A Graph-based Representation Framework for Trajectory Recovery via Spatiotemporal Interval-Informed Seq2Seq

Yaya Zhao\textsuperscript{1}, Kaiqi Zhao\textsuperscript{2}, Zhiqian Chen\textsuperscript{3}, Yuanyuan Zhang\textsuperscript{4}, Yalei Du\textsuperscript{1} and Xiaoling Lu\textsuperscript{1}\textsuperscript{*}

\textsuperscript{1}Center for Applied Statistics, School of Statistics, Renmin University of China
\textsuperscript{2}The University of Auckland
\textsuperscript{3}Mississippi State University
\textsuperscript{4}Beijing Baixingkefu Network Technology Co., Ltd.
\{zhaoyaya, xiaolinglu\}@ruc.edu.cn, kaiqi.zhao@auckland.ac.nz, zchen@cse.msstate.edu, zhang.huanzhiyuan@gmail.com, yaleidu@163.com

Abstract
The prevalent issue in urban trajectory data usage, notably in low-sample rate datasets, revolves around the accuracy of travel time estimations, traffic flow predictions, and trajectory similarity measurements. Conventional methods, often relying on simplistic mixes of static road networks and raw GPS data, fail to adequately integrate both network and trajectory dimensions. Addressing this, the innovative GRFTrajRec framework offers a graph-based solution for trajectory recovery. Its key feature is a trajectory-aware graph representation, enhancing the understanding of trajectory-road network interactions and facilitating the extraction of detailed embedding features for road segments. Additionally, GRFTrajRec’s trajectory representation acutely captures spatiotemporal attributes of trajectory points. Central to this framework is a novel spatiotemporal interval-informed seq2seq model, integrating an attention-enhanced transformer and a feature differences-aware decoder. This model specifically excels in handling spatiotemporal intervals, crucial for restoring missing GPS points in low-sample datasets. Validated through extensive experiments on three large real-life trajectory datasets, GRFTrajRec has proven its efficacy in significantly boosting prediction accuracy and spatial consistency.

1 Introduction
Low-sampling rate datasets present a formidable obstacle when it comes to analyzing and utilizing trajectory data in urban settings. Such limitations often lead to significant inaccuracies and spatial disparities in vital tasks such as estimating travel times [Wang et al., 2022; Zhang et al., 2018], forecasting traffic flow [Li and Zhu, 2021; Lan et al., 2022] and trajectory similarity measurement [Yao et al., 2022; Han et al., 2021]. Therefore, it becomes imperative to develop methods for enhancing the sampling rate by effectively recovering the missing points within a trajectory.

\textsuperscript{*}The Corresponding Author

Figure 1: An example demonstrating the significance of attributing dynamic trajectory information to road network representation.

Recently, there has been a surge in deep learning-based models for trajectory recovery, such as MTrajRec [Ren et al., 2021] and RNTrajRec [Chen et al., 2023]. These methods adopt a sequence-to-sequence [Sutskever et al., 2014] architecture, featuring an encoder model responsible for generating representations of the input trajectory and a decoder model tasked with recovering the trajectory point by point. Representation learning, a deep learning technique that automatically discovers meaningful patterns from raw data, has been widely used to model trajectory data [Jiang et al., 2023] and road networks [Zhang and Zhao, 2021; Fu and Lee, 2020], in a wide range of downstream tasks [Li et al., 2018; Yang et al., 2021]. Furthermore, it plays a vital role in trajectory recovery. For instance, RNTrajRec utilizes road network representation learning and trajectory representation learning within the encoder to effectively capture both temporal and spatial features for each GPS point in the trajectory.

