DP-CTGAN: Differentially Private Medical Data Generation using CTGANs

Mei Ling Fang $^{\star 1,2}$, Devendra Singh Dhami $^{\star 2,3}$, and Kristian Kersting[2,3]

[1] Merck KGaA, Darmstadt
[2] Technical University of Darmstadt, Germany
[3] Hessian Center for AI (hessian.AI)
julie.fang@merckgroup.com, {devendra.dhami,kersting}@cs.tu-darmstadt.de

Abstract. Generative Adversarial Networks (GANs) are an important tool to generate synthetic medical data, in order to combat the limited and difficult access to the real data sets and accelerate the innovation in the healthcare domain. Despite their promising capability, they are vulnerable to various privacy attacks that might reveal information of individuals from the training data. Preserving privacy while keeping the quality of the generated data still remains a challenging problem. We propose DP-CTGAN, which incorporates differential privacy into a conditional tabular generative model. Our experiments demonstrate that our model outperforms existing state-of-the-art models under the same privacy budget on several benchmark data sets. In addition, we combine our method with federated learning, enabling a more secure way of synthetic data generation without the need of uploading locally collected data to a central repository.

Keywords: Generative Adversarial Networks · Medical Data Generation · Differential Privacy · Federated Learning

1 Introduction

Machine learning has taken giant strides in recent years and has found wide applications affecting everyday life ranging from traffic accident prevention [25], spam filtering and phishing attacks detection [23] to several health care related tasks such as cancerous tumor detection [15], Parkinson's detection [8], drug-drug interaction prediction and discovery [7], and mortality prediction [5]. Since most clinical models are built on confidential patient records, data sharing is exceptionally rare. Protecting patient privacy is an essential obligation for researchers, but this also creates a bottleneck for fast, open, and accessible machine learning. Synthetic medical data that protects patients confidentiality could pave a way for machine learning to be extensively applied in high impact medical problems and enable researchers to design a new generation of reproducible clinical decision support models, along with standardized performance benchmarks for new

* equal contribution

methods. Due to these issues, generation of synthetic data sets has been studied extensively [4,24] and this area of research has received a significant push [11,19] after the introduction of generative adversarial networks (GANs) [13] but most of the research has focused on image data and the issue of privacy in GAN based models has not been extensively addressed.

Furthermore, the majority of existing research efforts in generating synthetic data focus on the performance of the synthetic data and ignore potential data leakage from the published model, which might suffer from adversarial attacks. Linkage attacks can be launched by linking a target record to a single or group of records in a sensitive medical data set without direct identifiers. Membership inference attacks can be instantiated against deep generative models to identify whether a specific data record was used for training. Sensitive attributes of an individual such as income or disease history can be inferred.

To remedy this drawback, in this work, we go beyond the image data and propose DP-CTGAN, which incorporates differential privacy (DP) [9] into a conditional tabular generative adversarial network, CTGAN [31] to generate medical data[4]. We achieve DP by clipping the training gradient thereby bounding the gradient norms and injecting carefully calibrated noise. This enables DP-CTGAN to generate "secure" synthetic data, which can be shared freely among researchers without privacy issues. We also acclimatize our model to federated learning, a decentralized form of machine learning [17], and introduce federated DP-CTGAN (FDP-CTGAN). This enables a more secure way of generating synthetic data without the need of uploading locally collected data to a central repository. We differ from the state-of-the-art GAN models such as PATE-GAN [16] and DPGAN [30] as we make use of a conditional generator and are thereby able to capture different distributions present in the data.

Overall, we make the following important contributions: (1) We introduce a differentially private CTGAN capable of generating secure tabular medical data. (2) We adapt our model to the federated learning setting thereby providing a more secure way of medical data generation. (3) We outperform several state-of-the-art generative algorithms on several benchmark data sets thereby empirically proving the effectiveness of our proposed models.

The rest of the paper is organized as follows: after briefly reviewing the related work, we present our proposed DP-CTGAN and FDP-CTGAN models. We then present our extensive empirical results in several different real data sets before concluding by outlining future research directions.

