On The Private Data Synthesis Through Deep Generative Models for Data Scarcity of Industrial Internet of Things

Yen-Ting Chen, Chia-Yi Hsu, Chia-Mu Yu, *Senior Member, IEEE*, Mahmoud Barhamgi, Charith Perera

***Abstract*—Due to the data-driven intelligence from the recent deep learning (DL)-based approaches, the huge amount of data collected from various kinds of sensors from industrial devices have the potential to revolutionize the current technologies used in the industry. To improve the efficiency and quality of machines, the machine manufacturer needs to acquire the history of the machine operation process. However, due to the business secrecy, the factories are not willing to do so. One promising solution to the above difficulty is the synthetic dataset and an informatic network structure, both through deep generative models such as differentially private GANs (DP-GANs). Hence, this paper initiates the study of the utility difference between the above two kinds. We carry out an empirical study and find that the classifier generated by private informatic network structure is more accurate than the classifier generated by private synthetic data, with approximately $0.31\% \sim 7.66\%$.**

***Index Terms*—Industrial Internet of Things, Deep Generative Model, Generative Adversarial Network, Data Synthesis, Differential Privacy**

I. Introduction

A. Industrial Internet of Things (IIoT)

Thanks to the rapid rise of the Internet of Things (IoT), there are increasing demands and novel user scenarios for human life. Smart appliances [1], autonomous driving [2], intelligent robots [3] are exemplar applications with a considerable number of devices connected to each other. For industry, wireless communications and artificial intelligence (AI) jointly promote the development of industrial IoT (IIoT) [4], [5]. In particular, IIoT has been witnessed to significantly improve manufacturing efficiency, reduce product cost, and upgrade the manufacturing process by integrating various sensors and controllers with intelligent analysis. While the intelligence in IIoT is the core component that leads to the above benefits, a critical part behind the scene is the abundance of the data.

B. Data Issue in IIoT

Due to the data-driven intelligence from the recent deep learning (DL)-based approaches, the massive amount of data collected from various sensors from industrial devices has already become the primarily productive force. They have the potential to revolutionize the current technologies used in the industry. For example, one can feed the data collected from IIoT into a productive decision-making model to achieve data-driven smart manufacturing. However, as current data-driven AI models widely used in IIoT (e.g., deep neural networks, DNN) require a considerable amount of high-quality data to achieve intelligence, data incompleteness, low data quality, and insufficient quantity have become the pain points. For example, for image classification that often sees industrial applications such as defect detection, a rule of thumb is at least 1000 images per class in DL. More specifically, first, possibly due to the malfunctioning of sensors, the IIoT data may have missing values, which may frustrate the training process. Second, the factors such as vibration and high-frequency interference in the factory may affect the sensors, leading to the low quality and uncertainty of IIoT data. Such IIoT data mostly compromise overall decision-making performance. Third, an insufficient amount of IIoT, due to scarcity of the events of interest, may easily make the DNN underfitting. Fourth, the valuable data lead to the owner's unwillingness to share the data. Despite the effort in using techniques such as federated learning to help data collection [6], [7], [8], only parts of the above problems can be handled.

The availability of large datasets has been a crucial factor in the success of DL-based classification and detection methods. While datasets for everyday objects can easily be collected, datasets for specific industrial use-cases (e.g., automated inspection and defect detection) can hardly be collected. In this paper, we mainly focus on the scarcity of image datasets in IIoT.

C. Key Challenges in Data Synthesis Through Deep Generative Models (DGM)

To handle the data scarcity and facilitate DL techniques in industrial applications, before fed into model training algorithm, the dataset needs to be either created from scratch or enhanced from a small-size initial dataset. In essence, to accomplish the above task, deep generative models (DGM), such as generative adversarial networks (GAN) [9], could be a promising solution for generating realistic "real" data in an unsupervised manner. Despite its powerful generative capability, GAN has limitations such as limited expressive power, poor interpretability, and weak discriminative ability. However, a fundamental problem for the data synthesis in IIoT data synthesis is that most models "memorize" the training data due to the potential overfitting issue. The synthetic data

Yen-Ting Chen and Chia-Yi Hsu are with Department of Computer Science and Engineering, National Chung Hsing University, Taiwan. Chia-Mu Yu is with Department of Information Management and Finance, National Yang Ming Chiao Tung University. Mahmoud Barhamgi is with Claude Bernard Lyon 1 University, France. Charith Perera is with Cardiff University, UK.

generated from such a model also leak information about the original sensitive data.

D. Motivating Example and Problem Statement

Motivating Example. Consider a motivating example as follows. There are many factories, each of which runs a machine of the same type (e.g., grinding machines and metal processing machine tools). The machine manufacturer wants to collect the history of the machine operation process from factories to perform the AI-based analysis to improve their future design's quality and efficiency. However, in operating the machine, trade secrets such as different combinations of parameters will be crafted and stored in the machine. The history of the machine operation process may reflect or leak the factory's trade secrets. As a consequence, each factory is unwilling to share the machine operation process.

Problem Statement. A promising solution to the above difficulty is that each factory privately constructs a synthetic dataset according to the machine operation process. A differentially private GAN (DP-GAN) could be the best choice to build a synthetic dataset because it strikes a balance between data privacy and data utility. After that, one can have two possible approaches for the factory to "share" the data with the machine manufacturer.

- **(A1)** Each factory individually constructs the synthetic dataset through DP-GAN according to the machine operation process and then shares the synthetic dataset with the machine manufacturer. Here, from the machine manufacturer's viewpoint, it receives a dataset. So the machine manufacturer can perform arbitrary analysis on the received dataset, hoping that the corresponding analytical conclusion is consistent with the one made from the original history of the machine operation process.
- **(A2)** With the assumption that the machine manufacturer has announced to factories the analytical algorithms such as convolutionary neural network (CNN) that will be used, each factory instead sends the differentially private trained model (e.g., differentially private convolutionary neural network, DP-CNN [10]) to the machine manufacturer. In this scenario, the machine manufacturer does not have the flexibility of adaptively choosing analytical algorithms, compared to **(A1)**.

As the machines in some areas such as the car and semiconductor industry may cost up to billions of dollars, the improvement of machines may profoundly impact the business of both the machine manufacturer and the factories that use machines. Thus, one may raise a research question that which one (**(A1)** or **(A2)**) will lead to a better data utility, given the same level of privacy.

E. Contribution

Our technical contribution can be summarized as follows.

- While there is no research effort devoted to investigating the utility difference between the above two options, this paper initiates the study of the utility difference between the above two kinds of private information-sharing mechanisms.
- After carrying out an extensive set of experiments, we find that **(A2)** is superior to **(A1)** in terms of data utility, at the cost of the flexibility in choosing arbitrary analytical algorithms. In particular, the classification accuracy by directly using differentially private models (e.g., DP-CNN) is more accurate than the classifier generated by differentially private synthetic data from DP-GAN, with approximately 0.31% $\sim$ 7.66%.

