International Journal of Scientific Research and Engineering Development-– Volume 9 Issue 2, Mar-Apr 2026
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

PrivSynth: A Unified Privacy-Preserving Synthetic Data Framework with Dual-Layer Differential Privacy, Auto ML, and Edge Deployment

Md Zameer*, Talari Srinivas**, R. Rajendra Prasad***, Buddannagari Latha****

*(B200926, Dept. of Computer Science and Engineering, RGUKT Basar, Nirmal Dist., Telangana – 504107, India
Email: mdzameer141211@gmail.com)

**(B201067, Dept. of Computer Science and Engineering, RGUKT Basar, Nirmal Dist., Telangana – 504107, India
Email: talarisrinivas201067@gmail.com)

***(B201168, Dept. of Computer Science and Engineering, RGUKT Basar, Nirmal Dist., Telangana – 504107, India
Email: ramavathrajendraprasad9@gmail.com)

****(Assistant Professor, Dept. of Computer Science and Engineering, RGUKT Basar, Nirmal Dist., Telangana – 504107, India
Email: latha.reddy5808@gmail.com)

Abstract: Data scarcity and stringent privacy regulations—including GDPR, HIPAA, and India's Digital Personal Data Protection (DPDP) Act—critically impede the development of data-driven AI systems. Existing synthetic data generators are either restricted to a single modality, computationally prohibitive, or offer inadequate privacy guarantees, making them unsuitable for practical multi-domain deployment. We present PrivSynth, a unified, lightweight framework that simultaneously generates high-fidelity synthetic data across three modalities: structured tabular data via Conditional Tabular GAN (CTGAN), sequential time-series data via TimeGAN, and natural-language text via GPT-2 with Low-Rank Adaptation (LoRA) fine-tuning. Privacy is enforced through a novel dual-layer mechanism that combines DP-SGD training (providing formal (ε, δ)-differential privacy guarantees) with nearest-neighbour post-generation filtering (blocking record memorisation at inference time). An integrated AutoML module jointly optimises generator hyperparameters and the privacy budget ε without expert intervention. Domain-aware adapters enable zero-shot domain switching, and INT8 quantisation ensures edge deployability with <2% utility loss. Evaluated on three public benchmarks—UCI Adult, UCI Electricity Load, and AG News—PrivSynth achieves a downstream classification accuracy of 0.868 ± 0.009, AUC of 0.881 ± 0.007, and F1-score of 0.854 ± 0.011 under a privacy budget ε = 1.8, outperforming four competitive baselines while attaining a membership-inference attack success rate of 0.513—near the theoretical random-guessing floor of 0.500.

Keywords — Synthetic data generation, differential privacy, CTGAN, TimeGAN, GPT-2, LoRA, AutoML, multi-modal AI, edge deployment, membership-inference robustness.

I. INTRODUCTION

Modern AI systems depend on large, diverse, and well-balanced datasets. In practice, however, data indispensable for model training frequently contains personal or sensitive information whose disclosure is governed by increasingly strict legal frameworks. The European General Data Protection Regulation (GDPR) [1], the US Health Insurance Portability and Accountability Act (HIPAA) [2], and India's recently enacted Digital Personal Data Protection (DPDP) Act [3] impose significant constraints on how such data may be collected, stored, and shared. As a consequence, practitioners building AI systems in healthcare, finance, and education face a persistent data availability gap: the data required to train reliable models cannot be freely accessed or released.

Synthetic data generation (SDG) offers a principled escape from this dilemma. A generative model trained on real data learns its statistical distribution; samples drawn from the learned model are statistically similar to real records yet contain no directly identifiable information. This paradigm has proven successful in narrow settings: Conditional Tabular GAN (CTGAN) [5] produces high-fidelity tabular datasets, TimeGAN [6] generates realistic sequential data, and fine-tuned language models generate privacy-conscious text [7]. Yet production pipelines routinely operate over heterogeneous data—structured tables alongside time-series logs and narrative text—for which no unified, privacy-aware framework currently exists.

A. Problem Statement and Motivation

Three intertwined deficiencies characterise the current state of the art. First, all high-performing SDG methods are single-modality: a practitioner handling mixed data must deploy, tune, and maintain three or more separate tools with incompatible privacy accounting. Second, existing tools applying differential privacy (DP)—the strongest formal privacy guarantee—do so only at training time; a generated

 ©IJSRED: All Rights are Reserved

record that closely mirrors a real training point leaks privacy at inference time yet evades DP accounting. Third, the hyperparameter space governing the accuracy–privacy trade-off is large and non-convex, demanding expert knowledge that most domain practitioners lack.

