ST-DPGAN: A Privacy-preserving Framework for Spatiotemporal Data Generation

Wei Shao, Rongyi Zhu, Cai Yang, Chandra Thapa, Muhammad Ejaz Ahmed, Seyit Camtepe, Rui Zhang, Du Yong Kim, Hamid Menouar, and Flora D. Salim

***Abstract*—Recent advancements have sparked a growing interest in integrating spatiotemporal analysis with large-scale language models. However, spatiotemporal data often contains sensitive information, making it unsuitable for open third-party access. To address this challenge, we propose a Graph-GAN-based model for generating privacy-protected spatiotemporal data. Our approach incorporates spatial and temporal attention blocks in the discriminator and a spatiotemporal deconvolution structure in the generator. These enhancements enable efficient training under Gaussian noise to achieve differential privacy. Extensive experiments conducted on three real-world spatiotemporal datasets validate the efficacy of our model. Our method provides a privacy guarantee while maintaining the data utility. The prediction model trained on our generated data maintains a competitive performance compared to the model trained on the original data.**

***Index Terms*—differential privacy, spatiotemporal data, generative adversarial network**

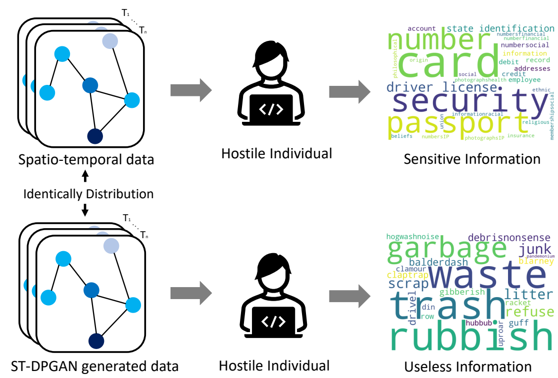

Fig. 1. A hostile individual can extract sensitive information from original spatiotemporal data. Our proposed method enables the generation of privacy-protected data while maintaining good data quality. Our method serves as a fundamental study for applying spatiotemporal analysis techniques on a large scale.

I. Introduction

Spatiotemporal data permeates various aspects of Internet of Things (IoT), ranging from everyday experiences like transportation [10] to larger-scale domains such as energy transmission [39], and currency flow [45]. Consequently, spatiotemporal data inherently include sensitive information linked to both time and space. For example, parking data, from a numerate remote sensor, in the central business districts of Melbourne has its geographic insights [29], which consists of the human mobility pattern, the traffic flow pattern, and the government traffic policies [37]. Similarly, wearable devices, including fitness trackers and smartwatches, utilize embedded sensors to collect continuous data on heart rate, physical activity, and environmental factors. These data points are timestamped and can be transmitted to central health management systems or cloud-based platforms for analysis.

As mentioned above, semantic information in spatiotemporal data poses a new challenge for fine-grained exploitation. Many regions have proposed data protection regulations due to privacy concerns. The European Union has prompted the General Data Protection Regulation (GDPR), which requires appropriate measures, such as pseudonymisation, to protect data privacy [1]. This regulation makes it important to find a de-identification method when red using more fine-grained spatiotemporal datasets. These de-identification methods are meaningful to other parts of the world.

In 2021, the United States set up an independent data protection agency to regulate specified high-risk data practices, such as using automated decision systems [2]. Thus, an answer to the question of how to protect data privacy is important to the further study of spatiotemporal analysis.

To enable large-scale spatiotemporal data analysis, we should provide a privacy guarantee. There has been a wide study on how to bring differential private mechanisms into various data generation schemes, such as GAN [18], [34], VAE [6], [33] and U-Net [8]. In this work, we focus on incorporating GAN into differentially private spatiotemporal data generation. GAN has more potential in spatiotemporal data generation than other methods, as shown in the recent study [14].

There are challenges in generating spatiotemporal data using GANs with differential privacy. **(1)** Spatiotemporal data are heterogeneous. The generated spatial information and temporal information are hard to align into spatiotemporal data, as it is demonstrated that spatial and temporal features belong to different Euclidean spaces [28]. **(2)** Due to the space-time duality property of spatiotemporal data, generating spatiotemporal data samples with a single variable drawn from a certain distribution is difficult, which is different from homogeneous

Wei Shao, Chandra Thapa, Muhammad Ejaz Ahmed and Seyit Camtepe are with the Data61, CSIRO, Australia. email: wei.shao@data61.csiro.au.

Rongyi Zhu is with the University of Rochester. email: rongyi.zhu@rochester.edu

Cai Yang is with Australian National University, Australia.

Rui Zhang is with School of Computer Science and Techology, Huazhong University of Science and Technology, China (www.ruizhang.info). email: rayteam@yeah.net

Du Yong Kim is with RMIT University, Australia.

Hamid Menouar is working at Qatar Mobility Innovations Center, Qatar

Flora D. Salim is with University of New South Wales, Australia.

Manuscript received April 19, 2005; revised August 26, 2015.

data such as pixel-based images [26]. **(3)** Simply adding noises to each spatial location or temporal point cannot guarantee a privacy-preserving dataset for training usage because it does not consider spatial correlation and temporal correlation, and the overall data distribution cannot be maintained.

To overcome the abovementioned challenges, we propose a novel differential private generative adversarial network for generating spatiotemporal data, emphasizing privacy preservation. In our framework, we introduce two key strategies for effectively extracting the distribution of spatiotemporal data and measuring privacy loss during the training process. First, we adopt the Transposed One-Dimensional convolution layer (transConv1d) to generate fake data samples from Gaussian noises, which can transform single-channel Gaussian noises into spatiotemporal data. Through adversarial learning with the discriminator, the generator refines its ability to transform the noise into images that resemble the data distribution. Second, we simplify existing spatiotemporal graph convolutional networks [40] and introduce a temporal attention block and a spatial attention block. These blocks are designed to model spatial and temporal dependencies, enabling the simultaneous extraction of spatiotemporal features from graph-based time series data and facilitating easier convergence during the training process. The contributions of this paper are summarized as follows:

- Introduce an end-to-end framework based on Generative Adversarial Networks to generate graph-based spatiotemporal data, incorporating a differential privacy mechanism.
- Construct a spatiotemporal convolution module capable of converting 1-D Gaussian noise data into 2-D data, enabling the generation of spatiotemporal data from a random variable.
- Propose to utilize two layers of the spatiotemporal attention block in response to the differential private training on the highly heterogeneous spatiotemporal data.
- Evaluate our proposed method on three extensive real-world spatiotemporal datasets and find that our methods maintain comparable data utilities with baseline approaches while providing a privacy guarantee.

