Towards Training Robust Private Aggregation of Teacher Ensembles Under Noisy Labels

Qiuchen Zhang*, Jing Ma*, Jian Lou*, Li Xiong* and Xiaoqian Jiang†
*Department of Computer Science, Emory University, Atlanta, GA
Email: {qiuchen.zhang, jing.ma, jian.lou, lxiong}@emory.edu
†University of Texas Health Science Center at Houston, Houston, TX
Email: xiaoqian.jiang@uth.tmc.edu

2020 IEEE International Conference on Big Data (Big Data) | 978-1-7281-6251-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/BigData50022.2020.9378234

***Abstract*—Deep learning models trained on large-scale data have achieved encouraging performance in many real-world tasks. Meanwhile, publishing those models trained on sensitive datasets, such as medical records, could pose serious privacy concerns. To counter these issues, one of the current state-of-the-art approaches is *Private Aggregation of Teacher Ensembles*, or PATE, which achieved promising results in preserving the utility of the model while providing a strong privacy guarantee. PATE combines an ensemble of "teacher models" trained on sensitive data and transfers the knowledge to a "student" model through the noisy aggregation of teachers' votes for labeling unlabeled public data which the student model will be trained on. However, the knowledge or voted labels learned by the student are noisy due to private aggregation. Learning directly from noisy labels can significantly impact the accuracy of the student model. In this paper, we propose the PATE++ mechanism, which combines the current advanced noisy label training mechanisms co-teaching(+) with the original PATE framework to enhance its accuracy. A novel structure of Generative Adversarial Nets with one generator and two discriminators is developed in order to integrate them effectively. Furthermore, we discuss the intrinsic limitations of the "update-by-disagreement" method in the co-teaching+ mechanism and develop a novel noisy label detection mechanism for semi-supervised model training to further improve student model performance when training with noisy labels. We evaluate our method on Fashion-MNIST and SVHN to show the improvements on the original PATE on all measures.**

***Index Terms*—Differential Privacy, Deep Learning, Noisy Labels**

I. Introduction

Training deep learning models requires large-scale data that may be sensitive and contain user's private information, such as detailed medical histories and personal messages or photographs [1]–[3]. Publishing or sharing those models trained on private data directly could cause information leakage and lead to serious privacy issues, as adversaries could exploit the trained models to infer or reconstruct (the features of) the training data [4], [5].

Differential privacy (DP) [6], [7] has demonstrated itself as a strong and provable privacy framework for statistical data analysis and recently been explored to protect privacy of training data when training deep learning models [8]–[10]. Phan et al. [11] explore the objective function perturbing method and use it to train a deep autoencoder satisfying DP. However, it may not be trivial to generalize to other deep learning models. One widely accepted way to provide a rigorous DP guarantee for training neural network models on sensitive data is to use differentially private Stochastic Gradient Descent (DP-SGD) which adds Gaussian noise to the gradients in each iteration during the SGD based optimization process [8]. However, as the model goes deeper, their method becomes less effective.

Another promising approach is *Private Aggregation of Teacher Ensembles*, or PATE, which trains multiple teacher models on disjoint sensitive data and transfers the knowledge of teacher ensembles to a student model by letting the teachers vote for the label of each record from an unlabeled public dataset [9], [12]. The teachers' votes are aggregated through a differentially private noisy-max mechanism, which is to add DP noise to the number of each label's votes first and then take the label with the majority count as the output. Finally, the student model is trained on the partially labeled public dataset in a semi-supervised fashion and published, while the teacher models are kept private. Compared to DP-SGD, PATE achieves higher accuracy with a tighter privacy guarantee. Meanwhile, the PATE method is independent of the learning algorithms and can be applied to different model structures and to datasets with various characteristics. However, the knowledge transferred from teachers to the student, which are noisy-max voted labels, contain a certain proportion of errors or noisy labels, and the proportion has a positive relationship with the level of privacy guarantee that PATE provides and a negative impact on the accuracy of the student model.

In this paper, we propose an enhanced framework PATE++ by incorporating the start-of-the-art noisy label training mechanism into PATE to further improve its practical applicability. PATE++ makes several novel contributions. First, we modify the student model in the original PATE, a generative adversarial network (GAN) [13] with a semi-supervised training strategy [14], by adding another discriminator to the structure of GAN. The purpose of the second discriminator is to enable co-teaching [15] with the first discriminator for robust training with noisy labels. Second, to further exploit the benefit of semi-supervised training, we propose a novel noisy label detection mechanism based on the co-teaching framework and move the data with detected noisy labels from labeled dataset to unlabeled dataset instead of excluding them completely from the training process. We evaluate our framework

on Fashion-MNIST and SVHN datasets. Empirical results demonstrate that our new PATE structure with additional noisy label detection and switching (from labeled data to unlabeled data) mechanism outperforms the original PATE in privacy-preserving model training. Our work further improves the practicality and operability to privately and safely train deep learning models on sensitive data.

II. Preliminaries

In this section, we introduce the definitions of differential privacy [6], and the two essential components of our approach: (1) the PATE framework which was first developed by Papernot et al. in [12] and later improved by Papernot et al. in [9]; (2) the co-teaching mechanism for robust model training with noisy labels and the improved co-teaching+ mechanism [16].

A. Differential Privacy

Differential Privacy (DP) ensures the output distributions of an algorithm are indistinguishable with a certain probability when the input datasets differ in only one record, which is achieved by adding some randomness to the output. Both Laplacian noise and Gaussian noise are widely used to achieve DP, and the scale of the noise is calibrated according to the privacy parameter(s) ϵ (and δ) as well as the sensitivity of the algorithm [7].

