


# LSTM-TrajGAN: A Deep Learning Approach to Trajectory Privacy Protection

Jinmeng Rao 

Geospatial Data Science Lab, University of Wisconsin-Madison, USA

Song Gao 

Geospatial Data Science Lab, University of Wisconsin-Madison, USA


Yuhao Kang 

Geospatial Data Science Lab, University of Wisconsin-Madison, USA

Qunying Huang 

Department of Geography, University of Wisconsin-Madison, USA

## Abstract

 The prevalence of location-based services contributes to the explosive growth of individual-level trajectory data and raises public concerns about privacy issues. In this research, we propose a novel LSTM-TrajGAN approach, which is an end-to-end deep learning model to generate privacy-preserving synthetic trajectory data for data sharing and publication. We design a loss metric function TrajLoss to measure the trajectory similarity losses for model training and optimization. The model is evaluated on the trajectory-user-linking task on a real-world semantic trajectory dataset. Compared with other common geomasking methods, our model can better prevent users from being re-identified, and it also preserves essential spatial, temporal, and thematic characteristics of the real trajectory data. The model better balances the effectiveness of trajectory privacy protection and the utility for spatial and temporal analyses, which offers new insights into the GeoAI-powered privacy protection.

**2012 ACM Subject Classification** Security and privacy → Privacy protections; Computing methodologies → Artificial intelligence

**Keywords and phrases** GeoAI, Deep Learning, Trajectory Privacy, Generative Adversarial Networks

**Acknowledgements** Support for this research was provided by the University of Wisconsin - Madison Office of the Vice Chancellor for Research and Graduate Education with funding from the Wisconsin Alumni Research Foundation.

## 1 Introduction

The increasing location-based services (LBS) have generated large-scale individual-level trajectory data (i.e., a sequence of locations with attributes) through mobile phones, wearable sensors, GPS devices, and geotagged social media [19]. Such trajectory big data provide new opportunities to study human mobility patterns and human-environment interactions [11], disaster responses [12, 27] and public health issues [17, 25]. It also introduces grand challenges regarding the protection of geoprivacy and broader behavioral, social, ethical, legal and policy implications [14]. Generally speaking, trajectory privacy refers to an individual's rights to prevent the disclosure of individual trajectory identity and associated personal sensitive locations [15, 2, 5].

Due to the data breach concerns and increasing public awareness of location privacy protection, many approaches have been proposed to prevent users' trajectories from being identified. A common practice is to remove the identifiers (e.g., user name or ID number) from the trajectory data. However, it turned out that such "de-identified" trajectories may

<sup>1</sup> a preprint and the final version will be available in the Proceedings of the 11th International Conference on Geographic Information Science (GIScience 2021) <https://www.giscience.org/>

still cause serious privacy threats since the spatial, temporal and thematic characteristics of trajectories can still be used as strong quasi-identifiers for linking the trajectories to their creators [2]. Another commonly used method is to aggregate trajectory points into geographic or administrative units so that their original locations are not revealed. Nevertheless, recent studies show that aggregation may not only fail to preserve user privacy, but also reduce the spatial resolution and effectiveness of spatial analysis [3] [28] [5]. For example, De Montjoye et al. [3] lower the resolution of a human mobility trace dataset through spatial and temporal aggregation to preserve the individuals from being identified, but the coarsened dataset still provides little anonymity. Thus, in order to achieve trajectory privacy protection more efficiently, we need to deal with the spatial and temporal characteristics of trajectory data more specifically.

Current trajectory privacy protection studies focus on two research streams. One is the differential privacy approach to grouping and mixing the trajectories from different users so that the identification of individual trajectory data is converted into a  $k$ -anonymity problem [23] [31]. For example, the spatial cloaking approach mixes together the trajectory points between  $k$  users using  $k$ -anonymous cloaked spatial regions, making these trajectories  $k$ -anonymized [7]. Also, the mix-zones approach anonymizes the trajectory points in a mix-zone using pseudonyms and breaks the linkage between the former segment and the latter segment of the same trajectory that passes through a mix-zone [24]. Alternatively, the generalization-based approach first divides the points of  $k$  trajectories into different  $k$ -anonymized regions, and then reconstructs  $k$  new trajectories by uniformly selecting points from each  $k$ -anonymized regions and linking them together [22].

Another research stream is called geomasking, which blurs the locations of original trajectory data by utilizing perturbation on the spatial dimension so that the original locations can be hidden or modified while spatial patterns may not be significantly affected [9] [5]. For example, Armstrong et al. [1] explored the privacy preservation ability and spatial analysis effectiveness of several types of geomasks. Kwan et al. [15] evaluated the spatial analysis effectiveness of three different random perturbation geomasks on lung-cancer deaths. Seidl et al. [26] applied grid masking and random perturbation on GPS trajectory data and evaluated the privacy protection performance. Gao et al. [5] investigated the effectiveness of random perturbation, gaussian perturbation, and aggregation on Twitter data as well as explored the privacy, analytics, and uncertainty level of each method.

While these approaches all show the capabilities to protect trajectory privacy, they also expose several limitations. First of all, despite the diversity, the goal of these approaches largely is to obfuscate the trajectory locations and add more uncertainty to preserve privacy. However, the trade-off between the effectiveness of trajectory privacy protection and the utility for spatial and temporal analyses is still hard to control [18], and this issue has not been fully discussed or evaluated. Besides, current studies mainly focus on the spatial dimension of trajectory data whereas other semantics (e.g., temporal and thematic attributes) are rarely considered. In fact, these characteristics have been proven to be crucial for trajectory user identification [21]. Moreover, current approaches rely heavily on manually designed procedures. Once the procedure is disclosed, one may have the chance to recover the original trajectory data [28] (e.g., using reverse engineering). The "black-box" machine learning models may help to solve this issue.

To this end, this research aims to explore the effectiveness of state-of-the-art deep learning approaches for trajectory privacy protection. We propose a novel LSTM-TrajGAN model that combines the Long Short-Term Memory (LSTM) recurrent neural network and the Generative Adversarial Network (GAN) structure together to generate privacy-preserving

synthetic trajectories as alternatives to real trajectories for trajectory data sharing and publication. Two research questions (RQ) will be investigated in this work.