II. Related Works

Differential Private SGD Differential privacy has been introduced into machine learning to secure the training data. The privacy of these training data could be compromised due to the machine learning model's strong representation ability. The differential privacy [9] has been proposed to measure the privacy loss of a certain data processing mechanism. This theory can be applied in the deep learning area to measure the maximum differential privacy loss σ of a certain deep learning process. The Differential Preserving SGD [3] is the first training mechanism that uses Differential Preserving in the training process of deep learning by clipping the training weights and adding noise to the clipped training weights. After that, privacy-preserving SGD [3] tried further to improve the computational efficiency of this training mechanism and published the TensorFlow implementation of the improved Differential Preserving SGD mechanism. These works focus on the general training procedure while ignoring the differences in various kinds of data. However, the real-world application of differential privacy mechanisms on different data domains, such as vision [7], language [31] and sounds [15], [38], are hard and need corresponding adjustments. In this work, we focus on adapting the differential privacy mechanism in the spatio-temporal domain.

Private Aggregation of Teacher Ensembles (PATE) Besides the differentially preserving SGD, there is another way to protect the privacy of the deep learning model. The Private Aggregation of Teacher Ensembles (PATE) [23] uses several teacher models, which are trained on different partitions of the privacy data, and a student model, which is trained on unlabelled public data. The label of the public data is obtained by querying the trained teacher models. The privacy-preserving and utility of the whole model can be further enhanced by making the teachers reach a consensus on query voting.

However, in many real applications [14], such as currency flow and social analysis, public data could be difficult to obtain. In our application, our model generates spatiotemporal data where the public spatial-temporal dataset is difficult to obtain. We want to provide a privacy guarantee for the spatio-temporal data in order to enable a wider range of studies. Thus, we adopted the DP-SGD structure and applied it in our discriminator to protect the privacy of the generated spatial-temporal data. Our approach serves as a balanced approach between the differential privacy noise and the generated data quality.

Spatial and temporal data analysis and management is an area that has attracted many studies such as indexing [42]–[44] and mining [35], [36]. Moreover, given the recent popularity of large language models, there have been many studies on various kinds of generation [21], [30], [32], including spatiotemporal data [14], but there has been little work on privacy-preserving spatiotemporal data generation.

III. Methodology

To address the challenges of generating privacy-preserving spatiotemporal data, we propose ST-DPGAN (Spatiotemporal Differential-Privacy Generative Adversarial Network), a novel framework designed to adapt the classical GAN architecture to handle the unique demands of spatiotemporal data while incorporating differential privacy (DP). This section outlines the strategies employed in ST-DPGAN, focusing on the mechanisms developed to overcome the inherent complexities of spatiotemporal data and ensure privacy preservation without compromising data utility.

The model accepts spatiotemporal data as input, represented as $\mathcal{G}(t) = (\mathbf{V}, \mathbf{E}, \mathbf{X}(t))$, where $\mathbf{V}$ denotes the nodes in the graph, $\mathbf{E}$ represents the edges connecting the nodes, and $\mathbf{X}(t)$ signifies the attributes associated with each node, incorporating temporal information. The output of our model is privacy-protected spatiotemporal data that maintains a high data quality.

Our proposed network, ST-DPGAN, comprises two primary components: a generator and a discriminator. In our framework, we have introduced two novel features to enhance the

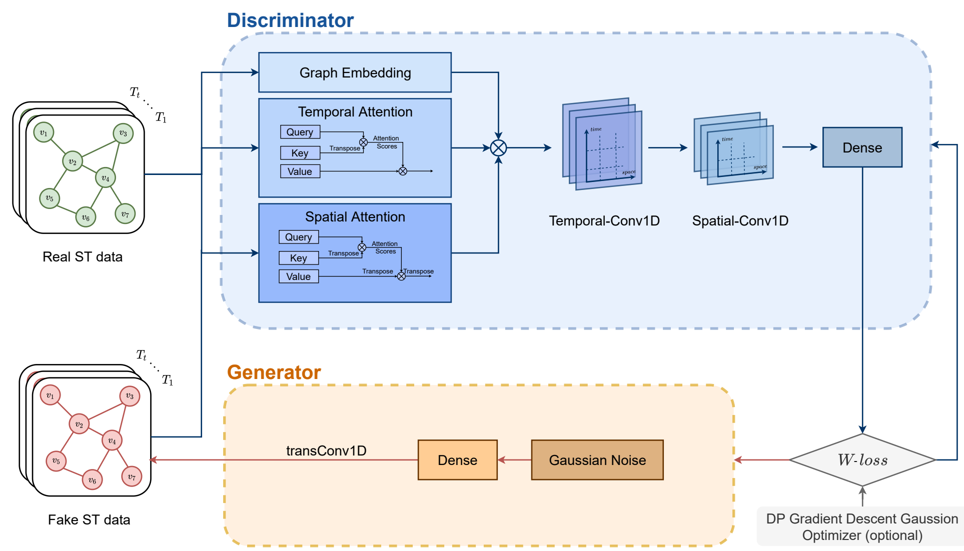

Fig. 2. Architecture of ST-DPGAN.

model performance. We summarized our approach with the pseudo-code in Algorithm 1. The generator is parameterized by weights θ with kernels K_g and biases b_g to transform noise samples into synthetic data, while the discriminator utilizes graph embeddings along with spatial and temporal attention mechanisms to distinguish real data from synthetic data. The transposed convolution has been shown useful in many domains, such as vision [12] and sound [5]. We incorporate a transConv1d module, which transforms 1-D Gaussian noise data into 2-D spatiotemporal data. During training, the discriminator is updated multiple times per iteration, followed by an update of the generator.

A. Generator

To address the challenge of synthesizing spatiotemporal data from 1-D Gaussian noises, we introduce a novel module in our GAN's generator, termed *transConv1d*. In a traditional GAN framework, the generator's role is to learn an approximation of the real data distribution. This learning process is achieved by mapping random noise to data samples, typically sampled from a Gaussian distribution. However, spatiotemporal data presents a unique complexity due to the inherent spatial and temporal correlations. These correlations pose a challenge in reconstructing the data distribution, particularly while maintaining a balance between privacy preservation and data quality.

To overcome this challenge, we designed the transConv1d module to recover spatial and temporal information from a learned distribution. Specifically, at each temporal data location, the data is multiplied by a kernel initialized with a Gaussian distribution to generate new sequential data. All the generated sequential data is then aggregated to form the final output of the transConv1d module. In our model, the filter parameter of the transConv1d module is set to the size of the spatial locations (N), and the kernel size is set to the length of the time series (T). Consequently, the initial random data with dimensions of $1 \times N$ can be processed into dimensions of $T \times N$. The transConv1d enables the model to capture the temporal variations in the generated data effectively. Our proposed transConv1d module can help the model recover the aligned spatiotemporal information from noise, which we will validate in the next section.

