled-gaussian mechanism.

The following is an asymptotic scaling of the the subsampled Gaussian mechanism.

Lemma 11 (Theorem 11 of [4]). *Let the global ℓ_2 sensitivity be Δ. Assume $\gamma \leq 0.1$, $\sigma/\Delta \geq \sqrt{5}$, then the Poisson-subsampled Gaussian mechanism obeys $(\alpha, \frac{6\gamma^2\Delta^2}{\sigma^2})$-RDP for all $\alpha \leq \frac{\sigma^2 \log(1/\gamma)}{2}$.*

The above lemma is implied by the original statement about tCDP [4] for randomly selecting a subset of a fixed size γn, because (1) tCDP is an upper bound of RDP; (2) the exact RDP calculation for the Poisson-subsampled Gaussian mechanism matches the RDP lower bound of the (Random subset) subsampled Gaussian mechanism [33, Proposition 10].

The global sensitivity of the Gaussian mechanism in the "noisy screening" step is 1 because we are releasing only $\max\{\text{Votes}\}$, while it is 2 in the Gaussian mechanism for releasing the Votes — the histogram. Check that the stated assumptions on $\gamma, \sigma_1, \sigma_2$ satisfy the conditions above.

By the composition rule of Renyi Differential Privacy in Lemma 5 which establish that the end-to-end algorithm obeys RDP with

$$\epsilon(\alpha) \leq \frac{6\gamma^2 m\alpha}{\sigma_1^2} + \frac{12\gamma^2 m_{\text{select}}\alpha}{\sigma_2^2}.$$

for all α in the range that are permitted by Lemma 11.

Finally, by Lemma 3, we can convert RDP to (ϵ, δ)-DP with

$$\epsilon = \alpha\left(\frac{6\gamma^2 m}{\sigma_1^2} + \frac{12\gamma^2 m_{\text{select}}}{\sigma_2^2}\right) + \frac{\log(1/\delta)}{\alpha - 1}.$$

Choose $\alpha = 1 + \frac{\sqrt{\log(1/\delta)}}{\sqrt{\frac{6\gamma^2 m}{\sigma_1^2} + \frac{12\gamma^2 m_{\text{select}}}{\sigma_2^2}}}$ we get that:

$$\epsilon = \frac{6\gamma^2 m}{\sigma_1^2} + \frac{12\gamma^2 m_{\text{select}}}{\sigma_2^2} + 2\gamma\sqrt{\log(\frac{1}{\delta})\left(\frac{6m}{\sigma_1^2} + \frac{12m_{\text{select}}}{\sigma_2^2}\right)}.$$

The proof is complete by checking that under our assumption m, the second term always dominates and the assumption on α in Lemma 11 no matter that m_{select} turns out to be. □

C. The utility and privacy trade-off on Market1501 dataset

Figure 9 shows the utility and privacy trade-off of PATE and ours by varying sampling ratio γ, the noisy scale σ_1 and the number of queries. For GNMAX in PATE, to push

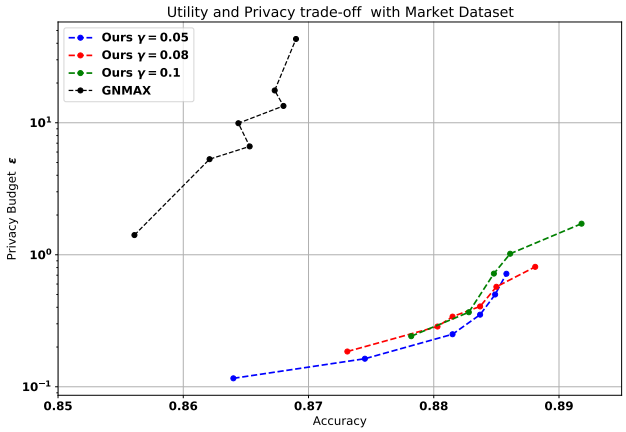

Figure 9. Trade-off between utility and privacy for PATE and ours on Market1501 dataset.

the accuracy from 86.80% to 86.90%, we need to increase the privacy budget from 13.41 to 43.14. In the low privacy cost regime, our method achieves accuracy 87.82% with privacy budget 0.2416. In the high privacy cost regime, our algorithm achieves 89.18% with $\epsilon = 1.72$ compared to $\epsilon = 5.298$ and accuracy =86.21% in PATE. Further by checking the same accuracy, i.e., 86.5% for both "GNMAX" and ours with $\gamma = 0.05$, our privacy cost is 0.116 while "GNMAX" is 6.62. Indeed, more than 90% privacy budget is saved from the baseline method.

Privacy and utility trade-off of GNMAX In all experiments of GNMAX, we set the number of teachers with respect to the performance of each teacher. For example, if we set the number of teachers to be 600, then the average non-private accuracy of each teacher is around 76%. Since every partitioned data should not be overlapped with each other regard to the identity, the total identity is 750, and $T = 600$ is the maximum number GNMAX algorithm can afford. If we set a small T for GNMAX, e.x. $T = 100, \sigma_1 = 40$, then the privacy loss of GNMAX achieves $\epsilon = 13.22$ even it only answers 80 queries.

D. Applying Private-kNN to multi-label classification tasks

So far, we have been primarily working with multi-class classification tasks where the global sensitivity of the voting results of the nearest neighbors are naturally bounded. But for multi-label tasks, this is no longer true. Potentially, for a problem with c-labels, any neighbor can potentially vote on all c-labels, which makes the naïve noisy-adding mechanisms inefficient. We propose to fix it by a "clipping" heuristic that limits contribution of every label to at most τ.