Nevertheless, it is important to note that all existing works still grapple with three significant limitations: (1) Ignorance of dynamic trajectory-road network interactions: Much of the existing research tends to either overlook the road network entirely or solely focus on static road networks, which contain only the fixed topology of the road network and do not account for the dynamic trajectory information traversing it. However, this trajectory-aware road network interplay plays a crucial role in the task of trajectory recovery. For instance, as illustrated in Figure 1, when provided with the road segments where the blue trajectory points are located, it becomes necessary to recover the missing road segments where the purple trajectory points circled in red are situated. In this context, road segments passing through or close to the blue trajectory gain higher consideration for selection compared.
to other road segments farther away from the blue trajectory points. (2) Neglect of extracting the overall and synergistic spatiotemporal trajectory representation: Most studies solely rely on grid information or raw GPS points as input, resulting in extracted trajectory representations that lack comprehensiveness and synergy. (3) Failure of considering the crucial spatiotemporal intervals when employing a seq2seq: Many prior studies have relied on simple seq2seq models, often overlooking the crucial role of spatiotemporal intervals between trajectory points. However, it is essential to recognize that these intervals hold significance. For instance, smaller intervals between the first and second points within a trajectory imply similar embeddings, while larger intervals between the first and last points indicate weaker correlations. Failing to account for these spatiotemporal intervals between points can impede the effective extraction of comprehensive spatiotemporal contextual information among trajectory points, thereby hindering the recovery accuracy.

To overcome the three limitations inherent in existing trajectory recovery methods, we introduce a groundbreaking graph-based representation framework with a spatiotemporal interval-informed seq2seq model, known as GRFTrajRec. GRFTrajRec takes its first stride by harnessing trajectory-aware graph representation, enhancing the understanding of trajectory road network interactions and facilitating the extraction of detailed embedding features for road segments while considering dynamic trajectory information. Furthermore, it employs trajectory representation to capture overall and synergistic spatiotemporal features at each trajectory point. Finally, accounting for spatiotemporal intervals between points, GRFTrajRec utilizes a spatiotemporal interval-informed seq2seq model to effectively integrate and leverage both road and trajectory representations, thereby enabling the precise recovery of missing GPS points. In summary, our key contributions can be outlined as follows:

- We propose a novel framework, namely GRFTrajRec\(^{1}\). To the best of our knowledge, GRNTrajRec is the first attempt to use trajectory-road network interaction for the task of trajectory recovery.
- We introduce a trajectory-aware graph representation for extracting advanced road embedding, enabling a deep understanding of the interplay between trajectories and roads, considering dynamic trajectory information. Additionally, we leverage trajectory representation to capture the spatiotemporal features of each trajectory point.
- We propose a spatiotemporal interval-informed seq2seq model that combines an attention-enhanced transformer and a feature differences-aware decoder, all considering the spatiotemporal intervals between points. This model enhances the integration of road and trajectory representations, extracting comprehensive spatiotemporal contextual information between trajectory points.
- We present extensive experimental results obtained from three real-life datasets conclusively and unequivocally demonstrate that GRFTrajRec outperforms all competitors in both prediction accuracy and spatial consistency.

\(^{1}\)Source codes: https://github.com/zhaoyya1234/GRFTrajRec.

2 Related Work

Road Network Representation Learning. Many existing works consider the road network as a directed graph. Models like Node2vec [Grover and Leskovec, 2016] and DeepWalk [Perozzi et al., 2014] have been introduced to represent road segments as latent embeddings. With the rapid advancement of graph neural networks (GNNs), various graph convolutional networks, such as GCN [Kipf and Welling, 2017], GraphSage [Hamilton et al., 2017], and GAT [Veličković et al., 2018], have proven to be suitable for road network representation. Recent studies have focused on road network representation learning, notably SGMP [Zhang and Zhao, 2021] and Trembr [Fu and Lee, 2020]. SGMP offers a framework for spatial network representation, while Trembr’s Road2Vec model learns embeddings for road segments by understanding the relationships among these segments.

Trajectory Representation Learning. In recent years, trajectory representation learning has garnered widespread attention, as evidenced in [Li et al., 2023; Jarboui and Perchet, 2021]. Prominent models in this field include DeepTTE [Wang et al., 2022], T2vec [Li et al., 2018], ST2vec [Fang et al., 2022], and Start [Jiang et al., 2023]. DeepTTE employs LSTM modules to capture temporal dependencies and generate trajectory representations. T2vec introduces a pioneering deep learning model for trajectory similarity learning, utilizing BiLSTM [Graves and Graves, 2012] to model temporal dependencies. ST2vec [Fang et al., 2022] focuses on encoding both spatial and temporal information within trajectories. Most recently, Start [Jiang et al., 2023] presents a graph-based trajectory representation method, complemented by an innovative spatial network based on GAT [Veličković et al., 2018].

