

DeepTimeGeo: Trajectory Reconstruction From Sparse Data With Transformer

Shangqing Cao¹, Jiaman Wu¹, Aparimit Kasliwal¹, Baoqi Chen¹, Giuseppe Perona, and Marta C. González¹

Abstract—The completion of sparse Location-Based Service (LBS) data for modeling urban-scale origin-destination (OD) flow is of great importance to transportation planning applications. Sparse trajectories lack realistic human mobility patterns. Only with completed trajectories one can derive urban-scale OD flow that resembles complete travel diaries as those gathered by surveys or actively collecting phone applications. We present DeepTimeGeo (DTG), a transformer encoder-only model that reconstructs complete trajectories from sparse LBS inputs. We adopt a rank-based representation of locations to preserve individual-level heterogeneity and take a sequence-to-sequence approach to address the issue of gradient back-propagation blockage when it comes to regulating human mobility patterns. We devise human mobility distribution-based loss functions and leverage auxiliary learning to model the dynamics of exploration versus returns in users’ spatial choices. Experimental results show the superiority of DTG in trajectory reconstruction compared to other start-of-the-art generative models for human mobility trajectories. We conducted a case study with LBS data in the city of Coral Gables, Florida. The case study reveals that DTG leads to a reduction of more than 15% (1.35 vs. 1.60), when compared to the state-of-art model, in the cross-entropy loss that measures the deviation from the ground truth departure time distribution. We further demonstrate through SUMO simulation that DTG-generated trip demand captures both morning and evening rush hours, enabled by the more accurate distribution of trip departure time with important implications for traffic estimates.

Index Terms—Trajectory reconstruction, sequence-to-sequence transformer, self-supervised learning, auxiliary learning, human mobility.

I. INTRODUCTION

RAPID advances in mobile communication technology over the past few decades have paved the way to a new understanding of the movement of people, that is, human mobility, in urban regions. Compared to traditional methods, such as travel surveys, Location-Based Service (LBS) data provide a cheaper and more efficient alternative to gather people’s travel history. However, LBS data are collected passively. The intermittent use of cell phone applications results in sparse datasets that are inadequate to recover the complete

time-varying origin-destination (OD) travel flow within urban regions [1], [2], [3]. At the same time, to leverage human mobility datasets in various applications such as infrastructure planning [4], [5], traffic analysis [6], and contingency management [7], a complete travel history of individuals is needed [8], [9]. The partial observation and the passive collection of users’ travel history obscures the underlying human mobility laws that govern when people travel and where they travel to [10] and [11].

Challenges emerge when one attempts to reconstruct complete trajectories from a sparse input. First, reconstructed trajectories must comply with universal human mobility patterns, characterized by metrics such as circadian rhythm, characteristic distributions of travel distances and of the number of visited locations per day [12]. One can choose to model only the active users, whose trajectories are complete, to extract these governing distributions, but disregarding the inactive users leads to a loss of geographical information contained in their trajectories [13], [14].

We present DeepTimeGeo (DTG), a transformer-based model that reconstructs complete trajectories from sparse LBS data. We want to emphasize that DTG is not a predictive model. The purpose of DTG is to generalize and characterize the travel flow pattern in an urban region using static data. That is, we perform a one-time reconstruction on the input data set to recover a characteristic travel demand in an urban region. We do not use dynamically updated LBS data to predict future travel flow. The main task of the model is to accurately recover the observed stay points. The remaining unobserved entries are then generated under the guidance of mobility-informed regularizers. Our main contributions are summarized as follows.¹

- We propose a sequence-to-sequence transformer encoder-only model to reconstruct sparse trajectories. We incorporate mobility-informed regularizers by exploiting the sequence-to-sequence architecture to jointly regulate the mobility patterns in completed trajectories.
- We introduce a rank-based representation of the locations in the input space. The rank-based representation not only reduces the dimension of the output space but also preserves the individual location history in the sparse input data. This generalized representation of locations allows the model to be expanded to more users.
- We use auxiliary task learning to model the dichotomy between explorations and returns. The ablation study shows that the use of an auxiliary task improves the performance of the model.

Received 18 February 2025; revised 6 October 2025; accepted 2 January 2026. This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via the Department of Interior/Interior Business Center (DOI/IBC) under Contract 140D0423C0033. The Associate Editor for this article was Z. Ma. (Corresponding author: Shangqing Cao.)

Shangqing Cao, Aparimit Kasliwal, Baoqi Chen, Giuseppe Perona, and Marta C. González are with the Department of Civil and Environmental Engineering, University of California, Berkeley, CA 94720 USA (e-mail: caoalbert@berkeley.edu).

Jiaman Wu is with the Department of Data Science, City University of Hong Kong, Hong Kong SAR, China.

Digital Object Identifier 10.1109/TITS.2026.3657275

¹We release our code and the model for Los Angeles at <https://github.com/humnetlab/DeepTimeGeo>

- We tested the model on two separate real-world LBS datasets against state-of-the-art benchmarks, showing superior results in creating completed trajectories that resemble the mobility patterns documented in travel surveys. We also include a case study in the city of Coral Gables, FL, to showcase the application of DTG in estimating the time-varying urban-scale OD flows (Section VI). The case study reveals that, compared to the state-of-the-art model, DTG excels at generating flow patterns that match the distribution of travel by time of the day and by travel purpose.

The remainder of the paper is organized as follows. Section II provides a review of the literature on the existing trajectory reconstruction methods. Section III defines the relevant terminology. Section IV explains the architecture of DeepTimeGeo. Section V validates DeepTimeGeo and compares it with 7 other state-of-the-art baselines. Section VI demonstrates the use of DeepTimeGeo with a case study in Coral Gables, FL.

II. RELATED WORK

While Luca et al. [15] categorizes human mobility tasks into generation and prediction, methods for trajectory reconstruction can be categorized into Generation or Completion. Generation learns the dynamics of people's travel behavior, represents the dynamics as latent probability distributions, and then generates a separate set of realistic trajectories, while Completion seeks to retain the information contained in the raw data and complete the missing observations.

A. Generation

One can find a wealth of work in the generation approach. Many tackled the generation task with stochastic processes such as Markov chains and temporal point processes to model users' spatial and temporal decisions [14], [16], [17], [18], [19]. Some of these methods leverage existing parametric models that describe the dynamics of human mobility, including density exploration and preferential return (d-EPR) models [17] and ranked exploration and preferential return (r-EPR) models [14]. Others use temporal point processes such as the Poisson and Hawkes process to model the duration of stay between consecutive movements and spatial choices [18], [20]. However, many of the stochastic process-based methods are Markovian, which means that the generated locations are independent of each other. It has been pointed out that the probability distribution of visiting a certain location is not only conditioned on the user, which has been assumed in models such as d-EPR and r-EPR, but also on previous locations the user has visited. These models may suffer from the inability to capture the long-term correlation among the locations visited by the user [21].

Generative Adversarial Networks (GANs) have also been widely adopted for trajectory generation [20], [21], [22], [23], [24], [25], [26]. GANs consist of a generator and a discriminator, and the two networks play a Min-Max game [21], [27]. The generator learns the latent representation of the input features and the complex sequential transitions between different locations to generate synthetic trajectories [13], [21]. Meanwhile, the discriminator performs binary classification to

distinguish fake trajectories, which are synthetic trajectories, from the real ones in the input data set. Many use Recurrent Neural Networks (RNNs) to capture the sequential correlation between locations and the regularity in human mobility patterns in the generator [22], [23], [24], [26], [28], [29], [30]. More recently, attention blocks and transformers have also been embraced by the trajectory generation community due to their ability to capture long-term correlations between tokens within sequences [21], [31], [32], [33], [34]. Features such as location, time, and mode [31] are projected to a high-dimensional space, in which the key, query, and value matrices compute the similarity scores among the various input tokens to inform the next location prediction for autoregressive generation. The attention mechanism allows the model to focus on tokens that are significant and effective in predicting the location of the next timestamp [29].

B. Completion

The completion approach uses the sparse trajectory as the skeleton on which the fitted model imputes the missing entries to complete the trajectory. A common approach is to identify anchor points on the user's trajectory [1], [35] and then fill in the missing positions using heuristics such as interpolation [36] or machine learning algorithms. The completion task has also been formulated as a low-rank matrix factorization problem, in which the difference between the sparse input and the low-rank reconstructed output is minimized [2], [37]. Transformer and attention-based models have also been developed for completing sparse trajectories; however, these models are trained on the masked tokens on small, relatively complete GPS datasets [38], [39].

Completion differs from generation in that existing completion models reconstruct trajectories on a per-user basis. While this allows for personalized outputs, it significantly limits scalability and hinders transferability across users, as a separate model is often required for each individual [2], [35]. In contrast, generation models learn a shared latent distribution that characterizes the overall trajectory patterns across users. These models incorporate user identity as an input attribute, which is then used in the training of a single, generalizable model [21], [29]. However, this approach can struggle with user-level heterogeneity. Since activity types such as home and work are highly user-specific, generation models may fail to accurately reproduce personalized activity classifications within the generated trajectories [40].

Nonetheless, a few key research gaps exist in the current literature. First, there is a lack of systematic integration of human mobility-informed loss functions into models' training process. The autoregressive sampling action that is required to generate trajectories from the learned distributions in GANs blocks the back-propagation of gradients [8], [21], [25], [41]. As a result, one cannot evaluate the human mobility patterns of the generated output during the training process, and therefore unable to tune the model accordingly. While some resorted to reinforcement learning (RL) algorithms and used the reward function of the Markov Decision Process to regulate the human mobility distributions [21], [42], [43], others explored variational inference and incorporated a separate neural network to approximate the latent features based on the generated results [41]. Our approach uses

sequence-to-sequence generation, which allows us to directly incorporate human mobility-informed loss function into the training process. Human mobility distributions are defined on a set of trajectories rather than a set of single points. Because the sequence-to-sequence approach directly outputs trajectories, we can update the learnable parameters in the model directly with loss functions that characterize human mobility patterns.