1) transConv1d: In this section, we provide our mathematical insight into transConv1d to validate the property of the proposed module.

Theorem 1. *Let the initial random data be X of size $1 \times N$, where each entry is x_j, and let the kernel be K of size $s \times s$, where each entry is k_{ij}. Assume $x_j \overset{i.i.d}{\sim} \mathcal{N}(\mu_1, \sigma_1^2)$, $k_{ij} \overset{i.i.d}{\sim} \mathcal{N}(\mu_2, \sigma_2^2)$, and x_j is mutually independent to k_{ij}. Then the output matrix is Y of size $T \times N$, where each entry of its vectorization form $vec(Y)_i$ satisfies*

$$vec(Y)_i = \frac{\sigma_1^2 + \sigma_2^2}{4}(Q_1 - Q_2) + G + m\mu_1\mu_2,$$

where $Q_1, Q_2 \sim \chi_m^2$, $G \sim \mathcal{N}(0, \frac{m^2(\sigma_1^2+\sigma_2^2)(\mu_1^2+\mu_2^2)}{2})$, and m is the number of nonzero entries in the i^{th} row of the transposed convolution matrix.

Considering the actual training, we specifically let the mean of the kernel be approximately the same as the mean of the training data. Hence, we provide two useful results below about the common setting and the setting we used.

Lemma 2. *Using the same setting as Theorem 1:*

1) *if $x_j \overset{i.i.d}{\sim} \mathcal{N}(0, 1)$ and $k_{ij} \overset{i.i.d}{\sim} \mathcal{N}(0, 1)$, then $vec(Y)_i$ is the difference of two chi-squared random variables with scaling.*

$$vec(Y)_j = \frac{1}{2}(Q_1 - Q_2)$$

where $Q_1, Q_2 \sim \chi_m^2$.

2) *if $x_j \overset{i.i.d}{\sim} \mathcal{N}(0, 1)$ and $K_{ij} \overset{i.i.d}{\sim} \mathcal{N}(\mu, 1)$, where μ is the mean of the training data, then*

$$vec(Y)_i = \frac{1}{2}(Q_1 - Q_2) + G$$

Algorithm 1: Spatiotemporal Differential-Privacy Generative Adversarial Network

Input: Spatiotemporal Dataset D with N samples, Laplacian matrix L, sampling probability q, Differential private optimizer $O_{(\epsilon,\delta)}$, learning rate l, noise level σ, overall iteration T, discriminator iteration k_{critic}
, coefficients α, β Initialize generator parameters θ with weight W_{g^z}, kernel K_g and bias b_g; discriminator parameters w with kernel K_d and bias b_d.

Generator setup:

$$\tilde{z} = W_g z + b_g$$
$$G_\theta(z) = \tilde{z} * K_g$$

Discriminator setup:

$$\tilde{x} = GraphEmb(x, L) + \alpha \cdot SpatialAttn(x) + \beta \cdot TemporalAttn(x)$$
$$\bar{x} = \tilde{x} * K_d$$
$$F_w(x) = W_d \bar{x} + b_d$$

for *i = 1,..., T* **do**
  for *t = 1, ..., k_{critic}* **do**
    Take a random sample D_q from D by probability q
    Sampling m example $\{z^{(i)}\}_{k=1}^m$ from noise prior $p(z)$, where $m = qN$.
    $g_w \leftarrow \nabla_w [\frac{1}{m}\sum_{k=1}^m F_w(x^{(i)}) - \frac{1}{m}\sum_{k=1}^m F_w(G_\theta(z^{(i)}))]$
    $w \leftarrow w + l \cdot O_{(\epsilon,\delta)}(g_w)$
  end
  Sampling m example $\{z^{(i)}\}_{k=1}^m$ from noise prior $p(z)$.
  $g_\theta \leftarrow \nabla_w - \frac{1}{m}\sum_{k=1}^m F_w(G_\theta(z^{(i)}))$
  $\theta \leftarrow \theta - l \cdot O_{(\epsilon,\delta)}(g_\theta)$
end

where $Q_1, Q_2 \sim \chi_m^2$ *and* $G \sim \mathcal{N}(0, m^2\mu^2)$

From the above explanation, we can see that transConv1d has the property to recover the distribution for both spatial and temporal information. We will discuss and validate this claim in our experimental session.

B. Discriminator

The conventional pipeline for discriminating spatiotemporal data, known as the spatiotemporal Graph Convolutional Network (STGCN) [40], typically employs three convolutional blocks to predict the temporal evolution of spatiotemporal data. However, in our research, we observed that the original STGCN structure is not ideally suited as a discriminator in a GAN framework, because of its heavy three-layer spatiotemporal structure, which can lead to computational issues such as gradient explosion and vanishing. These challenges necessitate reconsidering the discriminator design in the context of GANs for spatiotemporal data.

To address this problem, we propose the integration of two self-attention blocks within the discriminator as an enhancement to the ‘basic’ model. Firstly, while convolutional layers are commonly used in modern GAN architectures for feature extraction, they have limitations in capturing global dependencies between nodes, which are crucial in spatiotemporal data. We believe that each node is influenced by other nodes, both spatially and temporally. Therefore, we introduce attention structures to better capture and align the spatiotemporal information. The spatial self-attention block establishes relationships between nodes by attending to every other node simultaneously. Similarly, the temporal self-attention block links each node by attending to itself across all timestamps.

The feature extraction process is formulated as follows: The input data is transformed through the normalized Laplacian matrix, as shown in Equation 1, where $X_l \in \mathbb{R}^{N\times T}$. Equation 2 describes the process of spatial self-attention. The input $X \in \mathbb{R}^{T\times N}$ is first projected using W_s^q and $W_s^k \in \mathbb{R}^{T\times M}$ to obtain query and key matrices, where M is the hidden dimension. The value matrix $V_s \in \mathbb{R}^{N\times T}$ is obtained through 1D convolution with a kernel size 1. The raw attention scores are then passed through a softmax function and multiplied with V_s to produce the new representation $X_s \in \mathbb{R}^{N\times T}$. The process is similar for temporal self-attention, where W_t^q and $W_t^k \in \mathbb{R}^{T\times H}$ are the projection matrices for query and key. The value matrix $V_t \in \mathbb{R}^{T\times N}$ is multiplied with the attention scores to obtain the new representation $X_t \in \mathbb{R}^{T\times N}$.

