SIMULATING CONTINUOUS-TIME HUMAN MOBILITY TRAJECTORIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent pandemic events have greatly emphasized the need to understand how humans navigate in modern day cities for both urban planning and recommending effective public health policy. In this paper, we propose a two-stage generative model, *DeltaGAN*, to simulate realistic human mobility trajectories. Compared with existing work where time was discretized, *DeltaGAN* generates continuous visitation time which better captures temporal irregularity in human mobility behaviors. Conditioned on the generated time, *DeltaGAN* produces accurate locations via limiting the range of realizable candidates. Experimental results demonstrate our model achieves competitive distributional similarity to real-world GPS trajectories across 6 individual trajectory and geographical metrics. We further validate *DeltaGAN*'s utility on COVID-19 spread simulation and observe the diffusion process under generated trajectories is consistent with that under real data.

1 INTRODUCTION

Understanding how humans navigate in modern day cities is critical for urban planning and optimizing location-based services (Asgari et al., 2013), such as traffic congestions (Song et al., 2016a; Calabrese et al., 2010), disaster management (Aschenbruck et al., 2004; Song et al., 2016b), network support (Lee et al., 2009; Rhee et al., 2011), and epidemic modeling (Feng et al., 2020). However, it is often difficult to gain access to large-scale city-wise mobility trajectory data of high quality in practice due to privacy concerns and limited availability (Feng et al., 2020). Therefore, learning to simulate realistic mobile trajectories has become a major subject of many recent research efforts to better understand human mobility behaviors.

Based on highly simplified assumptions of human mobility patterns, previous work treated individuals' mobility behaviors as Markov chains, and calculated transitional probabilities for location generation (Song et al., 2004; Shokri et al., 2011). Motivated by the success of generative models in computer vision tasks (Goodfellow et al., 2014), recent work proposed a standard CNN-based GAN to generate trajectory images (Ouyang et al., 2018) or leveraged Reinforcement Learning algorithms to generate discrete locations (Feng et al., 2020). However, the majority of existing work (see A.1) addressed the trajectory generation problem by discretizing the temporal space with fixed-sequence generation for tractability consideration. Compared with fine-grained signals from GPS data, such binning approaches learn from coarse signals and often produce less faithful trajectories.

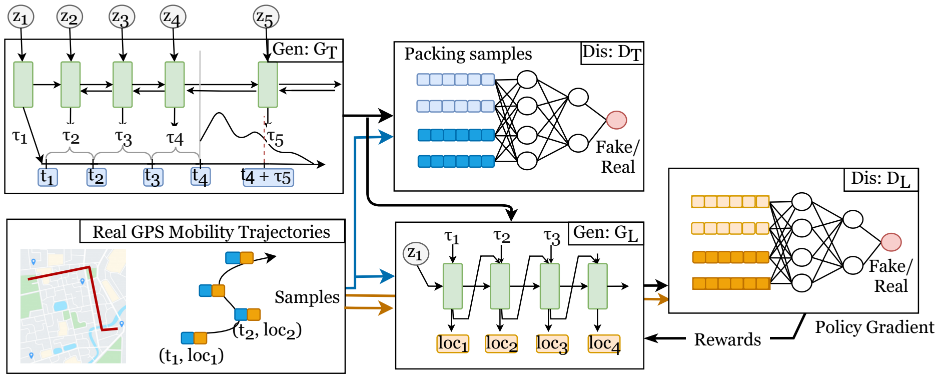
To simulate realistic mobility trajectories, we propose a deep generative model called *DeltaGAN*, which factorizes the trajectory generation problem into continuous-time and time-conditioned location generation. To focus on the most informative moments in the spatiotemporal sequence (Pertsch et al., 2020), we view each trajectory as a sequence of *person-entering-location* events and employ *Wasserstein GAN-GP* (Gulrajani et al., 2017) to generate travel time (or stay duration). The times are further utilized to train the location generator, which helps encode realistic mobility range in location modeling. Our contributions are as follows:

• **End-to-end Generation**: We propose a novel two-stage generative model *DeltaGAN* to simulate mobility trajectories with continuous-time and time-conditioned location generation.