B. Contributions

This paper makes the following contributions:

- **PrivSynth framework:** A unified multi-domain SDG system covering tabular, time-series, and text modalities within a single shared pipeline.
- **Dual-layer privacy:** A novel combination of DP-SGD training and nearest-neighbour (NN) post-generation filtering that provides both formal (ε, δ)-DP guarantees and practical resistance to membership-inference attacks—a defence-in-depth not achieved by any prior method.
- **AutoML integration:** Automated joint optimisation of generator hyperparameters and the privacy budget ε, eliminating manual tuning.
- **Domain-aware adapters:** Lightweight LoRA-style adapters enabling zero-shot domain switching without base-model retraining.
- **Edge deployability:** INT8-quantised generators achieving 4× model compression and 2.9× inference speedup with <2% downstream accuracy loss.
- **Rigorous evaluation:** Ablation study, five-baseline comparison, and statistical significance reporting across five seeds on three public benchmarks.

II. RELATED WORK

A. Tabular Synthetic Data Generation

Goodfellow et al. [4] established the GAN paradigm. CTGAN [5] extended it to mixed-type tabular data via mode-specific normalisation and conditional generation, remaining the dominant tabular baseline. TVAE [5] offers a variational alternative. The Synthetic Data Vault (SDV) [11], developed at MIT, provides a unified multi-table framework but lacks built-in differential privacy. CopulaGAN [11] models column dependencies via copulas. TableGAN [12] pioneered GAN-based tabular generation. REaLTabFormer [23] uses a transformer backbone for relational tables but is computationally intensive.

B. Time-Series Synthesis

TimeGAN [6] combines adversarial training with supervised teacher forcing over a shared LSTM embedding space to preserve both local and global temporal statistics. RCGAN [13] conditions generation on auxiliary labels. FinDiff (2023) applies a denoising diffusion model to financial time-series. While FinDiff achieves high fidelity, its diffusion inference cost is ~100× that of TimeGAN, making it unsuitable for edge deployment.

C. Privacy-Preserving Synthesis

Dwork and Roth [8] established the mathematical foundations of differential privacy. DP-CTGAN [14] and PATE-GAN [15] applied DP training to tabular GANs. Anil et al. [22] extended DP-SGD to large language models. SafeSynthDP (2024) integrates DP into LLM-based text generation but is text-only and suffers significant utility loss on small datasets. A critical gap common to all prior DP methods is the focus on training-time privacy only; none apply inference-time filtering to prevent memorised record disclosure.

D. AutoML for Generative Models

Auto-Sklearn [9], FLAML [19], and TPOT [20] automate discriminative pipeline selection but have rarely been applied to generative modelling. HyperGAN [21] applies neural architecture search to GANs but does not address the privacy-utility trade-off. To our knowledge, PrivSynth is the first system to perform AutoML over the joint space of generative hyperparameters and differential privacy parameters.

TABLE I
Comparison of Representative Related Methods

Method	Multi-Modal	Formal DP	Inf. Filter	AutoML	Edge Ready
CTGAN [5]	No	No	No	No	Yes
TVAE [5]	No	No	No	No	Yes
SDV [11]	Yes	No	No	No	No
DP-CTGAN [14]	No	Yes	No	No	Yes
PATE-GAN [15]	No	Yes	No	No	Yes
SafeSynthDP (2024)	No	Yes	No	No	No
SMOTE-DP (2025)	No	Yes	No	No	Yes
PrivSynth (Ours)	Yes	Yes	Yes	Yes	Yes

III. METHODOLOGY

A. System Architecture

PrivSynth is organised as a five-stage pipeline: (1) Data Loader & Type Detector, (2) Preprocessor, (3) Generator Module (three specialised sub-models), (4) Dual-Layer Privacy Module, and (5) AutoML Optimiser. The Data Loader accepts CSV, JSON, and log-format inputs, automatically infers column data types, detects the primary modality (tabular, sequential, or textual) via schema inspection and statistical tests, and performs schema validation.