We consider graph embedding as an initial means of obtaining features. Following a similar approach to Zhang et al. [41], the output from the two self-attention blocks is multiplied by scalars initialized with a value of 0 and added back to the basic features. This ensures the model begins with fundamental features derived from the graph properties and gradually learns other underlying features during training. The final representation is expressed as $Z = X_l + \alpha X_s + \beta X_t^\intercal$. To avoid confusion, we denote models with the additional self-attention blocks as ST-DPGAN-Attn and empirically validate the performance improvements brought by attention in our experiments.

$$X_l = LX^\intercal \tag{1}$$

$$\begin{aligned} &Q_s = X^\intercal W_s^q,\ \ K_s = X^\intercal W_s^k,\ \ V_s = (F_s * G_s)(X^\intercal) \\ &X_s = \sigma(Q_s K_s^\intercal)V_s \end{aligned} \tag{2}$$

$$\begin{aligned} &Q_t = X W_t^q,\ \ K_t = X W_t^k,\ \ V_t = (F_t * G_t)(X) \\ &X_t = \sigma(Q_t K_t^\intercal)V_t^\intercal \end{aligned} \tag{3}$$

Besides that, for the optimizer in the presence of differential privacy, DPSGD is used in the discriminator to add random noise in the gradients of the trainable parameters through back-propagation [3].

TABLE I
WE SHOW THE MSE AND MAE SCORES OF PREDICTIVE MODELS AND THE BEST VALUE IS MARKED IN BOLD. OUR MODELS ARE MARKED BY *. THE LOWER THE SCORE VALUE, THE BETTER THE DATA QUALITY. WE CHOOSE THE PRIVACY BUDGET ϵ AS 12 FOR THE DP-BASED MODEL TO COMPARE WITH OTHER NON-DP CONTAINED ALGORITHMS. THE AVERAGE ROW STANDS FOR THE AVERAGE LOSS UNDER DIFFERENT PREDICTION ALGORITHMS. THE COUNT ROWS REPRESENT THE TOTAL NUMBER OF BEST RESULTS FOR EACH COLUMN.

	Methods	Real		WGAN		ST-DPGAN*		ST-DPGAN-Attn*	
	Metrics	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Parking	LR	0.0004	0.0158	**0.0068**	**0.0608**	0.0074	0.0678	**0.0068**	0.0643
	Linear SVR	0.0003	0.0141	0.0069	0.0605	0.0066	0.0594	**0.0062**	**0.0576**
	MLP	0.0005	0.0188	0.0091	0.0743	0.0077	0.0677	**0.0068**	**0.0645**
	SGD	0.0004	0.0158	**0.0068**	**0.0608**	0.0076	0.0686	0.0069	0.0646
	Decision Tree	0.0046	0.0479	0.0218	0.1127	0.0150	0.0877	**0.0136**	**0.0828**
	Random Forest	0.0008	0.0218	0.0132	0.0887	0.0077	0.0667	**0.0069**	**0.0613**
	Gradient Boosting	0.0004	0.0166	0.0117	0.0838	0.0074	0.0676	**0.0071**	**0.0630**
	Bagging	0.0012	0.0255	0.0148	0.0927	0.0084	0.0691	**0.0076**	**0.0638**
	Ada Boosting	0.0027	0.0451	0.0134	0.0919	0.0100	0.0804	**0.0096**	**0.0774**
	LGBM	0.0003	0.0157	0.0103	0.0791	0.0074	0.0673	**0.0065**	**0.0603**
	LSTM	0.0118	0.0920	0.0162	0.1035	**0.0087**	0.0714	**0.0087**	**0.0689**
	Average	0.0021	0.0299	0.0119	0.0826	0.0085	0.0703	0.0078	0.0662
METR-LA	LR	91.8794	6.7027	**95.2423**	6.8065	97.1440	6.8816	97.0645	**6.7750**
	Linear SVR	89.4740	6.3760	**96.3206**	6.7831	97.7706	**6.6635**	97.2597	6.6976
	MLP	103.0017	6.9999	99.0773	7.4096	110.6378	7.2572	**97.6307**	**6.8797**
	SGD	91.8495	6.6269	**95.7337**	6.8376	96.8873	6.8355	97.0175	**6.7621**
	Decision Tree	225.9751	10.1180	286.1651	15.0811	191.0109	9.7920	**142.6260**	**8.0813**
	Random Forest	119.4058	7.6720	294.5921	15.4848	111.4184	7.5598	**108.7084**	**6.8320**
	Gradient Boosting	101.2091	7.0300	231.6227	13.8163	105.7373	7.1437	**102.2363**	**6.7753**
	Bagging	129.1552	7.9630	176.2332	11.9194	119.4350	7.8397	**113.0507**	**7.0063**
	Ada Boosting	159.0458	10.7724	244.9048	14.1454	141.7784	10.0757	**107.5073**	**8.0823**
	LGBM	101.7017	7.0309	241.4298	14.0983	102.7747	7.0127	**102.0630**	**6.9834**
	LSTM	123.6026	8.1991	151.5952	10.3412	126.8095	8.2104	**122.6244**	**8.0745**
	Average	121.4818	7.7719	182.9924	11.1567	118.3094	7.7520	107.9808	7.1772
Windmill	LR	0.0496	0.1739	0.0607	**0.1885**	0.0609	0.2130	**0.0581**	0.2022
	Linear SVR	0.0615	0.1620	0.0603	0.1968	**0.0570**	0.1778	0.0771	**0.1751**
	MLP	0.0514	0.1752	0.0920	0.2633	0.0609	0.2132	**0.0578**	**0.2011**
	SGD	0.0498	0.1737	0.0607	0.1884	0.0613	0.2140	**0.0582**	**0.2028**
	Decision Tree	0.1085	0.2356	0.0828	0.2493	0.1393	0.2817	**0.1353**	**0.2702**
	Random Forest	0.0530	0.1805	0.0814	0.2514	**0.0629**	0.2178	0.0637	**0.2086**
	Gradient Boosting	0.0509	0.1746	0.0734	0.2404	0.0623	0.2173	**0.0590**	**0.2063**
	Bagging	0.0580	0.1863	0.0835	0.2552	0.0792	0.2339	**0.0699**	**0.2137**
	Ada Boosting	0.0576	0.2085	0.0668	0.2235	0.0743	0.2455	**0.0604**	**0.2075**
	LGBM	0.0511	0.1750	0.0722	0.2395	0.0621	0.2168	**0.0592**	**0.2069**
	LSTM	0.0568	0.1864	**0.0603**	**0.1834**	0.0897	0.2560	0.0991	0.2703
	Average	0.0589	0.1847	0.0722	0.2254	0.0736	0.2261	0.0725	0.2150
	Count	-	-	6	4	3	2	26	28

C. Model Summary

We now summarise the main characteristics of our model as follows:

- ST-DPGAN is an end-to-end GAN-based framework to generate spatiotemporal data with quantitative privacy-preserving guaranteed by a differential privacy mechanism. The generated data could be used for regression tasks and other machine learning or deep learning applications.
- The TransConv1D focuses on transposing the spatiotemporal relation from one-dimensional Gaussian noise, which controls the initial distribution of the simulation dataset.
- The spatial and temporal blocks help extract underlying spatial and temporal relations among nodes, which further boosts the performance of the model in downstream tasks.