RQ 1: How effective is the proposed LSTM-TrajGAN model in protecting the trajectory creators from being re-identified? (i.e., privacy protection effectiveness)

RQ 2: Can the synthetic trajectories preserve the semantic features (spatial-temporal-thematic characteristics) compared to real trajectories? (i.e., utility)

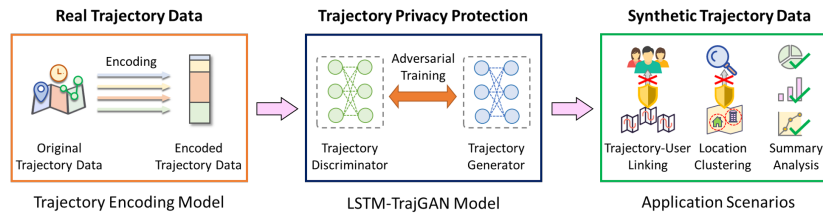
The main contributions of our work are fourfold: (1) we propose an end-to-end deep learning approach to generating privacy-preserving trajectory data. The procedure is simple and highly secure (a GeoAI "black-box"); (2) we introduce a trajectory encoding model for semantic trajectory encoding; (3) we design a new TrajLoss metric function to measure the trajectory similarity losses for training deep learning models; and (4) we evaluate the privacy protection effectiveness and the utility of the proposed model using real-world LBS data and explore the trade-off between them.

The remainder of the paper is organized as follows. Section 2 introduces our methodological framework, including a trajectory encoding model, the LSTM-TrajGAN model, and the TrajLoss function design. In section 3, we train and test our model using a city-scale weekly trajectory dataset and compare with other commonly used trajectory privacy protection methods. Both privacy protection effectiveness and utility are evaluated and compared with baseline approaches; In section 4, we discuss the factors affecting privacy protection effectiveness, the trade-off between privacy protection and utility, and the limitations of our model. Section 5 summarizes this research and outlines the future work.

## 2 Method

Inspired by the vision of the TrajGANs [18], we propose a new approach consisting of three main components: (1) a Trajectory Encoding Model, which encodes GPS location coordinates, temporal attributes, and other attributes such as point of interest (POI) category; (2) a Trajectory Generator, which takes random noise and original trajectories as inputs to generate synthetic trajectories as outputs; and (3) a Trajectory Discriminator, which takes trajectories as inputs and determines them as "real" or "synthetic".

The overall workflow is described in Figure 1. The goal is to train an "intelligent" trajectory generator that generates "realistic" synthetic trajectories to replace the original trajectories, which preserves differential privacy in trajectory analysis tasks such as Trajectory-User Linking (TUL) and trajectory data mining (e.g., work/home location clustering). Meanwhile, it ensures the quality of multiple spatial or temporal summary analysis tasks. Such a framework can serve as a trajectory privacy protection layer in trajectory data acquisition, processing, and publication pipelines, which publish the synthetic alternatives rather than the real trajectory data that may disclose individual privacy.



■ **Figure 1** The overall workflow of the proposed LSTM-TrajGAN approach.

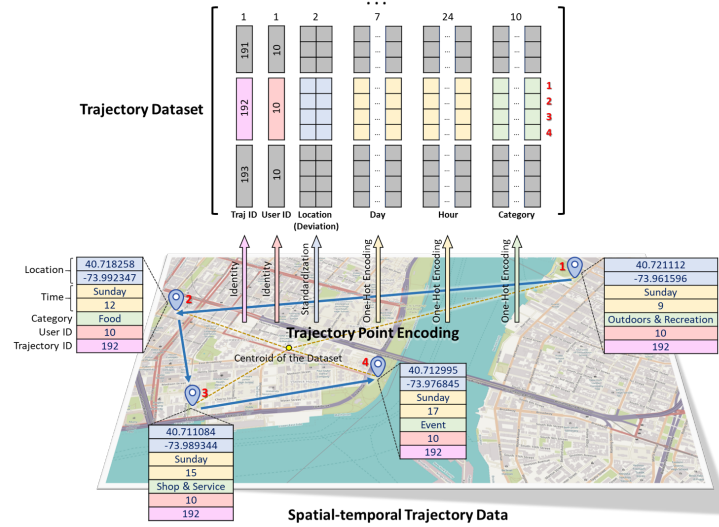
## 2.1 Trajectory Encoding

First, we introduce a trajectory encoding model that converts the original trajectory data to a specific format that serves as the inputs for the LSTM-TrajGAN model. The main reason for the encoding process is that the trajectory data usually contain various types of attributes such as interval data (e.g., GPS coordinates, date and time), nominal data (e.g., POI category), ordinal data (e.g., POI rating), and these data need to be converted into valid numerical representations for training the deep learning model. Our trajectory encoding model includes two parts: trajectory point encoding and trajectory padding.

### Trajectory Point Encoding

The trajectory point encoding process is illustrated in Figure 2. A semantic trajectory point contains the following attributes: location, time, user id, trajectory id, and other optional attributes such as POI category. For the location attribute, we standardize all the latitudes and longitudes using the centroid of all the trajectories in the dataset to obtain the deviations of the latitudes and longitudes from the centroid. In this way, the model can better learn the spatial deviation pattern between different trajectory points. These deviation values will be used as the numerical representations of the trajectory points for constructing spatial embeddings [20].

For the temporal attributes and categorical attributes, we use the one-hot encoders (i.e., a representation process using dummy variables in machine learning) to encode the attributes into high-dimensional binary vectors based on their vocabulary sizes. For example, the "Day" attribute is encoded into 7-dimensional binary vectors, and "Monday" is represented as  $[1, 0, 0, 0, 0, 0, 0]$ . Likewise, the "Hour" attribute is encoded into 24-dimensional binary vectors, and the "Category" attribute is encoded into 10-dimensional binary vectors. Note that we don't encode the User ID and the Trajectory ID since they are only used to indicate the user and the trajectory that the point belongs to.



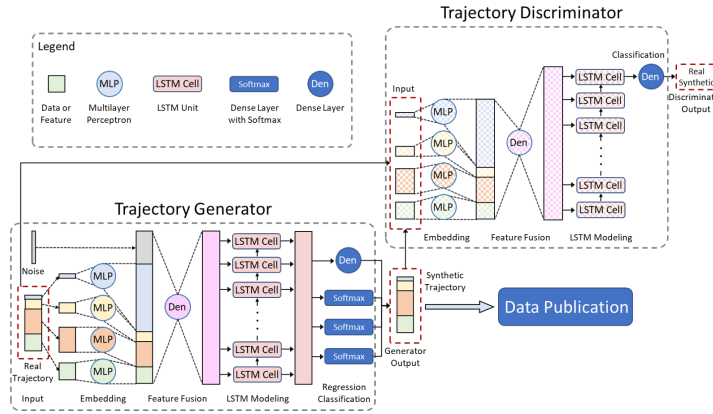
■ **Figure 2** An example for the trajectory point encoding process.