2 Related Work

In the past decade, synthetic data are generated using statistical approach [3,27]. Recently, generative models using GANs and its variations have been widely adopted due to its outstanding performance, flexibility and usability in generating tabular data, such as medical records. TableGAN [21] uses deep convolutional

[4] we consider only tabular medical data set generation.

GAN [22] with an auxiliary classifier to produce statistically similar tabular data while preserving privacy. CorGAN [26] builds on pretrained denoising autoencoders [28] and 1-dimensional convolutional GANs, which captures important inter-correlation between features. HA-GAN [6] took a different approach and uses symbolic knowledge representation derived from human experts as a constraint in training Wasserstein GANs [14].

CTGAN [31] built upon Wasserstein GAN with gradient penalty in the PacGAN [18] framework and introduced several innovative preprocessing steps. Motivated by the success of this mechanism, in our work, we extend the current framework of CTGAN by incorporating the differential privacy, in the hope of making the synthetic data generation more robust, secure semantic-rich and real-world usable. As for the state-of-the-art, DPGAN [30] and PATE-GAN [16] are among the most successful endeavors that incorporate differential privacy in synthetic data generation. The former injects noise to gradients during training, while the latter builds GANs on top the Private Aggregation of Teacher Ensembles framework [20] which provides a tighter privacy bound. A detailed discussion on other privacy preserving GAN architecture can be found in [10].

3 DP-CTGAN

We now introduce our our proposed model, `DP-CTGAN` (see fig.1). Before describing the architecture, it is important to justify the choice of using a CTGAN. The unique properties of tabular data pose difficulties for GANs to learn the tabular data distribution. These properties include correlated features, mixed data types such as discrete or continuous features, difficulty in learning from highly sparse vectors and potential mode collapse due to high class imbalance. To mitigate these issues, we choose CTGAN as the underlying generative model.

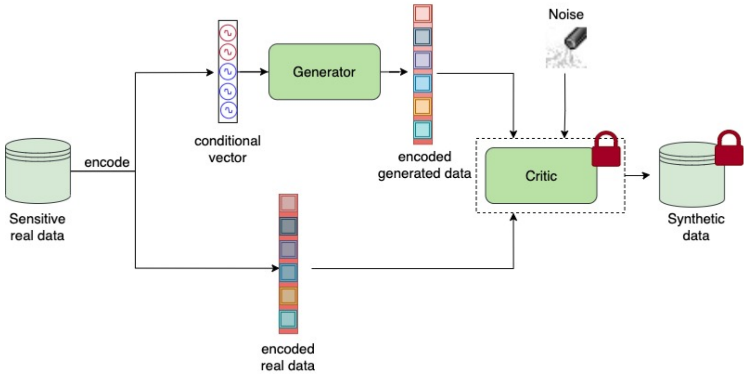

Fig. 1: `DP-CTGAN`. Sensitive training data is fed into a conditional generator to generate samples which can evenly cover all possible discrete values. At the same time, random perturbation is added to the critic to enforce privacy protection.

In DP-CTGAN, we employ the privacy accountant [1] to track the privacy loss and incorporate the differential private framework within a CTGAN model to capture correlated feature patterns as well as the complicated distributions. For preprocessing, mode-specific normalization is used for continuous columns so that the representation of the data learns the complicated distributions. Conditional generator tackles the imbalanced categorical columns and enables more efficient and even data generation. At the same time, fully-connected networks in generator and critic (or discriminator) are deployed to capture all possible correlations between columns. One of the intuitive approaches is to add noises in both the generator $\mathcal{G}$ and the critic $\mathcal{C}$. However, due to the minimax game of GAN's formulation, such an approach will increase the difficulties for convergence and privacy loss estimation, which lead to performance degradation of the models. We follow the rationale of [32] and opt to add random perturbation only in training $\mathcal{C}$. Since only the critic has access to the real data, it will be sufficient to control the privacy loss in training $\mathcal{C}$. In addition, the architecture of the critic in CTGAN is simpler than the generator, which utilizes batch normalization and residual layer to increase the performance. Thus, the critic has a relatively smaller number of parameters, which makes it easier to tightly estimate the privacy loss. We present DP-CTGAN training in algorithm 1.