IV. EXPERIMENTS

In this section, we present the experiments conducted on our ST-DPGAN model. We provide an overview of the experimental setup, including details about the datasets employed and the chosen evaluation metrics. Furthermore, we evaluate our methods in relation to data quality and privacy budgets. Subsequently, we assess the variance of our approach within this section. The results obtained from our experiments not only surpass those of alternative methods but also serve as validation for the efficacy of our proposed techniques.

A. Dataset

We conduct extensive experiments on three publicly available large-scale real-world datasets. **Melbourne parking dataset** [29], collected by sensors among parking bays; **METR-LA traffic dataset** [17], collected by Los Angeles County and **Windmill energy dataset** [11], which is provided by Pytorch Geometric. Each dataset contains large-scale spatial-temporal observations for long periods of time and has been widely used in spatiotemporal data mining applications [20], [27], [29].

B. Experimental Settings

Following the study conducted by Jordon et al. [18], we compare the generated data from the proposed model with real data and generated data from other methods. For real data, we employed a set of predictive models using real training and test data. For the generated data, the set of predictive models is trained on data generated by a generative model and evaluated using real test data.

In the ablation study, we introduce two proposed methods, namely ST-DPGAN and ST-DPGAN-Attn. In ST-DPGAN, we incorporate the transConv1D operation in the generator to facilitate information alignment while utilizing two spatiotemporal blocks from the discriminator side. In the case of ST-DPGAN-Attn, we replace the convolution layer within the spatiotemporal block with an attention block.

Baselines: The generation of spatiotemporal data using GANs with differential privacy is a relatively new and emerging area, as highlighted by Gao et al. [13]. Consequently, there is a limited number of established baselines that are specifically tailored to our research focus, particularly in the context of spatiotemporal data generation rather than image data. In this paper, we explore the application of spatiotemporal data generation tasks using alternative algorithms. We use DPGAN [34] as an important baseline for data generation under a privacy setting. We use WGAN [4] as a baseline for data generation without differential privacy and also an upper bound for DPGAN. Compared to our proposed model, the selected baseline models do not contain the transConv1D module, graph embedding, and two-layer spatiotemporal attention blocks.

Evaluation Metric: We make use of the following predictive models in our experiments: Linear Regression [22], Linear Support Vector Regressor [22], Multiple Layer Perceptron [25], Stochastic Gradient Descent [22], Decision Tree [22], Random Forest [22], Gradient Boosting, Bagging, Ada Boosting [24], Light Gradient Boosting Machine [19] and Long Short-term Memory [16]. The graph-based neural network is not chosen as a regressor here because we would like to remove the potential bias in comparison, since our model utilizes the graph embedding and convolution structure. We select 70%, 20%, and 10% data for training, validation, and testing, respectively.

We evaluate their performance using Mean Square Error (MSE) and Mean Absolute Error (MAE) because the downstream tasks are all regressions. The regression task can be defined as a typical time-series prediction problem, where we use a sliding window of size 6 across the data to make 3-step predictions. In the privacy aspect, we also use ϵ to measure the upper bounds of the privacy loss. A smaller privacy budget usually exhibits a better privacy guarantee over the data, while a higher value potentially brings more risks in privacy leakage.

Setup: All the models were compiled and tested using a single NVIDIA GeForce RTX 3090 24 GB GPU. To validate the performance of the proposed model with different privacy loss, we allocate five different levels of privacy budgets (ϵ) when training on three datasets. In our experiment, we set the sampling probability $q = 0.01$, noise level $\sigma = 2$, and relaxation $\delta = 10^{-7}$. We set the default epoch number as 400, and the training process stops if the epoch number exceeds 400. We also stop the training process for each DP-based model when the privacy budget is exhausted. Generators of all models are optimized using RMSProp optimizer. Discriminators of all DP-related models are optimized through DP-SGD optimizer [3]. In comparison, we experiment with different optimizers to train the discriminators of all non-DP-related models and choose the best one. The batch size in all experiments is set to 10, and the global seed is set for experiment reproduction.

C. Results and Analysis

This section discusses the experimental results obtained under different settings. Table I presents the results of our proposed model and baselines on the three mentioned datasets. We trained a Wasserstein GAN (WGAN) to compare its performance with our proposed method in the above tasks. The training and network settings for the WGAN follow the algorithm outlined in Arjovsky et al. [4].

In most cases, the data generated by ST-DPGAN demonstrate lower loss and outperform the WGAN in regression tasks. Moreover, incorporating the attention mechanism in ST-DPGAN-Attn further enhances the data quality by leveraging the inherent spatial and temporal relationships among nodes. These findings validate the efficacy of our methodology, as presented in Section IV. B.

Table II presents the high-level results of our proposed model under a privacy-preserving setting. We evaluate the quality of the generated data under different privacy budgets defined by ϵ. In our experiments, we compare our methods with DPGAN, which follows the same settings described in Xie et al. [34]. The results indicate that, with the same privacy budgets (represented by ϵ), ST-DPGAN and ST-DPGAN-Attn are more effective than DPGAN in capturing and recovering

TABLE II
AVERAGE RESULTS ON THREE DATASETS USING DIFFERENT PRIVACY BUDGETS ϵ. WE USE BOLD NUMBERS TO INDICATE THE BEST RESULTS FOR EACH ROW. OUR PROPOSED METHODS ARE MARKED BY *. WE USE LSTM AS THE PREDICTION MODEL IN THIS SET OF EXPERIMENTS. THE OTHER HYPER-PARAMETER SETTING FOLLOWS THE SAME MODE, WHICH IS DISCUSSED IN SECTION V.C. THE SETTINGS OF DPGAN ARE FOLLOWED TO THE [34]. THE COUNT ROWS STAND FOR THE TOTAL NUMBER OF BEST RESULTS IN EACH COLUMN.