### Trajectory Padding

After the trajectory point encoding process, all the spatial, temporal, and thematic attributes of a trajectory are stored in a multidimensional matrix, whose first dimension indicates the index for each trajectory. Since the length of each trajectory data (i.e., the number of the trajectory points) is a variable, we then apply the trajectory padding technique to make sure all the trajectories have the same length as the longest trajectory. Specifically, we use zero pre-padding to pad empty trajectory points (i.e., the points whose attributes are all set to zero) to each trajectory until all the trajectories reached the same length as the longest trajectory in the dataset. The main reason is that the data with the same size can be utilized for batch processing and training the deep learning model, which would speed up the training process. During the model training and inference processes, these padded trajectory points will be masked (i.e., cut) and they won't actually influence the neural network weight updates and the derived results.

### 2.2 LSTM-TrajGAN Model

Figure 3 describes the neural network structure of the LSTM-TrajGAN Model. The trajectory generator captures the data distribution and pattern of the real trajectory data and generates synthetic trajectory data based on their corresponding original trajectory data and random noise. In addition, the trajectory discriminator distinguishes whether the trajectory samples come from the training set (i.e., real trajectory data) or the trajectory generator (i.e., synthetic trajectory data). The goal of the trajectory generator is to generate "high-quality" synthetic trajectories that can "fool" the trajectory discriminator, which leads to a two-player minimax game between them. The generated synthetic trajectories aim to be competent for spatial and temporal summary analysis, while having some degree of uncertainty and randomness to protect the user privacy in trajectory analysis tasks with privacy issues involved. This idea is reflected in the design and optimization of the LSTM-TrajGAN model.



■ **Figure 3** The neural network structure of the LSTM-TrajGAN Model.

### Trajectory Generator

As is shown in Figure 3 the trajectory generator consists of five functional layers: the input layer, the embedding layer, the feature fusion layer, the LSTM modeling layer, and the regression/classification layer. The generator first takes the encoded real trajectories and random noise as inputs, and embeds trajectories using Multilayer Perceptrons (MLPs) [10].

For the spatial dimension of a trajectory (i.e., pairs of latitude and longitude deviations), we embed each pair of them using a MLP to get 64-dimensional vectors. For the temporal dimension (e.g., day and hour) and categorical attributes (e.g., POI category), we use MLPs to embed them respectively and get fixed-length vectors based on their vocabulary sizes:

$$e_i^{spatial} = \phi^s(\Delta lat_i, \Delta lon_i; W_{es}) \quad (1)$$

$$e_i^{day} = \phi^d(v_i^{day}; W_{ed}) \quad (2)$$

$$e_i^{hour} = \phi^h(v_i^{hour}; W_{eh}) \quad (3)$$

$$e_i^{category} = \phi^c(v_i^{category}; W_{ec}) \quad (4)$$

Where  $\Delta lat_i$  and  $\Delta lon_i$  stand for the latitude and longitude deviation of the  $i$ -th trajectory point;  $v_i^{day}$ ,  $v_i^{hour}$ ,  $v_i^{category}$  stand for the one-hot vectors for the day, hour, and category attributes of the  $i$ -th trajectory point;  $\phi^s$ ,  $\phi^d$ ,  $\phi^h$ , and  $\phi^c$  stand for the MLPs with an activation function – the Rectified Linear Unit (ReLU) for embedding the spatial, daily, hourly, and categorical attributes;  $W_{es}$ ,  $W_{ed}$ ,  $W_{eh}$ , and  $W_{ec}$  are the embedding weight matrices for these MLPs;  $e_i^{spatial}$ ,  $e_i^{day}$ ,  $e_i^{hour}$ , and  $e_i^{category}$  are the embedded vectors for each attribute respectively. Note that the embedding weight matrices are shared among all trajectory points.

After the embedding process, we further concatenate all the vectors and the random noise, and then use a dense layer to fuse them into 100-dimensional vectors. By leveraging the feature fusion, we take the advantage of all the spatial, temporal, and categorical characteristics of each trajectory point and fuse them together to support spatiotemporal trajectory modeling and generation. In the LSTM modeling layer, we use a many-to-many LSTM structure that takes a sequence with specific time steps as the input and generates a sequence with the same time steps as the output. Recurrent models such as LSTM are proven to be efficient in spatial-temporal sequence modeling and prediction [8, 21]. Given the dimension of the fused feature, we assign 100 units in the LSTM model and feed the fused features to the model:

$$H = LSTM(F; W_{lstm}) \quad (5)$$

Where  $F$  represents for the fused features of all the trajectory points in a trajectory (i.e.,  $F = [f_0, f_1, \dots, f_{maxlength-1}]$ ), in which  $f_i$  is the fused feature vector for the  $i$ -th trajectory point;  $H$  is the output of the LSTM model, which has the same time step dimensions as the input (i.e.,  $H = [h_0, h_1, \dots, h_{maxlength-1}]$ , in which  $h_i$  is the modeling output vector for  $f_i$ );  $W_{lstm}$  is the weight matrix of the LSTM model.

Finally, we decode the synthetic trajectory data from the output  $H$  of the LSTM modeling layer. Each feature vector  $h_i$  in  $H$  is a 100-dimensional vector containing the spatial, temporal, and categorical characteristics of a synthetic trajectory point. To decode the latitude and longitude deviations, we use a dense layer with two units and use the  $\tanh$  hyperbolic tangent function. In addition, we further stretch the output range to make sure its range covers all the possible deviation values. To decode the day, hour and category attributes, we use dense layers that have as many units as the vocabulary sizes, and use the  $\text{softmax}$  normalized exponential function to recover the one-hot representation of these attributes:

$$(\Delta lat'_i, \Delta lon'_i) = D^s(h_i; W_{ds}) \quad (6)$$

$$v_i^{day} = D^d(h_i; W_{dd}) \quad (7)$$

$$v_i^{hour} = D^h(h_i; W_{dh}) \quad (8)$$

$$v_i'^{category} = D^c(h_i; W_{dc}) \quad (9)$$

Where  $\Delta lat'_i$  and  $\Delta lon'_i$  are the latitude and longitude deviations of the  $i$ -th synthetic trajectory point;  $v_i'^{day}$ ,  $v_i'^{hour}$ ,  $v_i'^{category}$  represent the one-hot vectors for the day, hour, and category attributes of the  $i$ -th synthetic trajectory point;  $D^s$ ,  $D^d$ ,  $D^h$ , and  $D^c$  represent the dense layers with a *tanh* or *softmax* function for decoding the location, day, hour, and category attributes;  $W_{ds}$ ,  $W_{dd}$ ,  $W_{dh}$ , and  $W_{dc}$  are the decoding weight matrices for these dense layers; Note that the decoding weight matrices are shared among all trajectory points.