An implication in IIoT is that when the machine manufacturer has already determined the analysis tool (e.g., CNN), it would be preferred to ask the factories to return the differentially private models. In such a case, the machine manufacturer can have more accurate analysis results for future machine improvement. Nevertheless, when the machine manufacturer wants to keep the freedom of choosing arbitrary analysis tools, the machine manufacturer needs to trade the analysis accuracy for flexibility.

II. Related Work

A. Differential Privacy (DP)

In this paper, we use differential privacy to both generate synthetic data and train the privacy model. Differential privacy [11], [12], [13] comprises strong privacy guarantees for algorithms on aggregate databases. Two databases differ on a single record called neighboring databases, so the results of querying them are extremely similar. Base on this setting, if you cannot distinguish the result queried from which databases, the single record, the only difference between the two databases, will not leak the information. The strict definition of DP (ϵ-DP) is that a statistical release cannot compromise a member's privacy if their data are not in the database. Consequently, the statistical functions run on the database should not excessively rely on any individual's data. Dwork et al. [14] proposed a loose definition of DP named (ϵ, δ)-DP which allows for the probability that ϵ-DP is failed with probability δ and we show it as the following:

Definition 1. (ϵ, δ)–differential privacy. A randomized algorithm $\mathcal{M}$ takes a database as input. $\mathcal{M}$ satisfies (ϵ, δ)–DP if, for neighboring databases $\mathcal{D}_1$, $\mathcal{D}_2$ that all $\mathcal{S} \subseteq \text{Range}(\mathcal{M})$:

$$\Pr[\mathcal{M}(\mathcal{D}_1) \in \mathcal{S}] \leq \exp(\epsilon) \cdot \Pr[\mathcal{M}(\mathcal{D}_2) \in \mathcal{S}] + \delta, \quad (1)$$

where ϵ is a privacy budget that represents the degree of privacy protection and $\delta \in$ [0,1] is a probability of not satisfying differential privacy. The degree of privacy protection is higher when ϵ is smaller, that is, the utility of databases is lower. When δ = 0, $\mathcal{M}$ satisfies ϵ-DP. Given a deterministic function $f : D \rightarrow \mathbb{R}$ and differential privacy protection is achieved by adding noise to the output of f. The magnitude of noise influences on both the privacy degree and the utility of databases. Adding the quite small magnitude of noise does not provide sufficient protection. However, the excessive magnitude of noise tremendously reduces the utility of results. Therefore, f's sensitivity $\Delta f = \max_{d_1,d_2} \parallel f(d_1) - f(d_2) \parallel$ is the key parameter to determine how much noise to be added, where Δf represents the maximum impact of each record on the f's output, and d_1 and d_2 are adjacent inputs. Laplace and Gussain noises are used to be added to achieve DP guarantees and we show their mechanisms as the following:

Definition 2. Laplace mechanism. Given any function f :

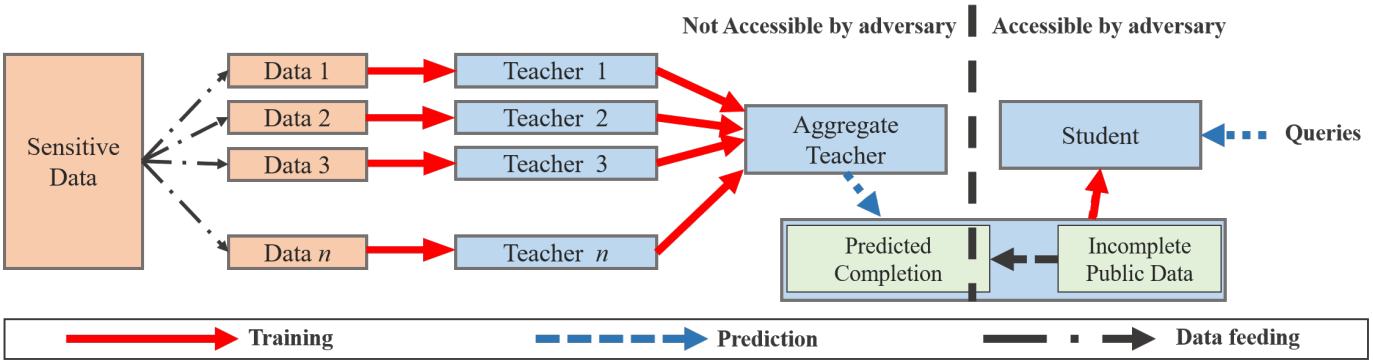


Fig. 1. The flowchart of PATE. First, we get the ensemble of teachers trained on disjoint subsets of sensitive data. Then, we train a student model with public data labeled by the ensemble.

$D \rightarrow \mathbb{R}$, the Laplace mechanism is $M_L(D) = f(D) + (Y_1, \cdots, Y_n)$, where the Y_i is independently identical distribution random noise drawn from Laplace distribution $Lap(\frac{\triangle f}{\epsilon})$.

Definition 3. Gaussian mechanism. Given any function $f : D \rightarrow \mathbb{R}$, the Gaussian mechanism is $M_G(D) = f(D) + (Y_1, \cdots, Y_n)$, where the Y_i is independently identical distribution random noise drawn from the Gaussian distribution $\mathcal{N}(0,\ \sigma^2)$ with the zero-mean and the scale showing as below:

$$\sigma \geq \sqrt{2ln(\frac{1.25}{\delta})}\frac{\triangle f}{\epsilon} \tag{2}$$

Because Gaussian mechanism can accept a more powerful composition property, both "Clean features with sloppy train" and DP-GAN use Gaussian mechanism to randomize f's output and define as $M(d_1) = f(d_1) + \mathcal{N}(0, (\Delta f\sigma)^2 I)$, where $\mathcal{N}(0, (\Delta f\sigma)^2 I)$ is a Gaussian distribution with zero mean and standard deviation $(\Delta f\sigma)^2 I$, where σ is the noise parameter and I is the identity matrix. An inherent assumption behind DP is that data records are independent. As the data generated and collected from IIoT devices might be correlated, the DP on correlated data is also developed [15].

B. Private Aggregation of Teacher Ensembles (PATE)

As shown in Fig. 1, PATE [16] partitions sensitive data into n disjoint subsets and each teacher model trains on the received data separately. After that, we get n classifiers f_i called teachers and the aggregate teacher gathers all teachers to predict the label based on the student's query. The ensemble of teachers counts the predictions for each teacher and generates the statistical result. The privacy guarantee is derived from aggregation so that it needs to add noise to the statistical result and return the prediction corresponding to the highest noisy vote to the student model:

$$f_{\text{en}}(\bar{x}) = \underset{c}{argmax}\left\{n_c(\bar{x}) + Lap\left(\frac{1}{\gamma}\right)\right\}, \tag{3}$$

where $f_{\text{en}}(\cdot)$ is the ensemble of teachers and $\bar{x}$ is an input which is the number of teachers classifying input $\bar{x}$ as class c: $n_c(\bar{x}) = |\{i\colon i \in [n], f_i(\bar{x}) = c\}|$. $Lap(a)$ is the Laplace distribution with location 0 and scale a. The privacy parameter γ affects the privacy guarantee. Instinctively, a large γ brings about a strong privacy guarantee but decreases the accuracy of the labels. Because of the additive noise on statistical results generated by teachers, the student model has privacy protection in the process of training.