Definition 1. *(ϵ, δ)-Differential Privacy) [7]. Let $\mathcal{D}$ and $\mathcal{D}'$ be two neighboring datasets that differ in at most one entry. A randomized algorithm $\mathcal{A}$ satisfies (ϵ, δ)-differential privacy if for all $\mathcal{S} \subseteq Range(\mathcal{A})$: $Pr[\mathcal{A}(\mathcal{D}) \in \mathcal{S}] \leq e^{\epsilon} Pr[\mathcal{A}(\mathcal{D}') \in \mathcal{S}] + \delta$, where $\mathcal{A}(\mathcal{D})$ represents the output of $\mathcal{A}$ with the input $\mathcal{D}$.*

Rényi Differential Privacy (RDP) generalizes (ϵ, 0)-DP in the sense that ϵ-DP is equivalent to (∞, ϵ)-RDP.

Definition 2. *(Rényi Differential Privacy (RDP)) [17]. A randomized mechanism $\mathcal{A}$ is said to guarantee (λ, ϵ)-RDP with $\lambda \geq 1$ if for any neighboring datasets $\mathcal{D}$ and $\mathcal{D}'$,*

$$D_{\lambda}(\mathcal{A}(\mathcal{D}) \| \mathcal{A}(\mathcal{D}')) = \frac{1}{\lambda - 1} \log \mathbb{E}_{x \sim \mathcal{A}(D)} \left[\left(\frac{Pr[\mathcal{A}(D) = x]}{Pr[\mathcal{A}(D') = x]} \right)^{\lambda - 1} \right] \leq \epsilon.$$

In the above definition, $D_{\lambda}(\mathcal{A}(\mathcal{D}) \| \mathcal{A}(\mathcal{D}'))$ indicates the Rényi divergence of order λ between $\mathcal{A}(\mathcal{D})$ and $\mathcal{A}(\mathcal{D}')$. RDP satisfies the adaptive sequential composition property of the privacy guarantee as stated in Proposition 1. The self-composition property of two RDP mechanisms can be generalized to the sequence of mechanisms as in Theorem 1.

Proposition 1. *(RDP Composition) [17] Let $f : \mathcal{D} \mapsto \mathcal{R}_1$ be (α, ϵ_1)-RDP and $g : \mathcal{R}_1 \times \mathcal{D} \mapsto \mathcal{R}_2$ be (α, ϵ_2)-RDP, then the mechanism defined as (X, Y), where $X \sim f(D)$ and $Y \sim g(X, D)$, satisfies $(\alpha, \epsilon_1 + \epsilon_2)$-RDP.*

Theorem 1. *(Sequence Composition) [9]. If a mechanism $\mathcal{A}$ consists of a sequence of adaptive mechanisms $\mathcal{A}_1$, ..., $\mathcal{A}_k$ such that for any $i \in [k]$, $\mathcal{A}_i$ guarantees (λ, ε_i)-RDP, then $\mathcal{A}$ guarantees $\left(\lambda, \sum_{i=1}^{k} \varepsilon_i\right)$-RDP.*

Theorem 2. *(From RDP to (ϵ, δ)-DP) [17]. If a mechanism $\mathcal{A}$ guarantees (λ, ϵ)-RDP, then $\mathcal{A}$ guarantees $\left(\epsilon + \frac{\log 1/\delta}{\lambda - 1}, \delta\right)$-DP for any $0 < \delta < 1$.*

Theorem 2 reveals the relationship between (ϵ, δ)-DP and (λ, ϵ)-RDP. Both of them are relaxed from pure ϵ-DP, while RDP equipped with Gaussian noise has better composition property when analyzing the accumulated privacy loss.

Corollary 1. *(Gaussian Mechanism for RDP) [17] Let $f : \mathcal{D} \mapsto \mathcal{R}$ be a real-valued function. If $\mathcal{A}$ has sensitivity 1, then the Gaussian mechanism $G_{\sigma}\mathcal{A} = f(D) + N\left(0, \sigma^2\right)$ satisfies $(\alpha, \alpha/(2\sigma^2))$-RDP, where $N\left(0, \sigma^2\right)$ is normally distributed random variable with standard deviation σ^2 and mean 0.*

B. The PATE Framework

Figure 1 illustrates the framework of PATE borrowed from [12]. It consists of an ensemble of teacher models and a student model. Each teacher is trained on a disjoint subset of sensitive data that contains user's private information that needs to be protected. Teacher models can be flexibly chosen to fit the data and task. After teachers are trained, the knowledge that teachers learned from sensitive data will be transferred to the student in a private manner. More specifically, at prediction, teachers independently predict labels for the queried data from an unlabeled public dataset. The votes assigned to each class will be counted to form a histogram. To ensure DP, Laplacian or Gaussian noise will be added to each count. The final prediction result for the queried data will be the label with the most votes after adding the noise.

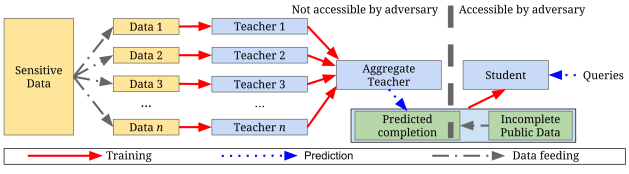


Fig. 1. Overview of the PATE framework: (1) an ensemble of teachers is trained on disjoint subsets of the sensitive data, (2) a student model is trained on public data labeled using the ensemble.

The student model in the PATE framework uses GAN with semi-supervised learning. During student model training, labeled public data are fed into the discriminator D of GAN to form the supervised cross-entropy loss while unlabeled data and generated data from generator G (labeled as an additional 'generated' class) are fed into D to form the unsupervised loss. Feature matching is used to increase the stability of GAN by involving a new objective for G, which requires the activations of real data and generated data on an intermediate layer of D to be as similar as possible through gradient-based optimization.

The initial PATE uses Laplacian noise for the perturbation and moments accountant [8] to compose the total privacy cost for multiple predictions. The improved PATE uses Gaussian noise based on RDP. Additionally, they proposed a *selective* aggregation mechanism called the confident Gaussian NoisyMax aggregator (Confident-GNMax) as in Algorithm 2. Teacher ensembles will only answer the queries if their

votes have strong consensus, which is checked privately. This mechanism benefits both privacy and utility. The privacy cost is small when most teachers agree on one vote. Meanwhile, when most teachers agree, the prediction result is more likely to be correct. However, even with the Confident-GNMax mechanism, the voted labels still contain a certain ratio of errors due to the noisy aggregation. Additionally, in order to achieve a tighter privacy guarantee, larger noise is needed for perturbing the votes, thus causing more noise in the student training dataset, which severely affects the utility of the trained student model.