### Trajectory Discriminator

As is shown in Figure 3 the trajectory discriminator has a very similar structure as the trajectory generator. The major differences between them are:

- (1) The discriminator only takes trajectory data as the input (no random noise needed);
- (2) We use a many-to-one LSTM model that takes the features with time steps as the input and make one scalar as the output:

$$h = LSTM(F; W_{lstm_d}) \quad (10)$$

Where  $F$  represents for the fused features of all the trajectory points in a trajectory (i.e.,  $F = [f_0, f_1, \dots, f_{maxlength-1}]$ ), in which  $f_i$  is the fused feature vector for the  $i$ -th trajectory point;  $W_{lstm_d}$  is the weight matrix of the LSTM model; and  $h$  is the output scalar of the LSTM model.

- (3) We use a one-unit dense layer with the *sigmoid* activation function to make binary classification (real or synthetic) on the scalar output:

$$O_d = D^{bc}(h; W_{bc}) \quad (11)$$

Where  $D^{bc}$  is the one-unit dense layer with a *sigmoid* function used to make binary classification, and  $W_{bc}$  is its weight matrix;  $O_d$  is the final output of the discriminator.

### 2.3 TrajLoss for Measuring Trajectory Similarity Losses

The original GAN is designed to optimize the following objective function [6]:

$$V(D, G) = \min_G \max_D (\mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (12)$$

Where  $p_{data}(x)$  represents the distribution of the real data samples;  $p_z(z)$  represents a prior on noise variables;  $D(x)$  represents the probability that  $x$  came from  $p_{data}(x)$ ;  $G(z)$  represents a mapping from  $p_z(z)$  to  $p_{data}(x)$ . The generator aims to minimize  $\mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$  while the discriminator aims to maximize  $\mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$ , leading to a two-player minimax game.

According to the objective function  $V(D, G)$ , the loss function for the discriminator can be considered as a Binary Cross-Entropy (BCE) loss function ( $L_{BCE}$ ), which will also be used in training the generator. However, different from the original GAN, we need the real trajectory data as inputs. Thus, we design a new loss metric function named TrajLoss to further measure the similarity losses between the real trajectory data and the synthetic trajectory data in spatial, temporal and categorical dimensions, and use this loss function to train the generator. The TrajLoss is defined as follows:

$$TrajLoss(y^r, y^p, t^r, t^s) = \alpha L_{BCE}(y^r, y^p) + \beta L_s(t^r, t^s) + \gamma L_t(t^r, t^s) + c L_c(t^r, t^s) \quad (13)$$

Where  $y^r$  and  $y^p$  represent the ground truth label and the prediction result of the trajectory by the discriminator, respectively;  $t^r$  and  $t^s$  represent the real trajectory and the

corresponding synthetic trajectory;  $L_{BCE}$  is the original binary cross-entropy loss from the discriminator;  $L_s$ ,  $L_t$ , and  $L_c$  are the spatial similarity loss, temporal similarity loss, and the categorical similarity loss between the real and synthetic trajectories, respectively;  $\alpha$ ,  $\beta$ ,  $\gamma$ , and  $c$  are the weights for these losses and can be assigned differently for different scenarios.

In this paper, we use the L2 loss (i.e., least square errors) for  $L_s$  as a recent study [8] shows that the L2 loss is effective in measuring trajectory spatial similarity. Besides, we choose the Softmax Cross-Entropy (SCE) as the loss function for  $L_t$  and  $L_c$  since they are all regarded as multi-classification problems in this framework, and thus can be optimized using SCE. During the model training, the weights of the generator will be updated by the TrajLoss to improve the quality of the synthetic trajectory data.

### 3 Experiments

To address the abovementioned RQ1, this section first evaluates the effectiveness of trajectory privacy protection using the proposed LSTM-TrajGAN model on a classic LBS task: Trajectory-User Linking (TUL), which identifies users from trajectories and link trajectories to them [4]. TUL is an essential task in geo-tagged social media applications and receives increasing privacy concerns [4, 30, 21]. The evaluation can be regarded as an adversarial experiment: we train the LSTM-TrajGAN model and use the generated synthetic trajectories to suppress the accuracy of a state-of-the-art TUL algorithm. We also compare our approach with the other two commonly used location privacy protection methods: Random Perturbation and Gaussian Geomasking.

Meanwhile, to address the RQ2 for verifying the utility of the proposed model (i.e., the usefulness of the synthetic trajectories in analysis), we also explore the spatial and temporal characteristics of the synthetic trajectories to see if they preserve sufficient information from the original trajectories to further support spatial and temporal analyses.

#### 3.1 Trajectory-User Linking

##### Dataset

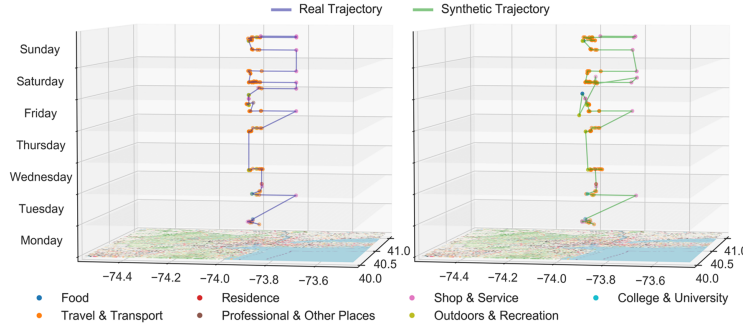
We use the Foursquare weekly trajectory dataset in New York City (NYC) provided by Petry et al. [21], which is extracted from the Foursquare NYC check-ins dataset [29]. We only keep the user ID, trajectory ID, location, hour, day, and category attributes and remove other attributes (e.g., price tier, rating, weather). The summary of the attributes is shown in Table 1. There are 193 users, 3,079 trajectories and 66,962 trajectory points in total in the dataset. We use 2/3 of the trajectories for training the LSTM-TrajGAN model and 1/3 for testing as suggested in [21].