When the number of teachers is small, the difference between the most votes and the second-highest number of votes is small. If the noise is arbitrarily selected, it will be hard to maintain the most votes' consistency after adding noise. Considering the utility of the student model, the noise must be strictly selected. However, this will result in the rapid consumption of privacy costs and decrease student queries. Eventually, the lack of training data with labels will cause low accuracy because of the reduction in the number of student queries. On the other hand, when the number of teachers is too large, each teacher has only a small amount of training data that makes their performance poor and finally causes the student learning badly.

C. Differentially Private SGD

Stochastic gradient descent (SGD) is an optimizer used to train the neural network. It computes the gradient of the loss function $\mathcal{L}$ w.r.t the model's parameters θ and updates θ for each training sample x^i and training label y^i:

$$\theta = \theta - \eta \cdot \nabla_\theta \mathcal{L}\ (\theta;\ x^i;\ y^i), \tag{4}$$

where η is the learning rate. It can converge faster than batch training. However, it may have information leakage via gradients. One of the "Clean features with sloppy training" achieves the privacy protection by using differentially private stochastic gradient descent (DP-SGD) [10] during optimization. Compared to the normal SGD, DP-SGD adds Gaussian noise on the gradients to achieve the privacy guarantees. To avoid the overflow occurring, it must clip the gradient before adding noise. Doing so can also prevent the exploding gradient [17] that happens when the gradient increases dramatically during training. It clips the ℓ_2 norm of each gradient in the same layer by a threshold C. Threshold C can limit the influence of individual data on the overall data and compute the sensitivity conveniently. After that, we compute the average gradients to update the parameters, which is the same as normal SGD. Finally, we use the privacy accountant to track the cumulative privacy loss. These processes iterate until it converges or the privacy budget runs out.

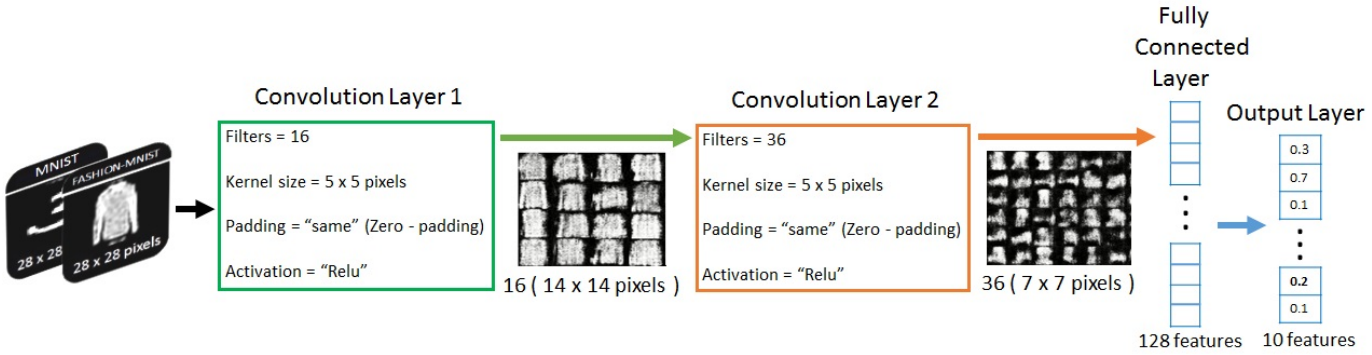


Fig. 2. The neural network architecture of the classifier for labeling synthetic data.

D. Differentially Private GAN (DP-GAN)

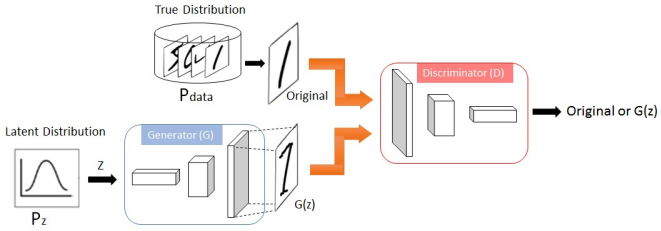


Fig. 3. The structure of generative adversarial networks (GAN). It consists of two neural networks, the generator and the discriminator. The generator takes the random noise as the input and generates the data resembling the training data. Discriminator compares synthetically generated data with the real data.

Goodfellow et al. [18] proposed a neural network architecture named generative adversarial network (GAN) shown in Fig. 3 composed of two neural networks, the generator G and the discriminator D, respectively. The generator G is in charge of generating the new synthetic data similar to the training data. The discriminator D tries to distinguish between the real data and the synthetic data generated by G. The competition between G and D, G can learn the latent distribution P_z well so that the synthetic data has similar statistical properties to the training data.

The DP-GAN proposed by Zhang et al. [19] is based on the improved WGAN [20] framework. There are various DP-GANs [21], [22]. According to Fig. 3, it shows that only D can directly access the original data, so DP-GAN adds noise on the gradients of D to achieve the privacy guarantees. However, G is still protected by differential privacy. Because any computation on the output of a differentially private mechanism does not increase the privacy leakage under the post-processing of differential privacy. Therefore, updating the parameters of G through D does not increase the privacy loss. Training G is also protected by differential privacy, and the releasing data are also secure. The process of DP-GAN is almost the same as DP-SGD. The slight difference between them is that DP-GAN also needs to update the generator G.

Recalling the process of DP-SGD, it needs to clip ℓ_2-norm of each gradient with the threshold C and then add Gaussian noise on gradients that DP-GAN conducts them too. These operations influence the training bringing about the low synthetic data quality generated by the generator. For instance, if C is too small, it will lead to excessive truncation of the gradient and slow polymerization. If C is too large, however, more noise will be added to the gradients. To enhance the performance of DP-GAN, Zhang et al. also propose some strategies for the setting of clipping bound C, and we show them below:

Basic. It sets the gradient clipping threshold of each layer to be the same.

Weight-Bias Separation. From $f(x) = wx + b$, we can intuitively observe that the weights w have a large influence on the input x, so the setting of C should be set separately for weights w and biases b.