• **Distribution Similarity**: Compared with existing deep predictive and generative approaches, *DeltaGAN* achieves consistently good performance with high distribution similarity to real-world GPS trajectories both in temporal and spatial aspects across 6 trajectory and geographical metrics.

• **Application Utility**: We further validate DeltaGAN's utility on a COVID-19 spread simulation and observe the diffusion process under generated trajectories is consistent with that under real data.

Figure 1: The proposed *DeltaGAN* includes a continuous-time generator G_T and a time-conditional location generator G_L with Gaussian noise z, together with their discriminators D_T and D_L.



2 Methodology

Figure 1 presents the proposed *DeltaGAN* architecture, which includes a continuous-time generator G_T and a time-conditional location generator G_L along with their discriminators D_T and D_L.

2.1 Setting

Human mobility data contains spatial-temporal trajectories $S = [x_1, x_2, \ldots, x_N]$ where each x_i is a tuple (t_i, l_i) representing a visiting record, with t_i $(0 \leq t_i < 24)$ denoting the i^{th} timestamp and l_i denoting the location $(\texttt{lat}, \texttt{long})$ of the record. Since it is often intractable to model the joint distribution $\mathbb{P}(S)$, especially for long sequences with large N, we made the common simplifying assumptions to factorize the joint probability, $\mathbb{P}(S) = \mathbb{P}(x_1)\prod_{t=2}^{N}\mathbb{P}(x_t|x_{1:t-1})$, treating the modelling approach as a sequential process. Following recent work (Feng et al., 2020), we discretize GPS coordinates into an $M \times M$ grid $\mathcal{L}$ containing up to 3 digits after decimal point of coordinates.

2.2 Continuous-time Generation

We view a mobility trajectory as a spatial-temporal point process with each event denoting a person entering a new location. Instead of binning timestamps into large discretized time slots, trajectories are viewed as sequences of events happening at irregular intervals, which allow us to generate fine-grained continuous trajectories unlike previous approaches. Formally, a temporal point process (TPP)[1] is a random process whose realizations consist of a sequence of strictly increasing arrival times $T = [t_1, ..., t_N]$, which can be equivalently represented as a sequence of strictly positive inter-event times $\tau_i = t_i - t_{i-1} \in \mathbb{R}_+$. The conditional intensity function $\lambda(t|H_{t_i})$ models the dependency of the next arrival time t on the history $H_{t_i} = \{t_j \in T | j < i\}$. By integrating, the conditional probability density function of the time τ_i until the next event, $\mathbb{P}(\tau_i|H_{t_i}) = \lambda^*(t_{i-1} + \tau_i)\exp(-\int_0^{\tau_i}\lambda^*(t_{i-1}+s)ds)$, $(*)$ denotes the dependence on H_{t_i} (Daley & Vere-Jones, 2007).

Our intensity-free approach leverages GANs to directly model $\mathbb{P}(\tau_i|H_{t_i})$, inspired by (Shchur et al., 2019). Instead of generating the continuous sequence T, generating the isomorphic sequence τ_i's let us explicitly enforce the monotonically increasing properties of the trajectory sequence. We can retrieve the generated continuous time sequence by taking the cumulative sum of τ. Since each trajectory represents a person's daily movements, τ_1 in the sequence denotes the starting time of the trajectory, where $\tau_1 = t_1 - t_0$, and t_0 is set to be 12:00 AM. This models the mobility pattern of when a person typically starts their day. We leverage the off-the-shelf implementation of WGAN-GP with a recurrent generator G_T and an MLP discriminator D_T. G_T takes a sequence of random variables $z = [z_1, \ldots, z_N], z_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and sequentially generates the sequence of duration $[\tau_1, \ldots, \tau_N]$ using a bidirectional LSTM. Importantly, we note an additional benefit of modelling the τ's, which allows for a natural interpolation between real and generated samples for gradient penalty. A toy example of the linear α interpolation between $\tau^{real} = [1.0, 0.0, 0.0]$ and $\tau^{gen} = [11.0, 0.5, 1.5]$ gives a meaningful sequence $\tau^{\alpha=0.5} = [6.0, 0.25, 0.75]$ with an "in-the-middle" starting time 6:00AM and shorter stay durations (15 mins, 45 mins). This drastically helps with providing stable gradients during training. We also leverage discriminator packing from