##### Training and Evaluation

We train the LSTM-TrajGAN model on the training set for 2,000 epochs with several default training hyperparameters (e.g., we use an adam optimizer with a learning rate of 0.001 and set the batch size to 256). After the training process, the trajectory data from the test set as well as random noise are then used as the input of the generator to get synthetic trajectory data. A visualization example of a real trajectory from the test data and its corresponding synthetic trajectory generated by our model is shown in Figure 4 as a comparison. Next, we use the MARC (Multiple-Aspect tRajjectory Classifier [21]), a start-of-the-art TUL algorithm, to perform the TUL task on both the test data and our synthetic data. Same as [21], we evaluate the TUL accuracy with five commonly used metrics: ACC@1 (Top-1 Accuracy, showing the model’s ability to have the correct label to be the most probable label candidate), ACC@5 (Top-5 Accuracy, showing the model’s ability to have the correct label among the top 5 most probable label candidates), Macro-P (Macro Precision, the mean precision among



all classes), Macro-R (Macro Recall, the mean recall among all classes), and Macro-F1 (the harmonic mean of Macro-P and Macro-R). For comparison, we also evaluate the privacy protection effectiveness of Random Perturbation (spatial filter: within 1 km; temporal filter: within 24 hours) and Gaussian Geomasking (spatial filter: standard deviation = 0.001; temporal filter: within 24 hours).



■ **Figure 4** The visualization example of a real trajectory from the test data and its corresponding synthetic trajectory generated by our model.

The results are shown in Table 2. The higher the TUL accuracy, the worse the capability for trajectory privacy protection. One can conclude that the synthetic data generated by the LSTM-TrajGAN successfully suppress the scores in the four metrics (ACC@1, Macro-P, Macro-R, and Macro-F) from over 0.900 to around 0.400. The Top-5 Accuracy is decreased from over 0.976 to 0.722. The results show that our model can effectively prevent users from being identified by analyzing the trajectories. Additionally, Random Perturbation has limited effectiveness in protecting trajectory privacy regarding the TUL task, and Gaussian Geomasking works better while still has higher scores than our model. The results also indicate that leveraging both spatial and temporal dimensions of the trajectories simultaneously leads to better privacy-preserving performance than using only the spatial dimension.

■ **Table 1** The summary of the Foursquare NYC weekly trajectory dataset.

Attribute	Type	Number / Range
Trajectory ID	integer	3,079
User ID	integer	193
Latitude	float	(40.550852, 40.988332)
Longitude	float	(-74.269644, -73.685767)
Hour	integer	24
Day	string	7
Category	string	10

### 3.2 Synthetic Trajectory Characteristics Analysis

Here, we analyze the spatial and temporal characteristics and other properties of the synthetic trajectories generated by the LSTM-TrajGAN to evaluate its utility (RQ2).

#### Spatial Characteristics

The spatial characteristics are explored based on two metrics: the Hausdorff Distance and the Jaccard Index. The Hausdorff Distance is a metric for measuring the distance between two

■ **Table 2** The privacy protection effectiveness of different privacy protection methods on the TUL task (RP stands for Random Perturbation; Gaussian stands for Gaussian Geomasking).

Method	ACC@1	ACC@5	Macro-F1	Macro-P	Macro-R
Original	0.938	0.976	0.925	0.937	0.927
RP (Spatial Only)	0.777	0.934	0.758	0.806	0.764
RP (Spatial-Temporal)	0.668	0.888	0.640	0.711	0.654
Gaussian (Spatial Only)	0.561	0.832	0.522	0.573	0.537
Gaussian (Spatial-Temporal)	0.486	0.766	0.431	0.488	0.470
LSTM-TrajGAN	<b>0.459</b>	<b>0.722</b>	<b>0.381</b>	<b>0.429</b>	<b>0.428</b>

point sets in a metric space and has been widely used for measuring the spatial dissimilarity between two trajectories. The Jaccard Index, also known as the Intersection over Union, is an efficient metric for measuring how much the two sample sets or regions overlap, and we use this to indicate the similarity of the activity spaces between two trajectories [18]. We calculate the Hausdorff Distance between each pair of the original and the synthetic trajectories. Likewise, we also calculate the Jaccard Index between the convex hulls of them since the convex hull can generally represent the activity space of LBS users [16]. Table 3 presents the summary of these metrics.

It shows that Random Perturbation has the smallest average Hausdorff Distance (0.004) and the largest average Jaccard Index (0.763), which makes sense since it only makes a limited influence on the spatial dimension of the trajectories. While such a method could preserve spatial similarity well, it sacrifices the location privacy. Our model performs better than Gaussian Geomasking on these two metrics and also better suppresses the abovementioned TUL metrics, which strikes a better balance between spatial similarity and location privacy.

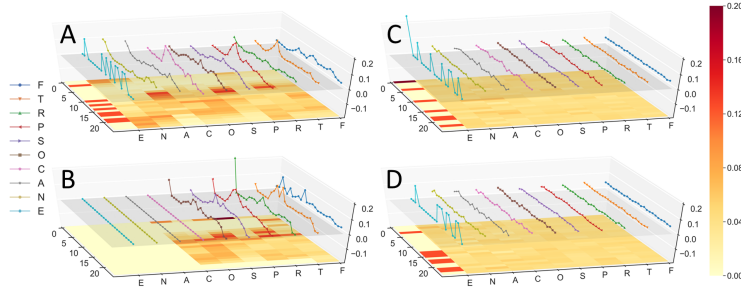
■ **Table 3** Spatial characteristics evaluation based on Hausdorff Distance and Jaccard Index (RP stands for Random Perturbation; Gaussian stands for Gaussian Geomasking).