Adaptive Clipping. Although the setting of C has been divided from the overall parameters into weights and biases, it is still set manually in each layer. Therefore, it still has a significant difference from the optimal C. To achieve the optimal C, the magnitudes of the gradients are monitored before and during training and finally set C according to the average magnitudes. We partition the training data into the private data $\mathcal{D}_{pri}$ and the public data $\mathcal{D}_{pub}$. During each training step, a batch of samples is randomly selected from $\mathcal{D}_{pub}$ and set the clipping bound as the average gradient norm w.r.t this batch of $\mathcal{D}_{pri}$. Because the clipping bound is computed from $\mathcal{D}_{pub}$ instead of being set manually. It is closer to the optimal C, but each iteration's progress direction is correct. Accordingly, this strategy accelerates the training convergence rate and has a higher data utility.

Weight Clustering. Since the weights of each layer vary greatly, the clipping bound should be different for overall weights. Therefore, Zhang et al. [19] proposed this strategy, as sketched in Algorithm 1 (see in Appendix A). First, we receive a set of gradients $\{c_{g_i}\}_i$ and each gradient forms its own group $\{(g_i, c_{g_i})\}_i$. Then, we sort each group from small to large and recursively find two groups $\hat{G}_i, \hat{G_{i+1}}$ with the most similar clipping bounds and combine into a new group. Because we clip the ℓ_2 norm for C, the clipping bound of the new group is computed as $\sqrt{c^2_{\hat{G}_i} + c^2_{\hat{G_{i+1}}}}$ using the ℓ_2 norm.

Warm-Start. Since DP-GAN adds a lot of noise during training, the convergence rate is slower than GAN. To improve the convergence rate and the utility, we extract a small proportion of $\mathcal{D}_{pub}$ (e.g., 2% of $\mathcal{D}_{pub}$ in [19]) to train several iterations

without DP. After that, based on the model trained in the above non-private manner, we use $\mathcal{D}_{pri}$ to train the model in an DP manner. In essence, this strategy can find a better weight initialization for the model training from the perspectives of the training efficiency and model utility budget by sacrificing the privacy of those data records used in the pre-training [1].

III. Our Approach

To avoid leaking sensitive data in machine learning, it in common uses the "Clean feature with sloppy training" approach to attain this goal. However, this method only releases a model of the fixed type, and third parties can not generate the corresponding model according to their needs. In recent years, Zhang et al. [19] has proposed a method with the same degree of privacy guarantees that can train the expected model according to their requirements. Suppose the accuracy of the model generated by the synthetic data is close to or higher than the model released by "Clean feature with sloppy training". In that case, we can use synthetic data extensively for various analyses and the model generated by these synthetic data with the same degree of privacy guarantees. In Section IV, we will show the performance of classifiers trained on the original data with the DP-SGD optimizer and trained on the synthetic data generated by DP-GAN normal SGD optimizer.

In this section, we show that how we train the classifier with the DP-SGD optimizer. We also show how we generate a classifier from synthetic data and evaluate its performance for the DP-GAN.

For DP-SGD, we train the classifiers on MNIST and FASHION-MNIST with the architecture: a 60-dimensional PCA projection layer, a single 1000-unit ReLU hidden layer, and a 10-unit output layer. Based on [10], PCA projection needs to access the sensitive data, so we must add noise to avoid the leakage of privacy. To maintain the overall privacy budget, ϵ is split into ϵ_{clip} and ϵ_{pca} whose noise scales are σ_{clip} and σ_{pca}, respectively.

We train the DP-GAN models with $\delta = 10^{-5}$ and various privacy budgets in this paper for DP-GAN. The DP-GAN model only generates differentially private synthetic data and does not label them. Hence, our strategy is to generate a highly accurate classifier at first and then use it to label the synthetic data generated by the generator of the DP-GAN model. To get a highly accurate classifier, we train the neural networks of the architecture shown in Fig. 2 on MNIST and FASHION-MNIST datasets whose accuracies are 99.16% and 92.58%, respectively. Then, we randomly select the same number of each class in the training data of "Clean features with sloppy training" from the synthetic data. For the sake of fairness, we train these synthetic data based on the neural network structure of "Clean features with sloppy training". We repeat to generate synthetic data and train the classifier fifty times and finally take the average accuracy.

[1]Warm-start is optional. In other words, if one cannot find any $\mathcal{D}_{pub}$ available for the pre-training, one can skip warn-start. In fact, warm-start can be seen as the transfer learning with full-model fine-tuning. Thus, even if one cannot find $\mathcal{D}_{pub}$ that shares the same distribution with the sensitive dataset and can only find $\mathcal{D}_{pub}$ that shares the somewhat similar distribution, then warm-start can still increase the training efficiency and utility.

IV. Experiments

In this section, we mainly compare the performance between the classifiers trained on the synthetic data and trained with "Clean features with sloppy training". Both methods are under an equal degree of privacy protection.

A. Experiments Setup

We trained classifiers on MNIST and FASHION-MNIST datasets. MNIST consists of 70,000 handwritten digital images of size 28×28 and divides them into 60,000 training and 10,000 test samples. FASHION-MNIST consists of 70,000 images of 10 categories: t-shirt, trousers, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot of size 28×28 divides them into 60,000 training and 10,000 test samples. Both two datasets are black and white images. All experiments are conducted using an Intel Xeon E5-2620v4 CPU, 125 GB RAM, and an NVIDIA TITAN Xp GPU with 12 GB RAM.

Each teacher and student model uses the same neural network structure in PATE: two convolution layers with max-pooling and one fully connected layer with ReLUs. The teachers can access 60,000 samples totally, of which 5,000 are used for validation. The batch size is set to 128, and the learning rate is initially set to 5. We found that when the number of teachers is 100 and 80 for MNIST and FASHION-MNIST, the student model has the highest accuracy shown in Table I. We show more experiments on number of teachers versus the student accuracy in Fig 10. (see in Appendix B). Since the FASHION-MNIST dataset is more complex than the MNIST dataset, each teacher model needs more training data so that the student model can attain higher accuracy. The amount of the training data for training the student model is a critical factor for the accuracy shown in Fig 4. We use 5000 test data labeled by the ensemble teacher to train the student model for all experiments.

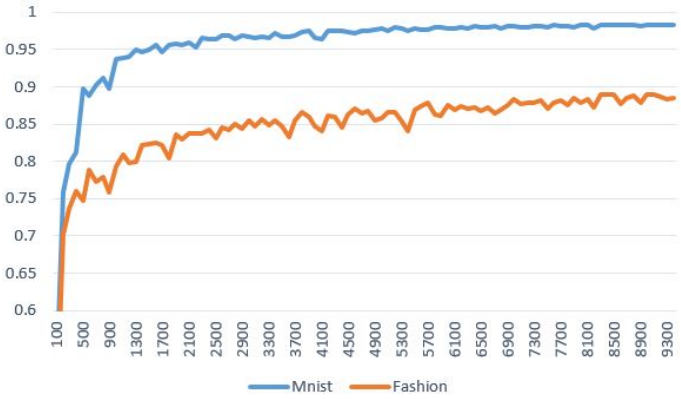


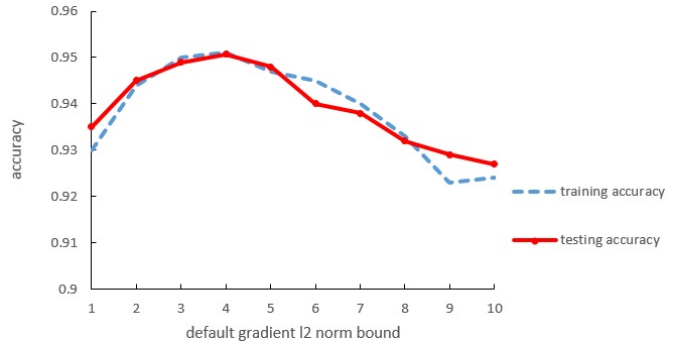
Fig. 4. The accuracy generated by the different numbers of training data under the student's training structure.