C. Co-teaching and Co-teaching+ Mechanisms

Deep learning models have enough capacity to remember all training instances even with noisy labels, which leads to bad generalization ability [18]. Han et al. [15] propose a simple but effective mechanism called co-teaching for training deep models with the existence of noisy labels. Their method is based on the observation that during training, models would first memorize or fit training data with clean labels and then those with noisy labels [19]. Co-teaching maintains two networks with the same structure but independent initialization. In each mini-batch of data, each network selects a ratio of small-loss instances as useful knowledge and teaches its peer network with such useful instances for updating the parameters. Intuitively, small-loss instances are more likely to be the ones with correct labels, thus training the network in each mini-batch using only small-loss instances is more robust to noisy labels.

In the early stage of co-teaching, due to independent and random parameter initialization, two networks have different abilities to filter out different types of error using the small-loss trick. However, this divergence between two networks will gradually diminish with the increase of training epochs, which decreases the ability to select clean data and increases the accumulated error. To solve this issue, Yu et al. introduce the "Update by Disagreement" strategy to co-teaching and name the improved mechanism co-teaching+ [16]. Similar to co-teaching, co-teaching+ maintains two networks simultaneously. In each mini-batch of training, two networks feed forward and predict the same batch of data independently first, and then a ratio of small-loss instances will be chosen by each network only from those data with disagreed predictions between two networks and fed to each other for parameter update. This disagreement-update step keeps the constant divergence between two networks and promotes the ability of them to select clean data.

III. Improved Training Mechanism For PATE

Inspired by co-teaching mechanism and its improved version co-teaching+, we modify the PATE framework to improve the student model's robustness when training with noisy labels provided by teachers.

A. PATE+: Student Model with Co-teaching+

The student model of PATE is a GAN trained under semi-supervised learning with both supervised and unsupervised losses while co-teaching(+) is originally used in the supervised model training. To utilize co-teaching(+) in the student model, our main idea is to add an additional discriminator in the GAN used in the student model, as shown in Figure 2. We do not use two GANs with both generator and discriminator as the peers for co-teaching(+) because the small-loss trick plays its role only in the supervised part, while the generator is involved in the unsupervised loss of GAN as well as the feature matching loss [14], which are both unsupervised and not associated with labels.

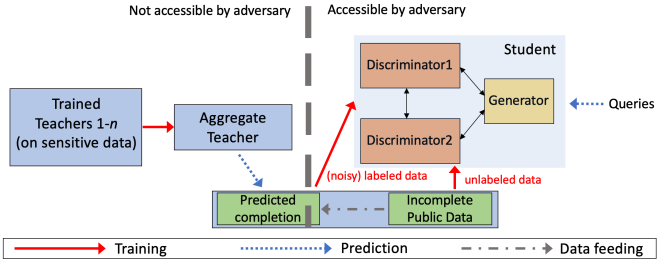


Fig. 2. Overview of the PATE+ framework. (1) an ensemble of teachers is trained on disjoint subsets of the sensitive data, (2) a semi-supervised GAN student model with one generator and two discriminators co-teaching+ with each other is trained on public data labeled using the ensemble.

Suppose there exist K possible classes in sensitive data as well as the labeled public data that the student model will be trained on. In the semi-supervised learning using GANs, the data generated by generator G are labeled with a new "generated" class $y = K + 1$. The discriminator D takes in a data sample $\boldsymbol{x}$ as input and outputs class probabilities distribution $p_D(y|\boldsymbol{x}, j < K + 1)$. For labeled data $\boldsymbol{x}$, the cross-entropy between the observed label and the predicted distribution $p_D(y|\boldsymbol{x}, j < K + 1)$ forms the supervised loss. For generated data, $p_D(y = K + 1|\boldsymbol{x})$ is used to supply the probability that x is not real. For those unlabeled data, since we know they come from one of the K classes of real data, we can learn from them by maximizing $\log p_D(y \in \{1, \ldots, K\}|\boldsymbol{x})$ [14].

For the student model in Figure 2, there are two discriminators and one generator. The supervised loss and unsupervised loss for `Discriminator1` and `Discriminator2` (D_1 and D_2) are expressed as:

$$L^{D_i}_{\text{supervised}} = -\{\mathbb{E}_{\boldsymbol{x},y\sim p_{\text{data}}(\boldsymbol{x},y)} \log p_{D_i}(y|\boldsymbol{x}, y < K + 1)\};$$

$$L^{D_i}_{\text{unsupervised}} = -\{\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})} \log [1 - p_{D_i}(y = K + 1|\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x}\sim G} \log [p_{D_i}(y = K + 1|\boldsymbol{x})]\}.$$

where $i = 1, 2$ and p_{data} indicates the real data distribution.

Feature matching loss in the semi-supervised GANs training is defined as: $\left\|\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}} \mathbf{f}(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})} \mathbf{f}(G(\boldsymbol{z}))\right\|_2^2$, where $p_{\boldsymbol{z}}(\boldsymbol{z})$ indicates the random distribution and $\mathbf{f}(\boldsymbol{x})$ is the activation output of an intermediate layer of the discriminator. In the structure of student model as shown in Figure 2, the generator takes the activations from two discriminators which are expressed as $\mathbf{f}_{D_1}(\boldsymbol{x})$ and $\mathbf{f}_{D_2}(\boldsymbol{x})$ respectively. We use the average of two feature losses associated with two discriminators as the objective for the generator. Therefore, the feature matching loss of the generator in the student model is defined as:

$$L_{\text{fm}}^{G} = \frac{1}{2}\big(\big\|\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}} \mathbf{f}_{D_1}(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}\mathbf{f}_{D_1}(G(\boldsymbol{z}))\big\|_2^2 + \big\|\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}} \mathbf{f}_{D_2}(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}\mathbf{f}_{D_2}(G(\boldsymbol{z}))\big\|_2^2\big).$$

The main steps for training student model with the "update by disagreement" strategy are illustrated in Algorithm 1.