[1]For simplicity, we now consider only the time dimension, which can be easily factored out in most spatial-temporal point process formulation.

Table 1: Distribution comparison between real and generated mobility data. For all the metrics, lower values indicate more realistic trajectories. We marked the best result with boldface.

		Individual Trajectory Metrics				Geographical Metrics	
	Models	*Distance*	*Radius*	*Duration*	*DailyLoc*	$P(r)$	$P(r,t)$
Markov	First-order MC	0.56113	*0.10059*	0.58858	0.37374	0.43219	0.81836
	HMM	0.45217	0.52043	*0.10166*	0.39246	0.38329	0.82717
	IO-HMM	0.30730	0.15118	0.72849	0.66639	0.60712	0.82690
Deep Prediction Models	GRUPred	0.11441	0.17767	0.25546	0.55544	0.48476	0.82401
	TransDecoder	**0.09735**	0.16273	0.28388	0.56912	0.51261	0.82423
	TransAutoencoder	0.16209	0.22480	0.22952	0.54911	0.47934	0.82441
Deep Generative Models	GRU-VAE	0.82830	0.57407	0.15602	0.71901	0.58838	0.82190
	TransVAE	0.83198	0.67098	0.20954	0.62373	0.51397	0.82079
	TrajGAN	0.82075	0.72006	0.16102	0.42136	0.47586	**0.79298**
	ARAE	0.67968	0.57447	0.60294	0.44594	0.50957	0.82129
	SeqGAN	0.11074	0.16360	0.27096	0.57523	0.57125	0.82806
	DeltaGAN (Ours)	0.10553	**0.06677**	**0.00561**	**0.35276**	**0.30523**	0.80262

PacGAN (Lin et al., 2018) for D_T in addition to gradient penalty to further help reduce mode collapse, which can occur when the model focuses on generating very realistic but short trajectories.

2.3 Conditional Spatial Generation

Since generating discrete locations breaks the gradient propagation from the discriminator to the generator, we follow widely used techniques in text generation (Yu et al., 2017) to bypass the differentiation problem via gradient policy updates. Formally, the generation procedure is viewed as a Markov Decision Process (MDP), where the agent is a generative model G_L that produces the locations $L = [l_1, \ldots, l_N]$, $l_t \in \mathcal{L}$. At time step t, the state is the partial trajectory $L_{t-1} = [l_1, \ldots, l_{t-1}]$, the action is the next location l_t, and reward is the loss from the discriminator D_L. We additionally condition the stochastic policy G_L on the generated duration d_t to get $G_L(l_t|L_{1:t-1}, d_t)$. We train G_L parameterized by θ via policy gradient with the gradient of the expected end reward R_N,

$$\nabla_\theta \mathbb{E}\big[R_N|l_0\big] = \sum_{t=1}^{N} \mathbb{E}_{l_t \sim G_L(l_t|L_{t-1},d_t)}\big[Q^{D_L}(L_{t-1}, l_t)\nabla_\theta \log P_\theta^{G_L}(l_t|L_{t-1}, d_t)\big]$$

where the expected cumulative reward Q^{D_L} is the estimated probability of being real or fake by the discriminator, $Q^{D_L}(s = L_{T-1}, a = l_T) = D_L(L_T)$. D_L is a recurrent network and $P_\theta^{G_L}$ is the probability of selecting the next location given the history and stay duration. Based on the above gradient ∇_θ, the generator G_L is updated by $\theta \leftarrow \theta + \alpha\nabla_\theta$, where α is the learning rate.

