Learning to Simulate Human Mobility

Jie Feng, Zeyu Yang, Fengli Xu, Haisu Yu, Mudan Wang, Yong Li [*]
Beijing National Research Center for Information Science and Technology,
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
liyong07@tsinghua.edu.cn

ABSTRACT

Realistic simulation of a massive amount of human mobility data is of great use in epidemic spreading modeling and related health policy-making. Existing solutions for mobility simulation can be classified into two categories: *model-based methods* and *model-free methods*, which are both limited in generating high-quality mobility data due to the complicated transitions and complex regularities in human mobility. To solve this problem, we propose a *model-free* generative adversarial framework, which effectively integrates the domain knowledge of human mobility regularity utilized in the *model-based methods*. In the proposed framework, we design a novel self-attention based sequential modeling network as the generator to capture the complicated temporal transitions in human mobility. To augment the learning power of the generator with the advantages of *model-based methods*, we design an attention-based region network to introduce the prior knowledge of urban structure to generate a meaningful trajectory. As for the discriminator, we design a mobility regularity-aware loss to distinguish the generated trajectory. Finally, we utilize the mobility regularities of spatial continuity and temporal periodicity to pre-train the generator and discriminator to further accelerate the learning procedure. Extensive experiments on two real-life mobility datasets demonstrate that our framework outperforms seven state-of-the-art baselines significantly in terms of improving the quality of simulated mobility data by 35%. Furthermore, in the simulated spreading of COVID-19, synthetic data from our framework reduces MAPE from 5% ∼ 10% (baseline performance) to 2%.

CCS CONCEPTS

• **Computing methodologies → Neural networks**; **Learning from demonstrations**; *Agent / discrete models*; • **Information systems → Spatial-temporal systems**; *Data mining*.

KEYWORDS

GAN; Mobility Simulation; Mobility Trajectory

ACM Reference Format:
Jie Feng, Zeyu Yang, Fengli Xu, Haisu Yu, Mudan Wang, Yong Li . 2020. Learning to Simulate Human Mobility. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, CA, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3394486.3412862

1 INTRODUCTION

Due to the outbreak of COVID-19 around the world since Jan. 2020, its spreading modeling becomes an emergent topic for health organizations and national government to support public health policy-making. To accurately model the spatial diffusion of COVID-19 [12, 23], the mobility trajectory of populace becomes a fundamental component when it determines the strength and speed of the spreading to a large extent. Although a massive amount of mobility data are generated and collected every day via smartphone, they are hard to be directly utilized in the practice due to the privacy issue and commercial concern. Moreover, the direct replay of collected data is also not enough for advanced modeling, e.g., counterfactual scenarios like what will happen if more people travel with public transportation. Thus, the realistic simulation of human movement to generate massive high-quality individual mobility trajectory becomes a valuable and important problem.

To capture the detailed spreading procedure of the epidemic, we need to simulate the fine-grained movement (e.g., whereabouts of individual in each hour of the day) of massive individuals in the daily life. From many years ago, this problem has been investigated by the researchers from transportation and physics fields and *model-based methods* [10, 11, 20, 30] are proposed with capturing the regularity of human mobility behaviors [7, 26]. These methods assume that human mobility can be modeled by limited parameters with explicit physical meaning and these parameters describe the key characteristics and patterns of human mobility, *e.g.*, the temporal periodicity, spatial continuity, etc. By extracting these patterns from real-world mobility data, researchers leverage the techniques of decision tree [11] and Markov process [30] to design various human mobility models. However, in practice, human mobility exhibits complex sequential transitions between locations, which can be time-dependent and high-order. Besides, other factors like irregular patterns and various exploration schemes are also important characteristics of individual mobility. Relying on the simplified assumptions of human mobility behaviors, *model-based methods* overlook these complicated human mobility patterns and thus fail to accurately model real-world mobility behaviors.

On the other hand, with the recent success of generative adversarial network (GAN) [8], researchers propose *model-free methods* to learn to simulate the human mobility behaviors from the real data [16, 24]. Different from the *model-based methods*, *model-free methods* give up explicitly extracting mobility patterns with physical meaning, and move the way to build a neural network-based generative model for directly learning to simulate human mobility behavior from the real-world mobility data. While these *model-free*

[*] Yong Li is the corresponding author.

KDD '20, August 23–27, 2020, Virtual Event, CA, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3412862

methods might have better performance since they do not rely on simplified assumptions of human mobility, the pure learning paradigm without utilizing the prior knowledge of human mobility patterns makes their learning procedure inefficient and ineffective in the practice. Besides, it is also hard for them to capture the hidden patterns of human mobility when learning with noisy and inaccurate raw data. In summary, while *model-based methods* succeed in modeling the human mobility patterns with simplified assumptions and *model-free methods* try to learn from data directly, both of these two types of existing methods fail to simulate the realistic human mobility effectively and efficiently. As a result, the epidemic models based on these unreal mobility simulation methods also fail to reproduce and predict the spreading of epidemic.

In this paper, we propose a novel framework to fundamentally combine the advantages of the *model-based methods* and *model-free methods* to achieve the high-quality simulation of human mobility. As mentioned before, human mobility simulation is challenging due to the complicated transitions and complex regularities. Based on the classic GAN framework, we propose specific designs in the generator, discriminator and the training procedure to solve these challenges. Concretely, we design a novel self-attention based sequential modeling network as the generator to capture the complex temporal transitions in human mobility including the long-term and time-dependent patterns. More importantly, we enhance the framework by introducing the prior knowledge of human mobility regularities considered in the *model-based methods*. First, we design an attention-based network in the generator to model the effects of the urban structure on shaping human mobility by considering the prior relations between locations from multi-view. Second, we introduce the mobility regularity-aware loss to inspire the discriminator to distinguish the mobility trajectory by considering the crucial mobility patterns including the temporal periodicity [18] and spatial continuity [7]. Finally, we propose to utilize the mobility regularities of spatial continuity and temporal periodicity to pre-train the generator and discriminator to further improve the learning efficiency and final performance.