While the majority of the existing learning-based methods adopt global location encoding, that is, each stay point is defined by a pair of latitude and longitude [21], [34], [41], [42], many Markovian model use user-specific location encodings [14], [17]. Often characterizing a location by its rank in the number of visits specific to a particular user, user-specific location encoding better preserves individual-level heterogeneity, as the set of locations that a user can explore is much smaller. It is important to emphasize that the choice of the encoding method should align with the intended application of the model. For many models that employ global location encoding, primary objective is to preserve and enhance user privacy [21], [22], [43]. Consequently, retaining user-specific information in the generated trajectories is less desirable. In these cases, the trajectory generation task is framed not as recreating plausible trajectories for each individual user, but rather as producing a dataset of synthetic trajectories that are statistically indistinguishable from the original dataset as a whole [22]. However, contextual information about trips, such as whether or not a trip is commute, depends on user-specific home and work locations. Therefore, it is still crucial to retain user-level information for applications beyond the realm of privacy enhancement. We introduce user-specific, rank-based encoding to a learning-based method, which retains both individual-level heterogeneity and leverages the ability of the transformer architecture to capture short- and long-term correlations between tokens in a dataset.

III. PRELIMINARIES

A. Definitions

LBS Dataset is a set of spatial-temporal observations. Each observation is a tuple $r = (i, t, x, y)$, where i is the user identification (ID) number, t is the timestamp, x is the longitude and y is the latitude.

Stay Point is a tuple $s = (i, l, \delta)$, where i is the user ID, l is the location ID, and δ is the stay duration. To convert an LBS dataset into a set of stay points, we adopt the method outlined in Zheng et al. [44], where observations in the LBS dataset are first filtered based on a set of spatial-temporal criteria. Consecutive observations that are less than 5 minutes apart in time and 300 meters apart in space are merged into one candidate stay point, of which the duration is the difference in time between the two consecutive observations in the same location. Then, a hierarchical mapping algorithm is applied to further cluster the candidate stay points and map the results to a square or hexagonal grid system to obtain a discrete representation of the locations [44].

Trajectory, $L_i = \{l_t^i | t \in T\}$, is a sequence of locations for user i within the observation time window. In this paper, we discretize the observation time window into hours, with T being the number of hours in the dataset. l_t^i denotes the location ID of the location user i visited in hour t . If no record is found, l_t^i is an empty set.

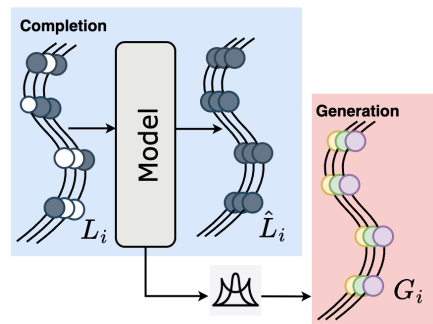


Fig. 1. Completion models fill in the missing stay points in a trajectory whereas generation models builds a probability space from which generated trajectories are sampled.

Sparse Trajectories, $\Omega_i \cup \Omega_i^c = L_i$, are trajectories that are not fully observed. That is, $\exists t \in T$ such that $l_t^i = \emptyset$, $l_t^i \in L_i$. We use $\Omega_i = \{l_t^i \in L_i | l_t^i \neq \emptyset\}$ to denote the observed portion of a trajectory L_i and use $\Omega_i^c = \{l_t^i \in L_i | l_t^i = \emptyset\}$ to denote the unobserved portion of the trajectory L_i . We can then define the sparsity, C_M , of a set of trajectories $M = \{L_1, \dots, L_N\}$ belonging to N users as:

$$C_M = \sum_i^N \frac{|\Omega_i^c|}{|L_i|} \quad (1)$$

B. Problem Statement

The sparse trajectory reconstruction problem is defined as the task of building complete trajectories from sparse inputs. Shown in Fig. 1, while both result in complete trajectories, the completion approach finds an approximation \hat{L}_i of L_i such that $\hat{L}_i \neq \emptyset \forall t \in T$. Although we want the reconstructed trajectories \hat{L}_i to resemble the observed portion of the sparse trajectory Ω_i to the greatest extent, a reconstruction solely based on Ω_i could distort human mobility patterns. Therefore, the challenge is to develop a method for reconstructing trajectories such that we can preserve as much information in Ω_i as possible, while ensuring that the reconstructed trajectories follow human mobility patterns.

In contrast, generation-based models output a probability distribution over a set of potential locations or trajectories. By sampling from this distribution, we can obtain trajectory realizations G_i that serve as a complete trajectory. In this paper, we formulate the training process of our model as a trajectory completion task. Meanwhile, the trained model can also be used as a zero-shot generator for inference, effectively bridging the gap between completion and generation.

IV. DEEPTIMEGEO

In this section, we present DeepTimeGeo (DTG) for sparse trajectory reconstruction. Fig. 2 illustrates the overall framework of our approach. We first encode stay points in trajectories through four features that characterize the stay points. We then apply a sequence-to-sequence transformer encoder to learn the spatial-temporal correlation among the stay points. The sequence-to-sequence model structure allows us to directly control the mobility patterns in the final output. This addresses the issue of gradient back-propagation blockage

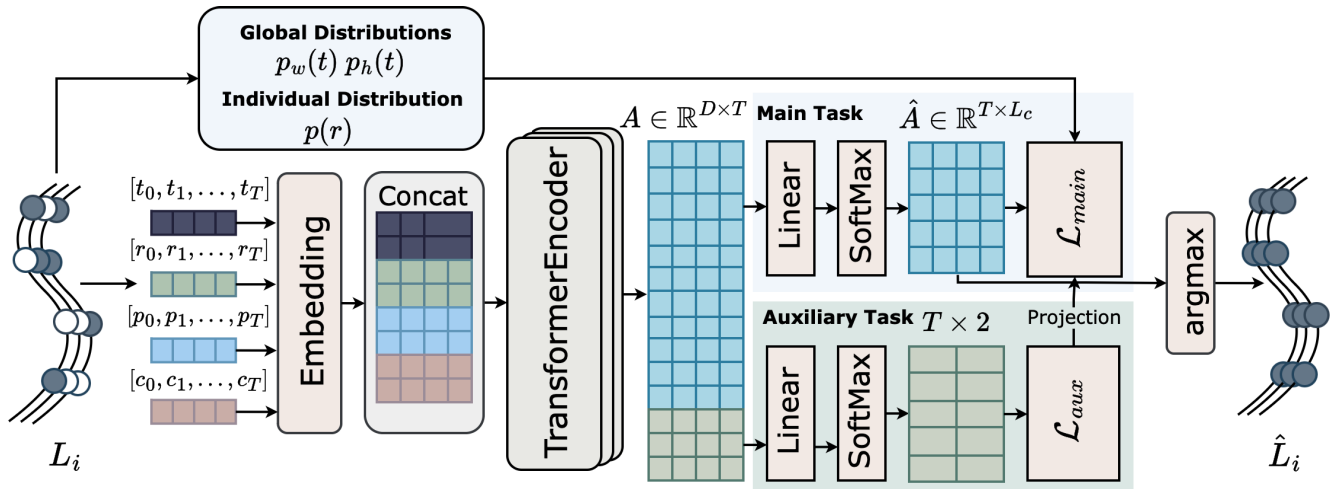


Fig. 2. The overall framework of DeepTimeGeo.

in conventional auto-regressive generation. The loss function of the main task is designed to reflect the mobility patterns. We introduce auxiliary learning to model exploration and returns to assist in the training process.

A. Rank-Based Input Representation

The main advantage of using the sequence-to-sequence transformer is that such a design gives us direct control over the mobility patterns implied by the completed trajectories. The parameters in the model are directly updated on the basis of the goodness of the mobility patterns evaluated by the loss function. However, instead of sampling one realization from the distribution over a set of locations, as is the case in auto-regressive generation, the sequence-to-sequence approach requires sampling multiple realizations that subsequently leads to the explosion of the size of the model. To address this issue, we adopt a rank-based representation of the candidate locations from which the completed trajectories are sampled. The rank-based representation significantly decreases the number of candidate locations, allowing us to retain the sequence-to-sequence structure.

Research in human mobility over the past decade has found the applicability of Zipf's law in modeling the number of visits to different locations, where the number of visits is inversely related to the rank of the location [33], [45]. The entire trajectory of a user over a period of time can be explained by some of the most frequently visited locations. Therefore, instead of sampling from all potential locations, which can reach hundreds of thousands in a large metro area, the model learns on the rank-based representation that is unique to each user.

Fig. 3A) shows that a mere number of 10 locations can account for more than 70% of all observed stay points in sparse trajectories. We adopt a rank-based representation of the locations for each user where the function $R_i(l)$ maps a location $l = (x, y)$ to the rank-space for user i . After completing a user's trajectory using DTG, we then apply the inverse function, $R^{-1}(r)$, to obtain the geographical coordinates. As users can have varying numbers of visited locations, in the implementation of DTG, we set a threshold of L_c , the upper

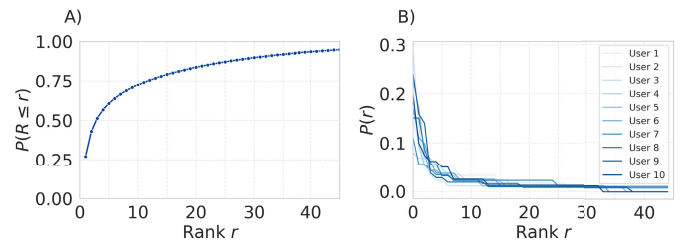


Fig. 3. Rank representation. A) The percentage of stay points in the Coral Gables dataset that can be explained with rank r locations. B) A sample of 10 users and the probability distribution, $p(r)$, of the number of stay points observed at their respective rank r .

bound of the rank of the locations we consider. We do not consider locations with a rank greater than L_c .

We extract four features from the sparse input trajectories: (1) time of visit t (as in hour of the day), (2) rank of the location r , (3) the category of Point of Interest (POI) of the location p , and (4) a binary indicator of whether or not a user is commuter c . DTG does not rely on external POI information. The locations are categorized into *home*, *work*, or *other* locations. Details on the home and work detection process, as well as the identification of commuters are outlined in Appx. A.

We concatenate the four features into a dense representation:

$$x_i = [W_t t_i, W_r r_i, W_p p_i, W_c c_i] \quad (2)$$

where W_t , W_r , W_p , and W_c are the respective learnable weights of the embedding matrix. The embedding matrix, x_i , has a dimension of $D \times T$, where $D = d_{time} + d_{loc} + d_{poi} + d_{comm}$ is the embedding dimension.