Algorithm 1 PATE+: Training Student Model in PATE with Discriminators Co-teaching+

Input: D_1, D_2, G, labeled public data M_l from private teachers aggregation, unlabeled data M_u, batch size B, learning rate η, epoch E, ratio R.

1: **Duplicate** M_l or M_u to make them have the same size.
2: **for** $e = 1, ..., E$ **do**
3: **Shuffle** M_l, M_u into $\frac{|M_l|}{B}$ mini-batches respectively.
4: **for** $b = 1, ..., \frac{|M_l|}{B}$ **do**
5: **Fetch** b-th mini-batch m_l (m_u) from M_l (M_u);
6: **Generate** B fake samples m_g from G;
7: **Select** samples with the different predicted results between D_1 and D_2 in m_l as $\hat{m_l}$
8: **for** $i = 1, 2$ **do**
9: **Fetch** the R% smallest-loss samples ${\hat{m_l}}^{(i)}$ of D_i: ${\hat{m_l}}^{(i)} = \text{argmin}_{\hat{m_l}':|\hat{m_l}'|\geq R|\hat{m_l}|} L_{\text{supervised}}^{D_i}(\hat{m_l}'; D_i)$
10: **end for**
11: **Update** $D_1 = D_1 - \eta\nabla(L_{\text{supervised}}^{D_1}({\hat{m_l}}^{(2)}; D_1) + L_{\text{unsupervised}}^{D_1}(m_u, m_g; D_1))$ // D_1 indicates parameters of Discriminator1 here
12: **Update** $D_2 = D_2 - \eta\nabla(L_{\text{supervised}}^{D_2}({\hat{m_l}}^{(1)}; D_2) + L_{\text{unsupervised}}^{D_2}(m_u, m_g; D_2))$ // D_2 indicates parameters of Discriminator2 here
13: **Update** $G = G - \eta\nabla L_{\text{fm}}^{G}(m_u, m_g; G)$ // G indicates parameters of Generator here
14: **end for**
15: **end for**

Output: Trained D_1, D_2 and G, where D_1 and D_2 satisfy rigorous DP guarantee.

B. Privacy Guarantee of PATE+

The privacy guarantee of Algorithm 1 is inherited from the privacy guarantee of the labeled public dataset M_l by the post-processing property of DP [7]. We analyze the RDP guarantee of generating M_l by the Confident-GNMax aggregator which is used in the scalable PATE framework [9]. To start with, we recall the two steps of the Confident-GNMax aggregator. Given a query sample x belonging to one of the classes from 1 to m, let $n_i(x)$ denote the vote count for the i-th class of x. Confident-GNMax aggregator first privately checks if there is enough consensus among teachers (line 1 in Algorithm 2). $\mathcal{N}\left(0, \sigma_1^2\right)$ is the Gaussian distribution with mean 0 and variance σ_1^2. If the check is passed, Confident-GNMax aggregator will output the class label with noisy plurality after adding Gaussian noise ($\mathcal{N}\left(0, \sigma_2^2\right)$) to each vote count (line 2 in Algorithm 2), while discarding this query without labeling it if the pass is failed. The sensitivity of private consensus check (line 1) is 1 because the private training data is divided without overlapping, and one data sample will only affect one teacher model which will change the maximum vote count ($\max_i \{n_j(x)\}$) by at most 1. Therefore, line 1 in Algorithm 2 guarantees $\left(\lambda, \lambda/2\sigma_1^2\right)$-RDP for all $\lambda > 1$ by corollary 1. Line 2 in Algorithm 2 is the GNMax mechanism in [9]. By the data-dependent privacy guarantee in Proposition 8 of [9], line 2 satisfies $\left(\lambda, \lambda/\sigma_2^2\right)$-RDP for all $\lambda > 1$. By using the composition property of RDP in Proposition 1, we can conclude the privacy guarantee for the Confident-GNMax Aggregator as in Theorem 3.

Algorithm 2 Confident-GNMax Aggregator [9]

Input: input x, threshold T, noise parameters σ_1 and σ_2.

1: **if** $\max_i \{n_j(x)\} + \mathcal{N}\left(0, \sigma_1^2\right) \geq T$ **then**
2: **return** $argmax_j \left\{n_j(x) + \mathcal{N}\left(0, \sigma_2^2\right)\right\}$
3: **else**
4: **return** $\perp$
5: **end if**

Theorem 3. *For any $\lambda > 1$, the Confident-GNMax Aggregator in Algorithm 2 satisfies (λ, β)-RDP where $\beta = \lambda/2\sigma_1^2 + \lambda/\sigma_2^2$ if the private consensus check in line 1 of Algorithm 2 is passed, or $\beta = \lambda/2\sigma_1^2$ if the check is failed.*

By using the privacy guarantee of Confident-GNMax aggregator in Theorem 3, we derive the privacy guarantee of the PATE+ algorithm.