Algorithm 1 Train DP-CTGAN

```
Input: Training data $\mathcal{D}_{train}$, conditional generator parameters $\Phi_G$, critic parameters $\Phi_C$, batch size $m$, step size $s$, gradient clipping bound $\mathcal{C}$, noise scale $\sigma$, privacy budget ($\epsilon_0$, $\delta_0$)
Output: parameters $\Phi_G$ of a differentially private generator $\mathcal{G}$
procedure TRAIN
    while $\epsilon \leq \epsilon_0$ do
        for $1 \leq j \leq m$ do
            $N_d \leftarrow$ number of discrete columns from $\mathcal{D}_{train}$
            $d_i \leftarrow$ one hot discrete vector                                ▷ $1 \leq i \leq N_d$
            create masks $\{m_1, \ldots, m_{i^*}, \ldots, m_{N_d}\}_j$
            create conditional vectors $cond_j$ from masks
            $z_j \sim \mathcal{MVN}(0, I)$                    ▷ sample from multi-variable normal dist.
            $\hat{r_j} \leftarrow$ Generator $(z_j, cond_j)$                      ▷ generate synthetic data
            $r_j \sim$ Uniform $(\mathcal{D}_{train}|cond_j)$                            ▷ get real data
            for $1 \leq k \leq s$ do
                sample $cond_k^j$, fake data $\hat{r_k^j}$, and real data $r_k^j$
            $\mathcal{L_C} \leftarrow \frac{1}{s}\sum_{k=1}^{s}(\text{Critic}(\hat{r}_k^j, cond_k^j) - \text{Critic}(r_k^j, cond_k^j)) + \mathcal{L_{GP}}$
            $\xi \sim \mathcal{N}(0, (\sigma C)^2\mathcal{I})$                                  ▷ generate noise
            $\Phi_C \leftarrow \Phi_C - 0.0002 \times \text{Adam}(\nabla_{\Phi_c}(\mathcal{L_C} + 10\mathcal{L_{GP}} + \xi))$
            $\mathcal{L_G} \leftarrow \frac{1}{m}\sum_{j=1}^{m}\text{CrossEntropy}(\hat{d}_{i^*,j}, m_{i^*}) - \frac{1}{m/s}\sum_{k=1}^{m/s}\text{Critic}(\hat{r}_k^s, cond_k^s)$
            $\Phi_G \leftarrow \Phi_G - 0.0002 \times \text{Adam}(\nabla_{\Phi_G}\mathcal{L_G})$
            $\epsilon \leftarrow$ query $\mathcal{A}$ with $\delta_0$                ▷ compute cumulative privacy loss
    return $\Phi_G$
```

We start with defining the differential privacy budget (ϵ_0, δ_0). The number of discrete columns in the underlying data N_d is identified and a 1-hot vector for each of the discrete columns [**lines 4-5**] is created. Then masks that provide information about the required discrete variables are created [**line 6**] and conditional vectors are sampled from these masks [**line 7**]. These conditional vectors force the generator to generate samples from the required discrete variables. To model the continuous variables, we sample from a multi-variate normal distribution [**line 8**]. Then synthetic data is created, while real data is sampled from an uniform distribution with specified constraints [**lines 9-10**]. Note that the generator is conditional in nature and can be interpreted as the conditional distribution of rows given that particular value at that particular column. A critic is then used to access the conditional distribution of generated data with respect to real data along with gradient penalty to avoid mode collapse [**line 13**].