Our contributions can be summarized as follows.

- We propose a novel generative adversarial framework [1], which combines the advantages of *model-free methods* and *model-based methods* to generate realistic and high-quality human mobility by directly learning to simulate from the real-world mobility data.
- We propose to utilize the prior knowledge of human mobility regularity to improve the learning efficiency and performance by introducing the urban structure modeling component in generator and mobility regularity-aware loss in discriminator. Besides, we also utilize the mobility regularities to design pre-train strategies to further improve the model efficiency and performance.
- We conduct extensive experiments on two real-life mobility datasets and demonstrate that the proposed framework outperforms all the state-of-the-art baselines significantly, *e.g.*, reducing the difference measured by four Jensen–Shannon divergence based metrics by more than 35%. Furthermore, we apply synthetic data in the COVID-19 spreading modeling and produce much closer simulation results to the real data by reducing MAPE from 5% ∼ 10% to 2%.

2 PROBLEM STATEMENT

Human mobility data can be defined as a spatial-temporal trajectory $\mathrm{S} = [x_1, x_2, \dots, x_n]$, where i_{th} element x_i is a spatial-temporal record defined as a tuple (l_i, t_i), l_i denotes the location identification, which can be GPS coordinates (lat, lon) or a region ID r_i. Besides, t_i is the timestamp of i_{th} visiting record. Based on the above notation, the mobility trajectory generation problem is defined as follows.

Definition 1 (Mobility Trajectory Generation). *Given a real-world mobility trajectory dataset, generate a realistic mobility trajectory $\hat{\mathrm{S}} = [\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n]$ with a θ-parameterized generative model G by the following formulation:*

$$p_{\boldsymbol{\theta}}(\hat{\mathrm{S}}) = \prod_{i=1}^{n} p_{\boldsymbol{\theta}}\left(\hat{x}_i | \hat{x}_1, \dots, \hat{x}_{i-1}\right),$$

where $p_{\boldsymbol{\theta}}$ denotes the generation distribution from generator G. The generation of trajectory $\hat{\mathrm{S}}$ via generator G is a sequential decision process, whose probability is described as the multiplication of the probability of each generated spatial-temporal point $\hat{x}_i$.

In terms of generative adversarial networks [8], it simultaneously trains a generator $G_{\boldsymbol{\theta}}$ and a discriminator $D_{\boldsymbol{\phi}}$ within a min-max game to learn to generate new data with the fed of real data. In a typical GAN framework, a generator is trained to fool the discriminator in distinguishing fake samples. Meanwhile, the discriminator is trained to distinguish the samples from real data distribution or fake data distribution from the generator. Following the original paper [8], the min-max game is optimized as follows,

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\phi}} \mathbb{E}_{\mathbf{x} \sim p_d} \left[\log D_{\boldsymbol{\phi}}(\mathbf{x})\right] + \mathbb{E}_{\mathbf{x} \sim G_{\boldsymbol{\theta}}} \left[\log\left(1 - D_{\boldsymbol{\phi}}(\mathbf{x})\right)\right],$$

where p_d denotes the real data distribution, $G_{\boldsymbol{\theta}}$ is the θ-parameterized generator, $D_{\boldsymbol{\phi}}$ is the ϕ-parameterized discriminator, $\mathbf{x}$ is the data sample. Following-up works propose various loss [1] and design specific structures [13, 31] for the generator and discriminator to improve the performance and stability of GAN for many generative tasks like image and text generation. However, due to the specific characteristics and regularities of human mobility, these existing methods cannot be directly applied in the mobility trajectory generation task.

3 METHODS

Figure 1 presents the proposed *MoveSim* framework, which includes a generator G for simulating the mobility behavior and a discriminator D for distinguishing the generated mobility trajectory from real mobility data.

3.1 Generator

As presented in Figure 1, the generator G consists of two components: *SeqNet* for directly generating the sequential transition among different locations and *RegNet* for modeling the effects of urban structure on human mobility which utilizes the prior knowledge of human mobility patterns.

3.1.1 SeqNet: Modeling the Temporal Transition of Mobility. To capture the complicated sequential transition regularities in mobility trajectories, we design a self-attention based sequential modeling

[1] Our codes: https://github.com/FIBLAB/MoveSim

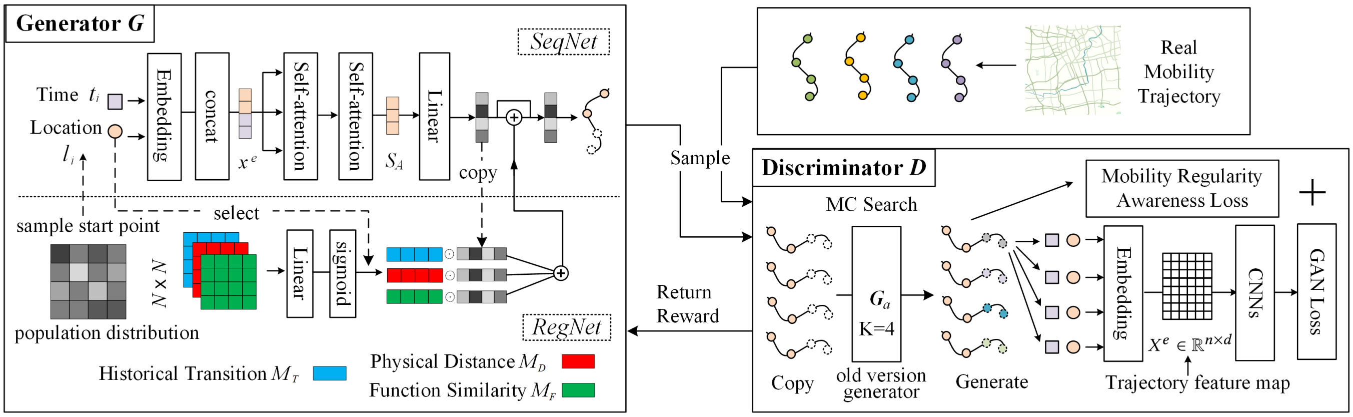


Figure 1: The *MoveSim* framework for human mobility simulation.