Methods		DPGAN		ST-DPGAN*		ST-DPGAN-Attn*	
Metrics		MSE	MAE	MSE	MAE	MSE	MAE
Parking	ϵ=1	0.0504	0.1894	0.0578	0.2014	**0.0349**	**0.1488**
	ϵ=4	0.0296	0.1403	0.0481	0.1828	**0.0315**	**0.1385**
	ϵ=8	0.0247	0.1176	0.0157	0.0994	**0.0134**	**0.0885**
	ϵ=10	0.0166	0.0996	**0.0118**	0.0852	0.0129	**0.0824**
	ϵ=12	0.0136	0.0893	**0.0087**	0.0714	**0.0087**	**0.0689**
METR-LA	ϵ=1	391.9801	16.8606	128.0023	8.2780	**124.9095**	**8.1993**
	ϵ=4	381.3257	16.6288	126.3622	8.1894	**122.9242**	**8.0959**
	ϵ=8	348.2818	15.7742	128.2958	8.2870	**122.1717**	**8.0710**
	ϵ=10	344.8490	15.7249	125.7851	**8.1732**	**124.9988**	8.1977
	ϵ=12	324.3259	15.2576	126.8095	8.2104	**122.6244**	**8.0745**
Windmill	ϵ=1	0.1195	0.3167	**0.0967**	**0.2688**	0.1112	0.2864
	ϵ=4	0.1195	0.3167	**0.0945**	**0.2616**	0.1085	0.2828
	ϵ=8	0.1247	0.3233	**0.0940**	**0.2618**	0.1048	0.2780
	ϵ=10	0.1234	0.3217	**0.0901**	**0.2561**	0.1042	0.2773
	ϵ=12	0.1186	0.3156	**0.0897**	**0.2560**	0.0991	0.2703
Count		0	0	7	6	9	9

spatiotemporal information. Furthermore, this performance superiority is consistently maintained across different privacy budgets, suggesting that our method is stable and robust to varying ϵ values. Overall, ST-DPGAN and ST-DPGAN-Attn have demonstrated the potential to generate high-quality data with privacy protection, surpassing existing models.

The significant gap observed between the baseline models and our proposed model can be primarily attributed to the inclusion of the transposed 1D convolutional layer and graph embedding, which facilitate the generation of high-quality spatiotemporal data. An in-depth analysis of these components is provided in the ablation study section.

By comparing the real data and the generated data, we can further explore the performance of ST-DPGAN in terms of noise resistance and the potential for improvement. From the table, it can be observed that a smaller privacy budget leads to relatively higher loss, while a larger privacy budget can potentially increase the utility of generated data but comes with a reduced guarantee of privacy. Additionally, we find that the utility of the generated data decreases in a nonlinear manner. The privacy loss represented by ϵ is not directly proportional to the changes in MSE and MAE. Therefore, certain high-cost-effective points can achieve good performance with acceptable privacy preservation.

D. Ablation Study

We conduct further experiments to study the significance of model components as an ablation study. In detail, we want to find out: (a) how much the transposed 1D convolutional layer in the generator and graph embedding in the discriminator help our task in terms of data quality, and (b) whether increasing the number of spatial and temporal blocks could help produce data of higher quality.

We specifically create 4 variants based on ST-DPGAN: *ST-DPGAN-TC*, where the transposed 1D convolutional layer is removed from generator; and *ST-DPGAN-GE*, where graph embedding is removed from discriminator are used for study (a); *ST-DPGAN-TST*, where an additional temporal block is added to form the sandwich structure from [40], and *ST-DPGAN-TSTS* where additional temporal and spatial blocks are appended for study (b), which have four temporal and spatial blocks in total.

The ablation study results are shown in Table III. We observe a serious performance degeneration for *ST-DPGAN-TC* and *ST-DPGAN-GE* after the transposed 1D convolutional layer and graph embedding are removed. It proves the significance of these components when generating high-quality spatial-temporal data. Increasing the number of spatial and temporal blocks does not bring any performance gain. We attribute this to the instability and difficulty in training brought by increased spatial and temporal blocks, and it has empirically validated our claim in section IV.B.

V. DISCUSSION

In this paper, we employ a Generative Adversarial Network (GAN) framework for generating privacy-preserving spatiotemporal data. While we demonstrated its significant potential in generating high-quality synthetic data, GAN presents inherent challenges such as mode collapse, instability during training, and sensitivity to hyperparameters. In future work, other alternative generative models, such as VAE and diffusion, can also be explored.

We tested the proposed method on limited publicly available datasets. For the privacy problem, we cannot perform the methods on more categories of spatio-temporal data in IoT, such as remote sensors, smart meters, GPS systems, and other

TABLE III
ABLATION STUDY RESULTS ON THREE DATASETS. OUR PROPOSED METHODS ARE MARKED BY *. THE REMAINING COLUMNS SHOW THE MSE AND MAE SCORES OF PREDICTIVE MODELS UNDER *setting B*. OUR MODELS ARE MARKED BY *. THE BEST VALUE IS MARKED IN BOLD. THE LOWER THE SCORE VALUE, THE BETTER THE DATA QUALITY. THE BEST-PERFORMING ENTRY UNDER *setting B* IS IN BOLD. WE CHOOSE THE PRIVACY BUDGET ϵ AS 12 FOR THE DP-BASED MODEL TO COMPARE WITH OTHER NON-DP CONTAINED ALGORITHMS. THE AVERAGE ROW STANDS FOR THE AVERAGE LOSS UNDER DIFFERENT PREDICTION ALGORITHMS. THE COUNT ROWS REPRESENT THE TOTAL NUMBER OF BEST RESULTS IN EACH COLUMN.