Method	Hausdorff Distance				Jaccard Index			
	Min	Max	Std	Mean	Min	Max	Std	Mean
RP	0.001	0.006	0.001	0.004	0.000	0.977	0.194	0.763
Gaussian	0.001	0.034	0.005	0.014	0.000	0.933	0.231	0.478
LSTM-TrajGAN	0.001	0.046	0.006	0.012	0.000	0.951	0.234	0.582

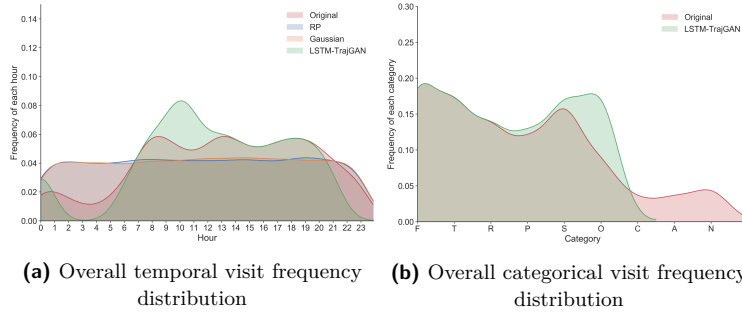
### Temporal Characteristics

We also explore the temporal characteristics based on the visualization of two summary indicators: temporal visit probability distribution for each POI category, and overall temporal visit frequency distribution. We count the frequencies of visits to each POI category at each hour in original trajectories and the synthetic trajectories using three different approaches, and convert them into probability distribution matrices (Figure 5), in which the temporal patterns and the temporal similarity can be analyzed and compared.

It shows that the temporal visit probability distribution from LSTM-TrajGAN shares a large commonality with that from the original data, which embodies a significant temporal similarity. Some parts of the result by LSTM-TrajGAN (i.e., categories C and E) have near zero visit probability since these categories rarely appear in training data and thus the model doesn't learn sufficient information to make intelligent predictions on them. As the comparisons, the temporal visit probability from Random Perturbation and Gaussian



**Figure 5** The hourly temporal visit probability distribution for each POI category by (A) Original data, (B) LSTM-TrajGAN, (C) Random Perturbation (within 24 hours), and (D) Gaussian Geomasking (within 24 hours) data (F: Food; T: Travel & Transport; R: Residence; P: Professional & Other Places; S: Shop & Service; O: Outdoors & Recreation; C: College & University; A: Arts & Entertainment; N: Nightlife Spot; E: Event).



**Figure 6** Overall temporal visit frequency distribution and overall categorical visit frequency distribution (RP stands for Random Perturbation; Gaussian stands for Gaussian Geomasking).

Geomasking show neither temporal similarity with the original data nor significant temporal patterns over 24 hours (except for the Event category).

Besides, we investigate the overall temporal and categorical visit frequency distribution (Figure 6a and Figure 6b). The overall temporal visit frequency distribution from our model can better fits the original data (Pearson Coefficient: 0.761) than Random Perturbation (0.536) and Gaussian Geomasking (0.535). The overall categorical visit frequency distribution also fits well (0.889). Hence, we conclude that our model generally well preserves both temporal and categorical characteristics.

## 4 Discussion

This section discusses the factors that may affect the privacy protection effectiveness of the LSTM-TrajGAN model, and the trade-off between the privacy protection effectiveness and the utility. Finally, we discuss the limitations of our approach.

### 4.1 Factors Affecting Privacy Protection Effectiveness

#### Training and Optimization Settings

We first explore how the different learning rates, loss metric functions, and random noise data affect the metric scores in the TUL task compared with the baseline setting (i.e., the

learning rate = 0.001; the spatial dimension = 64; and the TrajLoss metric function during training). As shown in Table 4, different random noise data have small influences on the metrics, which in fact contributes to the potential generalizability of the proposed approach for generating privacy-preserving trajectory data. We also found that the selection of the learning rate may have a great influence on the metrics. A higher learning rate (0.002) makes the model converge faster, generating the synthetic trajectories that have less uncertainty and share more characteristics with the original trajectories, leading to higher TUL metric scores and vice versa. Although this is not always the case, the learning rate should be carefully set to balance the trajectory utility and privacy protection effectiveness.

■ **Table 4** The metrics in the TUL task based on the synthetic trajectories by LSTM-TrajGAN using different training and optimization settings as well as different spatial embedding dimensions.

LSTM-TrajGAN	ACC@1	ACC@5	Macro-F1	Macro-P	Macro-R
Baseline	0.459	0.722	0.381	0.429	0.428
Different Random Noise	0.466	0.741	0.398	0.451	0.436
Higher Learning Rate (0.002)	0.841	0.959	0.824	0.855	0.828
Lower Learning Rate (0.00002)	0.055	0.157	0.029	0.047	0.054
Higher Spatial Dimensions (128)	0.510	0.811	0.504	0.513	0.513
Lower Spatial Dimensions (32)	0.426	0.703	0.396	0.402	0.392
TrajLoss without Spatial Loss	0.047	0.176	0.030	0.037	0.042
TrajLoss without Temporal Loss	0.093	0.252	0.076	0.119	0.089
TrajLoss without Categorical Loss	0.354	0.623	0.311	0.386	0.346
No TrajLoss	0.010	0.032	0.002	0.001	0.007

In addition, we also investigate how the TrajLoss metric function contributes to the training. When removing the Spatial Loss or the Temporal Loss from the TrajLoss function, the metric scores fall dramatically, implying that the synthetic trajectories fail to preserve the spatial or temporal characteristics of the original trajectories. By comparison, removing the Categorical Loss only has a limited impact on the metric scores. Not surprisingly, removing the whole TrajLoss function results in losing spatiotemporal characteristics and thus getting the lowest TUL metric scores. We conclude that the spatial and the temporal dimensions represent the essential characteristics of a trajectory and hence need to be taken into consideration explicitly in the privacy protection approaches.

### Spatial Embedding

Since the embedding of temporal attributes and categorical attributes is based on their vocabulary sizes, we mainly discuss the spatial embedding. The commonly used methods for spatial embedding are Multilayer Perceptron (MLP) and the Geohash algorithm. For example, Gupta et al. [8] use a MLP to embed the location of each person to obtain a fixed-length vector and use the vector as the input for an LSTM model to generate human trajectory. Petry et al. [21] introduce a binary Geohash algorithm, in which they first use the Geohash algorithm to divide the area into grid cells and then encode the latitude and longitude as a character string, and finally convert the string into a binary fixed-length vector as the representation for the spatial dimension of each trajectory point.