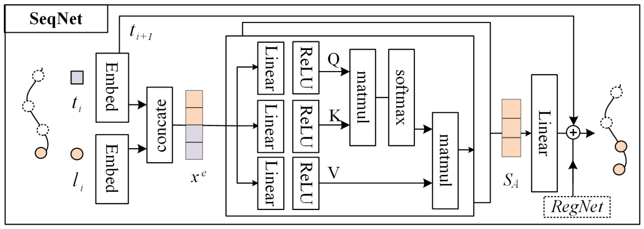


Figure 2: Illustration of the *SeqNet* architecture.

unit *SeqNet*. As the core component of the generator G, *SeqNet* generates the mobility trajectory sequence and captures the correlation in the sequence with above attention mechanism.

Figure 2 presents the architecture of *SeqNet*. The first spatial-temporal point is generated by a sampled location l_0 from the physical space based on the population density and a fixed start time $t_0 = 0$. Start from this point, *SeqNet* begins to sequentially generate spatial-temporal point to construct the mobility trajectory. We discrete space and time and encode them as one-hot vectors. Then, the location and time embedding modules convert and concatenate the raw input spatial-temporal point into a dense representation vector denoted as x^e,

$$x_i^e = tanh([W_l l_i; W_t t_i]),$$

where W_l and W_t are the learnable parameters of embedding table, x_i^e is the vector representation of i_{th} generated point.

Based on the spatial-temporal point representation x^e, we design a self-attention based network to generate the trajectory sequentially. While the self-attention mechanism [27] directly captures the correlation between each element in the sequence, it is able to better model the high order and long-term patterns in mobility behavior when compared with the widely used RNNs [5, 21, 29]. Besides, with the joint representation of the spatial-temporal point, *SeqNet* is also able to capture the time-dependent transitions. Here, we first project x_i^e with three independent non-linear operations into three vectors: query vector Q_i, key vector K_i, and value vector V_i. Then, we apply scaled dot-product attention on them to obtain the weighted sum of value vectors as the feature representation of the past trajectory. The formula is as follows,

$$Q = ReLU(X^e W_Q), K = ReLU(X^e W_K), V = ReLU(X^e W_V),$$

$$S_A = \text{softmax}\left(QK^T/\sqrt{d}\right)V,$$

where X^e is the batch set of x^e, S_A is the output of the self-attention layer, and W is the learnable weight. Furthermore, we use multi-head and stacking operations to model the relation from different subspaces to obtain a comprehensive representation of the past trajectory. Finally, we use a linear layer to process the feature and apply the soft-max function to obtain the probability distribution of the next location.

3.1.2 *RegNet: Capturing the Effects of Urban Structure.*

When individual moves around the city to generate the mobility trajectory, the attributes of location and relations between locations become the important factor influencing human mobility [2, 11]. While these patterns and relations can be left to the *SeqNet* to implicitly learn from the mobility data via the signal from the discriminator, experiments show that such pure learning method is hard to train in practice. Thus, we design a *RegNet* module to integrate the prior knowledge of the urban structure and model their effects on individual mobility effectively.

The structure of *RegNet* is presented in Figure 1. Based on the discrete partition of urban space, we build three $N \times N$ location relation matrix for N locations to represent the multi-view relation between locations. These three relation matrix describe the relation among locations from different dimensions: *physical distance* denotes the spatial distance of locations in free space, *function similarity* captures the effects of urban function, and *historical transition* is the direct evidence for human experience and preference. The details of the construction of these relation matrices are as follows,

- *Physical Distance Matrix (M_D)*: it is obtained by calculating the Euclidean distance between all location pairs.
- *Function Similarity Matrix (M_F)*: we first calculate the PoI category distribution of each location and then obtain the functional similarity between location pairs by calculating the correlation between their PoI distribution.
- *Historical Transition Matrix (M_H)*: it is the aggregation of all the transitions between locations in the real-world mobility data.

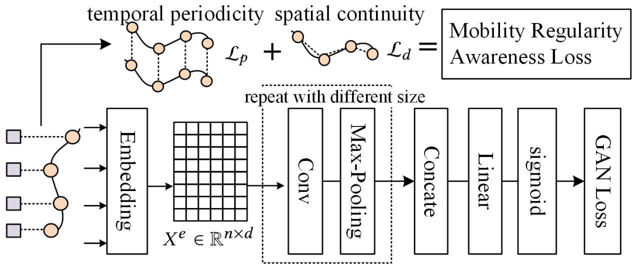


Figure 3: Illustration of the discriminator D.

All the relation matrices are calculated and normalized by the row. With these multi-view relation matrices, we utilize the intermediate output of *SeqNet* to select useful information from them and refine the next generated point respectively. To do this, we first use the probability vector $\hat{l}_{i-1}$ of last generated location l_{i-1} to "select" corresponding relation vector R from multi-view relation matrix $M_{D,F,H}$. Then, we utilize the element-wise multiplication operation to fuse the current generated location $\hat{l}_i$ feature vector with the multi-view relation vectors R. Finally, as a residual connection, we use this fused relation vector to refine the intermediate output $\hat{l}_i$ to generate the final result l_i. The formula is as follows,

$$[R_D; R_F; R_H] = \sigma(Linear([M_D; M_F; M_H])) * \hat{l}_{i-1},$$
$$l_i = softmax(\hat{l}_i + \hat{l}_i \odot R_D + \hat{l}_i \odot R_F + \hat{l}_i \odot R_H),$$

where $\hat{l}_{i-1}$ is the probability vector of the last location in the mobility trajectory, $\hat{l}_i$ is the probability vector for next generated location before the refining by *RegNet*, σ is the sigmoid function, R is the relation vector, $\odot$ is the element-wise multiplication.