To incorporate privacy, we sample noises from the normal distribution. The training gradients of the critic are then clipped along with the injection of sampled noise thereby bounding the gradient norms [**lines 14-15**] ensuring the sensitivity is bounded. The generator is then used to create synthetic samples [**lines 16-17**]. Finally, after each iteration, we use a privacy accountant $\mathcal{A}$ to track the cumulative privacy loss [**line 18**]. This process iterates until reaching the privacy budget or convergence.

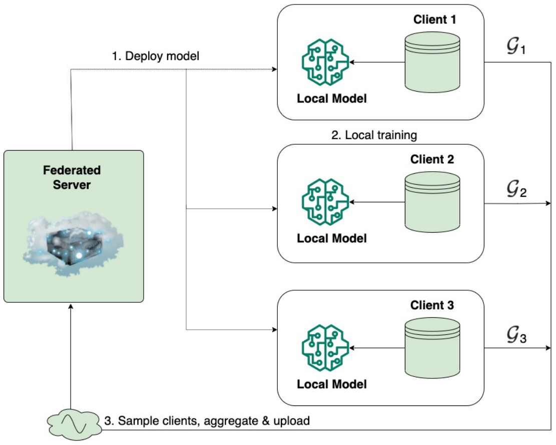

Fig. 2: Architecture of proposed **Federated** `DP-CTGAN`.

3.1 Federated DP-CTGAN

Traditionally, in federated learning, data is collected from different edge devices such as smartphones, censors or laptops, which will then be uploaded to a centralized server to train a machine learning model. However, such a model has access to the aggregated user-sensitive data, which makes it easier for attackers to break into the centralized server and steal the critical data.

We adapt our model to federated learning by incorporating the client sided differential privacy preserving federated optimization proposed by [12]. Each client has its own data set. After all clients have finished training, a set of clients are sampled randomly and their parameters are aggregated to approximate the true distribution parameters. This preserves the data privacy of the client even if the model is published. We present the overview of the federated DP-CTGAN (FDP-CTGAN) framework in Fig. 2. In each training round, the federated server deploys an initial model to each client. Local data set in the local client will be utilized for training the model. After the training is completed, the learned parameters of the generator will be aggregated with random mechanism and distributed back to the server.

Algorithm 2 Train FDP-CTGAN

```
Input: Training data $\mathcal{D}_{train}$, conditional generator parameters $\Phi_G$, critic parameters
    $\Phi_C$, batch size $m$, step size $s$, gradient clipping bound $\mathcal{C}$, noise scale $\sigma$, privacy
    budget ($\epsilon_0$, $\delta_0$), the set of clients (data owners) N
Output: parameters $\Phi_G$ of a differentially private generator $\mathcal{G}$
procedure TRAIN
    initialize generator parameters $\Phi_G$ and critic parameters $\Phi_C$
    for each $n \in N$ do
        train $n$ using algorithm 1
        store parameters $\Phi_{G_n}$ and $\Phi_{C_n}$
    $\Phi_G \leftarrow$ sample and aggregate a batch of stored parameters
    return $\Phi_G$
```

The algorithm for FDP-CTGAN is presented in algorithm 2 and the steps are:

1. The federated server initializes a DP-CTGAN model and sends it to all clients.
2. In each round of generator training, generator is updated after the training process of the critic is completed.
3. After all clients participate in the training, aggregate the learned parameters from a pool of randomly sampled clients and distribute the parameters back to the server.

We make use of the privacy accountant $\mathcal{A}$ as in algorithm 1 in order to keep track of the privacy loss of FDP-CTGAN. In this federated setting, the central server averages randomly selected client models after each communication round, in order to hide each client's contribution in the learning process and thus achieves the differential privacy on the client side.

4 Experiments

In this section, we first present the details of our experiment setup and then report the results of the data quality as well as compare with different state-of-the-art models. We aim to answer the following research questions: **Q1:** Can we effectively incorporate differential privacy in conditional tabular generative models? **Q2:** Does an extension to the federated learning setting beneficial for privacy-preservation?