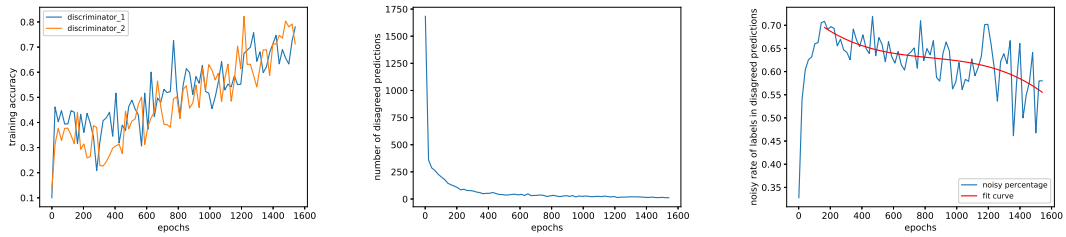
Proposition 2. *If querying the teacher ensembles with a public dataset M, and the teacher ensembles label M using Confident-GNMax aggregator in Algorithm 2 to generate a labeled dataset M_l, then the student model trained on M_l using PATE+ algorithm in Algorithm 1 satisfies (ϵ, δ)-DP for any $0 < \delta < 1$ and $\epsilon = \lambda\left(\frac{|M|}{2\sigma_1^2} + \frac{|M_l|}{\sigma_2^2}\right) + \frac{\log 1/\delta}{\lambda - 1}$, where $\lambda > 1$.*

Proof: Suppose the number of data samples in public dataset M and in labeled dataset M_l is $|M|$ and $|M_l|$ respectively. Therefore, the number of data samples that are discarded (without labeling) during Confident-GNMax aggregation is $|M| - |M_l|$. We use Theorem 3 in conjunction with Theorem 1 to derive the total (λ, β)-RDP privacy guarantee for answering M and generating M_l, where $\lambda > 1$ and $\beta = (|M| - |M_l|) * \frac{\lambda}{2\sigma_1^2} + |M_l| * (\frac{\lambda}{2\sigma_1^2} + \frac{\lambda}{\sigma_2^2}) = \frac{|M|\lambda}{2\sigma_1^2} + \frac{|M_l|\lambda}{\sigma_2^2}$. By Theorem 2, we can transfer $(\lambda, \frac{|M|\lambda}{2\sigma_1^2} + \frac{|M_l|\lambda}{\sigma_2^2})$-RDP into $\left(\lambda\left(\frac{|M|}{2\sigma_1^2} + \frac{|M_l|}{\sigma_2^2}\right) + \frac{\log 1/\delta}{\lambda - 1}, \delta\right)$ -DP for any $0 < \delta < 1$. By the post-processing property of DP, the student model trained on M_l will satisfies (ϵ, δ)-DP where $\epsilon = \lambda\left(\frac{|M|}{2\sigma_1^2} + \frac{|M_l|}{\sigma_2^2}\right) + \frac{\log 1/\delta}{\lambda - 1}$ and $0 < \delta < 1$ since it has no access to the private training data of teacher ensembles and therefore, can not obtain additional knowledge about the private dataset. ■

Notice that training PATE with co-teaching satisfies the same privacy guarantee with PATE+ because the discrepancy between co-teaching and co-teaching+, which is the "update by disagreement" strategy, is independent with the private training data of teacher ensembles and the privacy analysis.

C. PATE++: PATE+ with Noisy Label Cleansing

Potential Drawbacks in PATE+. "Update by disagreement" strategy actually has two potential drawbacks. First, in the late stage of training, two discriminators are going to achieve a similar capacity and consensus on the predictions with most data. Therefore, the number of "disagreed" data in each mini-batch is limited, which restricts the models from learning since



(a) training accuracy (b) number of disagreed predictions (c) noisy rate of labels in disagreed predictions

Fig. 3. A student model trained on 2,200 labeled (726 are noisy labels) and 6,800 unlabeled data from Fashion-MNIST dataset using PATE+ algorithm. (a) The training accuracy of two discriminators in the student model vs epochs (b) The number of labeled samples with different predictions by two discriminators vs epochs (c) The noisy rate of labels in disagreed predictions vs epochs.

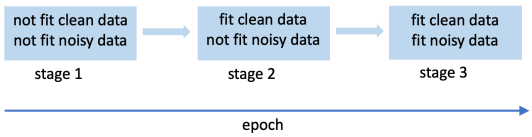


Fig. 4. Illustration of the three stages of model training process with the existence of noisy data.

they can only learn from limited data. Second, the proportion of noisy labels within the "disagreement" in the mini-batch is increasing with the epoch, and models' utility is sacrificed by learning from data with more noisy labels. According to the learning pattern of deep models [19], after the models have learned to fit easy (clean) data, they are more likely to agree on the predictions of clean data while disagreeing on noisy data because the predictions on noisy data have more randomness and errors before models fit them.

We demonstrate our hypothesis using an example. We train a student model using Algorithm 1. Each discriminator is a convolutional neural network (see Experiments on Fashion-MNIST for details). A total of 2,200 data labeled by Confident-GNMax aggregator and 6,800 unlabeled data is used as the training dataset, where 726 of 2,200 labeled samples are noisy data (the labels of them are different from their ground truth labels). Fig 3(a) shows the training accuracy of two discriminators in the student model. We can see that they follow different learning paths. Fig 3(b) shows the number of training samples with disagreed predictions by two discriminators. The number is decreased to a small value during the training process. This observation is consistent with our first hypothesis of the potential drawbacks of the "update by disagreement" strategy. When two discriminators gradually acquire a similar capacity, the number of "disagreed" data in each mini-batch is few (less than 50 out of 2,200 after 400 epochs). Therefore, discriminators can only learn from very few data in the late period, which seriously affects their learning capacity. We calculate the percentage of noisy labels within the "disagreement" in each epoch as shown by the blue line in Fig 3(c). The noisy label rate in all labeled data is 0.33, while the noisy label rate in the "disagreement" is much higher. This observation reflects our second hypothesis that the model's utility will be sacrificed by learning from the "disagreed" data which contains more noisy labels.

Model Training Stages. We roughly divide the model learning process into three stages based on the observations in [19]. In the early stage which is indicated as stage 1 in Figure 4, models have not fit either clean or noisy data. The disagreement on predictions between two peer models is mainly caused by randomness. The percentage of noisy labels within the "disagreement" roughly equals the percentage of noisy labels in the entire training dataset. In stage 2, models have fit the clean data (except for "hard examples") but not the noisy data. The peer models are more likely to have the same (and correct) predictions for clean data. For those noisy data, since the models have not fitted them, the predictions of them are more random and with more errors. Therefore, prediction disagreements are more likely to happen on the data with noisy labels during this stage. In stage 3, which is the late stage of training, due to the memorization effort, the models have learned to fit both the clean and noisy data. The peer models begin to be more consistent in the prediction of both types of data. Thus the ratio of noisy labels in the disagreed predictions decreases. We can observe this phenomenon from Fig 3(c). We fit part of the blue line which is the noisy label rate in the "disagreement" as the function of epochs using smoothing spline fit [20] to observe the general trend of the curve more clearly, which is shown in the red line. We can see the noisy percentage in the "disagreement" increases in the early stage of model training while decreases in the later stage.