	Methods	ST-DPGAN-TST		ST-DPGAN-TSTS		ST-DPGAN-GE		ST-DPGAN-TC		ST-DPGAN-Attn*	
	Metrics	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Parking	LR	0.0070	0.0634	0.0071	0.0662	**0.0066**	**0.0595**	0.0381	0.1535	0.0068	0.0643
	Linear Regression	0.0070	0.0607	0.0068	0.0603	0.0066	0.0589	0.0575	0.1934	**0.0062**	**0.0576**
	MLP	0.0079	0.0684	0.0083	0.0713	0.0105	0.0760	0.0369	0.1507	**0.0068**	**0.0645**
	SGD	0.0071	0.0638	0.0072	0.0664	**0.0066**	**0.0595**	0.0391	0.1551	0.0069	0.0646
	Decision Tree	0.0490	0.1661	0.0267	0.1170	0.0203	0.0961	0.0696	0.1997	**0.0136**	**0.0828**
	Random Forest	0.0153	0.0995	0.0129	0.0924	0.0099	0.0791	0.0530	0.1804	**0.0069**	**0.0613**
	Gradient Boosting	0.0105	0.0790	0.0118	0.0863	0.0081	0.0698	0.0382	0.1534	**0.0071**	**0.0630**
	Bagging	0.0125	0.0840	0.0154	0.0957	0.0109	0.0771	0.0267	0.1268	**0.0076**	**0.0638**
	Ada Boosting	0.0152	0.1003	0.0159	0.1043	0.0100	0.0807	0.0395	0.1584	**0.0096**	**0.0774**
	LGBM	0.0104	0.0787	0.0100	0.0797	0.0078	0.0683	0.0410	0.1589	**0.0065**	**0.0603**
	LSTM	0.0148	0.0982	0.0158	0.1015	0.0155	0.1006	0.0632	0.1953	**0.0087**	**0.0689**
	Average	0.0142	0.0875	0.0125	0.0855	0.0102	0.0751	0.0457	0.1660	0.0079	0.0662
METR-LA	LR	98.4122	7.0480	98.4824	7.0603	97.1667	6.8635	98.4795	7.0600	**97.0645**	**6.7750**
	Linear Regression	98.7342	7.0686	98.4922	7.0887	97.7149	**6.6708**	98.5268	7.0911	**97.2597**	6.6976
	MLP	126.1910	9.4255	120.8335	9.0971	100.3331	7.1784	120.4894	9.0752	**97.6307**	**6.8797**
	SGD	106.1432	7.9692	109.7475	8.2786	**96.9715**	6.8293	109.8588	8.2877	97.0175	**6.7621**
	Decision Tree	524.1221	21.1745	648.9779	23.6976	189.5849	9.7803	442.8109	19.3704	**142.6260**	**8.0813**
	Random Forest	544.3726	21.6705	555.5890	21.9025	135.0329	8.2709	554.9820	21.8896	**108.7084**	**6.8320**
	Gradient Boosting	456.9711	19.7746	480.9894	20.2982	104.7566	7.0779	473.9832	20.1490	**102.2363**	**6.7753**
	Bagging	479.9638	20.2789	460.4139	19.8466	116.7623	7.6960	477.7846	20.1838	**113.0507**	**7.0063**
	Ada Boosting	456.1204	19.7248	475.6228	20.1578	170.1427	11.4454	484.3269	20.3551	**107.5073**	**8.0823**
	LGBM	469.6165	20.0458	487.3716	20.4343	102.7120	7.0131	489.6298	20.4841	**102.0630**	**6.9834**
	LSTM	341.9394	17.0463	354.2284	17.3683	183.4939	12.0563	354.1376	17.3660	**122.6244**	**8.0745**
	Average	336.5988	15.5661	353.7044	15.9300	126.7882	8.2620	336.8190	15.5738	107.9808	7.1772
Windmill	LR	0.1098	0.3054	0.0608	0.1993	0.1272	0.3291	0.1176	0.3163	**0.0581**	**0.2022**
	Linear Regression	0.1031	0.2955	**0.0608**	0.1983	0.1275	0.3295	0.0995	0.2901	0.0771	**0.1751**
	MLP	0.1052	0.2987	0.0615	0.2053	0.1317	0.3350	0.1278	0.3298	**0.0578**	**0.2011**
	SGD	0.1097	0.3052	0.0692	0.2326	0.1277	0.3297	0.0928	0.2793	**0.0582**	**0.2028**
	Decision Tree	**0.1130**	0.2923	0.1247	0.3254	0.2195	0.3785	0.1460	0.3495	0.1353	**0.2702**
	Random Forest	0.1229	0.3233	0.1254	0.3267	0.1282	0.3304	0.1253	0.3265	**0.0637**	**0.2086**
	Gradient Boosting	0.1138	0.3102	0.1246	0.3257	0.1289	0.3310	0.1478	0.3534	**0.0590**	**0.2063**
	Bagging	0.1010	0.2863	0.1236	0.3243	0.1397	0.3363	0.1281	0.3290	**0.0699**	**0.2137**
	Ada Boosting	0.1223	0.3225	0.1226	0.3230	0.1283	0.3306	0.1221	0.3224	**0.0604**	**0.2075**
	LGBM	0.1230	0.3235	0.1236	0.3243	0.1280	0.3302	0.1263	0.3279	**0.0592**	**0.2069**
	LSTM	0.0942	**0.2026**	0.0932	0.2120	0.0788	0.2529	**0.0727**	0.2397	0.0991	0.2703
	Average	0.1107	0.2968	0.0991	0.2724	0.1332	0.3284	0.1187	0.3149	0.0725	0.2149
	Count	1	1	1	0	3	3	0	0	27	29

connected technologies. This data is crucial for applications like smart cities, traffic management, energy monitoring, and environmental sensing. We hope our work could enable these data to be published and offer a more realistic analysis for the real-world IoT applications.

In this work, we focus on how differential privacy mechanisms can be performed in the spatio-temporal domain. To provide a straightforward comparison with previous models, we maintain the same privacy definitions and budget counting methods as in prior studies. Our primary contribution lies in the design of a model architecture that strikes a balance between data quality and differential privacy. However, spatio-temporal data differs significantly from images or language data. Due to the complex spatial and temporal relationships inherent in such data, determining whether a tighter bound for privacy budgets can be proposed remains an area for future research.

VI. CONCLUSION

In this paper, we introduced ST-DPGAN, a novel generative adversarial model for creating differential privacy-compliant spatiotemporal graph data. Our experiments show that ST-DPGAN surpasses other models in generating high-quality, privacy-preserving spatiotemporal graph data on three real-world datasets. The ST-DPGAN model promises wide-ranging applications, offering comprehensive data for unbiased neural network training and robustness against abnormal data. Its capability to handle diverse data sources allows for its application beyond specific domains like traffic and weather prediction, potentially improving various aspects of daily life. Crucially, ST-DPGAN addresses significant privacy concerns and legal risks in spatiotemporal data analysis, marking a significant step in responsible data handling. ST-DPGAN addresses the privacy issue in spatiotemporal data generation.

References

[1] Art. 25 general data protection regulation (gdpr) data protection by design and by default. https://gdpr-info.eu/art-25-gdpr/.

[2] S.2134 - data protection act of 2021. www.congress.gov/bill/117th-congress/senate-bill/2134.

[3] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[5] Chandra Churh Chatterjee, Manjunath Mulimani, and Shashidhar G. Koolagudi. Polyphonic sound event detection using transposed convolutional recurrent neural network. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 661–665, 2020.