In DP-SGD, the learning rate initially sets as 0.1, linearly reduces to 0.052 in 10 epochs, and finally fixes at 0.052. To limit the sensitivity, we found that the gradient clipping threshold is set to 4 and 5 separately for MNIST and FASHION-MNIST with the best utility demonstrated in Fig 5. Besides, the total privacy budget ϵ is partitioned into ϵ_{clip} and ϵ_{pca} with

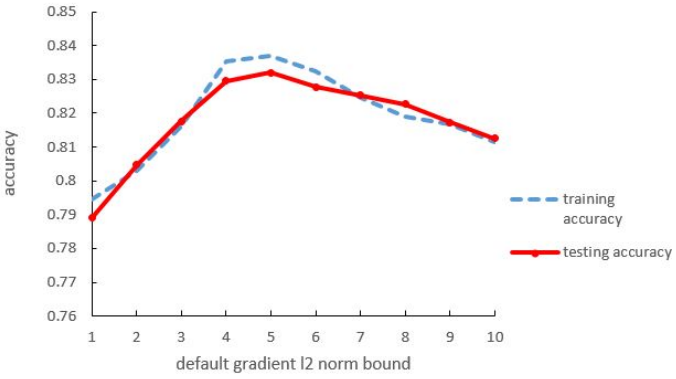
TABLE I
TRAINING PATE WITH DIFFERENT NUMBER OF TEACHERS SATISFIED $(2, 10^{-5})$-DP INFLUENCES THE PERFORMANCE OF STUDENT MODELS.

Dataset	Number of teachers	Average Accuracy of Teachers	Teacher Ensemble Accuracy	Student Accuracy
MNIST	10	97.49%	97.6%	93.33%
	100	90.27%	96.34%	**97.47%**
	250	83.7%	91.72%	92.48%
FASHION-MNIST	10	86.47%	87.72%	82.8%
	80	81.73%	85.76%	**86.78%**
	250	73.68%	74.37%	76.03%

noise scales σ_{clip} and σ_{pca} mentioned in Section III. We set $(\epsilon = 0.5,\ \sigma_{\text{clip}} = 8,\ \sigma_{\text{pca}} = 16)$, $(\epsilon = 2,\ \sigma_{\text{clip}} = 4,\ \sigma_{\text{pca}} = 7)$, $(\epsilon = 4,\ \sigma_{\text{clip}} = 3,\ \sigma_{\text{pca}} = 5)$ and $(\epsilon = 8,\ \sigma_{\text{clip}} = 2,\ \sigma_{\text{pca}} = 4)$ for both datasets.



(a) MNIST



(b) FASHION-MNIST

Fig. 5. The accuracies based on different gradient clipping thresholds C on MNIST and FASHION-MNIST.

In DP-GAN, we divide the training data into a publicly available dataset $\mathcal{D}_{pub}$ and a private dataset $\mathcal{D}_{pri}$ with a ratio of 2 : 98. We set the number of steps for updating the generator in the single iteration to be 4. We train the DP-GAN by setting the number of steps for updating generator in the single iteration$= 4$, batch size $= 64$, the initial learning rate$= 0.0002$, the coefficient of gradient penalty $\lambda = 10$, and hyper-parameters of Adam optimizer $(\alpha, \beta, \gamma) = (0.002, 0.5, 0.9)$. Based on Section II-D, there are five strategies for finding the optimal clipping bound C. The number of groups for weight clustering is set to be 5. The number of iterations of warm-start is set to 500. To limit the sensitivity, the settings of the gradient clipping threshold C for different models show as below:

1) Basic model: It is the traditional DP-GAN, we set the overall parameters of C to be 4 and 5 on MNIST and FASHION-MNIST, respectively.
2) Weight-Bias model: Because the initial parameters are very messy, it should be set the larger C at the beginning. As the training step increases, the model will tend to converge. Therefore, the setting of C should gradually become small.
3) Other strategies adopt adaptive clipping to set C through $\mathcal{D}_{pub}$.

B. Experimental Results

In this section, we show the performances on classifiers trained with "Clean features with sloppy training" and DP-GAN by varying privacy budgets ϵ on MNIST and FASHION-MNIST. Both DP-SGD and PATE belong to "Clean features with sloppy training". Besides, we perform the visual comparison of the synthetic data for five strategies of DP-GAN with the same privacy budget ϵ.

1) Performance on MNIST dataset: We train with $\delta = 10^{-5}$ and four different privacy budgets $\epsilon = \{0.5,\ 2,\ 4,\ 8\}$. The degree of privacy protection increases when ϵ reduces. We train eight classifiers: regular training, PATE, DP-SGD, and combinations of 5 strategies of DP-GAN models. We show the test accuracies of 8 classifiers in Fig 6. Original data meaning regular training; of course, it can get the highest accuracy which is more significant than 97%. Based on Equation 2., we know that the noise injection is related to the ϵ. If we add more noise during training, it will interfere severely with the parameters of the optimizer. When $\epsilon = 0.5$, we added more noise and the manual setting of the gradient clipping bound C has a large fluctuation bringing about the lower accuracy. When $\epsilon \geq 2$, the demand for noise is small. That is, it has a slight effect on the gradients. Consequently, the accuracy will become more stable and approach to the original data.

DP-SGD, it contains ϵ_{pca} and ϵ_{clip} so that the demand of the additive noise is more causing the poor performance. The poor performance of basic and weight-bias models demonstrates the importance of the gradient clipping bound C. Among all strategies of setting C, the adaptive clipping is closer to the varying gradients caused by each input than others. The adaptive clipping does not cause slow convergence and excessive truncation due to the small gradients. Furthermore, it does not add too much noise due to reducing the data utility of the generated data eventually. In addition to using adaptive clipping, the warm-start is also a good choice. Because it uses a little original data to train in several epochs without

adding noise, the latent space learned by the generator is closer to the original data. Compared to DP-GAN models without warm-start, we show images generated by generators trained with different steps shown in Fig 11 (see in Appendix C). The Model with warm-start can learn the latent space of the training data well in fewer steps than other models. The ability of generators influences the utility of the synthetic data. We visualize the synthetic data shown in Fig 7 and the qualities of images generated by estimation and warm-start models are better than others. Therefore, the performance of classifiers trained by them is higher than other DP-GAN models in Fig 6. In conclusion, training DP-GAN models with the combination of adaptive clipping and warm-staring is the most suitable method to train an accurate classifier.