2.4 Model Training

For stable learning, we perform a two-stage training pipeline, which includes a pre-training step for both the continuous-time generator G_T and the conditional location generator G_L followed by an iterative training step between G_T and G_L. Using real time samples from the mobility trajectories, we pre-train the time generator G_T and its discriminator D_T using the WGAN-GP loss. For the location generator G_L, we pre-train the network using maximum likelihood estimation (MLE) with the trajectories' stay durations and locations. In the iterative step, we alternate between training the pair G_T/D_T using WGAN-GP and the pair G_L/D_L conditioned on time using policy gradients.

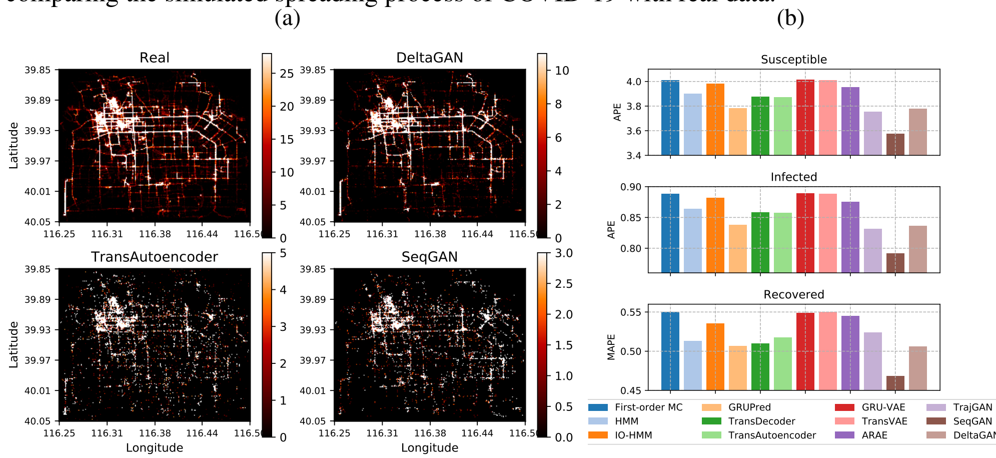
3 Evaluations

Dataset. We utilize the GPS trajectory dataset collected by MSRA Geolife project from 182 users in a period of over five years (Zheng et al., 2010). To generate city-wise mobility data, we keep the trajectories within the 5-th Ring Road of Beijing ($50,652$ grids covering $39.85°$N$\sim$$40.0°$N, $116.25°$E$\sim$$116.5°$E) and reduce the highly frequent sampling rate by considering records only when the person enters the location for the first time. There are $11,375$ trajectories with 31.531 records on average, and the average daily traveling duration and distance are 1.945 hours and 9.028 km.

Compared Approaches. We compare the performance of state-of-the-art baselines from *Markov*, *Deep Prediction Models* and *Deep Generative Models*. Details of each method are in Section A.2.

Evaluation Metrics Following the common practice in existing work (Ouyang et al., 2018; Feng et al., 2020), we adopt the following individual trajectory and geographical metrics to evaluate the distribution similarity (Jensen-Shannon divergence) between real and generated mobility data: 1)

Figure 2: (a) Geographical visualization of $10,000$ real and generated trajectories. We set the first 95% percentile of visit times as the colormap range for clarity. (b) Utility of generated mobility data by comparing the simulated spreading process of COVID-19 with real data.



Distance: the daily cumulative travel distance per trajectory; 2) *Radius*: the radius of gyration for a daily trajectory; 3) *Duration*: the total stay duration of each visited location; 4) *DailyLoc*: the number of unique locations in the daily trajectory; 5) *P(r)*: the visiting probability of one location r; 6) *P(r,t)*: the visiting probability of one location r at time t.