B. Sequence-to-Sequence Transformer Encoder

Sequence-to-sequence transformer models such as BART have been proven to be effective in various tasks such as language translation [46], audio transcription [47], and conversation modeling [48]. One key advantage of this approach is that it provides a self-contained, end-to-end training process

that enables the evaluation of a differentiable loss function embedded with patterns in the generated output [47]. We leverage the sequence-to-sequence transformer encoder and tackle the trajectory reconstruction problem with self-supervised learning.

Because the objective is reconstructing complete trajectories rather than predicting, we adopt an encoder-only model where self-attention has access to both the stay points that come before and after the current i th stay point in a length trajectory T . Therefore, when computing the attention values, we sum all tokens, j , on the trajectory instead of those that appear only after the i th token, as proposed in the masked attention scheme [33], [49]. The attention score between two input tokens is calculated as

$$\alpha_{ij} = \frac{\exp(q_i \cdot k_j / \sqrt{d_k})}{\sum_j \exp(q_i \cdot k_j / \sqrt{d_k})} \quad (3)$$

where $d_k = \frac{D}{N_{\text{heads}}}$ is the dimension of the query matrix in a single attention block.

The transformer encoder outputs a matrix $A = \text{TransformerEncoder}(\mathcal{X})$ and $A \in \mathbb{R}^{D \times T}$ where D is the dimension of the embedding across the four input features and T is the sequence length. We train both the main task and the auxiliary task from A . The main and auxiliary tasks attend to specific components of A , where $A = [A_m, A_a]$, with A_m the rows in A that attend the main task and A_a the rows in A that attend the auxiliary task. Then,

$$\begin{cases} P_l(i, t) = \text{Softmax}(\text{Linear}(A_m)) \\ P_a(i, t) = \text{Softmax}(\text{Linear}(A_a)) \end{cases} \quad (4)$$

where $P_l(i, t)$ is the probability distribution for selecting the location rank l at time t for trajectory i and $P_a(i, t)$ is the binary probability distribution for the auxiliary task, in which the model is asked to classify if the location is an exploration. Details on the auxiliary task are explained in sec. IV-D.

To create complete trajectories from the probability distribution over locations, we apply the following:

$$\hat{L}_i = \text{*argmax}_l P_l(i, t) \quad (5)$$

C. Main Task

DTG is a self-supervised learning model. The objective is to train a function, $f(L_i) = \hat{L}_i$, that maps a sparse trajectory to a complete output. The main task of the model consists of two parts: learning to recover the observed entries in the trajectory and regulating the complete output trajectory with respect to the human mobility laws. We define the overall loss function of the main task as follows:

$$\mathcal{L}_{\text{main}} = \mathcal{L}_{\text{sp}} + \sum_{i=1}^3 \theta_i \mathcal{L}_i \quad (6)$$

where \mathcal{L}_{sp} is the ‘‘single-prediction’’ loss, in which we evaluate the prediction for the set of observed stay points in the trajectories and \mathcal{L}_i s are the human mobility-informed regularizers.

1) \mathcal{L}_{SP} : *Recovery of Observed Stay Points*: We used the cross-entropy loss function to evaluate the recovery of observed stay points in the trajectories. We define \mathcal{L}_{sp} as:

$$\mathcal{L}_{\text{sp}} = - \sum_i^S \sum_l^{L_c} y_{il} \log(p_{il}) \quad (7)$$

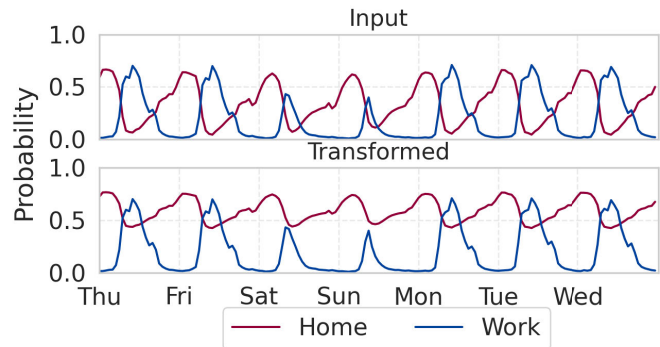


Fig. 4. Distribution of input and the transformed probability of being at home and work locations. We set $\alpha_h = 0.55$ and $\beta_h = 0.4$ while keeping $\alpha_w = 1$ and $\beta_w = 0$. The work and home distributions in the transformed plot are respectively $P'_h(t)$ and $Q'_w(t)$.

where $S = \bigcup_{i \in N} \Omega_i$ is the set of observed stay points in the number of trajectory samples N . y_{il} is a binary indicator that takes the value 1 if stay point i contains the stay location rank l . p_{il} is the probability of selecting location l for stay point i . The matrix characterizing p_{il} is the output of the transformer encoder model.

2) \mathcal{L}_1 : *Controlling Location Ranks*: We use \mathcal{L}_1 to account for the distribution of the number of visits to locations of various ranks, which can subsequently affect the number of daily visited locations. From the output probability matrix, $\hat{A} \in \mathbb{R}^{T \times L_c}$, we derive an explicit expression to regulate the distribution of location ranks. We define \mathcal{L}_1 as:

$$\mathcal{L}_1 = \sum_i^N P_i(l) \log \frac{P_i(l)}{Q_i(l)} \quad (8)$$

where $P_i(l)$ is the ground truth probability distribution of the number of stay points in location rank l as observed in the input data. To introduce heterogeneity among the samples, we calculate $P_i(l)$ for each individual user in the input data, shown in Fig. 3B). When evaluating the loss, we select the ground-truth distribution specific to the user. The predicted distribution as recovered from the output matrix \hat{A} is subsequently:

$$Q_i(l) = \frac{\sum_t^T \hat{A}_{tl}^i}{|T|} \quad (9)$$

where \hat{A}_{tl}^i is the probability of selecting location rank l at time t for sample i . T is the length of the trajectory.

3) \mathcal{L}_2 : *Controlling Departure Time*: To obtain a realistic time-varying OD flow on an urban scale, a crucial aspect is the distribution of departure time. The departure time can only be calculated autoregressively; that is, the distribution of departure time is a function of both the time and the location of the user. However, as mentioned in the introduction, neither the sampling action nor the **argmax* operator used in DTG is differentiable, resulting in no closed-form expressions for controlling the distribution of departure time. Hence, we use the following function to approximate the departure time loss:

$$\mathcal{L}_2 = \sum_i^N \sum_{t=0}^{T-1} \sum_l^L (\hat{A}_{tl}^i - \hat{A}_{(t+1)l}^i)^2 \cdot \frac{1}{p(t)} \quad (10)$$

By comparing the difference in the probability distributions between two consecutive time slots, we can approximate the likelihood of moving. Because in the reconstruction process, the location with the highest probability is selected, two probability distributions that are vastly different imply a higher chance of changing location. We then divide the loss by $p(t)$, which is the ground truth probability of moving in time slot t . $p(t)$ can be extracted either internally from the sparse input dataset, or imported from an external source.

4) \mathcal{L}_3 : *Modeling Circadian Rhythm*: The travel behavior of individuals is largely driven by commutes, reflected by stay points at home and work locations. The probability of being at home follows a circadian rhythm [50]. Individuals are more likely to be observed at the home location at night and on weekends. To reflect such a pattern, we compute the probability of being at home and work locations to reflect the commuting pattern. We define commute loss, \mathcal{L}_3 , as the sum of the KL-Divergence of being at and home and work locations:

$$\mathcal{L}_3 = \sum_t P'_h(t) \log \frac{P'_h(t)}{Q_h(t)} + \sum_t P'_w(t) \log \frac{P'_w(t)}{Q_w(t)} I_c \quad (11)$$

where $P'_h(t)$ and $P'_w(t)$ are the ground truth distributions of being at home and at work, and I_c is the binary indicator variable mentioned in Sec. IV-A that takes the value 1 if the user is deemed a commuter. As a result, while the loss on the distribution of being at home applies to all users, the loss on the distribution of being at work applies only to commuters. To obtain the ground truth distributions, we first recover the probability of being at home and at work locations, $P_h(t)$ and $P_w(t)$, from the observed stay points in the sparse input. The process of detecting home and work locations is explained in Appendix A. We then apply a linear transformation to obtain the ground truth distributions with $P'_h(t) = \alpha_h + \beta_h P_h(t)$ and $P'_w(t) = \alpha_w + \beta_w P_w(t)$. We use the findings in Song et al. [50] to calibrate the parameters, α_h , α_w , β_h , and β_w , for the transformation. Fig. 6 shows the probability distributions before and after the transformation.

The probability distribution of appearing at home and at work can be recovered from the output matrix \hat{A} as:

$$Q_h(t) = \sum_i^N \hat{A}_{th}^i \cdot \frac{1}{N} \quad (12)$$

$$Q_w(t) = \sum_i^N \hat{A}_{tw}^i \cdot \frac{1}{N} \quad (13)$$

where h and w is the location rank of the home and work location of sample i .

D. Auxiliary Task

Auxiliary task is a task learned simultaneously along with the main task to improve a model's overall performance [34]. Auxiliary task learning has been adopted for human mobility prediction in helping with predicting POI by considering the location prediction task as an auxiliary task [34]. Inspired by previous models such as TimeGeo [14] and empirical findings in Pappalardo et al. [51], we formulate the auxiliary task as a module that predicts whether or not a given observed stay point is an exploration or return. We define a visit as an exploration if the number of stay points at a location in the entirety of a

TABLE I
SUMMARY STATISTICS OF THE CORAL GABLES
AND LOS ANGELES LBS DATASET

Dataset	# Users	# Loc	# Stay Points	C_m
Coral Gables	46,952	106,853	8,139,004	0.882
Los Angeles	11,804	88,315	1,546,008	0.824

user's trajectory is less than n_c , a threshold of the number of visits. We then define the auxiliary loss, \mathcal{L}_{aux} , as:

$$\mathcal{L}_{aux} = - \sum_i^S e_i \cdot \log(p_i) + (1 - e_i) \cdot \log(1 - p_i) \quad (14)$$

where e_i is a binary indicator of whether or not an observed stay point i is an exploration. p_i is the predicted binary probability mass function of exploration for the observed stay point i . To ensure that the auxiliary task improves the model performance, we then project the gradient of \mathcal{L}_{aux} onto \mathcal{L}_{main} with weighted cosine [52]:

$$\nabla \mathcal{L} = \max(0, \cos(\nabla \mathcal{L}_{main}, \nabla \mathcal{L}_{aux})) \cdot \nabla \mathcal{L}_{aux} + \nabla \mathcal{L}_{main} \quad (15)$$

A summary of all the loss terms can be found in Appx. E. The mobility regularizers can either source the ground truth from the sparse input or use external distributions, depending on the input data quality.