3.2 Discriminator

With the real-world mobility trajectories and the synthetic trajectories from the generator G as input, we build a discriminator D to distinguish them and generate learning signals to guide the optimization of the generator G. Compared with the challenging task for generator G, the binary classification task for the discriminator D is much easier in most cases. To balance the training procedure for the efficient learning of the whole framework, we need to design a discriminator which is not so powerful and also computing efficiently. Otherwise, if the discriminator is too powerful, it is hard for the generator to learn progressively and efficiently. Thus, without using the complicated self-attention structure, we build a simple discriminator with convolution as a basic unit to extract features and distinguish input. The design of D is presented in Figure 3.

3.2.1 Sequential Discriminator. As Figure 3 shows, the CNN based discriminator consists of three components: 1) a spatial-temporal embedding module to convert raw trajectory sequence $S = [x_1, ..., x_n]$ into a 2D feature matrix $X^e \in \mathbb{R}^{n \times d}$; 2) several convolution layers to extract features from the feature matrix; 3) a linear layer with sigmoid activation function to produce the final score based on the flatten features from convolution layers. With the introduced CNN based design, our discriminator can generate learning signals on the complete trajectory from the real data or the generator G.

However, as discussed before, the trajectory sequence from generator is sequentially generated. While current discriminator can only produce quality signal for the entire trajectory, it cannot provide proper signal for the intermediate output of partial trajectory [31]. Without sufficient feedback for the intermediate generation result, it is hard for the generator to recognize where the error is generated and accumulated, and hence cannot be effectively improved and updated during training. Thus, as shown in the discriminator part of Figure 1, we use the Monte Carlo search (MC search) to solve this problem and generate sequential signals for the intermediate results from the generator. To do this, we first maintain an assistant generator G_a, which is the last step version of the current generator G, to complete the current partial trajectory by repeating K times with different random seed. Then, with these completed trajectories as the input, we require the discriminator to distinguish them and produce the loss. Finally, we average these losses and regard it as the quality signal for the original partial trajectory before completing.

3.2.2 Mobility Regularity-Aware Loss. While the standard discriminator generates learning signals for the generator by distinguishing the real and synthetic mobility trajectories, it fails to capture the regularity and constraints of the human mobility explicitly, which is the key point of high-quality mobility trajectory simulation. Human mobility shows a high degree of temporal and spatial regularity [7, 10, 26], such as the significant probability to return to few highly visited locations and explore to the nearby locations on the mobility trajectory. Such mechanisms have been extensively investigated in previous *model-based methods*. However, directly learning these mobility regularities from missing and noisy mobility data is not easy [19, 26], which is also the limitation of *model-free methods*. Thus, to help the model to explicitly capture the important mobility regularities from the noisy mobility trajectory data and improve the learning efficiency, we introduce the mobility regularity as the correction term of the discriminator's loss.

While the temporal periodicity and spatial continuity are the most essential human mobility regularity [7, 26], we construct mobility regularity-aware loss in the discriminator by considering both of them. On one hand, following the spatial continuity of human mobility, we define the distance aware loss $\mathcal{L}_d$ as the accumulated Euclidean distance of nearby transitions in the mobility trajectory. $\mathcal{L}_d$ works by encouraging the model to limit the travel distance between nearby mobility transitions. On the other hand, following the daily periodicity of human mobility, we define the periodicity loss $\mathcal{L}_p$ by calculating the Indicator distance of locations with fixed periodicity P. For example, with 1 hour as the basic time window, P is set as 24 which means that $\mathcal{L}_p$ is calculated as the number of different locations on different days at the same hour of the day. With the regularity of daily periodicity, people prefer to visit the same locations at the same time of different days which will also lead to the lower expected value of $\mathcal{L}_p$. The formulation of mobility regularity-aware loss of $\mathcal{L}_d$ and $\mathcal{L}_p$ is as follows,

$$\mathcal{L}_d = \sum\nolimits_{i=0,...,n-1} \sqrt{(l_i.x, l_{i+1}.x)^2 + (l_i.y, l_{i+1}.y)^2},$$
$$\mathcal{L}_p = \sum\nolimits_{i=0,...,n-1} D_I(l_i, l_{i+P}),\ D_I(l_1, l_2) = \begin{cases} 0, & \text{if } l_1 = l_2; \\ 1, & \text{if } l_1 \neq l_2, \end{cases}$$

where x and y denote the coordinates of location l in Mercator plane, D_I is the Indicator function, P is the discrete periodicity, and n is the length of the complete mobility trajectory.

3.3 Model Training

While the proposed framework owns a generator and a discriminator for min-max optimization, the standard training algorithm of GAN is not applicable due to the following two reasons: 1) the discrete output of generator with sampling operation on the probability distribution blocks the gradient back-propagation from discriminator [31]; 2) the complicated transitions and various noise in mobility trajectory sequence make it difficult to learn useful knowledge from raw mobility data efficiently.

3.3.1 *Reinforcement Learning based Training.*

For the first problem, while we can replace the discrete location with continuous representation like GPS coordinates to avoid the first problem of gradient back-propagation, this low dimension location representation will limit the capacity of mobility modeling and hard to achieve promising performance, which is testified in our experiment and past works [5, 29]. Thus, we follow the widely used technique in text generation [17, 31] to use the reinforcement learning technique to produce useful learning signals. In practice, we can regard the mobility trajectory generation procedure as a Markov decision-making process (MDP), where the agent is the generative model, the state is the generated partial mobility trajectory, the action is the next location, and reward is the loss from the discriminator D. With regarding the generative model as a stochastic parameterized policy, we can train the generative model with a policy gradient algorithm. We follow the REINFORCE algorithm [28] to generate the policy gradient by receiving the reward $R(\mathbf{x})$ from D,

$$\nabla_\theta = \nabla_\theta \mathbb{E}_{p_\theta(\mathbf{x})}[R(\mathbf{x})] = \mathbb{E}_{p_\theta(\mathbf{x})}\left[R(\mathbf{x})\nabla_\theta \log p_\theta(\mathbf{x})\right],$$

where θ is the parameter of generator G, $\mathbf{x}$ is the state (i.e., generated mobility trajectory), the reward $R(\mathbf{x})$ is the loss from the discriminator D. Based on the above gradient ∇_θ, parameters θ of generator G is updated by $\theta \leftarrow \theta + \alpha\nabla_\theta$, where α is the learning rate of generator G.