3.1 Distribution Similarity: Main Results

We list the performance of all generative methods in Table 1. With much lower distribution discrepancy over both individual trajectory and geographical metrics, the proposed *DeltaGAN* is able to generate more consistent human mobility data with the real ones both in spatial and temporal aspects. Focusing on generating continuous time as the first step, *DeltaGAN* can better capture the *Duration* for a person to stay in one location compared with other deep learning approaches, where event time is not dedicatedly learned and generated. Conditioned on the generated continuous event time, the location generator of *DeltaGAN* is capable of reducing the action space implicitly than deep learning approaches such as *GRUPred* and *SeqGAN*. Considering location visitation probability *P(r)* and *P(r,t)*, we also observe consistent location popularity from real and *DeltaGAN*.

In Fig. 2a, we show the qualitative performance of generative models by visualizing the distribution and popularity of visited locations. Compared with the real and other generative approaches, *Delta* successfully recognizes the relative popularity of different places in the city, e.g., main ring roads (bright horizontal and vertical lines), highways (bright lines spreading out in in the four corners), and the most popular Haidian District (brightest area in the northwest). Compared with *TransAutoencoder* and *SeqGAN*, we also notice that popularity intensity per location in synthetic data from *Delta* is much closer to the real case (color bars of different ranges in Fig. 2a, see Sec A.3 for details).

3.2 Application Utility: COVID-19 Spreading Simulation

We analyze the utility of generated mobility data in studying the spreading of COVID-19 with SIR model. We follow the recent work (Zeighami et al., 2020; Rambhatla et al., 2020) for epidemic diffusion simulation in 7 days: 1) $1,5000$ individuals start as either Susceptible (S) with probability 0.9 or Infected (I) with probability 0.1; 2) When an S individual u goes within 0.1 meter of an Infected and Spreading (IS) for at least 1 hour, then u immediately becomes Infected and Not Spreading (INS) with probability 0.5 at time t; 3) At time $t + t_{IS}$ with $t_{IS} \sim N(5, 10)$ in day, u becomes IS; 4) At time $t + t_R$ with $t_R \sim N(12, 24)$ in day, u becomes Isolated or Recovered (R). We run simulations with human mobility data, and calculate (mean) absolute percentage error between real and generated data on the number of different populations (S, I, R): the number of S or I individuals at the end of the 7-th day and the daily number of R individuals from the 7-th day till the day when all infected individuals become recovered. As shown in Fig. 2b, the proposed *DeltaGAN* model benefits COVID-19 Spreading study with small divergence in population distribution.

4 Conclusion

We propose the novel generative model $DeltaGAN$ to synthesize continuous-time mobile trajectories to better understand human mobility behaviors. By viewing human trajectories as sequences of events, *DeltaGAN* can generate realistic trajectories without discretizing visitation times and learn more accurate mobility dynamics, which is reflected in our evaluation and diffusion simulation.

REFERENCES

Nils Aschenbruck, Matthias Frank, Peter Martini, and Jens Tolle. Human mobility in manet disaster area simulation-a realistic approach. In *29th Annual IEEE International Conference on Local Computer Networks*, pp. 668–675. IEEE, 2004.

Fereshteh Asgari, Vincent Gauthier, and Monique Becker. A survey on human mobility and its applications. *arXiv preprint arXiv:1307.0814*, 2013.

Francesco Calabrese, Giusy Di Lorenzo, and Carlo Ratti. Human mobility prediction based on individual and collective geographical preferences. In *13th international IEEE conference on intelligent transportation systems*, pp. 312–317. IEEE, 2010.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

Daryl J Daley and David Vere-Jones. *An introduction to the theory of point processes: volume II: general theory and structure*. Springer Science & Business Media, 2007.

Jie Feng, Zeyu Yang, Fengli Xu, Haisu Yu, Mudan Wang, and Yong Li. Learning to simulate human mobility. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3426–3433, 2020.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

John Krumm and Eric Horvitz. Locadio: Inferring motion and location from wi-fi signal strengths. In *mobiquitous*, pp. 4–13, 2004.