V. EXPERIMENTS

In this section, we compare the performance of DTG with other state-of-the-art models. We include an ablation study that measures the effectiveness of the different components of DTG, including the auxiliary task and the three human mobility-informed regularizers.

A. Datasets

We tested our model and the benchmarks on two real-world datasets. We apply the stay point detection algorithm discussed in Sec. III-A to convert the raw LBS datasets into trajectories.

- **Coral Gables, FL**: The Coral Gables dataset is obtained from Cuebiq,² with 46,952 users whose activities span from 9/1/2022 to 10/31/2022. We use the home detection algorithm explained in Appx. VII-A to select users whose residence is identified as within the limit of the city of Coral Gables. We then select all of the identified users' stay points in the Miami-Dade County.
- **Los Angeles, CA**: The Los Angeles dataset is obtained from an anonymous provider, with 11,804 users whose activities span from 1/1/2019 to 1/31/2019.

A summary of the datasets can be found in Tab. I. The difference in the number of locations is attributed to the difference in the number of users. As we only consider

²Aggregated mobility data is provided by Cuebiq, a location intelligence and measurement platform. This first-party data is collected from anonymized users who have opted-in to provide access to their location data anonymously, through a CCPA and GDPR-compliant framework. Through Cuebiq's Social Impact program, Cuebiq provides mobility insights for academic research and humanitarian initiatives. Cuebiq's responsible data sharing framework enables Social Impact partners to query anonymized and privacy-enhanced data, by providing access to an auditable, on-premise sandbox environment. All final outputs provided to partners are aggregated in order to preserve privacy.

locations that appear in the set of observed stay points as candidates, the number of locations is correlated with the number of users. In general, the data are very sparse. Given an hourly representation of the trajectories, the sparsity of both datasets exceeds 80%, which means that more than 80% of the hourly slots are not observed.

B. Training

To train DTG, we set $L_c = 45$, which means that we consider the top 45 locations that each user visits. As shown in Fig. 3A), the top 45 locations cover more than 95% of all the stay points of the users. We set the sample length $T = 336$, which covers a two-week period. We obtain at most 4 samples from each user in the Coral Gables dataset because the time span of the dataset is 61 days. We obtain at most 2 samples for training from each user in the Los Angeles dataset because the time span of the dataset is 31 days. We set $\alpha_h = 0.55$, $\alpha_w = 1$, $\beta_h = 0.4$, and $\beta_w = 0$. The remaining hyper-parameters of the model can be found in Appx. VII-B. Due to the sparsity of the dataset, we import the ground truth departure time distribution, $p(t)$, from the American Time Use Survey (ATUS). Details on how we obtain $p(t)$ can be found in Appx. VII-D1.

C. Baseline and Metrics

We include the following baselines in our comparative study: Semi-Markov [16], Hawkes Process [53], SeqGAN [25], TrajSynVAE [18], LSTM [28], MoveSim [21], TrajGDM [13] and TimeGeo [14]. Details on the benchmark models can be found in Appx. VII-C. As the objective of DTG is to complete sparse trajectories for mimicking reality, we extract the ground truth distributions of the following metrics from travel surveys: DailyLoc, Departure, Duration and Distance.

- **DailyLoc:** DailyLoc is the probability density function (PDF) of the number of unique daily visited locations. Every 24-hour trajectory is treated as a sample in the calculation.
- **Departure:** Departure is the PDF of moving. The time of appearance at the new location is deemed the time of departure.
- **Duration:** Duration is the PDF of stay duration. Stay duration is the time elapsed between consecutive stay points at two different locations.
- **Distance:** Distance is the PDF of the haversine distance between two consecutive locations during a move.

To account for regional differences, we use the 2022 National Household Travel Survey (NHTS) and the 2017 NHTS for the Coral Gables and Los Angeles experiments, respectively [54]. For the Coral Gables experiment, we use surveys collected in the South Atlantic Census Region due to the small sample size in Florida and in Miami-Dade. For the Los Angeles experiment, we use surveys collected in the Los Angeles-Long Beach-Anaheim Core Base Statistical Area (CBSA). We chose to use the 2017 survey for Los Angeles because we assume that people’s travel behavior in 2019 is more consistent with the 2017 patterns rather than 2022, which is post-COVID-19. Details on how we compute the ground truth distributions from the travel surveys can be found in appx. VII-D2. We compute the Jensen-Shannon Divergence (JSD) in the aforementioned metrics between the ground truth

distribution, p , recovered from the travel surveys, and the distributions, q , from the completed trajectories. The JSD is defined as:

$$JSD(p, q) = \frac{1}{2}KL\left(p \parallel \frac{p+q}{2}\right) + \frac{1}{2}KL\left(q \parallel \frac{p+q}{2}\right) \quad (16)$$

where $KL(\cdot \parallel \cdot)$ is the Kullback-Leibler divergence.

D. Results

Tab. II details the performance of DTG and the benchmarks on the four metrics in the two datasets. We separate TimeGeo from the rest of the benchmark models, as it makes strong assumptions about the duration of work and the start time of work for commuters [14]. DTG excels in modeling the distribution of the number of daily visited locations. Except for TrajGDM in the Los Angeles dataset and TimeGeo, all other benchmarks demonstrate a high JSD between the generated output and the ground truth. Consistent with what has been documented in previous findings [41], TimeGeo performs well in Departure on the Coral Gables dataset because it explicitly uses the circadian rhythm to model the temporal decision of individual users. TimeGeo can still suffer from poor departure time distribution in the input dataset. As illustrated in Fig. 6B), the Los Angeles dataset lacks the early morning rush hour peak in departure time, leading to worse performance compared to DTG. For Distance, NHTS does not contain the spatial information of the locations that people visit. The surveys only provide the trip distance on the road instead of the haversine distance between the two locations, leading to a biased mismatch in the computation of Distance. Combined with the fact that we do not explicitly embed distance between stay points as a feature in DTG, models such as TrajSynVAE yield better results. Nonetheless, the JSD between the best performing models and DTG is within the same order of magnitude.

In addition to DTG and TimeGeo, two recently released models also demonstrate strong performance. TrajSynVAE and TrajGDM performs well in Duration and Distance. In both datasets, the two models closely trail the performance of DTG and TimeGeo. Unlike models such as MoveSim that adopt autoregressive generation, TrajSynVAE and TrajGDM sample an entire trajectory from a continuous latent presentation of the set of all potential trajectories [13]. This suggests that a sequence-to-sequence approach better regulates the human mobility patterns in the generated output compared to models that rely on autoregressive generation. In general, across the two datasets, DTG has the lowest average rank (AVG) of the four metrics.

Fig. 5 illustrates DailyLoc, Departure, and Duration in the Los Angeles dataset. As indicated in Fig. 5A), DTG recovers the distribution of the number of daily visited locations due to the implementation of the regularizer \mathcal{L}_1 that controls the percentage of visits to locations of different ranks. Fig. 5B) shows that DTG augments the distribution of departure time by introducing the early morning rush hour that is missing from the sparse input. By modeling commutes, DTG completes the sparse input by creating stays with duration around 6-10 hours as shown in Fig. 5C), which corresponds to the number of hours for a typical work day.

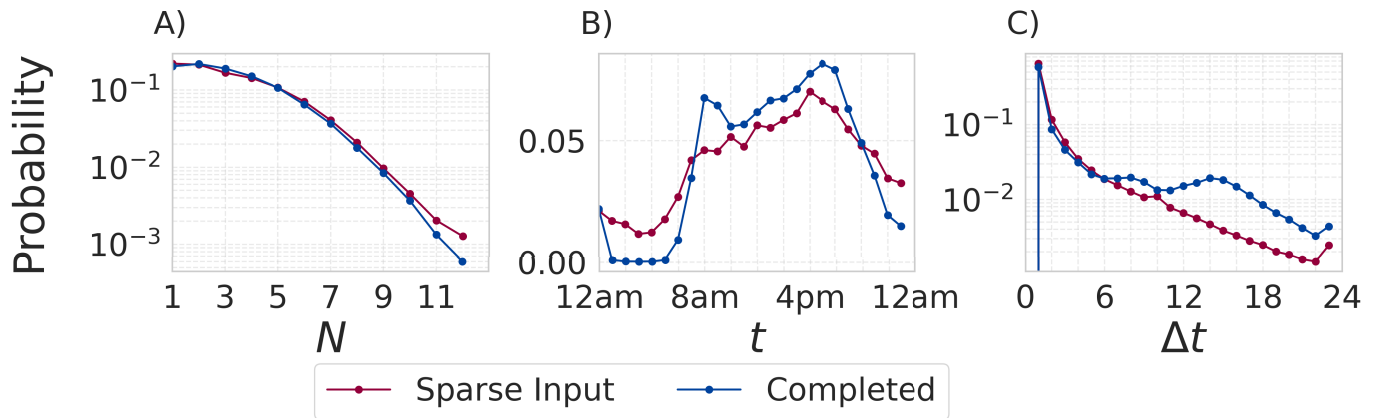


Fig. 5. Human mobility patterns in Los Angeles. A) The probability distribution of the number of daily visited locations. B) The probability distribution of departure time. C) The probability distribution of stay duration in hours.