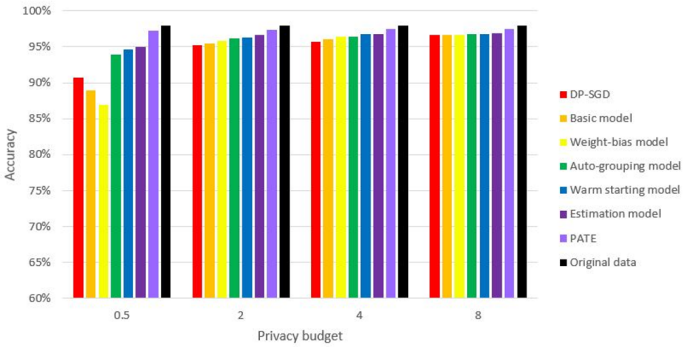
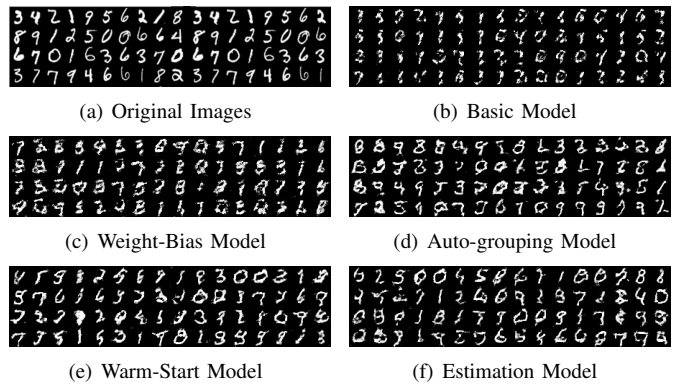


Fig. 6. Comparison test accuracies of 8 types of classifiers on MNIST. The auto-grouping model use strategies for the combination of adaptive clipping and weight clustering. The warm-start model is composed of warm-start and adaptive clipping. The estimation model consists of warm-start, adaptive clipping and weight-bias. The performance of Estimation model is closed to PATE when $\epsilon \geq 2$. However, DP-SGD belonging to "Clean features with sloppy training" cannot perform well until $\epsilon = 8$.

2) Performance on FASHION-MNIST dataset: We performed the same setting as MNIST dataset with $\delta = 10^{-5}$ and three different privacy budgets $\epsilon = \{2,\ 4,\ 8\}$ on FASHION-MNIST. Obviously, under the same degree of privacy protection, there is a significant gap in the accuracy of FASHION-MNIST compared to the MNIST. Because the FASHION-MNIST dataset is more complex than the MNIST dataset, the performance is arduously as good as MNIST. We overcome this barrier by using adaptive clipping and its accuracy is relatively high and stable. Since adaptive clipping constantly monitors the magnitude of the gradients in $\mathcal{D}_{pub}$ before and dynamically sets the clipping threshold C based on the average during training. Hence, it guarantees the maximum protection of the input sample to the gradients with minimal correlation error.



(a) Original Images (b) Basic Model

(c) Weight-Bias Model (d) Auto-grouping Model

(e) Warm-Start Model (f) Estimation Model

Fig. 7. Visual Comparison of the synthetic data generated by basic, weight-bias, auto-grouping, warm-start and estimation models trained by the same iterations under the privacy budget $\epsilon = 0.5$ and $\delta = 10^{-5}$ on MNIST. Estimation and warm-start models can generate the synthetic data being more similar with original images.

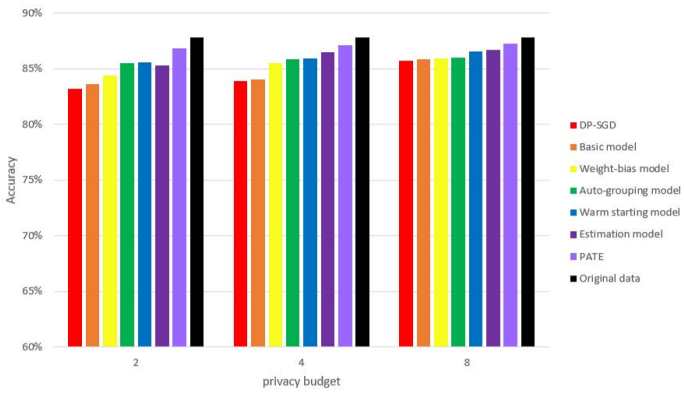
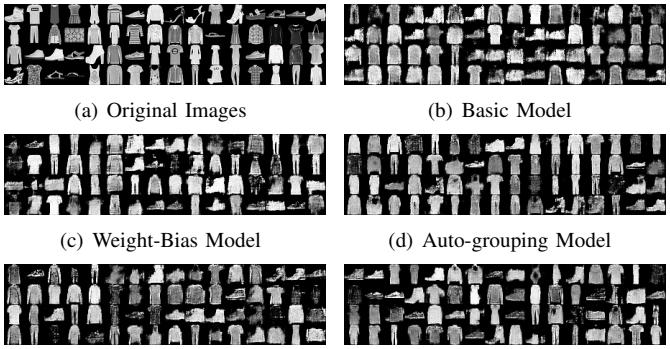


Fig. 8. Comparison test accuracies of 8 types of classifiers on FASHION-MNIST. The auto-grouping model use strategies for the combination of adaptive clipping and weight clustering. The warm-start model is composed of warm-start and adaptive clipping. The estimation model consists of warm-start, adaptive clipping and weight-bias. The performance of warm-start model shows that is closed to PATE when $\epsilon \geq 2$. It consistently shows that the utility of the synthetic data is brilliant.

As mentioned above, the amount of noise affecting the utility of data is connected to ϵ. When the privacy budget ϵ growing, the accuracy of each model increases shown in Fig 8. DP-SGD consistently cannot get the brilliant performance caused by partitions of ϵ for the gradient clipping and PCA projection. As ϵ increases, performances of DP-GAN models are closer to PATE and regular training. The reason for basic and weight-bias models gaining low accuracy mentioned in Section IV-B1 is that both do not use warm-start. We also perform the synthetic images generated by different DP-GAn models trained with and without warm-start with the privacy budget $\epsilon = 2$ and different steps shown in Fig 12 (see in Appendix C). Consider the results on MNIST; warm-start plays a vital role in the accuracy of classifiers trained on the synthetic data. We also perform the synthetic data generated by 5 DP-GAN models under the same privacy budget shown in Fig 9. The images generated by auto-grouping and warm-start models are closer to the original images.