Noisy Label Cleansing. Based on our analysis and observations, we hypothesize that the noisy label ratio is the highest within the "disagreement" during stage 2. We propose to filter out noisy labels using this criterion, i.e., the data that has different prediction results by the two peer models during stage 2. However, there is a critical situation that we need to consider. Two peer models do not always have the same learning speed, and they follow different paths during the optimization (as shown in Figure 3(a)). Therefore, it could happen that one model already fits the clean data while the other does not. In this situation, suppose there is a data record with the true (clean) label, the first model gives the correct prediction with high probability, while the second model with the weaker capability predicts it as other labels incorrectly and causes the variation in predictions. Thus clean data could also be chosen by the "disagreement" criterion. To avoid this situation, we further refine our criterion. Notice that in the

Algorithm 3 PATE++: PATE+ with Noisy Label Cleansing

Input: D_1, D_2, G, labeled public data M_l from private teachers aggregation, unlabeled data M_u, batch size B, learning rate η, epoch E, ratio R, removal percentage τ, decay factor α.

1: **Step 1: Filter out noisy label in M_l based on PATE+ framework**
2: **Duplicate** M_l or M_u to make them have the same size.
3: **Initialize** the filtered out noisy dataset M_n as $\emptyset$.
4: **Initialize** a count table T for each data in M_l to be 0.
5: **for** $e = 1, ..., E$ **do**
6: **Shuffle** M_l, M_u into $\frac{|M_l|}{B}$ mini-batches respectively.
7: **for** $b = 1, ..., \frac{|M_l|}{B}$ **do**
8: **Fetch** b-th mini-batch m_l (m_u) from M_l (M_u);
9: **Generate** B fake samples m_g from G;
10: **Select** samples with the different predicted results between D_1 and D_2 in m_l as $\hat{m}_l$
11: **Select** samples in $\hat{m}_l$ whose prediction results by D_1 and D_2 are both different with its observed label as $\overline{m_l}$.
12: **Set** the count of data in $\overline{m_l}$ to 1.
13: **Fetch** the $R\%$ smallest-loss samples $\hat{m_l}^{(1)}$ ($\hat{m_l}^{(2)}$) of D_1 (D_2) as in line 8-10 in Algorithm 1
14: **Update** D_1, D_2, G as in line 11-13 in Algorithm 1
15: **end for**
16: **Multiply** the count of each labeled data in this epoch with α and add to the count table T.
17: **end for**

18: **Step 2: Remove filtered out noisy labels**
19: **Remove** $\tau\%$ data with the most count from M_l to form M_l^{san}.
20: **Add** those removed data to the unlabeled dataset to form M_u^{san}.

21: **Step 3: Retrain the PATE+ on sanitized datasets M_l^{san} and M_u^{san} using Algorithm 1**

Output: Trained D_1, D_2 and G, where D_1 and D_2 satisfy rigorous DP guarantee.

above-mentioned circumstance, the "disagreement" happens when the first model with the stronger capability predicts the true label for the clean data (the predicted label is the same as the observed label) while the second model with the weaker capability predicts a wrong label (the predicted label is different from the observed label). Therefore, we further filter out noisy data whose predicted labels by peer models are both different from the observed label from the "disagreement" in stage 2. That is, our noisy label cleansing mechanism has two criteria: 1) peer models disagree on the predictions for this data, and 2) the prediction results by two peer models are both different from the observed label of the data.

The last question is, how can we know when the models change from stage 1(2) to stage 2(3). One possible solution is to use the validation utility to help us decide. In stage 1, models have very low utility since they fit neither clean nor noisy data. In stage 2, the utility of models increases as the models have learned useful knowledge from clean and easy-to-fit data. In stage 3, models' utility can still increase but with relatively slower speed compared to stage 2, since noisy labels are hard to fit. However, due to the uncertainty of the gradient-based optimization process, it is not efficient to separate these stages using the validation utility. We solve this problem using the weighted decay count. We count the number of epochs for each data when it satisfies the previously mentioned two criteria. Clean data tend to satisfy those two criteria during stage 1, while noisy data tend to satisfy those two criteria in both stage 1 and stage 2. Therefore, data with more counts at the end of training are determined as the data with noisy labels. To further reduce the effects of stage 1, we multiply a weight (smaller than 1) to the counts at the end of each epoch before adding them to the new counts of the next epoch. Weighted decay count smooths the decision process and makes the criteria more robust to the randomness caused by the gradient-based optimization process.

PATE++. In Algorithm 3, we present the complete PATE++ framework for training more robust PATE by filtering out noisy labels based on the PATE+ framework first, and then retraining PATE+ on the sanitized dataset, which is formed by removing the top $\tau\%$ data with the most count as introduced above. τ indicates the removal percentage. The privacy analysis for Algorithm 3 follows Proposition 2 by the post-processing property of DP. Notice the noisy label cleansing procedure does not involve additional privacy leakage since it does not depend on the private training dataset of teacher ensembles.

IV. Experiments

We performed experiments on Fashion-MNIST and SVHN to demonstrate the efficiency of our proposed PATE+ and PATE++ frameworks compared to the original PATE for training the student model on noisy data provided by private teachers aggregation.