TABLE II

EVALUATION OF MODELS BASED ON THE TWO LBS DATASET. THE BOLD NUMBERS INDICATES THE BEST (LOWEST SCORE) MODEL AND THE UNDERLINED NUMBERS INDICATE THE SECOND BEST MODEL FOR THE GIVEN METRIC. AVG IS THE AVERAGE RANK OF THE MODEL ACROSS THE 4 METRICS. A LOWER AVERAGE RANK INDICATES A BETTER MODEL. THE GROUND TRUTH IN DISTANCE* IS LACKING IN SURVEYS, WE USE THE DISTRIBUTION OF TRIP DISTANCE AS THE GROUND TRUTH INSTEAD OF THE HAVERSINE DISTANCE

Dataset	Coral Gables					Los Angeles				
	DailyLoc	Departure	Duration	Distance*	AVG	DailyLoc	Departure	Duration	Distance*	AVG
SMM	0.6277	0.1113	0.2624	0.0138	5.5	0.2970	0.0999	0.2098	0.0116	5.75
Hawkes	0.4242	0.4339	0.2779	0.0157	6	0.2722	0.2013	0.2295	0.0101	5.5
SeqGAN	0.4376	0.1867	0.1783	0.0507	5.75	0.1960	0.1064	0.068	0.1776	5.75
TrajSynVAE	0.4755	0.0904	0.2073	<u>0.0154</u>	<u>4</u>	0.3866	<u>0.0234</u>	0.1518	<u>0.0107</u>	4.25
LSTM	0.5644	0.1037	0.2386	0.0159	5.25	0.3606	0.0926	0.2568	0.0112	6
MoveSim	0.6639	0.1117	0.3924	0.1689	8.5	0.4334	0.0922	0.3798	0.2145	8
TrajGDM	0.4117	0.0886	0.2494	0.0647	5	<u>0.0648</u>	0.0483	<u>0.0439</u>	0.1755	3.75
TimeGeo	<u>0.0563</u>	0.0182	0.0171	0.0432	2.5	0.0934	0.0318	0.0284	0.0301	<u>3.25</u>
DeepTimeGeo	0.0321	<u>0.0492</u>	<u>0.1144</u>	0.0213	2.5	0.0615	0.0161	0.1296	0.0287	2.75

To further demonstrate the value of a rank-based approach, we computed the Jaccard index between the set of unique locations in the training data and in the generated output. We define the Jaccard index for a user i as:

$$J(U(L_i), U(\hat{L}_i)) = \frac{U(L_i) \cap U(\hat{L}_i)}{U(L_i) \cup U(\hat{L}_i)} \quad (17)$$

where the function $U(\cdot)$ returns the set of unique locations given a trajectory. We compute the average Jaccard index on the users and report them in Tab III. We see that DTG, by definition, results in a Jaccard index of one, but other models based on global encoding, including TrajSynVAE, MoveSim, and LSTM result in extremely low Jaccard index. This means that a user cannot be characterized by the set of locations he or she visits. Instead, the trajectories are generated for a global population that shares a homogeneous set of potential locations to visit. In other words, although global encoding methods can be good at generating trajectories that follow mobility distributions globally, they have a low level of individual-level characterization.

However, DTG, Hawkes, and SMM sampled from a fixed and individualized set of candidate locations, which yielded a much higher Jaccard index. This means that the generated trajectories are in fact specific to individuals. Because individual-level heterogeneity is inherently captured in the

training data, these models effectively transfer individual-level heterogeneity from the training data to the generated output. This is not to say that the generated output from models with a low Jaccard index is produce inaccurate results; instead, the Jaccard Index can be used as a simple surrogate to measure how well the model performs for the reconstruction task, of which the goal is to complete a user's trajectory rather than generating a distinctively new one.

E. Ablation Study

To understand the compartmental contribution of each design in DTG, we construct the following case studies:

- **Case 1:** We retain all loss terms in the main task but exclude the auxiliary task.
- **Case 2:** We cease to control the probability of being at home and work locations, i.e. $\theta_3 = 0$.
- **Case 3:** We cease to control the distribution of departure time, i.e. $\theta_2 = 0$.
- **Case 4:** We cease to control the distribution of the number of stay points at locations of different ranks, i.e. $\theta_1 = 0$.

Tab. IV shows that all components of DTG are essential to replicating the ground-truth human mobility patterns in the finished output. Turning off the auxiliary task leads to poorer performance in DailyLoc, Departure, and Distance. Setting $\theta_1 = 0$ leads to a significant degradation in DailyLoc, as

TABLE III

THE JACCARD INDEX MEASURING THE AVERAGE PERCENTAGE OVERLAP OF THE UNIQUE LOCATIONS OBSERVED IN THE TRAINING AND GENERATED TRAJECTORIES OVER USERS. WE EXCLUDE SEQGAN AND TRAJGDM, THE OUTPUT OF WHICH DO NOT INCLUDE USER IDENTIFICATION

Model	Coral Gables	Los Angeles
TrajSynVAE	0.010	0.000
MoveSim	0.002	0.001
LSTM	0.075	0.059
SMM	0.899	0.899
Hawkes	0.900	0.896
DTG	1.000	1.000

TABLE IV
ABLATION STUDY

Metrics (JSD)	DailyLoc	Departure	Duration	Distance*
Case 1	0.0675	0.0140	0.1377	0.0290
Case 2	0.0629	0.0205	0.1275	0.0289
Case 3	0.2797	0.0463	0.2514	0.0194
Case 4	0.4223	0.0421	0.0995	0.0252
DTG	0.0615	0.0161	0.1296	0.0287

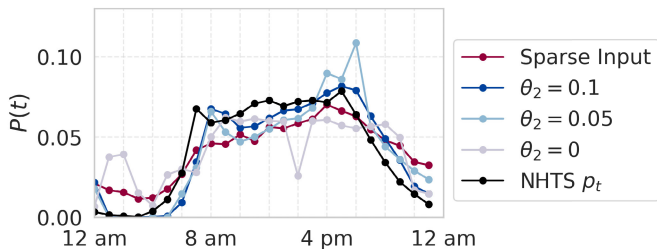


Fig. 6. Sensitivity analysis on the distribution of departure time.

the JSD increases from 0.0615 to 0.4223. Controlling the distribution of number of visits to different ranks helps regulate the number of locations users visit on a daily basis. Turning off the departure time regularizer leads to an increase in the metric Departure and turning off the commuting regularizer leads to an increase in DailyLoc, Duration, and Distance.

F. Sensitivity Analysis

We perform a sensitivity analysis on the departure time loss weight, θ_2 , to demonstrate the sensitivity of the completed output to the loss function. Fig. 6 shows that as we increase the weight, the distribution in the generated output becomes more aligned with the ground truth. With such a design, the generated distribution takes both the input distribution and the ground truth distribution into account when completing the trajectories.

VI. CASE STUDY: CORAL GABLES, FL

To showcase the capability of DTG in generating urban-scale time-varying OD flow, we present a case study set in the city of Coral Gables, Miami-Dade County, Florida. We compare our model to TrajSynVAE [18], which has demonstrated strong performance in Sec. V-D. We use the Southeast Florida Regional Planning Model (SERPM) to obtain the ground truth

TABLE V

PERCENTAGE OF TRIPS RECORDED ON DIFFERENT TIMES OF THE DAY AND TRIPS OF DIFFERENT TYPES. TRIPS ARE CLASSIFIED INTO ONE OF THE THREE CATEGORIES: HOME-BASED WORK (HBW) WHERE A USER TRAVELS BETWEEN HIS *home* AND *work* LOCATION, HOME-BASED OTHER (HBO) WHERE A USER TRAVELS BETWEEN HIS *home* AND *other* LOCATION, AND NON-HOME-BASED (NHB), TRIPS NOT COVERED BY HBW AND HBO

Time	AM	MD	PM	RD
TrajSynVAE	0.0331	0.2469	0.3433	0.3767
DTG	0.2076	0.4083	0.2900	0.0944
SERPM	0.1808	0.3645	0.2949	0.1598
Type	HBW	HBO	NHB	
TrajSynVAE	0.0039	0.1560	0.8398	
DTG	0.1962	0.5174	0.2865	

time-varying OD flow in the region [55]. We divide a day into four different time periods, AM [7 a.m.-10 a.m.], MD (10 a.m.-3 p.m.), PM (3 p.m.-7 p.m.) and RD (the rest of the day). We compute the OD flow as generated by the two models at three spatial resolutions. Levels 5 to 7 hexagons, as defined by the Uber h3 package, have respective areas of 252.9 square kilometers (km²), 36.1 km², and 5.2 km² [56]. We compute the coefficient of correlation r between the generated flow and the SERPM flow as the evaluation metric.

Fig. 8 reveals that both DTG and TrajSynVAE perform well at low resolution. As the resolution increases, both models see a decreasing coefficient of correlation. DTG outperforms TrajSynVAE at resolution levels 5 and 6. Specifically, DTG records a higher coefficient of correlation in the early mornings compared to TrajSynVAE. The ability of DTG to model morning commutes can be attributed to the fact that we actively regulate both the distribution of departure time (\mathcal{L}_2) and, more importantly, the probability of going to work and staying at home (\mathcal{L}_3), during the training process. As Tab. V shows, DTG demonstrates a more reasonable break down of the time of move. TrajSynVAE only records around 3.31% of trips between 7 am and 10 am, which is clearly an underestimation that affects the resulting ODs for traffic planning.

The strong correlation between the two models, DTG and TrajSynVAE, and SERPM is also illustrated in Fig. 7. Both models successfully identify areas with high travel flow. Edges with dense flow in the SERPM are also reflected in the respective generated flows. The figure also explains the decreasing correlation coefficient between the completed results and SERPM as we increase the resolution of the hexagons. Because we select trajectories of users who reside in Coral Gables as training input, parts of the northeast of Miami-Dade county are not covered by the training data.

Except for the early morning, the performance of DTG lags behind that of TrajSynVAE at resolution level 7. While the rank-based representation in DTG reduces the number of candidate locations for each user to at most $L_c = 45$, TrajSynVAE samples from a global representation of locations [18] during the generation process. The generated trajectories of individual users can include stay points from any location on the grid map, which contributes to the modeling of flow at a finer resolution.