To train the accurate classifier, the performance of both "Clean features with sloppy training" and DP-GAN are almost the same on both MNIST and FASHION-MNIST. However, using DP-GAN to generate synthetic data with appropriate strategies has widespread applications. For example, we can do statistical analysis on synthetic data. "Clean features with sloppy training" can only be used to predict the user's data. In practice, instead of releasing secure models, the synthetic data can be applied in broad fields.



(a) Original Images (b) Basic Model (c) Weight-Bias Model (d) Auto-grouping Model (e) Warm-Start Model (f) Estimation Model

Fig. 9. Visual Comparison of the synthetic data generated by basic, weight-bias, auto-grouping, warm-start and estimation models trained by the same iterations under the privacy budget $\epsilon = 2$ and $\delta = 10^{-5}$ on FASHION-MNIST.

V. Conclusion

In our work, we use two types of Clean feature with sloppy training" based on the location of noise injection and compare it with the classifier generated by the synthetic data under the same level of privacy protection. According to the experimental results, we can observe that the setting of the clipping threshold has a considerable influence on accuracy. In DP-SGD, it not only sets the clipping threshold manually but adds more noise than other models during training, so its accuracy is usually the lowest. However, in DP-GAN, we dynamically adjust the value of the clipping threshold in each step by using adaptive clipping. This strategy prevents the clipping threshold from being too small, causing slow aggregation; it also prevents it from being too large, adding more noise. Therefore, when the noise is added to the gradients, the model generated by differentially private synthetic data has higher utility. Another Clean feature with sloppy training" adds noise to the voting's statistical results, so we focus on whether there are enough correct labels. We increase the number of teachers to ensure that the correct label has an overwhelming number of votes. Therefore, the model generated by this method has a higher utility than the model developed by differentially private synthetic data.

On the other hand, due to the business secrecy, the factory owners are not willing to share the data collected from IIoT with the other one. However, from our empirical experiments, one can know that a DP synthetic dataset learned from DP-GANs or a DP deep neural network can be a surrogate for the shared data. With the DP synthetic datasets or models available for the public, the factory owners can also benefit from such a data sharing (e.g., the external machine learning experts can make an improvement to the production process when the data about the production process is available) without compromising the data privacy and business secrecy. Moreover, though DP-GANs has slightly worse utility than DP-CNNs, the factory owner may still prefer DP-GANs because of the high versatility.

Acknowledgments

Chia-Yi Hsu and Chia-Mu Yu were supported by MOST 110-2636-E-009-018, and we also thank National Center for High-performance Computing (NCHC) of National Applied Research Laboratories (NARLabs) in Taiwan for providing computational and storage resources.

Appendix A
The algorithm of Weighting Clustering.

Algorithm 1. summarize that how to accomplish weight-clustering group into k categories.

Algorithm 1 Weight-Clustering

Require: Number of groups: k, A set of gradients: $\{c_{g_i}\}_i$
Ensure: Grouping of parameters, G

G ,$\{(g_i, c_{g_i})\}_i$
while $|G| > k$ **do**
  // Sort G from small to large by c_{g_i}
  $\hat{G}_1, \hat{G}_2, \cdots, G_{\hat{|G|-1}}, G_{\hat{|G|}}, Sort(G)$
  $\hat{G}_i, G_{\hat{i+1}}, max\left(\frac{c_{\hat{G}_1}}{c_{\hat{G}_2}}, \frac{c_{\hat{G}_2}}{c_{\hat{G}_3}}, \cdots, \frac{c_{G_{\hat{|G|-1}}}}{c_{G_{\hat{|G|}}}}\right)$
  // Merge them and update the clipping threshold
  merge $\hat{G}_i, G_{\hat{i+1}}$ with clipping bound as $\sqrt{c^2_{\hat{G}_i} + c^2_{G_{\hat{i+1}}}}$
end while
return G

Appendix B
More sets of number of teachers versus the accuracy of student models

In Fig 10, we conducted more experiments with different number of teachers. The number of teaches significantly affect on the accuracy. Fixed the total number of the training data, each teacher needs more amount of the training data for the complicated dataset.

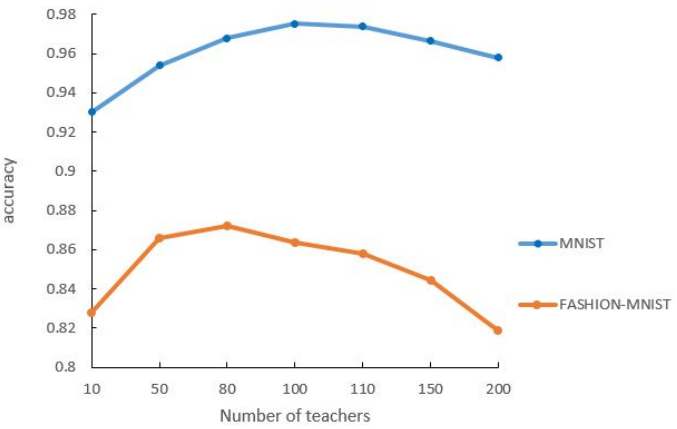


Fig. 10. The accuracies of student models trained with different number of teachers on MNIST and FASHION-MNIST. 100 and 80 teachers for MNIST and FASHION-MNIST can attain the highest accuracy.

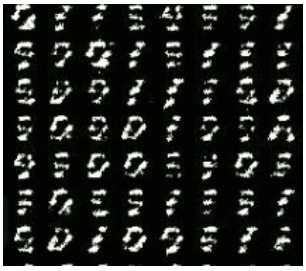
APPENDIX C
VISUAL COMPARISON OF SYNTHETIC IMAGES ON MNIST AND FASHION-MNIST.

Fig 11. and Fig 12. showed that using warm-start in the training learned more quickly than other models. Thus, the performance of generators trained with warm-start is better than others when we fixed the training iterations. As the result, classifiers trained on the synthetic data associated with warm-start gain higher accuracy.

(a) 0 step of Estimation model

(b) 360^{th} step of Estimation model

(c) 1460^{th} step of Auto-grouping model

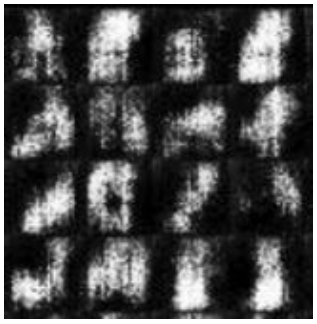


(d) 2040^{th} step of Weight-Bias model

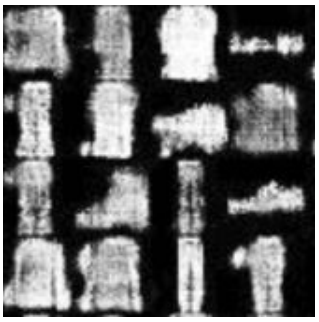
Fig. 11. Visual comparison of synthetic images generated by generators trained with different steps, the privacy budget $\epsilon = 2$, and disparate combinations of strategies on MNIST. Only Estimation model uses warm-start. It shows that the models trained without warm-start need more steps to learn the features of the training data.