Kyunghan Lee, Seongik Hong, Seong Joon Kim, Injong Rhee, and Song Chong. Slaw: A new mobility model for human walks. In *IEEE INFOCOM 2009*, pp. 855–863. IEEE, 2009.

Zinan Lin, Ashish Khetan, Giulia Fanti, and Sewoong Oh. Pacgan: The power of two samples in generative adversarial networks. *Advances in neural information processing systems*, 2018.

Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198*, 2018.

Kun Ouyang, Reza Shokri, David S Rosenblum, and Wenzhuo Yang. A non-parametric generative model for human trajectories. In *IJCAI*, pp. 3812–3817, 2018.

Karl Pertsch, Oleh Rybkin, Jingyun Yang, Shenghao Zhou, Konstantinos Derpanis, Kostas Daniilidis, Joseph Lim, and Andrew Jaegle. Keyframing the future: Keyframe discovery for visual prediction and planning. In *Learning for Dynamics and Control*, pp. 969–979. PMLR, 2020.

Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.

Sirisha Rambhatla, Sepanta Zeighami, Kameron Shahabi, Cyrus Shahabi, and Yan Liu. Towards accurate spatiotemporal covid-19 risk scores using high resolution real-world mobility data. *arXiv preprint arXiv:2012.07283*, 2020.

Injong Rhee, Minsu Shin, Seongik Hong, Kyunghan Lee, Seong Joon Kim, and Song Chong. On the levy-walk nature of human mobility. *IEEE/ACM transactions on networking*, 19(3):630–643, 2011.

Oleksandr Shchur, Marin Biloš, and Stephan Günnemann. Intensity-free learning of temporal point processes. *arXiv preprint arXiv:1909.12127*, 2019.

Reza Shokri, George Theodorakopoulos, Jean-Yves Le Boudec, and Jean-Pierre Hubaux. Quantifying location privacy. In *2011 IEEE symposium on security and privacy*, pp. 247–262. IEEE, 2011.

Libo Song, David Kotz, Ravi Jain, and Xiaoning He. Evaluating location predictors with extensive wi-fi mobility data. In *Ieee Infocom 2004*, volume 2, pp. 1414–1424. IEEE, 2004.

Xuan Song, Hiroshi Kanasugi, and Ryosuke Shibasaki. Deeptransport: Prediction and simulation of human mobility and transportation mode at a citywide level. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pp. 2618–2624, 2016a.

Xuan Song, Quanshi Zhang, Yoshihide Sekimoto, Ryosuke Shibasaki, Nicholas Jing Yuan, and Xing Xie. Prediction and simulation of human mobility following natural disasters. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(2):1–23, 2016b.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

Mogeng Yin, Madeleine Sheehan, Sidney Feygin, Jean-François Paiement, and Alexei Pozdnoukhov. A generative model of urban activities from cellular data. *IEEE Transactions on Intelligent Transportation Systems*, 19(6):1682–1696, 2017.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

Sepanta Zeighami, Cyrus Shahabi, and John Krumm. Estimating spread of contact-based contagions in a population through sub-sampling. *arXiv preprint arXiv:2012.06987*, 2020.

Junbo Zhao, Yoon Kim, Kelly Zhang, Alexander Rush, and Yann LeCun. Adversarially regularized autoencoders. In *International conference on machine learning*, pp. 5902–5911. PMLR, 2018.