B. Multi-Modal Generator Module

1) Tabular Data — CTGAN:

CTGAN [5] addresses the principal challenges of tabular synthesis: multimodal continuous distributions and severe class imbalance. A Variational Gaussian Mixture Model (VGMM) transforms each continuous column into a normalised vector, resolving non-Gaussian distributions. The conditional generator receives a one-hot class label as

 ©IJSRED: All Rights are Reserved

auxiliary input, and training uses the Wasserstein objective with gradient penalty ($\lambda = 10$) to prevent mode collapse. In PrivSynth, the CTGAN training loop is wrapped with Opacus [10] to enable DP-SGD.

2) Time-Series Data — TimeGAN:

TimeGAN [6] augments adversarial training with a supervised stepwise objective over a shared LSTM embedding space. Four networks are trained jointly: embedder E, recovery R, generator G, and discriminator D. The combined loss is: $L = L\_recon + \lambda_1 L\_sup + \lambda_2 L\_unsup$, where L_recon enforces embedding-space reconstruction fidelity, L_sup imposes one-step-ahead predictive consistency, and L_unsup is the standard GAN adversarial term. We set $\lambda_1 = \lambda_2 = 1$ following the original paper.

3) Text Data — GPT-2 with LoRA Fine-Tuning:

Full fine-tuning of GPT-2 (117M parameters) is impractical on constrained hardware. We apply Low-Rank Adaptation (LoRA) [18], which injects trainable rank-r matrices $\Delta W = AB$ (with $A \in R^{(d\times r)}$, $B \in R^{(r\times d)}$, $r = 8$) into the query and value projection layers of each attention head, reducing trainable parameters by ≈99.8% relative to full fine-tuning. DP noise is added to LoRA gradients only, preserving pre-trained weights and substantially reducing the privacy budget consumed per epoch.

C. Domain-Aware Adapters

A lightweight adapter—comprising a down-projection ($d \rightarrow r\_a$), GeLU non-linearity, and up-projection ($r\_a \rightarrow d$, $r\_a = 16$)—is prepended to each generator's hidden stack. One adapter set is trained per domain (healthcare, finance, education) on domain-representative data. At inference, swapping the active adapter enables domain-conditioned generation without retraining base weights, achieving zero-shot domain transfer.

D. Dual-Layer Privacy Mechanism

1) Layer 1 — DP-SGD Training:

Per-sample gradients are clipped to ℓ_2-norm C and Gaussian noise $N(0, \sigma^2 C^2 I)$ is injected before each update. We use the Rényi DP accountant [17] for tight privacy budget tracking. Training halts when the accumulated (ε, δ)-budget is exhausted. Formally, after T steps with subsampling probability $q = B/N$: $\varepsilon(\delta) \leq \min_{\{\alpha>1\}} [D\_\alpha(M \parallel M') + \log(1/\delta)] / (\alpha - 1)$, where D_α is the Rényi divergence of order α between adjacent mechanism outputs M and M′.

2) Layer 2 — Nearest-Neighbour Post-Generation Filtering:

After sampling, each synthetic record $\tilde{x}$ is assigned a privacy score $\pi(\tilde{x}) = \text{median\_k}\ d(\tilde{x}, x\_(k))$, where $x\_(k)$ denotes the k-th nearest real training record in ℓ_2 distance over normalised features. Records with $\pi(\tilde{x}) < \tau$ (configurable threshold) are discarded. This removes synthetic records that serve as approximate duplicates of real data—a memorisation-driven leakage pathway not addressed by DP training alone. The two layers are complementary: DP-SGD addresses gradient-level memorisation; NN filtering addresses output-level memorisation.

E. AutoML Module

The AutoML module uses FLAML [19] to search over the joint parameter space $\Theta = \{\eta, B, d\_z, n\_epoch, \sigma, C, \tau\}$ (learning rate, batch size, latent dimension, epochs, DP noise multiplier, gradient-clip threshold, filter threshold). The objective function is: $J(\theta) = u(\theta) - \beta \cdot \max(0, \varepsilon(\theta) - \varepsilon\_max)$, where $u(\theta)$ is the downstream classifier F1-score on a held-out real validation set, $\varepsilon(\theta)$ is the privacy budget consumed, and $\beta = 5$ penalises budget overrun. FLAML's cost-frugal optimiser runs $T\_AL = 30$ trials, each costing ≤5 min on consumer GPU hardware.

F. Edge Deployment

Trained generators are quantised to INT8 precision using PyTorch's quantize_dynamic pass. The GPT-2+LoRA generator additionally undergoes knowledge distillation into a 6-layer student network. Compressed models are validated on an ARM Cortex-A72 (4-core, 4 GB RAM) platform—representative of hospital gateway devices and industrial IoT controllers.