Trajectory Recovery. Various studies, including [Xia et al., 2022; Zhang et al., 2022; Si et al., 2023; Chen et al., 2023], have proposed innovative solutions to address the trajectory recovery challenge. Notably, DHT [Wang et al., 2020] proposes a two-stage solution that first recovers a high-sample trajectory and then uses a map matching algorithm (i.e., HMM [Newson and Krumm, 2009]) to recover the actual GPS locations. Another significant advancement, MTrajRec, as detailed in [Ren et al., 2021], employs a sequence-to-sequence model [Sutskever et al., 2014]. It has outperformed two-stage techniques and is widely adopted in subsequent trajectory recovery research. However, most of these studies overlook the crucial road network structure aspect, which could enhance accuracy. RNTrajRec [Chen et al., 2023] addresses this by introducing GridGNN for road segment feature learning and GPSFormer for detailed trajectory analysis. These components enable a multi-task decoder to efficiently reconstruct missing GPS points using encoder outputs.

3 Preliminaries

Definition 1 (Road Network). A Road Network is modelled as a directed graph \(\mathcal{G} = (\mathcal{V}, \mathcal{E})\), where \(\mathcal{V}\) represents the set of road segments and \(\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}\) captures the connectivity of these road segments.
Definition 2 (Trajectory). A trajectory $\tau$ can be defined as a sequence of GPS positions with timestamps, i.e., $\tau = (p_1, p_2, \cdots, p_n)$, where $p_i = (\text{lat}(i), \text{lng}(i), t(i))$, $\forall i, 1 \leq i \leq n$, which captures the latitude and longitude of the GPS position at timestamp $t(i)$.

Definition 3 (Road Segment Candidate Probability ($RCP^j$)). Given a raw trajectory $\tau = (p_1, p_2, \cdots, p_n)$ and a specified radius $R$, the road segment candidate frequency $N_j^R$ for road segment $v_j \in V$ is defined as follows: $N_j^R = \sum_{i=1}^{n} I[v_j]$ is within a radius of $R$ from $p_i$, where $I[\cdot]$ is the identity function. The standardized $N_j^R$ is defined as the road segment candidate probability $RCP_j^R$, with $\sum_{j=1}^{|V|} RCP_j^R = 1$.

Definition 4 (Map-matched Trajectory Point). Given a trajectory point $p_j$ and a map-matching function $M(\cdot)$, the map-matched trajectory point $a_j$ for $p_j$ is defined as $a_j = M(p_j) = (e_j, r_j, t_j, \text{lat}_{pre}(j), \text{lng}_{pre}(j))$, where $e_j$ is the matched road segment, $r_j$ is the moving ratio (representing the ratio of moving distance over the length of the road segment), and $t_j$ is the timestamp. The coordinates $(\text{lat}_{pre}(j), \text{lng}_{pre}(j))$ are the map-matched latitude and longitude on $e_j$.

Definition 5 (Map-matched $\epsilon$-Sampling Rate Trajectory). A map-matched trajectory $\tilde{\tau}$ with $\epsilon$-sampling rate is a sequence of map-matched trajectory points, i.e., $\tilde{\tau} = (a_1, a_2, \cdots, a_m)$, where $a_j = (e_j, r_j, t(j), \text{lat}_{pre}(j), \text{lng}_{pre}(j))$, $\forall j, 1 \leq j \leq m$ and $a_{j+1}.t(j+1) - a_j.t(j) = \epsilon$. For simplicity, we name $\tilde{\tau}$ as $\epsilon$-MM trajectory, with $\epsilon$ representing the sampling rate.