4.1 Baselines and Datasets

We compare our models with various state-of-the-art methods. **1. CTGAN** [31]: is used as a non-differentially private baseline. **2. PATE-GAN** [16]): uses the Private Aggregation of Teacher Ensembles (PATE) framework to obtain high quality private synthetic data. **3.DPGAN** [30]: applies a combination of designed noise and clipping of weights, and uses the Wasserstein distance as an approximation of distance between real and generated probability distributions.

To evaluate the performance, we use 9 real data sets mostly (8/9) from medical domain. Out of all, 3 data sets adult, breast and seizure are from UCI repository [2], while cardio and cervical are from Kaggle. We also choose 4 real medical data sets specified in [6]: adni, mimic, nephrotic and ppmi.

4.2 Experimental Setup

We split the data set to train $\mathcal{D}_{train}$ and test $\mathcal{D}_{test}$ sets. First, we train the generative model using $\mathcal{D}_{train}$ and produce a synthetic data set $\mathcal{D}_{syn}$. We set $|\mathcal{D}_{train}| = |\mathcal{D}_{syn}|$. Then, we train various classifiers with $\mathcal{D}_{train}$ and $\mathcal{D}_{syn}$ respectively. We then evaluate each model's performance using $\mathcal{D}_{test}$ and average the result. The generative model is trained for 300 epochs with a batch size of 500. The differential privacy parameters used are given in table 1.

Table 1: Differential Privacy configurations for our proposed methods

	adni	adult	breast	cardio	cervical	mimic	nephrotic	ppmi	seizure
clip_coef	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.15
σ (noise scale)	3	1	4	1	8	2	3	2	1
ϵ	2.5	3	4	2	2	1	2.5	3	4.7
δ	1e-5	1e-5	1e-5	1e-5	1e-5	1e-5	1e-5	1e-5	1e-5

4.3 Evaluation Metrics

Given that synthetic data aims to replace the real data, where the distribution of synthetic data should approximate the real one as possible, it is however difficult to empirically compare the distribution of generative models [29]. In our evaluation, we address this problem by training predictive binary machine learning models on $\mathcal{D}_{syn}$ and test it on $\mathcal{D}_{test}$, so that we can compare the efficacy of classification tasks using AUROC and AUCPR. We aggregate the metrics of multiple prediction models to evaluate the synthetic data generators. By comparing the performance, we see how well the generative model capture the characteristic of the real data. Fig. 3 illustrates the evaluation framework.

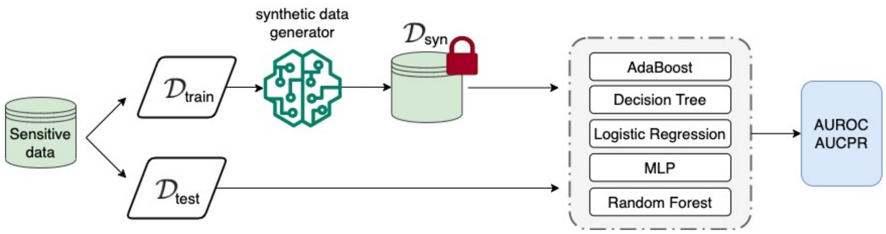

Fig. 3: Evaluation framework on synthetic data sets

4.4 Experimental Results

We are now ready to answer the posed research questions. We make our code publicly available: `https://github.com/juliecious/CTGAN/tree/DP`.

Q1. Incorporating DP in CTGAN: Tables 2 and 3 present the results of using `DP-CTGAN` to generate differentially private synthetic data. We can observe that in majority of the data sets, the performance decreases when compared to vanilla CTGAN with no privacy-preservation as expected. When compared to other privacy preserving GAN models, `DP-CTGAN` easily outperforms, thus answering **Q1:** Incorporating privacy in CTGAN leads to higher quality synthetic data compared to state-of-the-art GAN methods and is an effective alternative.