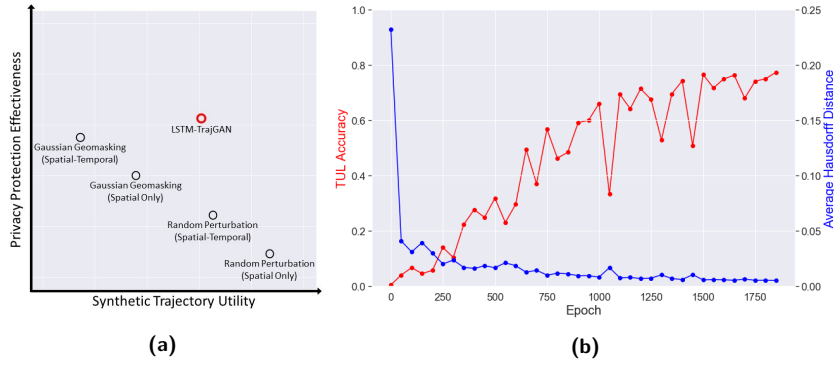
We use MLPs in the generator and the discriminator to embed the spatial dimension, but we implement them in a different way. Instead of directly embedding the coordinates, we first derive the deviations of latitudes and longitudes from the centroid of all trajectory locations, and then we embed these deviations into 64-dimensional vectors using MLP. There

are two considerations: (1) On the one hand, unlike the trajectory classification task in [21], our goal is to generate synthetic trajectories, which means we need to decode the coordinates out from the hidden features in the model, and therefore using binary Geohash may lead to difficulties in learning the valid representation of coordinates, in designing the proper spatial loss, and in back-propagating the errors; and (2) On the other hand, unlike the restricted prediction area described by a Cartesian coordinate system in [8], the prediction area in our task is on the city scale, and the difference between two GPS coordinates only appears after the decimal point. It would be a grand challenge for the model to learn and predict the coordinates with only subtle changes. As such, we standardized the coordinates to make the difference between two locations more significant for the model to learn. Recent studies also indicate that scattering the locations based on deviations may help preserve privacy [5].

We also explore how the spatial embedding dimensions affect the metrics in the TUL task. As is presented in Table 4 embedding the location information into a vector with higher dimensions (e.g., 128) improves the TUL metric scores and vice versa. This makes sense since vectors in a higher-dimensional space are usually able to extract and embed more information than that in a lower-dimensional space. However, this also involves a trade-off between location accuracy and computational effort due to the limitation of physical devices.

#### 4.2 The Trade-off between Privacy Protection Effectiveness and Utility

Generally speaking, specific trajectory analysis tasks may rely on different types of trajectory data (e.g., POI-based or road network-based) or different requirements (e.g., road extraction requires the location of each trajectory point to be precise), making it challenging to design a generic privacy protection method. However, we can evaluate a method by some specific criteria to determine its application scenarios, and even design a method based on this consideration to cover as many scenarios as possible. Inspired by the evaluation framework that involves the privacy, analytics, and uncertainty [5], we investigate the relationship between privacy protection effectiveness and utility. Figure 7a demonstrates the performance of each method. It is worth noting that the placement of each method is estimated from our experiment. We believe that the consideration of this relationship would help choose and design proper trajectory privacy protection methods for specific scenarios.



**Figure 7** (a) The performance of each method in privacy protection effectiveness and utility; (b) The trade-off between the effectiveness of privacy protection (presented by TUL Top-5 Accuracy) and the preservation of spatial characteristics preservation (presented by Average Hausdorff Distance)

Sometimes the relationship between the privacy protection effectiveness and the utility is somewhat contradictory: we hope that the synthetic data are less similar to the original data

to protect privacy while still preserve some similarities as good alternatives for spatiotemporal modeling or analyses. This may result in a “catch-22 situation”. As an end-to-end deep learning model, the LSTM-TrajGAN is able to monitor and quantify this relationship during training and help to find the best-balanced parameter settings. For example, as training progresses, the TUL accuracy (Top-5 Accuracy) increases while the Average Hausdorff Distance decreases (Figure 7b). Carefully selecting the model weight from different epochs based on this relationship could ensure that the synthetic trajectories preserve spatiotemporal characteristics to some extent while maintaining a low TUL accuracy as needed, thereby balancing the privacy protection effectiveness and the utility of synthetic trajectories.

### 4.3 Limitations

Several limitations exist in our current approach. First, compared to traditional geomasking techniques that blur the existing trajectories, our deep learning model that generates new trajectories leads to a much higher computational effort and also needs an additional training process before its deployment in applications. Second, we focus on the TUL task and analyzed spatial and temporal characteristics of the synthetic trajectories, which reflects their potential for privacy-preserving trajectory analysis, but more specific evaluations are not investigated yet. Third, our model generates only the synthetic trajectories that have the same length as the original trajectories. Finally, our model currently focuses on city-scale trajectories, and the deviation-based location representation may not be suitable for global-scale trajectories. These limitations will be further explored in our future work.

## 5 Conclusion and Future Work

This research proposes a novel LSTM-TrajGAN approach, i.e., a deep learning model that combines the LSTM recurrent neural network and the GAN structure to generate privacy-preserving synthetic trajectories for trajectory data publication. We utilize the idea of adversarial training in the model design, train our model on a Foursquare NYC weekly trajectory dataset, and evaluate its privacy protection effectiveness in the TUL task. To answer the two research questions we posed at the beginning of this research, the results show that (RQ1) our model can generate the spatial-temporal synthetic trajectories that prevent the trajectory creators (i.e., users) from being re-identified to certain degree and (RQ2) keep some spatial, temporal, and thematic characteristics of the original trajectories. Additionally, the results show that the model has the potentials for supporting further spatial or temporal analyses. Lastly, we explored the factors affecting the privacy protection effectiveness and discussed the trade-off between model effectiveness and utility in general. The design of a new loss function TrajLoss offers new insights into the development of spatially explicit artificial intelligence techniques for advancing GeoAI [13].

Our future work will focus on improving the trajectory similarity loss metric function, extending our framework to global-scale trajectory datasets, generating custom variable-length synthetic trajectory data, exploring potential privacy attack and defense strategies, and evaluating the privacy protection effectiveness and utility of our model in other trajectory data mining and analysis tasks.