[6] Qingrong Chen, Chong Xiang, Minhui Xue, Bo Li, Nikita Borisov, Dali Kaarfar, and Haojin Zhu. Differentially private data generative models. *arXiv preprint arXiv:1812.02274*, 2018.

[7] Rishav Chourasia, Jiayuan Ye, and Reza Shokri. Differential privacy dynamics of langevin diffusion and noisy gradient descent. *Advances in Neural Information Processing Systems*, 34:14771–14781, 2021.

[8] Tim Dockhorn, Tianshi Cao, Arash Vahdat, and Karsten Kreis. Differentially private diffusion models. *arXiv preprint arXiv:2210.09929*, 2022.

[9] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[10] Alireza Ermagun and David Levinson. Spatiotemporal traffic forecasting: review and proposed directions. *Transport Reviews*, 38(6):786–814, 2018.

[11] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[12] Hongyang Gao, Hao Yuan, Zhengyang Wang, and Shuiwang Ji. Pixel transposed convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 42(5):1218–1227, 2019.

[13] Nan Gao, Hao Xue, Wei Shao, Sichen Zhao, Kyle Kai Qin, Arian Prabowo, Mohammad Saiedur Rahaman, and Flora D Salim. Generative adversarial networks for spatio-temporal data: A survey. *arXiv preprint arXiv:2008.08903*, 2020.

[14] Nan Gao, Hao Xue, Wei Shao, Sichen Zhao, Kyle Kai Qin, Arian Prabowo, Mohammad Saiedur Rahaman, and Flora D Salim. Generative adversarial networks for spatio-temporal data: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(2):1–25, 2022.

[15] Yaowei Han, Sheng Li, Yang Cao, Qiang Ma, and Masatoshi Yoshikawa. Voice-indistinguishability: Protecting voiceprint in privacy-preserving speech data release. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2020.

[16] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 11 1997.

[17] H. V. Jagadish, Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M. Patel, Raghu Ramakrishnan, and Cyrus Shahabi. Big data and its technical challenges. *Commun. ACM*, 57(7):86–94, July 2014.

[18] James Jordon, Jinsung Yoon, and Mihaela van der Schaar. Pategan: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*, 2018.

[19] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 3149–3157, Red Hook, NY, USA, 2017. Curran Associates Inc.

[20] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations (ICLR '18)*, 2018.

[21] Qijiong Liu, Jieming Zhu, Yanting Yang, Quanyu Dai, Zhaocheng Du, Xiao-Ming Wu, Zhou Zhao, Rui Zhang, and Zhenhua Dong. Multimodal pretraining, adaptation, and generation for recommendation: A survey. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining KDD*, 2024.

[22] Tom Michael Mitchell et al. *Machine learning*, volume 1. McGraw-hill New York, 2007.

[23] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755*, 2016.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[25] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[26] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

[27] Sebastian Schmoll and Matthias Schubert. Semi-markov reinforcement learning for stochastic resource collection. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3349–3355. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

[28] Wei Shao, Flora D Salim, Andy Song, and Athman Bouguettaya. Clustering big spatiotemporal-interval data. *IEEE Transactions on Big Data*, 2(3):190–203, 2016.

[29] Wei Shao, Sichen Zhao, Zhen Zhang, Shiyu Wang, Mohammad Saiedur Rahaman, Andy Song, and Flora Dilys Salim. Fadacs: A few-shot adversarial domain adaptation architecture for context-aware parking availability sensing, 2021.

[30] Xiaoteng Shen, Rui Zhang, Xiaoyan Zhao, Jieming Zhu, and Xi Xiao. PMG : Personalized multimodal generation with large language models. In *Proceedings of the ACM on Web Conference 2024, WWW*, 2024.

[31] Weiyan Shi, Ryan Shea, Si Chen, Chiyuan Zhang, Ruoxi Jia, and Zhou Yu. Just fine-tune twice: Selective differential privacy for large language models. *arXiv preprint arXiv:2204.07667*, 2022.

[32] Bayu Distiawan Trisedya, Jianzhong Qi, Rui Zhang, and Wei Wang. GTR-LSTM: A triple encoder for sentence generation from RDF data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL*, 2018.

[33] Benjamin Weggenmann, Valentin Rublack, Michael Andrejczuk, Justus Mattern, and Florian Kerschbaum. Dp-vae: Human-readable text anonymization for online reviews with differentially private variational autoencoders. In *Proceedings of the ACM Web Conference 2022*, pages 721–731, 2022.

[34] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*, 2018.

[35] Andy Yuan Xue, Jianzhong Qi, Xing Xie, Rui Zhang, Jin Huang, and Yuan Li. Solving the data sparsity problem in destination prediction. *VLDB J.*, 24(2):219–243, 2015.

[36] Andy Yuan Xue, Rui Zhang, Yu Zheng, Xing Xie, Jin Huang, and Zhenghua Xu. Destination prediction by sub-trajectory synthesis and privacy protection against such prediction. In *29th IEEE International Conference on Data Engineering, ICDE*, pages 254–265, 2013.

[37] Hao Xue, Flora Salim, Yongli Ren, and Nuria Oliver. Mobtcast: Leveraging auxiliary trajectory forecasting for human mobility prediction. *Advances in Neural Information Processing Systems*, 34:30380–30391, 2021.

[38] Qingqing Ye, Haibo Hu, Ninghui Li, Xiaofeng Meng, Huadi Zheng, and Haotian Yan. Beyond value perturbation: Local differential privacy in the temporal setting. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pages 1–10. IEEE, 2021.

[39] Zonggen Yi and Peter H Bauer. Spatiotemporal energy demand models for electric vehicles. *IEEE Transactions on Vehicular Technology*, 65(3):1030–1042, 2015.

[40] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

[41] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019.
[42] Rui Zhang, H. V. Jagadish, Bing Tian Dai, and Kotagiri Ramamohanarao. Optimized algorithms for predictive range and KNN queries on moving objects. *Inf. Syst.*, 35(8):911–932, 2010.
[43] Rui Zhang, Beng Chin Ooi, and Kian-Lee Tan. Making the pyramid technique robust to query types and workloads. In *Proceedings of the 20th International Conference on Data Engineering, ICDE*, 2004.
[44] Rui Zhang, Jianzhong Qi, Dan Lin, Wei Wang, and Raymond Chi-Wing Wong. A highly optimized algorithm for continuous intersection join queries over moving objects. *VLDB J.*, 21(4):561–586, 2012.
[45] Xiaoguang Zhou and Xinmeng Tang. Spatiotemporal consistency effect of green finance on pollution emissions and its geographic attenuation process. *Journal of Environmental Management*, 318:115537, 2022.