3.3.2 *Pre-training Mechanism.*

While we have designed specific structures in the generator and discriminator for the characteristics of human mobility trajectory including the physical regularities, the unsupervised learning paradigm, random noise in the real data and complicated characteristics of mobility trajectory make it not easy to obtain the promising performance when training the whole framework from scratch [8, 13, 31]. To accelerate the training procedure and improve the performance of the whole framework, we propose to pre-train it with human mobility modeling based tasks. In this way, we can make full use of the mobility data and enable the framework to preview the important regularity of human mobility before GAN training. Based on the regularity of human mobility, we design two pre-train tasks: mobility prediction task and mobility regularity-aware task.

For the generator, the most important task is to generate realistic location choices sequentially. The mobility prediction task can be regarded as the weak version of this sequential generation task, which only needs to predict the next location with knowing all the previous ground-truth trajectory. Follow the similar settings to previous works [5, 21], we utilize the mobility prediction task to pre-train the *SeqNet* in our generator G. To do this, we randomly divide the whole mobility trajectory data into a training set and testing set with one trajectory as an instance. For each trajectory with n points, we choose the first $n - 1$ points as the input and the last $n - 1$ points as the target. Furthermore, we design mobility regularity-aware task to pre-train the discriminator D. Mobility regularity-aware task is designed as a binary classification task to distinguish whether the input mobility trajectory exhibits important mobility regularities including the temporal periodicity and spatial continuity. To complete this classification task, we construct the fake mobility trajectory by destroying the mobility pattern in the real trajectory. We disturb the real trajectory in two ways: 1) random select one location from the real trajectory and replace it with one location in the whole physical space which is distant to the original location; 2) random disturb the order of locations in one periodicity of the trajectory. With the constructed fake trajectory and real trajectory, our discriminator is pre-trained to distinguish them with the binary classification loss.

4 EXPERIMENTS

4.1 Datasets

We use two large-scale real-world mobility datasets to evaluate the performance of our framework, whose basic statistics of them are in Table 2. Detailed information of two datasets is as follows.

- **Mobile Operator:** This dataset was collected in a major city by a major mobile network operator in China. It is a large-scale dataset including 100,000 mobile users with a duration of 1 week, between April 1st and 7th, 2016. It records the anonymous user ID, accessed base station, and timestamp of each accessing.
- **GeoLife-GPS [35]:** This GPS trajectory dataset was collected by the MSRA Geolife project with 182 users in a period of over five years (from April 2007 to August 2012). It contains 17,621 trajectories, where each trajectory is defined as a sequence of points including the timestamp, latitude, and longitude.

We split the whole dataset into three parts: a training set for training generative model, a validation set for finding the best parameters of models, and a testing set for the final evaluation on various metrics. For Mobile Operator dataset, we use the base station as the basic spatial unit and the partition of it is set as 1:1:1. For GeoLife-GPS dataset, we project GPS coordinates into the grids by containing up to 3 digits after the decimal point. Besides, due to the limited size of GeoLife-GPS dataset, the partition of it is set as 7:2:1. Finally, we set the basic time slot as half an hour of the day for the convenience and universality of modeling.

4.2 Baselines and Metrics

We compare the performance of our model with seven state-of-the-art baselines.

- **Markov [15]**: It regards all the visited locations of users as states and builds a transition matrix to capture the first-order transition probabilities between these locations.
- **DeepMove [5]**: This is the state-of-the-art method for mobility prediction, which combines neural attention with the recurrent network to capture the periodical patterns in mobility.

Table 1: Performance comparison of our model and baselines on two mobility datasets, where lower results are better. Bold denotes best(lowest) results and underline denotes the second-best results.

Dataset	**Mobile Operator**						**GeoLife-GPS**					
Metrics(JSD)	*Distance*	*Radius*	*Duration*	*DailyLoc*	*G-rank*	*I-rank*	*Distance*	*Radius*	*Duration*	*DailyLoc*	*G-rank*	*I-rank*
Markov	**0.0023**	0.0659	0.0239	0.4212	0.1301	0.0431	0.0176	0.1452	0.0746	0.2845	0.2502	0.0964
DeepMove	0.0497	0.0206	0.0206	0.2599	0.6172	0.1163	0.1654	0.1038	0.0356	0.2279	0.3463	0.0567
TimeGeo	0.0040	0.0105	0.0074	0.0891	0.3079	0.0125	0.0193	0.1124	0.0093	0.1235	0.2764	0.0495
IO-HMM	0.0027	0.0118	0.0085	**0.0724**	0.2715	0.0196	0.0158	0.0972	0.0096	0.0986	0.2867	0.0557
GAN	0.0146	0.0299	0.043	0.1534	0.2981	0.0653	0.0233	0.0946	0.0834	0.2645	0.2876	0.0457
TrajGAN	0.1152	0.2194	0.0921	0.2843	0.1028	0.1514	0.1326	0.1954	0.0692	0.3362	0.1742	0.1034
SeqGAN	0.0052	0.0158	0.0026	0.1193	0.0998	0.0129	0.0165	0.0757	0.0079	0.0846	0.0964	0.0242
Ours	0.0025	**0.0068**	**0.0014**	0.0844	**0.0501**	**0.0087**	**0.0088**	**0.0539**	**0.0018**	**0.0635**	**0.0833**	**0.0096**

Table 2: Statistic information of two mobility datasets.