DTG outperforms TrajSynVAE in modeling individual travel patterns. Tab. V shows that with DTG, HBW trips

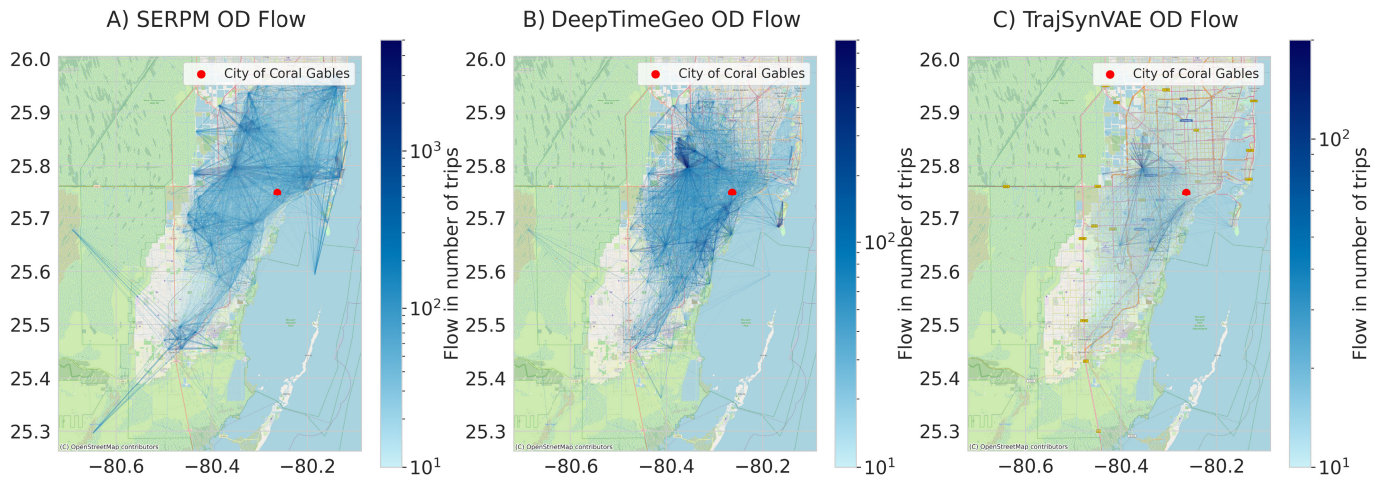


Fig. 7. Comparing spatial distribution of flow generated by the three models. A) The SERPM OD flow is a single-day model consisting 1.8 million trips [55]. B) The DeepTimeGeo OD flow is aggregated over completed trajectories in a 14-day period, consisting 1.2 million trips. C) The TrajSynVAE OD flow is computed based on the generated trajectory of 1,660 users on a single-day, consisting of 646,000 trajectories. The difference in time span is attributed to the fact that both SERPM and TrajSynVAE are both single-day models whereas DTG performs multi-day generation due to its sequence-to-sequence nature.

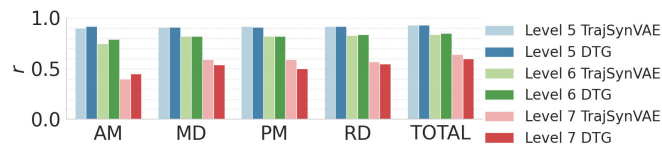


Fig. 8. Coefficient of correlation at 3 different spatial resolutions during different time periods of the day.

account for around 20% of all trips and NHB trips only account for around 29%. In comparison, of all trips generated by TrajSynVAE are HBW. NHB trips account for more than 83% of all trips. Such a breakdown of trip types does not align with the fact that most people's travel revolves around the home location. The disparity in the ratios of the trip types between DTG and TrajSynVAE shows DTG's superiority in modeling individual mobility history. Although TrajSynVAE generates trajectories that demonstrate good mobility patterns, as shown in Sec. V-D, and realistic OD flow, shown in Sec. VI, it suffers from the inability to generate trajectories that could incorporate advantage of DTG also lies in its ability to generate OD flow that matches the temporal distribution of the travel flow. For example, MD contains more than 36% of the total number of trips according to SERPM. While DTG arrives at 41%, TrajSynVAE records 25% of the trips being in MD. The difference in the percentage of trips at different times of the day between SERPM and DTG is much smaller than between SERPM and TrajSynVAE. This result further showcases DTG's superiority in modeling realistic urban-scale We further demonstrate using the trajectories generated by DTG can reflect morning and evening rush hours when used as inputs to traffic simulation. We employ software Simulation of Urban Mobility (SUMO) version 1.12.0 on a Linux machine with x86_64 architecture and 24 CPUs to simulate traffic congestion using the generated output of DTG and TrajSynVAE. We obtained the Miami-Dade road network from OpenStreetMap (OSM). To address the difference in the number of users captured by the TrajSynVAE and DTG output,

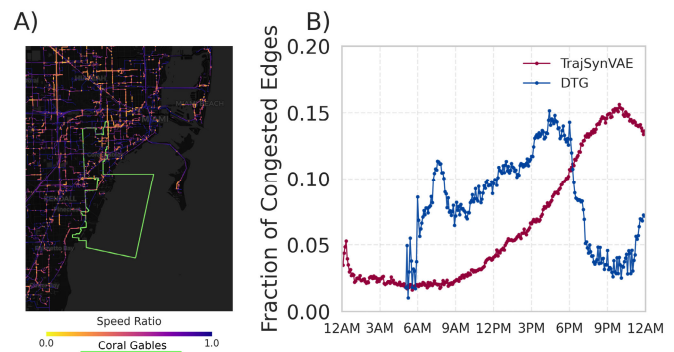


Fig. 9. SUMO simulation results. A) Simulated congestion at 9 AM using trip demand derived from DeepTimeGeo. Speed ratio is defined as ratio between the average speed observed in simulation and the speed limit of the road link. B) The fraction of edges that are congested, which describes road links in which the average density exceeds the critical density of the link.

we applied different expansion factors to the two datasets. The 2020 census reports that Coral Gables has a population of 49,248. The expansion factor is defined as the ratio between the number of residents in Coral Gables and the number of users in a given dataset. In the process of creating trip demand for the SUMO simulation, we multiplied each trip by the expansion factor to ensure that the number of trips derived from the two datasets reflects the travel demand of the same level of population in Coral Gables. The expansion factors for the DTG and TrajSynVAE datasets are 3 and 30, respectively.

Figure 9A) illustrates the road conditions at 9 AM using the trip demand generated by DTG. The speed ratio is defined as the ratio between the average travel speed on a particular link and the speed limit of that link. Trip demand generated by DTG lead to both a morning and evening rush hour, as shown in Fig. 9B), where drastic increases in the fraction of congested edges can be observed between 7 AM and 8 AM, as well as between 4 PM and 6 PM. The ability of DTG to control the departure time distribution as a part of the loss function is well translated to real-world applications in congestion estimation, in which departure time is one of the most crucial factors.

VII. CONCLUSION

In this paper, we address the trajectory reconstruction problem by bridging Generation and Completion. By defining the main task of the model as a self-supervised stay point recovery problem, we seek to retain the original sparse trajectory with the Completion approach. Incorporating Generation, we design mobility-informed regularizers that control the probability distribution of locations for the unobserved stay points. Numeric experiments demonstrate that DeepTimeGeo not only preserves the mobility patterns found in travel surveys in the reconstruction of sparse trajectories, but also generalizes to realistic time-varying urban-scale OD flows. For future research, the model can benefit from more individual-level input representation. Although DTG embeds whether or not a user is a commuter and uses a rank-based representation, the model can still benefit from embedding the activity types in training. Different sequence of activity types can explain a user’s travel motif, thus leading to more individual-level heterogeneity that can be captured by the trained model.

APPENDIX

A. Commuter Detection

The home location is defined as the most frequently visited location between 7 pm and 7 pm for each user. The work location is defined as the most frequently visited location between 7 am and 7 pm on weekdays only. After obtaining a candidate work location, we compute the distance between the home and the work location. To model commutes, we assume that the work location of an individual is more than 500 meters away from the home location. A threshold is also implemented. We conclude the work location of a user if the number of visits to this work location during the day on weekdays over the entire period of observation is more than 24 for the Coral Gables dataset or 8 for the Los Angeles dataset. The difference in the thresholds is due to the difference in the time-span of the two datasets. We label users with a valid work location found as commuters and the rest as non-commuters. While all users have home locations, those labeled as non-commuters do not have work locations.

B. Hyperparameters

The hyperparameters used in training DTG for the Coral Gables and Los Angeles datasets are summarized in Tab. VI.

C. Benchmark Models

We include the following models in the comparative experiment:

- **Semi-Markov** [16]: The Semi-Markov process models the dwell time between consecutive stay points as an exponential distribution. The probability transition matrix among various locations is estimated using Bayesian inference.
- **Hawkes** [53]: Hawkes processes are temporal point processes that model the occurrence of an event as a probability density function that depends on both a background intensity and observations collected in the past time period.

TABLE VI

HYPERPARAMETERS USED IN TRAINING DTG

Dataset	Coral Gables	Los Angeles
Number of Epochs	30	30
Learning Rate	0.0005	0.0005
Batch Size	128	128
N_{heads}	8	8
Dimension of FNN Layers	1024	1024
Number of Encoder Layers	2	2
n_c	4	4
θ_1	10	25
θ_2	0.4	0.1
θ_3	0.15	0.05
d_{loc}	26	24
d_{time}	8	6
d_{poi}	8	6
d_{comm}	8	4

- **SeqGAN** [25]: SeqGAN is a Generative Adversarial Network (GAN) that has been adapted for sequential data. SeqGAN uses Reinforcement Learning (RL) to address the gradient differentiation problem. The generator is updated directly with the policy gradient computed in the RL network.
- **TrajSynVAE** [18]: TrajSynVAE models the stay duration at a location as a temporal point process and uses Variational Auto-Encoder (VAE), in which the difference between the latent representations of the trajectories in the encoder and the decoder is minimized, to address the gradient differentiation problem.
- **LSTM** [28]: This model uses the LSTM network to predict the next location in a trajectory and uses autoregressive generation to reconstruct complete trajectories.
- **MoveSim** [21]: MoveSim uses GAN and RL to update the parameters in the generator. It incorporates regularity in human mobility patterns during the training process.
- **TrajGDM** [13]: TrajGDM is a diffusion model for trajectory reconstruction. Gaussian noises are added at each step during the process of encoding the sparse trajectories.
- **TimeGeo** [14]: TimeGeo is a model-based mechanistic framework for generating complete trajectories. We used the official implementation of the 2016 paper, which is not used in benchmark studies in other trajectory completion papers.

D. Survey Data

1) *American Time Use Survey*: The American Time Use Survey (ATUS) is an annual survey conducted by the U.S. Bureau of Labor Statistics about people’s time use throughout the day [57]. It interviews a demographically representative sample of the U.S. population and details information about each activity participants engage in from 4 a.m. Day 1 to 4 a.m. Day 2, including location types and start/end times. We use ATUS for model training and to calculate departure time distributions $p(t)$ on weekdays and weekends with hourly resolution. A move is defined as a reported change in location type.