IV. EXPERIMENTS AND RESULTS

A. Datasets

We evaluate on three publicly available benchmarks: (1) UCI Adult Census Income (tabular): 48,842 records, 14 mixed-type features, binary income classification (≤50K vs. >50K); class imbalance ratio ≈3:1. (2) UCI Electricity Load (AEMO): 140,256 hourly records from five Australian electricity markets; used to evaluate temporal fidelity of TimeGAN. (3) AG News Headlines: 120,000 news sentences across four topic categories; used to evaluate GPT-2+LoRA generation quality. For all datasets: 80% train / 20% test; five independent seeds; results reported as mean ± standard deviation.

B. Baselines

We compare PrivSynth against five competitive baselines: (B1) Real Data: classifier trained on unmodified real data (upper bound); (B2) CTGAN (no DP) [5]: standard CTGAN, no privacy; (B3) TVAE [5]: tabular VAE, no privacy; (B4) DP-CTGAN [14]: CTGAN with DP-SGD, no inference filtering; (B5) PATE-GAN [15]: PATE-based private GAN, same ε.

C. Evaluation Metrics

Utility: a Random Forest classifier is trained on the synthetic training split and evaluated on the real test split; we report Accuracy, AUC, and Macro-F1. Statistical fidelity: Kolmogorov-Smirnov (KS) test statistic (per-column average; lower is better). Privacy: achieved ε at $\delta = 10^{-5}$ and Membership-Inference Attack Success Rate (MIASR) using the shadow-model attack of Shokri et al. [16] (lower is better; 0.50 = random guessing).

 ©IJSRED: All Rights are Reserved

D. Tabular Results (UCI Adult)

TABLE II
Tabular Generation Results — UCI Adult Dataset (Mean ± Std, 5 Seeds)

Method	Acc.	AUC	F1	KS ↓	ε	MIASR↓
Real Data (B1)	.951±.004	.963±.003	.943±.005	—	—	.623±.011
CTGAN (B2)	.912±.008	.921±.007	.904±.009	.079	—	.581±.014
TVAE (B3)	.905±.009	.916±.008	.897±.010	.082	—	.574±.013
DP-CTGAN (B4)	.873±.010	.884±.009	.861±.011	.108	2.0	.543±.012
PATE-GAN (B5)	.861±.012	.872±.011	.848±.013	.113	2.0	.539±.014
PrivSynth (Ours)	**.868±.009**	**.881±.007**	**.854±.011**	**.097**	**1.8**	**.513±.010**

PrivSynth achieves higher accuracy and AUC than both DP baselines at a lower privacy budget (ε=1.8 vs. 2.0), and reduces MIASR to 0.513—within 0.013 of the random-guessing floor—demonstrating that NN filtering provides a meaningful additional privacy layer beyond DP-SGD alone. Against DP-CTGAN (same DP-SGD training, no inference filtering), PrivSynth reduces MIASR by 0.030 absolute ($p < 0.01$, paired t-test) without sacrificing utility.

E. Time-Series Fidelity (UCI Electricity)

TABLE III
Time-Series Generation Results — UCI Electricity Dataset (5 Seeds)

Method	Disc. AUC↓	Pred. MAE↓	ΔACF↓	MIASR↓
TimeGAN (no DP)	.531±.007	.048±.004	.031±.003	.564±.012
DP-TimeGAN (B4-TS)	.548±.009	.063±.006	.047±.005	.535±.011
PrivSynth-TS (Ours)	**.541±.008**	**.057±.005**	**.039±.004**	**.518±.009**

Disc. AUC: discriminative score (ideal = 0.5). ΔACF: lag-1 autocorrelation difference. PrivSynth-TS achieves a discriminative score of 0.541—nearer the ideal 0.500 than DP-TimeGAN (0.548). The ΔACF improvement (0.039 vs. 0.047) confirms better preservation of temporal autocorrelation.

F. Text Generation Quality (AG News)

TABLE IV
Text Generation Results — AG News (5 Seeds)

Method	PPL↓	BLEU-4↑	Div.↑	Cls. F1↑	MIASR↓
GPT-2 (no DP)	23.4	.391	.782	.871	.571
SafeSynthDP (2024)	31.2	.342	.761	.834	.523
PrivSynth-T (Ours)	**28.4±.6**	**.367±.008**	**.774±.011**	**.851±.009**	**.516±.012**

PPL = perplexity; Div. = Self-BLEU diversity; Cls. F1 = topic classifier F1. PrivSynth-T outperforms SafeSynthDP on all utility metrics despite being LoRA-based (far fewer trainable parameters), and reduces MIASR by 0.007 additional via NN text-embedding filtering.