Datasets	**Mobile Operator**	**GeoLife-GPS**
Duration	1 week	5 years
#Users	100,000	182
#Records/User	261	453
#Locations	9000 base stations	GPS coordinates

- **TimeGeo [11]**: It defines the weekly home-based tour number, dwell rate, burst rate to model the temporal choices and utilizes a r-EPR mechanism to model the spatial choices of human mobility.
- **IO-HMM [30]**: This method first annotates user activities from trajectory with IO-HMM and then generate sequences of mobility for each user with the manual assigned home and work.
- **GAN [8]**: Based on the general design of GAN, LSTM is selected as the generator and CNN as the discriminator.
- **TrajGAN [24]**: This method flattens and embeds a trajectory in the 2D matrix form and then uses a standard GAN to generate the matrix-form trajectory.
- **SeqGAN [31]**: It proposes to combine reinforcement learning with GAN to solve the discrete sequence generation problem, we directly apply this method to generate the location sequence.

Following the common practice in previous works [11, 24], we define 6 metrics to evaluate the quality of generated data by comparing the distribution of important mobility patterns between the simulated mobility trajectory and the real mobility trajectory.

- ***Distance***: travel distance, which is calculated as the cumulative travel distance of per user in the fixed time interval.
- ***Radius***: radius of gyration (r_g) [7], which represents the spatial range of user daily movement.
- ***Duration***: stay duration, which is calculated as the stay duration of per location visiting.
- ***DailyLoc***: daily visited locations, which is calculated as the number of visited locations per day for each user.
- ***G-rank***: the number of visits per location, which calculated as the visiting frequency of top-100 locations.
- ***I-rank***: an individual version of G-rank.

To get the quantitative results, we use Jensen–Shannon divergence (JSD) to measure the similarity between the mobility pattern distributions of generated trajectory and real trajectory data, which is defined as $JSD(p;q) = h\left((p+q)/2\right) - (h(p)+h(q))/2$, where h is the Shannon information, p and q are distributions.

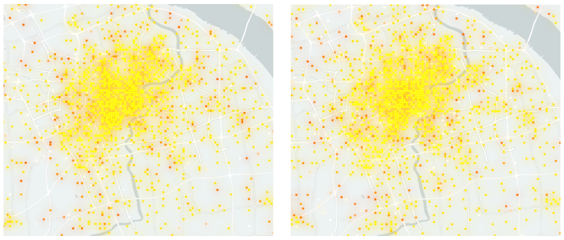
4.3 Performance Comparison

We compare our framework with 7 state-of-the-art baselines on two mobility datasets. To evaluate the quality of the generated mobility trajectory, we calculate its mobility pattern distributions on 6 metrics and compare them with the distribution from real data by JSD. The comparison results are presented in Table 1.

Performance on Mobile Operator Dataset: As Table 1 shows, *mobility prediction methods* (Markov and DeepMove) perform worst on the mobility generation task when both of them are trained with the short-term goal (next step prediction). While Markov performs well on single *Distance* metric, we find that it is achieved by only visiting very limited locations with higher error in *DailyLoc* metirc. With explicitly considering the mobility regularity in the parameters, *model-based methods* achieve the best results of baselines with ranking 1st on *DailyLoc* metric (IO-HMM) and ranking 2nd on two metrics (TimeGeo). For *model-free methods*, SeqGAN performs best with ranking 2nd on two metrics of *Duration* and *G-rank*, and other two GAN based methods perform worse when failing to model the specific characteristics of mobility trajectory with image format (TrajGAN) and GPS coordinates (GAN). Our proposed *model-free* and *model-based* combined framework achieves the best results with ranking 1st on 4 metrics and ranking 2nd on 2 metrics. For 4 ranking 1st metrics, compared with the best baseline, our method reduces the JSD with more than 35%. For 2 ranking 2nd metrics of *Distance* and *DailyLoc*, our method also obtains competitive performance with the best baseline.

Performance on GeoLife-GPS Dataset: The results of baselines on the GeoLife-GPS dataset are different from on the Mobile Operator dataset, which can be explained by the small data volume, dense sampling rate and short time window of GeoLife-GPS dataset. In the GeoLife-GPS dataset, mobility regularity is not so obvious and becomes less important in mobility modeling. Thus, *model-based methods*, TimeGeo and IO-HMM, which need to estimate accurate mobility regularity parameters from the raw data, becomes worse, and the *model-free methods* (especially SeqGAN) performs better. While the characteristics of mobility dataset changes, our model still performs best in most of the metrics and reduces the JSD by more than 14%.

We also present the global spatial population distribution of Mobile Operator dataset in Figure 4. In Figure 4, we aggregate the population from base stations into nearby grids and the grids with more population will be brighter with a yellow color. We can find



(a) Ground Truth Population@6 P.M. (b) Synthetic Population@6 P.M.

Figure 4: Spatial distribution of the aggregated population.

that the spatial distribution of our synthetic population is very similar to the ground truth population distribution from the real data. In summary, the above results on two mobility datasets demonstrate the superiority of our model on mobility simulation from different views. With explicitly modeling the mobility regularity in the GAN based framework design, our model achieves promising results on individual mobility simulation.

4.4 COVID-19 Spreading Simulation

We conduct a simulation experiment on the spreading of COVID-19 with SEIR model to testify the utility of synthetic mobility trajectory from Mobile Operator dataset. We follow recent work [12] to implement the spreading model of COVID-19 and detailed parameters of the simulation are presented in Table 3.

Table 3: Detailed parameters for COVID-19 simulation [12].