Definition 6 (Trajectory Recovery). Given a low-sampling-rate trajectory $\tau = (p_1, p_2, \cdots, p_n)$ with corresponding map-matched road segment sequence $\tau^\prime = (M(p_1), M(p_2), \cdots, M(p_n))$ and a target sampling rate $\epsilon$, we aim to recover the real map-matched $\epsilon$-sampling-rate trajectory $\tilde{\tau} = (a_1, a_2, \cdots, a_m)$. This is to say, for each low-sampling-rate trajectory, we will infer its missing points and map match it onto the road network simultaneously.

4 Methodology

In this section, we detail the components of the proposed framework GRFTrajRec for trajectory recovery, depicted in Figure 2. (1) We employ the Figure 2.(a) trajectory-aware graph representation and Figure 2.(b) spatiotemporal trajectory representation to derive an overall representation $Z^\tau$ for the low-sampling trajectory. (2) Using the attention-enhanced transformer encoder described in the Figure 2.(c) spatiotemporal interval-informed seq2seq model with $Z^\tau$, we generate the initial GRU hidden cell $h_{\text{gru}}^{(0)}$. (3) Utilizing the feature differences-aware decoder in the Figure 2.(c) based on $h_{\text{gru}}^{(0)}$, we decode the high-sampling trajectory $\tilde{\tau}$ point by point.

4.1 Trajectory-Aware Graph Representation

In this subsection, we initially analyze the static road network topology to derive a geographical graph representation. Next, we introduce CandiGNN (Candidate Graph Neural Network).
to acquire a candidate road network representation. This enables a more profound and nuanced comprehension of the intricate relationship between trajectories and roads by leveraging dynamic trajectory information. Lastly, we integrate the geographical representation with the dynamic candidate representation to obtain the ultimate trajectory-aware graph representation.

Geographical Representation: GeoGAT. The topology of the road network plays a crucial role in the formation of a trajectory. To capture the structural information, we first utilize Node2Vec [Grover and Leskovec, 2016] to obtain a geographical representation \( \mathbf{g}_i \) for road segment \( v_i \). Next, we feed \( \mathbf{g}_v \) to a graph attention network (GAT [Veličković et al., 2018]) step by step to obtain a smoothed road geographical representation \( \tilde{\mathbf{g}}_i \in \mathbb{R}^{d_{hid}} \).

Candidate Representation: CandiGNN. While GeoGAT effectively captures the geographic characteristics of road segments, it has limitations in extracting dynamic recovery information from the original trajectory sequence. To address this limitation, we introduce road segment candidate probability (RCP) in Definition 3. This probability represents the likelihood that a road segment either passes through a trajectory point or is close to a trajectory point. The higher the RCP value for a road segment, the greater the probability that it will pass through a trajectory point, and subsequently, the greater the probability that it will be selected as the road segment where the missing points are located.

To extract the candidate representation of the graph effectively, we propose a novel graph neural network called CandiGNN. In its message-passing process, CandiGNN leverages candidate probability in measuring the correlations between candidate road segments. First, we take random vector \( \epsilon(l) \in \mathbb{R}^{d_{can}} \) as the initialized candidate representation for node \( v_i \). We use the difference in candidate probabilities to represent the information mobility between road segments, then the attention weight \( \alpha(l)_{ij} \) between nodes \( v_i \) and \( v_j \) in the \( l \)-th update are computed as:

\[
\begin{align*}
D(l)_ij & = (RCP - RCP^*) \ast (c(l)_{ij} - c(l)_{ij}), \\
\epsilon(l)_{i} & = \text{ReLU} \left( D(l)_i W(l)_b \right) W(l)_1, \\
\alpha(l)_{ij} & = \frac{\exp(\text{LeakyReLU}(\epsilon(l)_{ij}))}{\sum_{k \in N_i} \exp(\text{LeakyReLU}(\epsilon(l)_{ij}))},
\end{align*}
\]