A. Fashion-MNIST

Fashion-MNIST dataset [21] consists of 10 classes with 60,000 training examples and 10,000 testing examples. Similar to in the original PATE, we use 60,000 training examples to train the teachers and 10,000 testing examples as the public dataset for training the student. We divide the 60,000 training examples randomly into 250 disjoint subsets equally. Each subset is used to train one teacher model, which is a convolutional neural network with seven convolutional layers followed by two fully connected layers and an output layer (same as the deep model in the original PATE). After 250 teachers are trained, we use Confident-GNMax aggregator to label 2,200 data from the public dataset twice. For the first time, we use the smaller noise which leads to (5.04, 10^{-5})-DP guarantee. For the second time, we use the larger noise which leads to (4.05, 10^{-5})-DP guarantee. Adding the larger noise during the private teacher aggregation leads to a tighter privacy guarantee (smaller ϵ), while the trade-off is that there will be more noisy labels within the labeled dataset. The structure of the discriminators in the student model is the same as the structure of teachers. The generator of the student model is a three-layer fully connected neural network. The 10,000 testing examples are further divided into the first 9,000 (where 2,200 are labeled by teachers as labeled data and 6,800 are used as unlabeled data) for training and the last 1,000 for testing. We compare the test accuracy of the

student models trained by 1) the original PATE (traditional semi-supervised training); 2) PATE with co-teaching between two peer discriminators; 3) PATE+ (PATE with co-teaching+ between two peer discriminators); and 4) PATE++ (PATE+ with noisy label cleansing). We train student models with batch size 100 using Adam optimizer with the learning rate set to 0.01. In PATE++, the decay factor α is set to 0.9 by grid search. Table I shows the experimental results.

From Table I, we can observe that PATE++ achieves the best performance on training the student model. The improvement is even higher (4.8% vs. 0.8%) when the privacy budget is tight (4.05 vs. 5.04). This further motivates our proposed mechanism PATE++, since there is an inevitable trade-off between utility and privacy in the PATE framework, The stronger privacy requires adding larger noise during the private teacher aggregation which leads to a higher noise ratio in the student training data. PATE++ mitigates this by making the student model more robust when trained with noisy labels.

TABLE I
TEST ACCURACY OF THE STUDENTS UNDER DIFFERENT FRAMEWORKS TRAINED ON FASHION-MNIST DATASET.

	Student Accuracy			
Privacy budget (ϵ, δ)	**Original PATE**	**PATE with co-teaching**	**PATE+ (Alg.1)**	**PATE++ (Alg.3)**
(4.05, 10^{-5})	74.8%	77.3%	76.5%	**79.6**%
(5.04, 10^{-5})	82.1%	82.5%	82.7%	**82.9**%

Selection of R and τ. As suggested in [15], the ratio of small-loss instances R should be chosen increasingly during the training since when the number of epochs goes large, the model will gradually overfit on noisy labels. Thus, more instances can be kept in the mini-batch at the start while less should be in the end. We use their proposed scheduling: $R(e) = 1 - \beta \min\left\{\frac{e}{15}, 1\right\}$ where e is the epoch and β is the estimated noise rate which can be determined by manually verifying a small sampled subset. We report the student accuracy of 1) PATE with co-teaching, and 2) PATE+ (the same as PATE++ with τ=0) under the different noise ratio estimation values in Table II. We show the setting with $(4.05, 10^{-5})$-DP guarantee. We can see that the estimated noise rate for the scheduling has an effect on student performance. How to best estimate the noise rate and set the optimal scheduling function is still an unsolved problem in the co-teaching and co-teaching+ works [15], [16].

TABLE II
TEST ACCURACY OF THE STUDENT MODELS WITH VARYING R (BOLD RESULTS COINCIDE WITH TABLE I).

Estimated Noise Ratio β	0.1	0.2	0.3	0.4
PATE with co-teaching	76.2%	**77.3**%	77%	76.2%
PATE+ (Alg.1)	76.4%	**76.5**%	77.3%	77.4%

TABLE III
TEST ACCURACY OF THE STUDENT MODELS WITH VARYING τ (BOLD RESULT COINCIDES WITH TABLE I).

Removal Ratio τ	0.091	0.182	0.227	0.273	0.318	0.364
PATE++ ($\beta = 0.2$)	78.1%	78.6%	79%	79.5%	**79.6**%	78.2%

We fix β = 0.2 and report the student accuracy of PATE++ with different τ values for the noisy label cleansing ratio in Table III. Increasing the removal ratio τ will increase the chance to move more noisy labels from the labeled dataset to the unlabeled dataset and lead to better student performance because the student model is trained on the dataset with less noisy labels. However, the tradeoff is that with the higher removal ratio, less labeled data will be left as well as data with clean labels that the student can learn useful knowledge from. In practice, we choose the removal ratio by grid search.

B. SVHN

SVHN [22] contains 10 classes with 73,257 training examples and 26,032 testing examples. We use the same structure for the student model as in Fashion-MNIST experiments. The 26,032 testing examples are divided into 10,000 for student training and 16,032 for student testing. We use the clean teacher votes made available online by the authors of PATE to do the Confident-GNMax aggregation for labeling student's training data. 3,000 data are labeled privately using the smaller noise corresponding to (4.93, 10^{-6})-DP guarantee and the larger noise corresponding to (3.96, 10^{-6})-DP guarantee. The student models are trained the batch size 100 inputs using the Adam optimizer with the learning rate set to 0.003 and the decay factor α set to 0.9 in PATE++. Table IV shows the experimental results on SVHN with the estimated noise rate $\beta = 0.2$ and the removal percentage $\tau = 0.4$.

TABLE IV
TEST ACCURACY OF THE STUDENTS UNDER DIFFERENT FRAMEWORKS TRAINED ON SVHN DATASET.

	Student Accuracy			
Privacy budget (ϵ, δ)	**Original PATE**	**PATE with co-teaching**	**PATE+ (Alg.1)**	**PATE++ (Alg.3)**
(3.96, 10^{-6})	80.5%	86.1%	79.8%	**91.5**%
(4.93, 10^{-6})	91.7%	92.8%	91.6%	**93.7**%

We can observe in Table IV that PATE++ significantly outperforms the original PATE, especially under the tight privacy budget. The student accuracy of PATE+ is shy when compared with PATE with co-teaching. The reason could be the drawbacks of the "update by disagreement" strategy that we mentioned previously.