Parameters	Value	Parameters	Value
close contact ratio (c)	0.2	R_0	2.2
transmission period (T)	5.8 days	β	R_0/T
incubation period (IC)	5.2 days	a	1/IC
infection period (IF)	11 days	γ	1/IF

During each day of simulation, infected or exposed people meet r susceptible people while they occur in a same base station. We assume that the probability of any two people in a same base station coming into close contact is c. We calculate transmission probability β using the basic reproduction rate $R_0 = 2.2$ divided by the average days (5.8 days) from onset to first medical visit and isolation [12]. We estimate the daily probability of becoming infectious from exposed is a, which is the inverse of incubation period (5.2 days in [12]). The daily probability of becoming removed from infectious γ is calculated based on average infection period (11 days in [12]). The formulas of infection processes are as follows,

$$\frac{dS}{dt} = -rc\beta, \frac{dE}{dt} = rc\beta - aE, \frac{dI}{dt} = aE - \gamma I, \frac{dR}{dt} = \gamma I.$$

In the simulation experiment, we first initialize 50 people as exposed randomly and label their status individually. In each day, we record the user ID of susceptible people who visit a same base station with those exposed or infectious people according to their trajectories in each time window. Then, these susceptible people become exposed at probability β, weighted by probability of close contacts c. Meanwhile, we update their status labels from susceptible to exposed. For other status updating, we do the same thing in each day. Besides, because different trajectories caused by random initial exposed people will result in different transmissions, we run at least 10 simulations for each experiment.

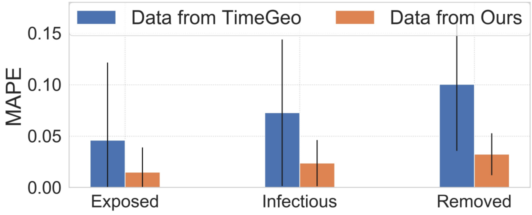


Figure 5: MAPE of the spreading modelling of COVID-19 with different synthetic data.

We conduct above experiments on different synthetic data and regard results from real Mobile Operator dataset as the ground truth. Then, we calculate MAPE of different synthetic data on the estimation of different kinds of population (E, I, R) and average results over 7 days are presented in Figure 5. As Figure 5 shows, synthetic data from our framework produce more similar spreading curves to the real data by reducing MAPE from 5% ~ 10% to 2%.

5 RELATED WORK

In the past decade, with detailed GPS records and large-scale data from the mobile network, the general characteristics, and regularities of human mobility have been well studied and deployed [7, 26, 34]. Furthermore, based on these mobility analyses, researchers designed different methods to predict the next location of individuals with knowing the movement history [5, 21, 29, 32]. Recently, RNN has been utilized to predict the next location by joint modeling the spatial-temporal transition [4–6, 21]. While these prediction models can be directly applied to generate the trajectory, they were trained with short-term goals (next step) and failed to generate high-quality long-term trajectory (multi-steps).

Generative adversarial network (GAN) [8] is a method for learning generative models by a min-max game. With its simple but effective design, GAN has been widely applied in many generative tasks like image generation [13], language generation [31], and time series [22]. GAIL [9] developed the imitation learning by replacing the implicit reward function with the GAN framework and has been successfully applied in driving behavior modeling [33], recommendation [25], and so on. Different from these works, we extend the GAN framework by designing several specific components for the unique characteristics and regularities of human mobility, which greatly improve the performance on simulating human mobility.

Existing works of mobility simulation can be classified into two groups: *model-based methods* and *model-free methods*. Based on the mobility regularity, *model-based methods* assume the individual mobility can be described by limited parameters with explicit physical meaning. Early works from transportation [3, 14] relied on the large-scale detailed user survey to calculate these parameters. Recently, researchers [10, 11, 30] proposed to estimate these parameters from the large-scale mobility data and then simulate individual mobility via Markov based models with simplified assumption of human mobility. While achieves promising performance in some cases, these methods with simplified assumptions cannot model the complex mobility in reality.

Recent works [16, 24] tried to apply GAN to generate individual mobility trajectory, which can be regarded as *model-free methods*. They do not assume any regularity of human mobility and just leave the model to learn from the data directly. With mapping the sequential trajectory into an image, Ouyang et al. [24] utilized the standard CNN based GAN to generate the virtual trajectory image. Kulkarni et al. [16] testified several natural language GANs on the short-term continuous trajectory generation task. With ignoring the important prior knowledge of human mobility, the learning efficiency and performance of these methods are limited. In our work, we introduce the pattern and prior knowledge of human mobility in the GAN based framework to achieve better performance on mobility simulation.

6 CONCLUSION

In this paper, we investigate the individual mobility simulation problem by proposing a novel generative adversarial based framework to simulate human mobility with explicitly modeling the prior knowledge and physical regularities. Following the main design philosophy of combing the advantage of model-based and model-free methods, our proposed human mobility simulation framework, which captures the complicated transitions and complex regularities of mobility, benefits the generation of high-quality mobility trajectory data. The extensive results on two real-life mobility datasets demonstrate that our framework outperforms seven state-of-the-art baselines significantly, and it also achieve better performance in the simulation of spreading of COVID-19. As future work, we will consider to model the underlying motivation of human mobility and extend the simulation to various applications.

7 ACKNOWLEDGMENTS

This work was supported in part by The National Key Research and Development Program of China under grant 2018YFB1800804, the National Nature Science Foundation of China under U1936217, 61971267, 61972223, 61941117, 61861136003, Beijing Natural Science Foundation under L182038, Beijing National Research Center for Information Science and Technology under 20031887521, and research fund of Tsinghua University - Tencent Joint Laboratory for Internet Innovation Technology.

REFERENCES

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

[2] M Batty. 2008. The size, scale, and shape of cities. *Science* 319, 5864 (2008).

[3] Jan Drchal, Michal Certický, and Michal Jakob. 2019. Data-driven activity scheduler for agent-based mobility models. *Transportation Research Part C-emerging Technologies* 98 (2019), 370–390.

[4] Jie Feng, Yong Li, Zeyu Yang, Qiang Qiu, and Depeng Jin. 2020. Predicting Human Mobility with Semantic Motivation via Multi-task Attentional Recurrent Networks. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2020).