Definition 12. (τ approximation) For a traditional classification task, the global sensitivity of our model in the noisy aggregation process is 2 from Theorem 9. However, consider the more general multi-label task in vision, e.x. Facial attribute classification task, where one face image could have at most 40 attributes and the global sensitivity will increase to 80. To limit the global sensitivity in the multi-label task, we introduce the τ-approximation method, where the basic idea is that each neighbor could vote no more than τ attributes. For simplicity's sake, we only consider binary multi-label tasks here. In a multi-label task, the vote of neighbor j upon query x is $f_j(x) \in \mathcal{N}^c$ now becomes a $c-$way vector. To impose τ approximation on it, we apply

$$\hat{f}_{j,i} = f_{j,i} \cdot \min(\frac{\tau}{|f_j(x)|}, 1), i \in [1, c],$$

with $|f_j(x)|$ the L_1 norm of original neighbor j's voting and $\hat{f}_j$ the neighbor j' prediction upon x with τ approximation.

Theorem 13 below provides a practical privacy bound to guide the analysis for multi-label classification task.

Theorem 13. *Let $\mathcal{M}_\tau$ be a randomized algorithm for a multi-label task with τ-approximation method, the global sensitivity of $f(x)$ here is $2 \cdot \tau$, then we have for integer $\alpha \geq 2$,*

$$D_\alpha(\mathcal{M}_\tau(X)||\mathcal{M}_\tau(X')) = \frac{\alpha \cdot \tau}{\sigma_1^2}$$

Regression problems. Similar clipping tricks can be applied to regression problems so private-kNN applies. We can also use median, rather than the mean. Careful experimental evaluation on regression problems are left as a future work.

E. Architecture of networks

We plot the network architecture of MNIST in Table 4. The MNIST model contains two convolutional layers with max-pooling and two fully connected layers with ReLUs. For the SVHN task, Table 5 shows that the SVHN model stacks seven convolutional layers with two fully connected layers, which replicates the experimental setup as in [25]. The source code of MNIST and SVHN experiments and a Pytorch implementation of [25] are available on Github.[3]

Table 4. Network architecture of MNIST task

Conv	64 filters of size 5×5
Max pool	2×2
Conv	128 filters of size 5×5
Max pool	2×2
FC	$(384, 192, 10)$

[3] https://github.com/jeremy43/Private_kNN

Table 5. Network architecture of SVHN task

Conv	96 filters of size 3×3
Conv	96 filters of size 3×3
Conv	96 filters of size 3×3
Conv	192 filters of size 3×3
Conv	192 filters of size 3×3
Conv	192 filters of size 3×3
Conv	192 filters of size 5×5
FC	$(192, 192, 10)$

F. More discussion about data-dependent noisy-screening

Noisy-Screening vs. Sparse Vector Technique. The noisy screening is closely related to the Sparse Vector Technique [12, 14] (SVT) that screens a sequence of online queries $f_1, f_2, ...$ with global sensitivity 1 and output $\{\top, \perp\}$ with the hope of approximately selecting those queries with value greater than a threshold T and essentially paying only the privacy loss for those that are selected.

The key steps of an SVT include adding Laplace noise to the threshold and also adding Laplace noise to $f_i(x)$ when deciding whether to output $\top$ or $\perp$. When the large majority of the queries have either $\top$ or $\perp$ with sufficiently high margin from the threshold T, then SVT is able to handle an exponentially large set of queries.

"Noisy-screening" is different in two ways. First, it does not aim at "calibrating noise to stability" to achieve a pre-defined privacy budget. Instead the version that we used pays the same amount for every query. Second, we can use Gaussian mechanism on $f_i(x)$ while keeping the threshold T unchanged. This method at a glance does not resemble SVT at all because it does not adapt to the input sequence, and pay only an amount proportional to the $\sqrt{\min\{\#\text{ of }\perp, \#\text{ of }\top\}}$ as in SVT.

That said, the data-dependent RDP of "Noisy-screening" is in fact a lot more closely related to SVT. If a query f_i obeys that either $f_i(x) \gg T$ or $f_i(x) \ll T$, then the data-dependent RDP is going to be exponentially smaller than that is coming from the Gaussian mechanism. Directly composing the data-dependent RDP will lead to qualitatively the same behavior as SVT.

For example, for a sequence of queries where SVT can answer exponentially many without using up a budget of (ϵ, δ), we can answer the same sequence with "noisy-screening" while paying a "data-dependent" privacy loss that is likely to be smaller than (ϵ, δ).

Consider another example, if the sequence of queries are close to $f_i(x) = T$, then the data-dependent calculations for "noisy screening" will arrive at about the same privacy losses as the data-independent counterpart. Similarly, SVT will also stop within just a few rounds because essentially it pays every other iteration on average.

In summary, the data-dependent RDP calculations of "noisy screening" can be thought of as a versatile alternative of SVT, when satisfying a fixed pre-specified privacy budget is not too important and when we do not have to reveal the final privacy loss that is realized (because its value depends on the data). This allows us to use a more concentrated Gaussian noise, and to take advantage of the RDP for a tighter composition.

Both limitations can be resolved by privately releasing the data-dependent RDP using smooth sensitivity [23] as in what was proposed in the appendix of [25]. Details of this procedure and how "noisy screening" compares to SVT in general is left as a future direction of research.

Open problem: Data-dependent RDP of subsampled mechanism. Privacy-amplification by subsampling is not compatible with data-dependent RDP because implicitly, the amplification is coming from the fact that for any subset that is selected, the same RDP bound holds.

A trap is to amplify the data-dependent RDP calculated through the specific sample that is chosen. This is because value probably cannot hold for other subsets.

It remains an open problem how to correctly calculate the data-dependent RDP for a subsampled mechanism. The exact calculation would require enumerating over all subsets and calculating their corresponding data-dependent RDP.