where \( c(l)_{ij} \in \mathbb{R}^{d_{can}} \) represent the candidate representations for nodes \( v_i \) and \( v_j \) respectively during the \( l \)-th update of parameters, \( W(l)_b \in \mathbb{R}^{d_{can} \times d_{can}} \), \( W(l)_1 \in \mathbb{R}^{d_{can} \times 1} \) are learnable parameters. The negative input slope of LeakyReLU is set to 0.2 [He et al., 2015]. Then we obtain the road candidate representation \( \tilde{\mathbf{c}}_i \in \mathbb{R}^{d_{can}} \) for node \( v_i \) through combining the features of its neighbors using the attention weights \( \alpha(l)_{ij} \):

\[
\begin{align*}
\mathbf{O}(l)_i & = \epsilon(l)_{i} \mathbf{W}(l)_2 + \sum_{j \in N_i} \alpha(l)_{ij} \epsilon(l)_{j} \mathbf{W}(l)_3, \\
\tilde{\mathbf{c}}_i(l+1) & = \left( \text{ReLU} \left( \mathbf{O}(l)_i \mathbf{W}(l)_4 \right) \right) \mathbf{W}(l)_5,
\end{align*}
\]

where \( \mathbf{W}(l)_2 \in \mathbb{R}^{d_{can} \times d_{mid}}, \mathbf{W}(l)_3 \in \mathbb{R}^{d_{can} \times 1} \in \mathbb{R}^{d_{mid} \times 1} \) and \( \mathbf{W}(l)_5 \in \mathbb{R}^{d_{mid} \times d_{hid}} \) are the learnable parameters.

Graph Fusion Layer. We propose a co-attention fusion module to fuse the geographical representation and candidate representation to get the final trajectory-aware graph representation. Given a raw trajectory \( \tau \) with corresponding map-matched road segment IDs \( \text{src} \), for simplicity, its static geographical representation and dynamic candidate representation acquired by \( \text{src} \) are denoted by \( G^* = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_n) \) and \( C^* = (c_1, c_2, \ldots, c_n) \), where \( G^*, C^* \in \mathbb{R}^{n \times d_{hid}} \). We integrate \( G^*, C^* \) to get the final road segment representation. Specifically, we concatenate \( G^* \) and \( C^* \) to get \( H^* \in \mathbb{R}^{n \times 2 \times d_{hid}} \) and calculate the fusion attention:

\[
H^*W_q(l) \preceq Q^*q(l), \quad H^*W_q(l) \preceq K^*q(l), \quad H^*W_q(l) \preceq V^*q(l),
\]

\[
\text{Attention}(Q^*q(l), K^*q(l), V^*q(l)) = \text{Softmax} \left( \frac{Q^*q(l)K^*q(l)^T}{\sqrt{d}} \right) V^*q(l),
\]

where \( W_q(l), W_k(l), W_v(l) \in \mathbb{R}^{d_{hid} \times d_{hid}} \) are learnable parameters. Denote the fusion result as \( \hat{H}^* \in \mathbb{R}^{n \times 2 \times d_{hid}} \) and we obtain the final output \( \hat{H}_{road} \in \mathbb{R}^{n\times d_{road}} \), where \( d_{road} = 2d_{hid} \), using a feedforward network \( F \) and batch normalization \( \text{Norm} \) as does in [Vaswani et al., 2017]:

\[
\hat{H}_{road} = \text{Reshape}(\text{Norm}(F \hat{H}^* + \hat{H}^*)).
\]

4.2 Spatiotemporal Trajectory Representation

In this subsection, building upon existing approaches relying on grid information or raw GPS points, integrating overall spatial sequences and relative positions in high-sampling trajectories, we present a spatiotemporal trajectory representation meticulously designed to capture overall and synergistic spatiotemporal information inherent in trajectory sequences.

Time-Aware Representation. To capture the temporal information of a trajectory \( \tau \), we first convert its timestamp sequence \( \langle t_1, t_2, \ldots, t_n \rangle \) into a sequence in seconds and normalize each timestamp to the range of \([0, 1]\) using min-max normalization. Denote \( (t_{s_1}, t_{s_2}, \ldots, t_{s_n}) \) as the standardized time sequence. Then, we obtain the time representation of the sequence \( T^* \in \mathbb{R}^{n \times d_{time}} \) via linear projection:

\[
T^* = (t_{s_1}, \ldots, t_{s_n}) \mathbf{W}^*(l),
\]

where \( \mathbf{W}^*(l) \in \mathbb{R}^{1 \times d_{time}} \) are learnable parameters.