[5] Jie Feng, Yong Li, Chao Zhang, Funing Sun, Fanchao Meng, Ang Guo, and Depeng Jin. 2018. DeepMove: Predicting Human Mobility with Attentional Recurrent Networks. In *Proceedings of the 2018 World Wide Web Conference.*

[6] Jie Feng, Can Rong, Funing Sun, Diansheng Guo, and Yan-Ping Li. 2020. PMF: A Privacy-preserving Human Mobility Prediction Framework via Federated Learning. *IMWUT* 4 (2020), 10:1–10:21.

[7] M. C. González, C. A. Hidalgo, and A. L. Barabási. 2008. Understanding individual human mobility patterns. *Nature* 453, 7196 (2008), 779.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems.* 2672–2680.

[9] Jonathan Ho and Stefano Ermon. 2016. Generative adversarial imitation learning. In *Advances in neural information processing systems.* 4565–4573.

[10] Sibren Isaacman, Richard Becker, Ramon Caceres, Margaret Martonosi, James Rowland, Alexander Varshavsky, and Walter Willinger. 2012. Human mobility modeling at metropolitan scales. (2012), 239–252.

[11] Shan Jiang, Yingxiang Yang, Siddharth Gupta, Daniele Veneziano, Shounak Athavale, and Marta C González. 2016. The TimeGeo modeling framework for urban mobility without travel surveys. *PNAS* 113, 37 (2016).

[12] Sheng jie Lai, Nick W. Ruktanonchai, Liangcai Zhou, Olivia Prosper, Wei Luo, Jessica R Floyd, Amy Wesolowski, Mauricio Santillana, Chi Zhang, Xiangjun Du, Hongjie Yu, and Andrew J Tatem. 2020. Effect of non-pharmaceutical interventions to contain COVID-19 in China. *Nature* (2020).

[13] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2017. Progressive growing of gans for improved quality, stability, and variation. (2017).

[14] Takehiro Kashiyama, Yanbo Pang, and Yoshihide Sekimoto. 2017. Open PFLOW: Creation and evaluation of an open dataset for typical people mass movement in urban areas. *Transportation Research Part C-emerging Technologies* 85 (2017).

[15] Marc Olivier Killijian. 2012. Next place prediction using mobility Markov chains. In *The Workshop on Measurement, Privacy, and Mobility.* 3.

[16] Vaibhav Kulkarni, Natasa Tagasovska, Thibault Vatter, and Benoit Garbinato. 2018. Generative Models for Simulating Mobility Trajectories. (2018).

[17] Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial Learning for Neural Dialogue Generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing.* 2157–2169.

[18] Zhenhui Li, Bolin Ding, Jiawei Han, Roland Kays, and Peter Nye. 2010. Mining periodic behaviors for moving objects. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining.* 1099–1108.

[19] Zhenhui Li, Jingjing Wang, and Jiawei Han. 2012. Mining event periodicity from incomplete observations. In *KDD.* 444–452.

[20] Ziheng Lin, Mogeng Yin, Sidney Feygin, Madeleine Sheehan, Jean-Francois Paiement, and Alexei Pozdnoukhov. 2017. Deep Generative Models of Urban Mobility. (2017).

[21] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the next location: A recurrent model with spatial and temporal contexts. In *AAAI.*

[22] Yonghong Luo, Xiangrui Cai, Ying Zhang, Jun Xu, et al. 2018. Multivariate time series imputation with generative adversarial networks. In *Advances in Neural Information Processing Systems.* 1596–1607.

[23] Benjamin F. Maier and Dirk Brockmann. 2020. Effective containment explains subexponential growth in recent confirmed COVID-19 cases in China. *Science (New York, N.y.)* (2020).

[24] Kun Ouyang, Reza Shokri, David S Rosenblum, and Wenzhuo Yang. 2018. A Non-Parametric Generative Model for Human Trajectories. In *IJCAI.* 3812–3817.

[25] Jing-Cheng Shi, Yang Yu, Qing Da, Shi-Yong Chen, and Anxiang Zeng. 2019. Virtual-Taobao: Virtualizing Real-world Online Retail Environment for Reinforcement Learning. In *AAAI.*

[26] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albertlaszlo Barabasi. 2010. Limits of Predictability in Human Mobility. *Science* 327, 5968 (2010), 1018–1021.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems.* 5998–6008.

[28] Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* 8, 3-4 (1992), 229–256.

[29] Hao Wu, Ziyang Chen, Weiwei Sun, Baihua Zheng, and Wei Wang. 2017. Modeling trajectories with recurrent neural networks. IJCAI.

[30] Mogeng Yin, Madeleine Sheehan, Sidney Feygin, Jean-François Paiement, and Alexei Pozdnoukhov. 2017. A generative model of urban activities from cellular data. *IEEE Transactions on Intelligent Transportation Systems* 19, 6 (2017).

[31] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. SeqGAN: Sequence generative adversarial nets with policy gradient. In *AAAI.*

[32] Chao Zhang, Keyang Zhang, Quan Yuan, Luming Zhang, Tim Hanratty, and Jiawei Han. 2016. GMove: Group-Level Mobility Modeling Using Geo-Tagged Social Media. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD).* 1305–1314.

[33] Guanjie Zheng, Hanyang Liu, Kai Xu, and Zhenhui Li. 2020. Learning to Simulate Vehicle Trajectories from Demonstrations. *2020 IEEE 36th International Conference on Data Engineering (ICDE)* (2020), 1822–1825.

[34] Yu Zheng, Quannan Li, Yukun Chen, Xing Xie, and Wei-Ying Ma. 2008. Understanding mobility based on GPS data. In *Proceedings of the 10th international conference on Ubiquitous computing.* ACM, 312–321.

[35] Yu Zheng, Xing Xie, Wei-Ying Ma, et al. 2010. Geolife: A collaborative social networking service among user, location and trajectory. *IEEE Data(base) Engineering Bulletin* 33, 2 (2010), 32–39.