V. RELATED WORK

[23] proposed to transfer the knowledge learned from a publicly available non-private dataset to the teachers in order to alleviate the problem that the training data assigned for each individual teacher maybe not enough to achieve an ideal performance for some complex datasets and tasks. [24] exploited knowledge distillation [25] to further transfer the knowledge from teacher ensembles to the student model privately through the representations from intermediate layers of teacher models. [26] developed a new semi-supervised learning algorithm called MixMatch, which achieves state-of-the-art performance in several benchmark datasets by combining several dominant approaches for semi-supervised learning together into a unified framework. They demonstrate that MixMatch improves the performance of PATE with respect to the accuracy-privacy trade-off, which is unsurprising because PATE is a general framework with the student model

trained by the semi-supervised learning paradigm in order to reduce the total privacy cost induced by each individual query. Any improved semi-supervised learning algorithm is expected to improve the original PATE framework. Different from these previous works, our work improves PATE from another perspective by incorporating the novel noisy label training and cleansing mechanism under the semi-supervised learning framework to improve the student model accuracy without additional privacy cost.

Learning with noisy examples has a long research history [27]. Currently, training deep learning models with noisy labels has received increasing attention [15], [16], [28]–[31]. A comprehensive review of all the works within this area is beyond the scope of this paper. Our proposed mechanisms incorporate the co-teaching and co-teaching+ methods into the PATE framework to better train the student model with noisy labels and achieve promising results. Investigation of other noisy label training methods to further enhance the performance will be an interesting research direction.

VI. Conclusions

We proposed the PATE+ mechanism for robust training of the student model in PATE, and PATE++ mechanism based on PATE+ which combines co-teaching+ between two discriminators within the structure of GAN and noisy label cleansing. Experimental results demonstrate the advantage of our mechanisms compared to the original PATE, especially when the privacy budget is tight. Our proposed mechanisms enhance the utility and privacy trade-off in private model training and further improve the practicality to achieve meaningful privacy guarantees when training deep models on sensitive data. We leave applying PATE++ to other applications such as sequence-based models and graph models as future work.

Acknowledgments

This research is supported by National Science Foundation under CNS-1952192 and IIS-1838200, National Institutes of Health (NIH) under UL1TR002378 and R01GM118609, and Air Force Office of Scientific Research (AFOSR) DDDAS Program under FA9550-12-1-0240. XJ is CPRIT Scholar in Cancer Research (RR180012), and he was supported in part by Christopher Sarofim Family Professorship, UT Stars award, UTHealth startup, NIH under award number R01AG066749, U01TR002062, and NSF RAPID 2027790.

References

[1] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley, "Deep patient: an unsupervised representation to predict the future of patients from the electronic health records," *Scientific reports*, vol. 6, p. 26094, 2016.

[2] O. Vinyals, Ł. Kaiser, T. Koo, S. Petrov, I. Sutskever, and G. Hinton, "Grammar as a foreign language," in *NIPS*, 2015, pp. 2773–2781.

[3] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," 2015.

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

[5] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 1322–1333.

[6] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of cryptography conference*. Springer, 2006, pp. 265–284.

[7] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

[8] 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*. ACM, 2016, pp. 308–318.

[9] N. Papernot, S. Song, I. Mironov, A. Raghunathan, K. Talwar, and Ú. Erlingsson, "Scalable private learning with pate," in *ICLR*, 2018.

[10] M. Abadi, U. Erlingsson, I. Goodfellow, H. B. McMahan, I. Mironov, N. Papernot, K. Talwar, and L. Zhang, "On the protection of private information in machine learning systems: Two recent approches," in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. IEEE, 2017, pp. 1–6.

[11] N. Phan, Y. Wang, X. Wu, and D. Dou, "Differential privacy preservation for deep auto-encoders: an application of human behavior prediction," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[12] 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.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, 2014, pp. 2672–2680.

[14] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *NIPS*, 2016, pp. 2234–2242.

[15] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *NIPS*, 2018, pp. 8527–8537.

[16] X. Yu, B. Han, J. Yao, G. Niu, I. W. Tsang, and M. Sugiyama, "How does disagreement benefit co-teaching?" *ICLR*, 2019.

[17] I. Mironov, "Rényi differential privacy," in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. IEEE, 2017, pp. 263–275.

[18] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," in *ICLR*, 2017.

[19] D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio *et al.*, "A closer look at memorization in deep networks," in *ICML*, 2017, pp. 233–242.

[20] M. P. Wand, "A comparison of regression spline smoothing procedures," *Computational Statistics*, vol. 15, no. 4, pp. 443–462, 2000.

[21] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[22] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, in *NIPS Workshop*, 2011.

[23] L. Wang, J. Zheng, Y. Cao, and H. Wang, "Enhance pate on complex tasks with knowledge transferred from non-private data," *IEEE Access*, vol. 7, pp. 50 081–50 094, 2019.

[24] L. Sun, Y. Zhou, J. Wang, J. Li, R. Sochar, P. S. Yu, and C. Xiong, "Private deep learning with teacher ensembles," *arXiv preprint arXiv:1906.02303*, 2019.

[25] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *NIPS Workshop*, 2014.

[26] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," in *NIPS*, 2019, pp. 5050–5060.

[27] J. Quinlan, "Learning from noisy data," in *Proc. of the International Machine Learning Workshop*. Citeseer, 1983, pp. 58–64.

[28] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, "Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels," in *ICML*, 2018, pp. 2304–2313.

[29] E. Malach and S. Shalev-Shwartz, "Decoupling" when to update" from" how to update"," in *NIPS*, 2017, pp. 960–970.

[30] J. Li, R. Socher, and S. C. Hoi, "Dividemix: Learning with noisy labels as semi-supervised learning," *ICLR*, 2020.

[31] D. Karimi, H. Dou, S. K. Warfield, and A. Gholipour, "Deep learning with noisy labels: exploring techniques and remedies in medical image analysis," *Medical Image Analysis*, p. 101759, 2020.