Table 2: Performance comparison (AUC-ROC)

	CTGAN	PATE-GAN	DPGAN	DP-CTGAN	FDP-CTGAN
adni	0.5052	-	0.5565	**0.5593**	0.5453
adult	**0.6739**	0.5008	0.5189	05097	0.5044
breast	**0.6820**	-	0.3740	0.6601	0.4694
cardio	0.4958	0.4648	0.4876	**0.6827**	0.5298
cervical	**0.6412**	0.4920	0.4617	0.5084	0.4956
mimic	**0.6992**	0.5871	0.5284	0.6312	0.5101
nephrotic	0.7188	-	-	**0.7292**	0.6354
ppmi	**0.6251**	0.4605	0.4784	05240	0.4712
seizure	0.4912	0.4869	0.4902	0.5000	**0.5187**

Q2. Adapting `DP-CTGAN` to federated setting: The results of extending `DP-CTGAN` to the federated setting is also shown in tables 2 and 3. It can be concluded from the performance that `FDP-CTGAN` achieves a very similar performance to the `DP-CTGAN` and also outperforms the baselines in most cases. This answers **Q2:** adapting `DP-GAN` to the federated setting is certainly beneficial.

5 Conclusion

We proposed a differentially private framework for synthetic medical data generation using CTGANs. The model aimed to capture the complicated distribution of the columns and reproduce an approximate synthesized version. We empirically show that our model can learn better distribution and thus outperform the state-of-the-art models in all scenarios. Furthermore, we attempt to find a flexible

Table 3: Performance comparison (AUC-PR)

	CTGAN	PATE-GAN	DPGAN	DP-CTGAN	FDP-CTGAN
adni	0.2603	-	0.2452	**0.2946**	0.2742
adult	**0.8511**	0.7500	0.7633	0.7653	0.7515
breast	**0.7619**	-	0.5520	0.7377	0.6416
cardio	0.5072	0.4825	0.4987	**0.6709**	0.5297
cervical	**0.1714**	0.1343	0.0776	0.1325	0.1207
mimic	**0.4210**	0.3692	0.3421	0.3833	0.3474
nephrotic	0.9081	-	-	**0.9330**	0.8729
ppmi	**0.4889**	0.3478	0.3453	0.3984	0.3692
seizure	0.4017	0.2000	0.1992	**0.4086**	0.3602

yet secure way to learn the distribution of locally stored data under the federated learning framework with calibrated randomized mechanism. Deriving theoretical justification on the privacy-preserving performance of DP-CTGAN and improving the performance of FDP-CTGAN with tighter privacy bound is an important future direction. Furthermore, applying our proposed methods on large-scale medical data sets is an interesting future avenue.

Acknowledgements The authors thank the anonymous reviewers for their valuable feedback. This work was supported by the ICT-48 Network of AI Research Excellence Center “TAILOR” (EU Horizon 2020, GA No 952215) and the Nexplore Collaboration Lab “AI in Construction” (AICO). It benefited from “safeFBDC - Financial Big Data Cluster” (FKZ:01MK21002K), funded by the BMWK as part of the GAIA-X initiative, and the HMWK cluster projects “The Third Wave of AI” and “The Adaptive Mind”.

References

1. Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. In: CCS (2016)
2. Asuncion, A., Newman, D.: Uci machine learning repository (2007)
3. Aviñó, L., Ruffini, M., Gavaldà, R.: Generating synthetic but plausible healthcare record datasets. arXiv preprint arXiv:1807.01514 (2018)
4. Buczak, A.L., Babin, S., Moniz, L.: Data-driven approach for creating synthetic electronic medical records. BMC medical informatics and decision making (2010)
5. Deprez, P., Shevchenko, P.V., Wüthrich, M.V.: Machine learning techniques for mortality modeling. European Actuarial Journal (2017)
6. Dhami, D.S., Das, M., Natarajan, S.: Beyond simple images: Human knowledge-guided gans for clinical data generation. In: KR (2021)
7. Dhami, D.S., Kunapuli, G., Das, M., Page, D., Natarajan, S.: Drug-drug interaction discovery: kernel learning from heterogeneous similarities. Smart Health (2018)
8. Dhami, D.S., Soni, A., Page, D., Natarajan, S.: Identifying parkinson’s patients: A functional gradient boosting approach. In: AIME (2017)
9. Dwork, C., Roth, A., et al.: The algorithmic foundations of differential privacy. Found. Trends Theor. Comput. Sci. **9**(3-4), 211–407 (2014)
10. Fan, L.: A survey of differentially private generative adversarial networks. In: The AAAI Workshop on Privacy-Preserving Artificial Intelligence (2020)