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

A Appendix

A.1 Related Work

Earlier literature treats an individual's movement behavior as a Markov chain, where the probability of visiting one location at the next time step only depends only on the current location. For example, *First-order MC* (Song et al., 2004) generated trajectories by calculating the transition matrix between locations and *Time-dependent MC* (Shokri et al., 2011) went one step further by constructing separate matrices for different periods in a day. General Hidden Markov Models *(HMM)* (Krumm & Horvitz, 2004) and its variant *IO-HMM* (Yin et al., 2017) were also leveraged to generate human mobility data by assuming another unobservable process whose behavior depends on individuals' movement. Motivated the success of deep generative models in computer vision tasks, recent work proposed to develop generative adversarial networks (*GANs*) to generate synthetic trajectories which were indistinguishable from real ones by a discriminator. For instance, *TrajGAN* (Ouyang et al., 2018) mapped trajectories into 2D images and leveraged standard CNN-based GANs to generate virtual trajectory images. (Feng et al., 2020) treated human mobility as a partially observable Markov Decision Process (POMDP) and built upon *SeqGAN* (Yu et al., 2017) — a Reinforcement Learning approach to generate sequences of visited locations. However, the majority of existing work split trajectories with a coarse-grained time interval and then treat the mobility synthesization as a time series generation task. In contrast, we propose *DeltaGAN* to better capture the underlying dynamics of human mobility by generating continuous-time human mobility trajectory via inter-event durations.

A.2 Compared Approach Details

All of the baselines learn from and generate discrete sequences per half an hour except *IO-HMM* and *TrajGAN*:

•**Markov**: 1) *First-order MC* (Song et al., 2004): It defines the state as the visited location and assumes the next location only depends on the current one, so that a transition matrix is constructed to capture the first-order transition probability among locations; 2) *HMM* (Krumm & Horvitz, 2004): it sets up with discrete emission probability and is optimized using the Baum-Welch algorithm (Rabiner, 1989); 3) *IO-HMM* (Yin et al., 2017): Initial, transition and emission models work together to maximize the likelihood of observed sequences.

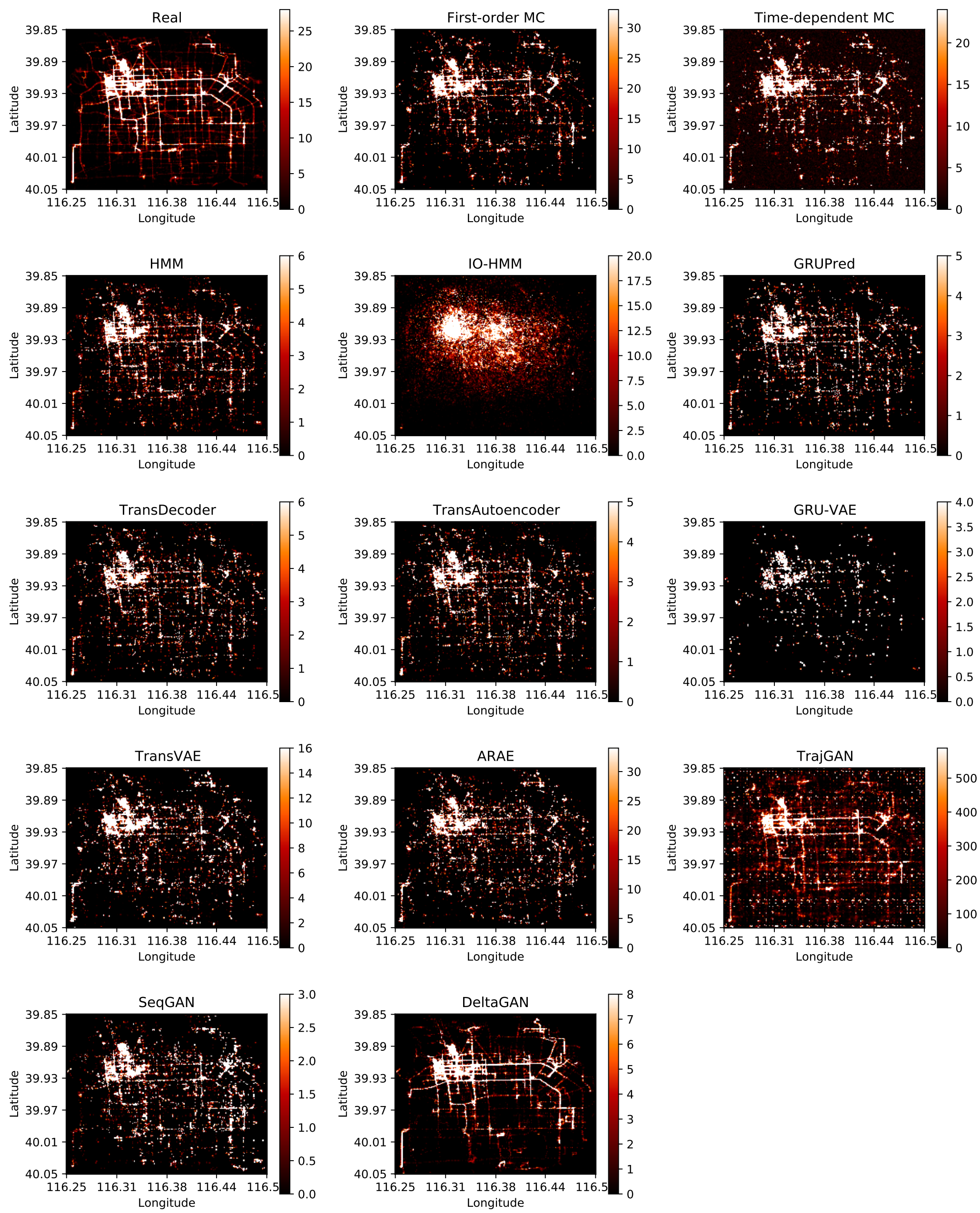
•**Deep Prediction Models**: Motivated by text generation with language models (Radford et al., 2018; Raffel et al., 2019), predictive models can be utilized to generate trajectories starting with a special token in an autoregressive way : 1) *GRUPred* (Cho et al., 2014): Gated Recurrent Units are utilized to predict next location given historical visited locations; 2) *TransDecoder* (Liu et al., 2018): A multi-layer Transformer decoder is utilized for location prediction; 3) *TransAutoencoder* (Vaswani et al., 2017): It builds an encoder to extract information from historical time data and feeds them to a decoder for sequential location generation.