---

### References

- 1 Marc P Armstrong, Gerard Rushton, Dale L Zimmerman, et al. Geographically masking health data to preserve confidentiality. *Statistics in Medicine*, 18(5):497–525, 1999.
- 2 Chi-Yin Chow and Mohamed F Mokbel. Trajectory privacy in location-based services and data publication. *ACM SIGKDD Explorations Newsletter*, 13(1):19–29, 2011.

- 3 Yves-Alexandre De Montjoye, César A Hidalgo, Michel Verleysen, and Vincent D Blondel. Unique in the crowd: The privacy bounds of human mobility. *Scientific reports*, 3:1376, 2013.
- 4 Qiang Gao, Fan Zhou, Kunpeng Zhang, Goce Trajcevski, Xucheng Luo, and Fengli Zhang. Identifying human mobility via trajectory embeddings. 17:1689–1695, 2017.
- 5 Song Gao, Jinmeng Rao, Xinyi Liu, Yuhao Kang, Qunying Huang, and Joseph App. Exploring the effectiveness of geomasking techniques for protecting the geoprivacy of twitter users. *Journal of Spatial Information Science*, 2019(19):105–129, 2019.
- 6 Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
- 7 Marco Gruteser and Dirk Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of the 1st international conference on Mobile systems, applications and services*, pages 31–42. ACM, 2003.
- 8 Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, 2018.
- 9 Kristen H Hampton, Molly K Fitch, William B Allshouse, Irene A Doherty, Dionne C Gesink, Peter A Leone, Marc L Serre, and William C Miller. Mapping health data: Improved privacy protection with donut method geomasking. *American Journal of Epidemiology*, 172(9):1062–1069, 2010.
- 10 Simon Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.
- 11 Qunying Huang and David WS Wong. Modeling and visualizing regular human mobility patterns with uncertainty: An example using twitter data. *Annals of the Association of American Geographers*, 105(6):1179–1197, 2015. doi:10.1080/00045608.2015.1081120.
- 12 Qunying Huang and Yu Xiao. Geographic situational awareness: Mining tweets for disaster preparedness, emergency response, impact, and recovery. *ISPRS International Journal of Geo-Information*, 4(3):1549–1568, 2015.
- 13 Krzysztof Janowicz, Song Gao, Grant McKenzie, Yingjie Hu, and Budhendra Bhaduri. GeoAI: Spatially explicit artificial intelligence techniques for geographic knowledge discovery and beyond, 2020.
- 14 Carsten Keßler and Grant McKenzie. A geoprivacy manifesto. *Transactions in GIS*, 22(1):3–19, 2018.
- 15 Mei-Po Kwan, Irene Casas, and Ben Schmitz. Protection of geoprivacy and accuracy of spatial information: How effective are geographical masks? *Cartographica: The International Journal for Geographic Information and Geovisualization*, 39(2):15–28, 2004.
- 16 Jae Hyun Lee, Adam W Davis, Seo Youn Yoon, and K. G. Goulias. Activity space estimation with longitudinal observations of social media data. *Transportation*, 43(6):955–977, 2016.
- 17 Mingxiao Li, Song Gao, Feng Lu, Huan Tong, and Hengcai Zhang. Dynamic estimation of individual exposure levels to air pollution using trajectories reconstructed from mobile phone data. *International journal of environmental research and public health*, 16(22):4522, 2019.
- 18 Xi Liu, Hanzhou Chen, and Clio Andris. trajGANs: Using generative adversarial networks for geo-privacy protection of trajectory data (vision paper). In *Location Privacy and Security Workshop 2018 in conjunction with GIScience '18*, pages 1–7, 2018.
- 19 Yu Liu, Xi Liu, Song Gao, Li Gong, Chaogui Kang, Ye Zhi, Guanghua Chi, and Li Shi. Social sensing: A new approach to understanding our socioeconomic environments. *Annals of the Association of American Geographers*, 105(3):512–530, 2015.
- 20 Gengchen Mai, Krzysztof Janowicz, Bo Yan, Rui Zhu, Ling Cai, and Ni Lao. Multi-scale representation learning for spatial feature distributions using grid cells. In *The Eighth International Conference on Learning Representations*. openreview, 2020.
- 21 Lucas May Petry, Camila Silva, Andrea Esuli, Chiara Renso, and Vania Bogorny. Marc: a robust method for multiple-aspect trajectory classification via space, time, and semantic embeddings. *International Journal of Geographical Information Science*, pages 1–23, 2020.

- 22 Mehmet Ercan Nergiz, Maurizio Atzori, and Yucel Saygin. Towards trajectory anonymization: a generalization-based approach. In *Proceedings of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS*, pages 52–61, 2008.
- 23 Ben Niu, Qinghua Li, Xiaoyan Zhu, Guohong Cao, and Hui Li. Achieving k-anonymity in privacy-aware location-based services. In *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*, pages 754–762. IEEE, 2014.
- 24 Balaji Palanisamy and Ling Liu. Mobimix: Protecting location privacy with mix-zones over road networks. In *2011 IEEE 27th International Conference on Data Engineering*, pages 494–505. IEEE, 2011.
- 25 Yoo Min Park and Mei-Po Kwan. Individual exposure estimates may be erroneous when spatiotemporal variability of air pollution and human mobility are ignored. *Health & place*, 43:85–94, 2017.
- 26 Dara E Seidl, Piotr Jankowski, and Ming-Hsiang Tsou. Privacy and spatial pattern preservation in masked GPS trajectory data. *International Journal of Geographical Information Science*, 30(4):785–800, 2016.
- 27 Zheyue Wang, Xinyue Ye, and Ming-Hsiang Tsou. Spatial, temporal, and content analysis of twitter for wildfire hazards. *Natural Hazards*, 83(1):523–540, 2016.
- 28 Fengli Xu, Zhen Tu, Yong Li, Pengyu Zhang, Xiaoming Fu, and Depeng Jin. Trajectory recovery from ash: User privacy is not preserved in aggregated mobility data. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1241–1250, 2017.
- 29 Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1):129–142, 2014.
- 30 Fan Zhou, Qiang Gao, Goce Trajcevski, Kumpeng Zhang, Ting Zhong, and Fengli Zhang. Trajectory-user linking via variational autoencoder. In *IJCAI*, pages 3212–3218, 2018.
- 31 Huaijie Zhu, Xiaochun Yang, Bin Wang, Leixia Wang, and Wang-Chien Lee. Private trajectory data publication for trajectory classification. In *International Conference on Web Information Systems and Applications*, pages 347–360. Springer, 2019.