2) *National Household Travel Survey*: The National Household Travel Survey (NHTS) is a periodic survey conducted by the U.S. Department of Transportation about people’s daily

TABLE VII
SUMMARY OF SURVEYS

# Users	Temporal Resolution	Temporal Coverage	Spatial Resolution	Spatial Coverage	If Complete
ATUS 2019 (7,779)	1 Minute	1 Day	US	US	Yes
NHTS 2017 (5,534 out of 219,194)	1 Minute	1 Day	Los Angeles-Long Beach-Anaheim	US	Yes
NHTS 2022 (2,126 out of 10,592)	1 Minute	1 Day	South Atlantic	US	Yes

TABLE VIII
SUMMARY OF THE LOSS FUNCTIONS EMPLOYED IN DTG

Name	Purpose	Function	Loss Type	Sources of Labels	Resolution
\mathcal{L}_{sp}	Recover Observed Stay Points	Main Task	Cross-Entropy	Internal	Individual
\mathcal{L}_1	Control PDF of Location Ranks	Regularization	KL-Divergence	Internal	Individual
\mathcal{L}_2	Control PDF of Departure Time		Mean Squared Error	Internal or External	Population
\mathcal{L}_3	Control Commutes		KL-Divergence	Internal or External	Population
\mathcal{L}_{aux}	Exploration Classification	Auxiliary Task	Cross-Entropy	Internal	Individual

travel behavior, with the latest available data from 2017 and 2022 [54]. NHTS details each trip made by an individual between 4 a.m. Day 1 and 4 a.m. Day 2, including trip purpose, visited location type, travel time and distance. We use NHTS for model validation by calculating the ground-truth Daily-Loc, Departure, Duration, and Distance. We use hourly and 30-minute resolutions for Departure and Duration, respectively. Since NHTS documents types but not coordinates of locations, we adopt the following assumption in our calculation: When an individual reports multiple visits to the same location type, we record one home and one work location but take each visit under other types (shopping, recreation, *etc.*) as a unique location. For example, with a daily location chain “home - work - home - shopping - shopping,” we derive four visited locations, “home, work, shopping1, shopping2.” As doing so results in a long-tail distribution of visitation frequency with single-visit non-home/work locations, we do not derive ground truth on location rank from NHTS. A summary of the surveys used can be found in tab. VII. The metrics are calculated at the level of the core base statistical area (CBSA) for Los Angeles using the 2017 NHTS and at the level of the census division for Florida using the 2022 NHTS, due to the smaller survey samples [58].

E. Summary of Loss Terms

Tab. VIII provides a summary of all loss terms. “Sources of labels” represents the source from which the labels are extracted. Labels can be recovered from either the sparse LBS input or obtained from an external source. Ideally, all labels should come from the input dataset; however, due to sparsity, which distorts human mobility patterns, external labels could provide correction. Resolution is the level at which the loss is evaluated. *Individual* means that the loss is performed individually in which both the labels and the prediction are

specific to a particular user. *Population* means that the loss is evaluated after we aggregated the completed trajectories across all users.

ACKNOWLEDGMENT

The authors would like to extend their gratitude to Junzhe Cao for providing constructive comments in the writing of the article. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or U.S. Government.

REFERENCES

- [1] G. Chen, S. Hoteit, A. C. Viana, M. Fiore, and C. Sarraute, “Enriching sparse mobility information in call detail records,” *Comput. Commun.*, vol. 122, pp. 44–58, Jun. 2018. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0140366417309234>
- [2] G. Chen, A. C. Viana, M. Fiore, and C. Sarraute, “Complete trajectory reconstruction from sparse mobile phone data,” *EPJ Data Sci.*, vol. 8, no. 1, p. 30, Dec. 2019. [Online]. Available: <https://epjdatascience.springeropen.com/articles/10.1140/epjds/s13688-019-0206-8>
- [3] X. Kong, Q. Chen, M. Hou, H. Wang, and F. Xia, “Mobility trajectory generation: A survey,” *Artif. Intell. Rev.*, vol. 56, no. S3, pp. 3057–3098, Dec. 2023. [Online]. Available: <https://link.springer.com/10.1007/s10462-023-10598-x>
- [4] J. Wu, S. Powell, Y. Xu, R. Rajagopal, and M. C. Gonzalez, “Planning charging stations for 2050 to support flexible electric vehicle demand considering individual mobility patterns,” *Cell Rep. Sustainability*, vol. 1, no. 1, Jan. 2024, Art. no. 100006. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S294979062300006X>
- [5] Y. Xu, S. Çolak, E. C. Kara, S. J. Moura, and M. C. González, “Planning for electric vehicle needs by coupling charging profiles with urban mobility,” *Nature Energy*, vol. 3, no. 6, pp. 484–493, Apr. 2018. [Online]. Available: <https://www.nature.com/articles/s41560-018-0136-x>

- [6] Y. Liu et al., "Exploiting heterogeneous human mobility patterns for intelligent bus routing," in *Proc. IEEE Int. Conf. Data Mining*, vol. 327, Dec. 2014, pp. 360–369. [Online]. Available: <http://ieeexplore.ieee.org/document/7023353/>
- [7] T. Yabe, N. K. W. Jones, P. S. C. Rao, M. C. Gonzalez, and S. V. Ukkusuri, "Mobile phone location data for disasters: A review from natural hazards and epidemics," 2021, *arXiv:2108.02849*.
- [8] C. N. Mayemba et al., "A short survey of human mobility prediction in epidemic modeling from transformers to LLMs," 2024, *arXiv:2404.16921*.
- [9] J. Wu, S. Cao, G. Perona, and M. C. Gonzalez, "Imitate the right data: City-wide mobility generation with graph learning," in *Proc. 32nd ACM Int. Conf. Adv. Geograph. Inf. Syst.*, Nov. 2024, pp. 609–612, doi: [10.1145/3678717.3691299](https://doi.org/10.1145/3678717.3691299).
- [10] H. Barbosa et al., "Human mobility: Models and applications," *Phys. Rep.*, vol. 734, pp. 1–74, Mar. 2018. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S037015731830022X>
- [11] H. Wang, S. Zeng, Y. Li, P. Zhang, and D. Jin, "Human mobility prediction using sparse trajectory data," *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 10155–10166, Sep. 2020. [Online]. Available: <https://ieeexplore.ieee.org/document/91117072/>
- [12] C. M. Schneider, V. Belik, T. Couronné, Z. Smoreda, and M. C. González, "Unravelling daily human mobility motifs," *J. Roy. Soc. Interface*, vol. 10, no. 84, Jul. 2013, Art. no. 20130246. [Online]. Available: <https://royalsocietypublishing.org/doi/10.1098/rsif.2013.0246>
- [13] C. Chu, H. Zhang, P. Wang, and F. Lu, "Simulating human mobility with a trajectory generation framework based on diffusion model," *Int. J. Geographical Inf. Sci.*, vol. 38, no. 5, pp. 847–878, May 2024. [Online]. Available: <https://www.tandfonline.com/doi/full/10.1080/13658816.2024.2312199>
- [14] S. Jiang, Y. Yang, S. Gupta, D. Veneziano, S. Athavale, and M. C. González, "The TimeGeo modeling framework for urban mobility without travel surveys," *Proc. Nat. Acad. Sci. USA*, vol. 113, no. 37, Sep. 2016. [Online]. Available: <https://www.pnas.org/doi/10.1073/pnas.1524261113>
- [15] M. Luca, G. Barlacchi, B. Lepri, and L. Pappalardo, "A survey on deep learning for human mobility," 2020, *arXiv:2012.02825*.
- [16] L. A. Maglaras and D. Katsaros, "Social clustering of vehicles based on semi-Markov processes," *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 318–332, Jan. 2016. [Online]. Available: <http://ieeexplore.ieee.org/document/7015614/>
- [17] L. Pappalardo and F. Simini, "Data-driven generation of spatio-temporal routines in human mobility," *Data Mining Knowl. Discovery*, vol. 32, no. 3, pp. 787–829, May 2018. [Online]. Available: <http://link.springer.com/10.1007/s10618-017-0548-4>
- [18] H. Wang et al., "Synthesizing human trajectories based on variational point processes," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 4, pp. 1785–1799, Apr. 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/10239531/>
- [19] M. Yin, M. Sheehan, S. Feygin, J.-F. Paiement, and A. Pozdnoukhov, "A generative model of urban activities from cellular data," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 6, pp. 1682–1696, Jun. 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/7932990/>
- [20] Y. Yuan, J. Ding, H. Wang, D. Jin, and Y. Li, "Activity trajectory generation via modeling spatiotemporal dynamics," in *Proc. 28th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2022, pp. 4752–4762. [Online]. Available: <https://dl.acm.org/doi/10.1145/3534678.3542671>
- [21] J. Feng, Z. Yang, F. Xu, H. Yu, M. Wang, and Y. Li, "Learning to simulate human mobility," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 3426–3433. [Online]. Available: <https://dl.acm.org/doi/10.1145/3394486.3412862>
- [22] X. Chen, J. Xu, R. Zhou, W. Chen, J. Fang, and C. Liu, "TrajVAE: A variational AutoEncoder model for trajectory generation," *Neurocomputing*, vol. 428, pp. 332–339, Mar. 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0925231220312017>
- [23] W. Jiang, W. X. Zhao, J. Wang, and J. Jiang, "Continuous trajectory generation based on two-stage GAN," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2023, vol. 37, no. 4, pp. 4374–4382. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/25557>
- [24] X. Liu, H. Chen, and C. Andris, "trajGANs: Using generative adversarial networks for geo-privacy protection of trajectory data (vision paper)," in *Proc. Location Privacy Secur. Workshop (LoPaS) Conjunct. GISci.*, Melbourne, VIC, Australia, 2018, pp. 1–7.
- [25] L. Yu, W. Zhang, J. Wang, and Y. Yu, "SeqGAN: Sequence generative adversarial nets with policy gradient," in *Proc. AAAI Conf. Artif. Intell.*, San Francisco, CA, USA, Feb. 2017, pp. 2852–2858. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/10804>
- [26] Y. Zhan, H. Haddadi, A. Kylo, and A. Mashhadi, "Privacy-aware human mobility prediction via adversarial networks," in *Proc. 2nd Int. Workshop Cyber-Phys.-Hum. Syst. Design Implement. (CPHS)*, May 2022, pp. 7–12. [Online]. Available: <https://ieeexplore.ieee.org/document/9804533/>
- [27] I. Goodfellow et al., "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, Montreal, QC, Canada, 2014, pp. 2672–2680.
- [28] Y. Abu Farha, A. Richard, and J. Gall, "When will you do what?—Anticipating temporal occurrences of activities," 2018, *arXiv:1804.00892*.
- [29] J. Feng et al., "DeepMove: Predicting human mobility with attentional recurrent networks," in *Proc. World Wide Web Conf. World Wide Web*, 2018, pp. 1459–1468. [Online]. Available: <http://dl.acm.org/citation.cfm?doi=10.1145/3178876.3186058>
- [30] A. V. Solatorio, "GeoFormer: Predicting human mobility using generative pre-trained transformer (GPT)," in *Proc. 1st Int. Workshop Human Mobility Predict. Challenge*, Nov. 2023, pp. 11–15.
- [31] Y. Hong, H. Martin, and M. Raubal, "How do you go where?: Improving next location prediction by learning travel mode information using transformers," in *Proc. 30th Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2022, pp. 1–10.
- [32] W. Wang and T. Osaragi, "Learning daily human mobility with a transformer-based model," *ISPRS Int. J. Geo-Inf.*, vol. 13, no. 2, p. 35, Jan. 2024. [Online]. Available: <https://www.mdpi.com/2220-9964/13/2/35>
- [33] Y. Wang, T. Zheng, Y. Liang, S. Liu, and M. Song, "COLA: Cross-city mobility transformer for human trajectory simulation," in *Proc. ACM Web Conf.*, May 2024, pp. 3509–3520. [Online]. Available: <https://dl.acm.org/doi/10.1145/3589334.3645469>
- [34] H. Xue, F. D. Salim, Y. Ren, and N. Oliver, "MobTCast: Leveraging auxiliary trajectory forecasting for human mobility prediction," 2021, *arXiv:2110.01401*.
- [35] M. Li, S. Gao, F. Lu, and H. Zhang, "Reconstruction of human movement trajectories from large-scale low-frequency mobile phone data," *Comput., Environ. Urban Syst.*, vol. 77, Sep. 2019, Art. no. 101346. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0198971519300237>
- [36] S. Hoteit, S. Secci, S. Sobolevsky, C. Ratti, and G. Pujolle, "Estimating human trajectories and hotspots through mobile phone data," *Comput. Netw.*, vol. 64, pp. 296–307, May 2014. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1389128614000656>
- [37] N. Yang and P. S. Yu, "Efficient hidden trajectory reconstruction from sparse data," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2016, pp. 821–830. [Online]. Available: <https://dl.acm.org/doi/10.1145/2983323.2983796>
- [38] J. Ma, C. Yang, S. Mao, J. Zhang, S. C. Periaswamy, and J. Patton, "Human trajectory completion with transformers," in *Proc. ICC*, May 2022, pp. 3346–3351. [Online]. Available: <https://ieeexplore.ieee.org/document/9838743/>
- [39] A. Nawaz, Z. Huang, S. Wang, A. Akbar, H. AlSalman, and A. Gumaiei, "GPS trajectory completion using end-to-end bidirectional convolutional recurrent encoder–decoder architecture with attention mechanism," *Sensors*, vol. 20, no. 18, p. 5143, Sep. 2020. [Online]. Available: <https://www.mdpi.com/1424-8220/20/18/5143>
- [40] X. Fu, Y. Zhang, J. D. D. Ortúzar, and G. Lü, "Activity-travel pattern inference based on multi-source big data," *Transp. Rev.*, vol. 45, no. 1, pp. 26–48, Jan. 2025. [Online]. Available: <https://www.tandfonline.com/doi/full/10.1080/01441647.2024.2400341>
- [41] Q. Long et al., "Practical synthetic human trajectories generation based on variational point processes," in *Proc. 29th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2023, pp. 4561–4571. [Online]. Available: <https://dl.acm.org/doi/10.1145/3580305.3599888>
- [42] A. Haydari, D. Chen, Z. Lai, M. Zhang, and C.-N. Chuah, "MobilityGPT: Enhanced human mobility modeling with a GPT model," 2024, *arXiv:2402.03264*.
- [43] H. Wang, C. Gao, Y. Wu, D. Jin, L. Yao, and Y. Li, "PateGail: A privacy-preserving mobility trajectory generator with imitation learning," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2023, pp. 14539–14547. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/26700>
- [44] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang, "Collaborative location and activity recommendations with GPS history data," in *Proc. 19th Int. Conf. World Wide Web*, Apr. 2010, pp. 1029–1038. [Online]. Available: <https://dl.acm.org/doi/10.1145/1772690.1772795>
- [45] M. C. González, C. A. Hidalgo, and A.-L. Barabási, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, Jun. 2008. [Online]. Available: <https://www.nature.com/articles/nature06958>