Spatial-Aware Representation. To enhance the extraction of spatial information from the raw sequence \( \tau = \langle p_1, p_2, \ldots, p_n \rangle \), we divide the entire road network area into grids identified by coordinates \( (x_i, y_i) \). In this way, each point \( p_i \) is located in a specific grid. We also define \( t_id_i = \left( \frac{t_i - t_0}{\epsilon} \right) \) as the index for the points in the target \( \epsilon \)-MM trajectory \( \hat{\tau} \), where \( \epsilon \) denotes the desired sampling rate and \( t_0 \) is the timestamp of \( i \)-th point in the low-sampling-rate trajectory \( \tau \). Consequently, each raw point \( p_i \) is represented as a triplet \( \langle x_i, y_i, t_id_i \rangle \), \( V_i, 1 \leq i \leq n \).

Initially, we retrieve the spatial grid representation \( \text{Grid}_i \in \mathbb{R}^{1 \times d_{grid}} \) using the grid coordinates \( (x_i, y_i) \):

\[
\text{grid}_i = x_i + \max(x) \times (y_i - 1),
\]

\[
\text{Grid}_i = \text{Embedding} (\text{grid}_i),
\]

Graph Neural Network...
where $\max(x)$ represents the maximum value among all $x$ coordinates within the area, and Embedding refers to the embedding technique in [Mikolov et al., 2013]. Then we use $tid$, to get the spatial positional encoding $POS_i \in \mathbb{R}^{1 \times d_{pos}}$ using the following equation:

$$
POS_i[2j] = \sin \left( \frac{tid_i \times 10000^{2j/d_{pos}}} \right),
$$
$$
POS_i[2j + 1] = \cos \left( \frac{tid_i \times 10000^{2j/d_{pos}}} \right).
$$

In conclusion, note $d_{sp} = d_{grid} + d_{pos} + 3$, the final spatial-aware representation $S^T \in \mathbb{R}^{n \times d_{sp}}$ is obtained by:

$$
S^T = Grid_{x_i,y_i,tid_i,POS_i},
$$
$$
S^T = Concatenate(S^T_i), \quad i = 1, 2, \ldots, n.
$$

4.3 Spatiotemporal Interval-Informed Seq2Seq

In this subsection, we present a spatiotemporal interval-informed seq2seq model that combines an attention-enhanced transformer with a feature-aware decoder. This model leverages spatiotemporal intervals between points to enhance the extraction of contextual information between trajectory points, effectively integrating road and trajectory representations for improved performance.

Attention-Enhanced Transformer Encoder. For trajectory $\tau$, we concatenate its graph representation $H^{r}_{road}$, trajectory representation $H^{t}_{traj}$, road features $RF \in \mathbb{R}^{n \times d_{RF}}$, road segment IDs $src \in \mathbb{R}^{n \times 1}$, and trajectory velocity $speed \in \mathbb{R}^{n \times 1}$, culminating in the formation of the full representation of the trajectory $\tau$, i.e., $GST^\tau \in \mathbb{R}^{n \times (d_{graph} + d_{traj} + d_{RF} + 2)}$. Then we apply a linear layer on $GST^\tau$ and add the position embedding [Vaswani et al., 2017] to obtain the spatial-temporal input $Z^\tau \in \mathbb{R}^{n \times d_{hid}}$ to the transformer encoder. Considering the time interval $\Delta tid = |tid_i - tid_j|$ and geographic distance interval $dis = \text{distance}((lat(i), lng(i)), (lat(j), lng(j)))$ between samples in the trajectory, we propose an enhanced self-attention mechanism to obtain the contextual trajectory representation $\tilde{Z}^\tau = (\tilde{Z}^\tau_1, \tilde{Z}^\tau_2, \ldots, \tilde{Z}^\tau_n) \in \mathbb{R}^{1 \times d_{hid}}$:

$$
\tilde{Z}^\tau - AQ^\tau + Q^\tau, \quad \tilde{Z}^\tau - AK^\tau + K^\tau, \quad \tilde{Z}^\tau - AV^\tau + V^\tau,
$$
$$
\tilde{Z}^\tau = \text{Attention}(Q^\tau, K^\tau, V^\tau),
$$
$$
= \text{Softmax} \left( \frac{Q^\tau K^\tau}{\sqrt{d}} + \alpha f(\Delta tid) + \beta g(dis) \right) V^\tau,
$$

where $A^\tau, K^\tau, V^\tau \in \mathbb{R}^{d_{hid} \times d'}$ are learnable parameters, $d' = d_{hid} \times h_{num}$, $h_{num}$ is the number of attention heads, $\alpha$ and $\beta$ are hyperparameters, $f$ and $g$ are the corresponding probability transformation function.

Feature Differences-Aware Decoder. When decoding the $j$-th point $a^j$ in $\tau$, the spatiotemporal interval between the $(j - 1)$-th predicted map-matched trajectory point $a_{j - 1}$ = $(c(j - 1), r(j - 1), t(j - 1), lat_{pre}^{(j - 1)}, lng_{pre}^{(j - 1)})$ and an input raw point $p_i = (lat(i), lng(i), t(i))$ varies across all i in the range $1 \leq i \leq n$. To take this into account, we define a spatiotemporal feature difference vector $f_i \in \mathbb{R}^{1 \times 3}$, which represents differences in sequence position, spatial location, and road segment candidate probability:

$$
f_i = \text{Concatenation}((tid_i, tid_j, RCP_{M(p_i)}^{\tau}), \text{distance}((lat(i), lng(i), lat_{pre}^{(j - 1)}, lng_{pre}^{(j - 1)}))),
$$

where $RCP_{M(p_i)}^{\tau}$ and $RCP_{M(p_i)}^{\tau}$ are the candidate probabilities for road segment $c(j - 1)$ and matched road segment $M(p_i)$ for $p_i$. We propose a differences-aware decoder that incorporates a GRU model [Cho et al., 2014] with the feature difference vector $f_i$. In the GRU model, the hidden-state vector at timestamp $j$ is denoted as $h^{(j)}_{gru}$, initially set to $h^{(0)}_{gru} = \text{Mean}(\tilde{Z}^\tau) \in \mathbb{R}^{1 \times d_{hid}}$. To capture the correlation between $a_{j - 1}$ and $p_i$, we introduce an attention mechanism $\text{att}(\tau) = \mathbb{R}^{1 \times d_{hid}}$ based on the feature difference $f$:}

$$
u^{(j)}_i = \left( W^{(t)}_{tf} \left( \text{tanh} \left( h^{(j-1)}_{gru} \| \tilde{Z}^\tau_i \| W^{(l)}_{attn} \right) \right) \right),
$$
$$
\mu^{(j)}_i = \exp \left( \mu^{(j)}_i \right) / \sum_{i=1}^{n} \exp \left( \mu^{(j)}_i \right),
$$
$$
\text{att}(\tau) = \sum_{i=1}^{n} \alpha^{(j)}_i \tilde{Z}^\tau_i,
$$

where $W^{(t)}_{tf} \in \mathbb{R}^{d_{hid} \times 1}$, $W^{(l)}_{attn} \in \mathbb{R}^{(d_{hid} + 3) \times d_{hid}}$ are learnable weights. The hidden-state vectors $h^{(j)}_{gru}$ by:

$$
h^{(j)}_{gru} = \text{GRU} \left( \left( p^{(j-1)} \| r^{(j-1)} \| R^{(j-1)} \| \text{att}(\tau) \right) \right),
$$

where $j \in [1, 2, \ldots, m]$, $p^{(j-1)} \in \mathbb{R}^{1 \times d_{hid}}$ is the embedding for the predicted road segment $c(j - 1)$, $r^{(j-1)} \in \mathbb{R}^{1 \times 1}$ is moving ratio at the $(j - 1)$-th timestamp, $R^{(j-1)} \in \mathbb{R}^{1 \times d_{RF}}$ is corresponding road features, $\text{att}(\tau)$ is the attention value.