APPENDIX D
NOTATION TABLE

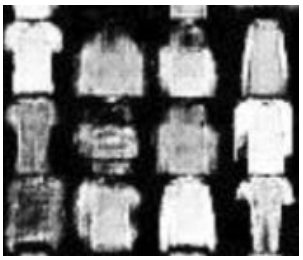
Notation	Description
ϵ, δ	privacy parameters in DP
Δ_f	global sensitivity
$Lap(b)$	Laplace distribution with zero mean and scale b
$\mathcal{N}(a, b)$	Gaussian distribution with mean a and variance b
θ	model parameter
$\mathcal{L}$	loss function
η	learning rate in SGD and DP-SGD
C	clipping threshold in DP-SGD
G	generator in GAN and DP-GAN
D	discriminator in GAN and DP-GAN
$\hat{G}$	group of similar gradients in Zhang et al. [19]



(a) 0 step of Estimation model



(b) 360^{th} step of Estimation model



(c) 1460^{th} step of Auto-grouping model

(d) 2040^{th} step of Weight-Bias model

Fig. 12. Visual comparison of synthetic images generated by generators trained with different steps, the privacy budget $\epsilon = 2$, and disparate combinations of strategies on FASHION-MNIST. Only Estimation model uses warm-start. It shows that the models trained without ting need more steps to learn the features of the training data.

REFERENCES

[1] M. Ma, W. Lin, J. Zhang, P. Wang, Y. Zhou, and X. Liang, "Toward energy-awareness smart building: Discover the fingerprint of your electrical appliances," in *IEEE Transactions on Industrial Informatics*, 2018.

[2] J. Wang, J. Liu, , and N. Kato, "Networking and communications in autonomous driving: A survey," in *IEEE Communications Surveys and Tutorials*, 2019.

[3] K. Morioka, J. H. Lee, and H. Hashimoto, "Human-following mobile robot in a distributed intelligent sensor network," in *IEEE Transactions on Industrial Informatics*, 2004.

[4] B. Jiang, J. Li, G. Yue, and H. Song, "Differential privacy for industrial internet of things: Opportunities, applications and challenges," in *arXiv:2101.10569*, 2021.

[5] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: Challenges, opportunities, and directions," in *IEEE Transactions on Industrial Informatics*, 2018.

[6] X. Zhang, X. Chen, J. Liu, and Y. Xiang, "Deeppar and deepdpa: Privacy preserving and asynchronous deep learning for industrial iot," in *IEEE Transactions on Industrial Informatics*, 2020.

[7] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Differentially private asynchronous federated learning for mobile edge computing in urban informatics," in *IEEE Transactions on Industrial Informatics*, 2020.

[8] B. Zhao, K. Fan, K. Yang, Z. Wang, H. Li, and Y. Yang, "Anonymous and privacy-preserving federated learning with industrial big data," in *IEEE Transactions on Industrial Informatics*, 2021.

[9] I. Goodfellow, M. Pouget-Abadie, Jean; Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," in *Proceedings of the International Conference on Neural Information Processing Systems (NIPS)*, 2014.

[10] 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*, 2016, pp. 308–318.

[11] C. Dwork, "A firm foundation for private data analysis," *Communications of the ACM*, vol. 54, no. 1, pp. 86–95, 2011.

[12] ——, "The differential privacy frontier," in *Theory of Cryptography Conference*. Springer, 2009, pp. 496–502.

[13] 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.

[14] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in

Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, 2006, pp. 486–503.
[15] T. Zhang, T. Zhu, P. Xiong, H. Huo, Z. Tari, and W. Zhou, "Correlated differential privacy: Feature selection in machine learning," in *IEEE Transactions on Industrial Informatics*, 2020.
[16] N. Papernot, M. Abadi, U. Erlingsson, I. Goodfellow, and K. Talwar, "Semi-supervised knowledge transfer for deep learning from private training data," in *International Conference on Learning Representations (ICLR)*, 2016.
[17] R. Pascanu, T. Mikolov, and Y. Bengio, "Understanding the exploding gradient problem."
[18] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014.
[19] X. Zhang, S. Ji, and T. Wang, "Differentially private releasing via deep generative model (technical report)," *arXiv preprint arXiv:1801.01594*, 2018.
[20] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*. PMLR, 2017, pp. 214–223.
[21] U. Tantipongpipat, C. Waites, D. Boob, A. Ankit Siva, and R. Cummings, "Differentially private mixed-type data generation for unsupervised learning," in *arXiv:1912.03250*, 2019.
[22] R. Torkzadehmahani, P. Kairouz, and B. Paten, "Differentially private synthetic data and label generation," in *arxiv:2001.09700*, 2020.

Yen-Ting Chen received his MS degree from National Chung Hsing University, Taiwan. His research interests include smart grid security and differential privacy.

Chia-Yi Hsu received her MS degree from National Chung Hsing University, Taiwan. She had an academic stay in IBM Thomas J. Watson Research Center from 2018 to 2019. She received the MS thesis award from Chinese Cryptology and Information Security Association. She is pursuing the Ph.D. degree at National Yang Ming Chiao Tung University, Taiwan. Her research interests include smart adversarial examples and differential privacy.

Chia-Mu Yu is currently an assistant professor at National Yang Ming Chiao Tung University, Taiwan. He had academic visits at IBM Thomas J. Watson research center, Harvard University, Imperial College London, University of Padova, and the University of Illinois at Chicago. He received Hwa Tse Roger Liang Junior Chair Professor, MOST Yong Scholar Fellowship, ACM/IICM K. T. Li Young Researcher Award, Observational Research Scholarship from Pan Wen Yuan Foundation, and MOST Project for Excellent Junior Research Investigators, Taiwan. He serves as an Associate Editor for IEEE Internet of Things Journal. His research interests include differentially private mechanism design, cloud storage security, and IoT security.

Mahmoud Barhamgi is an Associate Professor (HDR) at Claude Bernard Lyon 1 University, France, and is a data and software engineering researcher with several years of experience in research problems that directly touch the lives of people. His research interests are broadly in data and software engineering, particularly in the areas of service-oriented architectures, privacy-preserving data integration and analytics and cyber-physical systems security and privacy.

Charith Perera received the B.Sc. (Hons.) degree in computer science from Staffordshire University, U.K., the M.B.A. in business administration from the University of Wales, Cardiff, U.K., and the Ph.D. degree in computer science from The Australian National University, Canberra, Australia. He is currently a Senior Lecturer with Cardiff University, U.K. Previously, he worked at the Information Engineering Laboratory, ICT Centre, CSIRO. His research interests include the Internet of Things, sensing as a service, privacy, middleware platforms, and sensing infrastructure. He is a member of the ACM.