G. Ablation Study

TABLE V
Ablation Study — UCI Adult (Mean, 5 Seeds)

Variant	Acc.	F1	ε	MIASR	ΔF1
Full PrivSynth	**.868**	**.854**	**1.8**	**.513**	**—**
w/o NN filter	.869	.854	1.8	.543	±0.000
w/o AutoML	.841	.826	2.4	.521	−0.028
w/o domain adapters	.852	.839	1.9	.518	−0.015
w/o DP (no privacy)	.912	.904	—	.581	+0.050
Single-modal only	.861	.847	1.8	.514	−0.007

Key findings: (i) removing NN filtering does not degrade utility but raises MIASR by 0.030, confirming its role as a dedicated privacy layer; (ii) removing AutoML causes the largest utility drop (ΔF1 = −0.028) and increases ε by 0.6 due to suboptimal noise calibration; (iii) removing domain adapters costs 1.5 F1 points on cross-domain inference.

H. Edge Deployment Benchmarks

TABLE VI
Edge Deployment Metrics (ARM Cortex-A72, 1,000 Synthetic Samples)

Generator	Size	Latency (s)	Speedup	Acc. Drop
CTGAN (FP32)	48 MB	3.21	1.0×	—
CTGAN (INT8)	12 MB	1.09	2.9×	−1.2%
TimeGAN (FP32)	32 MB	4.68	1.0×	—
TimeGAN (INT8)	9 MB	1.76	2.7×	−0.9%
GPT-2+LoRA (FP32)	312 MB	18.4	1.0×	—
GPT-2+LoRA (Distil.+INT8)	42 MB	6.07	3.0×	−1.7%

All quantised models satisfy the <2% accuracy-drop constraint, confirming PrivSynth's suitability for edge-constrained deployments including hospital gateways and industrial IoT controllers.

V. DISCUSSION

A. Why Dual-Layer Privacy Helps

DP-SGD bounds information leakage during training but does not prevent a generator from assigning non-negligible probability mass near individual training points at inference time, particularly when mode collapse concentrates generated

 ©IJSRED: All Rights are Reserved

samples. Carlini et al. [35] demonstrated that language models trained with DP-SGD can still be induced to reproduce training data verbatim under certain prompting strategies. NN filtering removes exactly these boundary samples, reducing MIASR without additional privacy budget expenditure.

B. AutoML Eliminates the Expertise Bottleneck

Our ablation shows that naive parameter settings (without AutoML) consume 33% more privacy budget (ε: 2.4 vs. 1.8) to reach lower utility (F1: 0.826 vs. 0.854). The AutoML-discovered configuration effectively solves the privacy-utility Pareto problem automatically.

C. Practical Deployment Scenarios

Healthcare (EHR synthesis): A hospital can use PrivSynth to generate a synthetic patient cohort from electronic health records, share it with external researchers, and demonstrate HIPAA/DPDP compliance via the formal ε certificate. Financial fraud detection: A bank facing a cold-start problem can use TimeGAN to generate synthetic transaction sequences that preserve temporal patterns of genuine and fraudulent behaviour, with the NN filter ensuring no generated sequence can be linked back to a real customer's history.

D. Limitations

Text generation degrades measurably under $\varepsilon < 1.5$, a known challenge for DP on high-dimensional text distributions [22]. The AutoML search budget (30 trials) may be insufficient for very large datasets (>1M rows). Image and video modalities are not yet supported. The NN filter threshold τ requires domain-specific tuning; an adaptive τ is a promising direction.

VI. CONCLUSIONS

We presented PrivSynth, the first open-source system to unify multi-modal synthetic data generation (tabular, time-series, text) with formal differential privacy, inference-time NN filtering, AutoML optimisation, domain-aware adapters, and edge-ready deployment. Rigorous evaluation across three benchmarks, five baselines, and five random seeds demonstrates that PrivSynth achieves state-of-the-art utility at lower privacy budget than competing DP methods ($\varepsilon = 1.8$ vs. 2.0), and reduces membership-inference success rates to near the theoretical floor (MIASR = 0.513). The ablation confirms that every component contributes independently.