•**Deep Generative Models**: We evaluate following variants of variational autoencoders (VAEs) (Kingma & Welling, 2013) and generative adversarial networks (GANs) (Goodfellow et al., 2014): 1) *GRU-VAE*: it adopts the vanilla VAE architecture equipped with GRU for sequence generation; 2) *TransVAE*: Both the encoder and decoder in the vanilla VAE are designed with the Transformer architecture (Vaswani et al., 2017); 3) *ARAE* (Zhao et al., 2018): It trains a GAN model to generate a prior which indistinguishable from the real latent representations learned by an autoencoder; 4) *TrajGAN* (Ouyang et al., 2018): A standard CNN-based GAN model is utilized to generate the trajectories in 2D matrices; 5) *SeqGAN* (Yu et al., 2017): Discrete location data is generated by combining Reinforcement Learning and GAN.

A.3 Trajectory Visualization and Analysis

We visualize both real and generated mobility data from all baselines in Fig. 3. In general, the majority of generative approaches can recognize the ring roads and the most popular Haidian District in Beijing. We observe that *TrajGAN* can capture the busiest locations in the real world much better than other methods, but the scale of visitation frequency is much larger than the reality, e.g., the brightest location has visitations expanded from 25 to 500. This indicates that *TrajGAN* can distinguish POIs (Point of Interest) in the city, but the underlying mobility pattern is not fully learned

Figure 3: Geographical visualization of $10,000$ real and generated trajectories from different models. To clearly show both the most and least visited locations, we set the first 95% percentile of visit times as the colormap range for mobility data from all models except *GRU-VAE*, which uses 98% percentile instead due to scarce location visitations.



and hence misrepresented in the generated trajectories. That also explains why poor performance from *TrajGAN* is shown in Table 1 from most of the spatial and temporal metrics. Although *IO-HMM* achieves moderate performance when evaluated by metric *Distance* and *Radius* in Table 1, most of the popular locations are missing in the generated trajectories. We attribute its failure to the loose assignment of home and work locations without any prior knowledge or post-checking.