11. Frid-Adar, M., Klang, E., Amitai, M., Goldberger, J., Greenspan, H.: Synthetic data augmentation using gan for improved liver lesion classification. In: ISBI (2018)
12. Geyer, R.C., Klein, T., Nabi, M.: Differentially private federated learning: A client level perspective. arXiv preprint arXiv:1712.07557 (2017)
13. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. NeurIPS (2014)
14. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.: Improved training of wasserstein gans. NeurIPS (2017)
15. Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.M., Larochelle, H.: Brain tumor segmentation with deep neural networks. Medical image analysis (2017)
16. Jordon, J., Yoon, J., Van Der Schaar, M.: Pate-gan: Generating synthetic data with differential privacy guarantees. In: ICLR (2018)
17. Konečnỳ, J., McMahan, H.B., Yu, F.X., Richtárik, P., Suresh, A.T., Bacon, D.: Federated learning: Strategies for improving communication efficiency. arXiv preprint arXiv:1610.05492 (2016)
18. Lin, Z., Khetan, A., Fanti, G., Oh, S.: Pacgan: The power of two samples in generative adversarial networks. NeurIPS (2018)
19. Mahmood, F., Chen, R., Durr, N.J.: Unsupervised reverse domain adaptation for synthetic medical images via adversarial training. IEEE T-MI (2018)
20. Papernot, N., Abadi, M., Erlingsson, U., Goodfellow, I., Talwar, K.: Semi-supervised knowledge transfer for deep learning from private training data. ICLR (2017)
21. Park, N., Mohammadi, M., Gorde, K., Jajodia, S., Park, H., Kim, Y.: Data synthesis based on generative adversarial networks. VLDB Endowment (2018)
22. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. ICLR (2016)
23. Salihovic, I., Serdarevic, H., Kevric, J.: The role of feature selection in machine learning for detection of spam and phishing attacks. In: IAT (2018)
24. Shamsuddin, R., Maweu, B.M., Li, M., Prabhakaran, B.: Virtual patient model: an approach for generating synthetic healthcare time series data. In: ICHI (2018)
25. Tango, F., Botta, M.: Real-time detection system of driver distraction using machine learning. IEEE Transactions on Intelligent Transportation Systems (2013)
26. Torfi, A., Fox, E.A.: Corgan: Correlation-capturing convolutional generative adversarial networks for generating synthetic healthcare records. In: FLAIRS (2020)
27. Tucker, A., Wang, Z., Rotalinti, Y., Myles, P.: Generating high-fidelity synthetic patient data for assessing machine learning healthcare software. NPJ digital medicine (2020)
28. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A., Bottou, L.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. JMLR (2010)
29. Walonoski, J., Kramer, M., Nichols, J., Quina, A., Moesel, C., Hall, D., Duffett, C., Dube, K., Gallagher, T., McLachlan, S.: Synthea: An approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. JAMIA (2018)
30. Xie, L., Lin, K., Wang, S., Wang, F., Zhou, J.: Differentially private generative adversarial network. arXiv preprint arXiv:1802.06739 (2018)
31. Xu, L., Skoularidou, M., Cuesta-Infante, A., Veeramachaneni, K.: Modeling tabular data using conditional gan. NeurIPS (2019)
32. Zhang, X., Ji, S., Wang, T.: Differentially private releasing via deep generative model (technical report). arXiv preprint arXiv:1801.01594 (2018)