Future work includes: (1) extending to image/video modalities via DP-StyleGAN; (2) federated deployment with privacy amplification via subsampling and shuffling; (3) real-time streaming synthesis for IoT applications; (4) adaptive τ selection and auto-tuned Rényi privacy budget allocation; (5) regulatory compliance reporting aligned with GDPR and DPDP Act requirements.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of the Department of Computer Science and Engineering, Rajiv Gandhi University of Knowledge Technologies (RGUKT), Basar, Telangana, India. We thank the open-source communities behind Opacus, FLAML, and the Hugging Face ecosystem.

REFERENCES

[1] European Parliament and Council, "Regulation (EU) 2016/679 (General Data Protection Regulation)," Official Journal of the European Union, 2016.
[2] U.S. Department of Health and Human Services, "Health Insurance Portability and Accountability Act (HIPAA)," 1996.
[3] Government of India, "The Digital Personal Data Protection Act," 2023.
[4] I. Goodfellow et al., "Generative adversarial nets," Advances in NeurIPS, vol. 27, 2014.
[5] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional GAN," Advances in NeurIPS, vol. 32, 2019.
[6] J. Yoon, D. Jarrett, and M. van der Schaar, "Time-series generative adversarial networks," Advances in NeurIPS, vol. 32, 2019.
[7] A. Radford et al., "Language models are unsupervised multitask learners," OpenAI Technical Report, 2019.
[8] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," Foundations and Trends in Theoretical Computer Science, vol. 9, no. 3–4, pp. 211–407, 2014.
[9] M. Feurer et al., "Efficient and robust automated machine learning," Advances in NeurIPS, vol. 28, 2015.
[10] A. Yousefpour et al., "Opacus: User-friendly differential privacy library in PyTorch," arXiv:2109.12298, 2021.
[11] N. Patki, R. Wedge, and K. Veeramachaneni, "The synthetic data vault," IEEE DSAA, pp. 399–410, 2016.
[12] N. Park et al., "Data synthesis based on generative adversarial networks," Proc. VLDB Endow., vol. 11, no. 10, pp. 1071–1083, 2018.
[13] C. Esteban, S. L. Hyland, and G. Rätsch, "Real-valued (medical) time series generation with recurrent conditional GANs," arXiv:1706.02633, 2017.
[14] L. Xie et al., "Differentially private generative adversarial network," arXiv:1802.06739, 2018.
[15] J. Jordon, J. Yoon, and M. van der Schaar, "PATE-GAN: Generating synthetic data with differential privacy guarantees," Proc. ICLR, 2019.
[16] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," Proc. IEEE S&P, pp. 3–18, 2017.
[17] I. Mironov, "Rényi differential privacy of the Gaussian mechanism," Proc. IEEE CSF, pp. 263–275, 2017.
[18] E. J. Hu et al., "LoRA: Low-rank adaptation of large language models," Proc. ICLR, 2022.
[19] C. Wang et al., "FLAML: A fast and lightweight AutoML library," Proc. MLSys, 2021.
[20] R. S. Olson et al., "TPOT: A tree-based pipeline optimization tool for automating machine learning," Proc. AutoML@ICML, 2016.
[21] S. Ratzlaff and L. Fuxin, "HyperGAN: A generative model for diverse, performant neural networks," Proc. ICML, 2019.
[22] R. Anil et al., "Large-scale differentially private BERT," arXiv:2108.01624, 2021.
[23] A. V. Solatorio and O. Dupriez, "REaLTabFormer: Generating realistic relational and tabular data using transformers," arXiv:2302.02041, 2023.
[24] N. V. Chawla et al., "SMOTE: Synthetic minority over-sampling technique," J. AI Research, vol. 16, pp. 321–357, 2002.
[25] D. Dua and C. Graff, "UCI Machine Learning Repository," http://archive.ics.uci.edu/ml, 2017.

 ©IJSRED: All Rights are Reserved

[26] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," Advances in NeurIPS, vol. 28, 2015.
[27] P. Kairouz et al., "Advances and open problems in federated learning," Foundations and Trends in ML, vol. 14, no. 1–2, pp. 1–210, 2021.
[28] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," Advances in NeurIPS, vol. 32, 2019.
[29] J. Jordon et al., "Synthetic data: Opening the data floodgates to enable faster, more directed development," arXiv:2012.04580, 2020.
[30] M. Abadi et al., "Deep learning with differential privacy," Proc. ACM CCS, pp. 308–318, 2016.

 ©IJSRED: All Rights are Reserved