- [46] M. Lewis et al., "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," 2019, *arXiv:1910.13461*.
- [47] C. Hawthorne, I. Simon, R. Swavely, E. Manilow, and J. Engel, "Sequence-to-sequence piano transcription with transformers," 2021, *arXiv:2107.09142*.
- [48] J. Weston, E. Dinan, and A. Miller, "Retrieve and refine: Improved sequence generation models for dialogue," in *Proc. EMNLP Workshop SCAI, 2nd Int. Workshop Search-Oriented Conversational AI*, 2018, pp. 87–92. [Online]. Available: <http://aclweb.org/anthology/W18-5713>
- [49] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Long Beach, CA, USA, 2017, pp. 5998–6008.
- [50] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, Feb. 2010. [Online]. Available: <https://www.science.org/doi/10.1126/science.1177170>
- [51] L. Pappalardo, F. Simini, S. Rinzivillo, D. Pedreschi, F. Giannotti, and A.-L. Barabási, "Returners and explorers dichotomy in human mobility," *Nature Commun.*, vol. 6, no. 1, p. 8166, Sep. 2015. [Online]. Available: <https://www.nature.com/articles/ncomms9166>
- [52] L. M. Dery, Y. Dauphin, and D. Grangier, "Auxiliary task update decomposition: The good, the bad and the neutral," 2021, *arXiv:2108.11346*.
- [53] P. Protter, Q. Wu, and S. Yang, "Order book queue hawkes-Markovian modeling," 2021, *arXiv:2107.09629*.
- [54] (Mar. 2018). *National Household Travel Survey 2017*. [Online]. Available: <https://nhts.ornl.gov/documentation>
- [55] (May 2024). *Southeast Florida Regional Planning Model (SERPM)*. [Online]. Available: <https://www.fsutmsonline.net/index.php?modelpages/modD44/index/>
- [56] (Jun. 2018). *H3: Uber's Hexagonal Hierarchical Spatial Index*. [Online]. Available: <https://www.uber.com/blog/h3/>
- [57] (Jun. 2020). *American Time Use Survey 2019*. [Online]. Available: <https://www.bls.gov/tus/>
- [58] (Nov. 2023). *2022 NextGen NHTS Compatibility With Prior Data*. [Online]. Available: <https://nhts.ornl.gov/assets/2022/doc/2022%20NextGen%20NHTS%20Technical%20Release%20Notes%20V1.pdf>



Shangqing Cao received the Bachelor of Science degree in statistics and environmental science from the University of California, Los Angeles, in 2022. He is currently pursuing the Ph.D. degree in transportation engineering with the University of California, Berkeley. His research focuses on generating urban travel demand with human mobility datasets and fleet scheduling and pricing problems for Urban Air Mobility (UAM) operations.

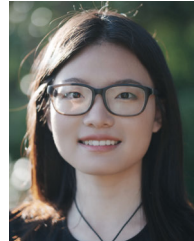


include operation and planning for the power system and transportation system.

Jiaman Wu received the bachelor's degree in geographical monitoring and census from the School of Remote Sensing and Information Engineering, Wuhan University, the master's degree in computer science and technology from the Institute for Interdisciplinary Information Sciences, Tsinghua University, and the Ph.D. degree in systems engineering from the Department of Civil and Environmental Engineering, UC Berkeley. She is currently an Assistant Professor in data science with the City University of Hong Kong. Her research interests



Aparimit Kasliwal received the bachelor's degree in civil engineering from Indian Institute of Technology (IIT) Delhi and the master's degree in systems engineering from the University of California, Berkeley, where he is currently pursuing the Ph.D. degree in systems engineering with the Department of Civil and Environmental Engineering. His research focuses on modeling, simulation, and optimization of urban mobility systems.



Baoqi Chen received B.S. and B.A. degrees (summa cum laude) and the master's degree in city planning from the University of Southern California, Berkeley, where she is pursuing the Ph.D. degree. Her research interests include human mobility behavior and the spatial organization of cities.



Giuseppe Perona received the B.A. degree in computer science from the University of California, Berkeley, in 2024. He is currently pursuing the Ph.D. degree with the Laboratory of Prof. Marta González. His research interests focus on learning human mobility patterns and planning transportation systems.



Marta C. González received the Licentiate degree in physics from the Universidad Simón Bolívar, in 1999, the Magister Sc. degree in physics from the Central University of Venezuela, in 2001, and the Ph.D. (Dr. rer. nat) degree in physics from Stuttgart University in 2006. She is currently a Full Professor of city and regional planning, civil and environmental engineering with the University of California, Berkeley. She is a Physics Research Faculty in energy technology area (ETA) with the Lawrence Berkeley National Laboratory (Berkeley Lab). Prior to joining Berkeley, she was an Associate Professor of civil and environmental engineering with MIT. She has more than 60 publications, and her work has been published in the *Nature*, *Nature Physics*, *Physics A: Statistical Mechanics and its Applications*, *Journal of the Royal Society Interface*, *Physical Review Letters*, *Scientific Reports*, *Transportation Research Part C: Emerging Technologies*, and *Data Mining and Knowledge Discovery*. She is a member of the Operations Research Center and the Center for Advanced Urbanism. She is a member of the scientific council of technology companies, such as Gran Data, PTV, and Pecan Street Project consortium.