4.4 Training

We employ a multi-task loss for training the model. Specifically, we consider predicting both the road segment ID and the moving ratio. In accordance with MTrajRec [Ren et al., 2021], we employ the cross-entropy loss $L_1(\theta)$ for the road segment ID prediction task and mean squared error (MSE) loss $L_2(\theta)$ for the moving ratio prediction task:

$$
L_1(\theta) = -\sum_{j=1}^{m} \sum_{j=1}^{V} 1 \{ a_{j, e} = e \} \log \left( P_{\theta} \left( a_{j,e} = e \mid h^{(j)}_{gru} \right) \right),
$$
$$
L_2(\theta) = \sum_{j=1}^{m} \sum_{j=1}^{V} 1 \{ a_{j, r} = r \} \log \left( P_{\theta} \left( h^{(j)}_{gru} \mid h^{(j)}_{gru} \right) \right),
$$

where $m$ is the length $\tilde{r}$, $|V|$ is the size of road segments, $a_{j, e}$ is the ground truth of road segment ID, $a_{j, r}$ is the prediction, $D$ means the testing dataset consisting of low-sampling-rate trajectories and e-MM trajectories, $a_{j, r}$ is the ground truth of the real moving ratio, $P(\tilde{r})$ represents the road segment embedding for the predicted road segment at the $j$-th timestamp, $P_{\theta}$ and $R_{\theta}$ represents the neural network for predicting road segments and the moving ratio. Overall, the final
Table 1: Dataset statistics.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>road network latitude range</th>
<th>road network longitude range</th>
<th>road segments</th>
<th>trajectories</th>
<th>sample ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>NanJing</td>
<td>[32.115211,32.215211]</td>
<td>[118.69454,118.84454]</td>
<td>3582</td>
<td>10267</td>
<td>15</td>
</tr>
<tr>
<td>YanCheng</td>
<td>[33.1696824,33.4401188]</td>
<td>[120.1070088,120.3540647]</td>
<td>7061</td>
<td>95420</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 2: Performance evaluation for different methods in trajectory recovery (random missing).

<table>
<thead>
<tr>
<th>Method</th>
<th>Porto (missing rate = 7/8)</th>
<th>YanCheng (missing rate = 7/8)</th>
<th>YanCheng (missing rate = 15/16)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy</td>
<td>Recall</td>
<td>Precision</td>
</tr>
<tr>
<td>HMM+linear</td>
<td>0.3045</td>
<td>0.4071</td>
<td>0.4195</td>
</tr>
<tr>
<td>T2vecRec</td>
<td>0.5085</td>
<td>0.6047</td>
<td>0.7553</td>
</tr>
<tr>
<td>ST2vecRec</td>
<td>0.5413</td>
<td>0.6423</td>
<td>0.7600</td>
</tr>
<tr>
<td>MTrajRec</td>
<td>0.5688</td>
<td>0.6620</td>
<td>0.7820</td>
</tr>
<tr>
<td>RNTrajRec</td>
<td>0.5584</td>
<td>0.6521</td>
<td>0.7377</td>
</tr>
<tr>
<td>StartRec</td>
<td>0.5635</td>
<td>0.6554</td>
<td>0.7371</td>
</tr>
<tr>
<td>GRTFrajRec</td>
<td>0.6143</td>
<td>0.6993</td>
<td>0.8044</td>
</tr>
</tbody>
</table>

5 Evaluation

5.1 Experimental Setup

Datasets. We validate the effectiveness of our model on three real-world trajectory datasets along with their corresponding road network sources from OpenStreetMap. The Porto dataset is openly available, while the YanCheng and NanJing datasets are provided by a company partner. Detailed statistics of these datasets are presented in Table 1. For each dataset, we divided the data into training, validation, and testing sets, following a 7:2:1 split ratio. To obtain low-sample trajectories, we set three distinct data missing types: random missing (where the missing trajectory points are scattered randomly), uniform missing (where the missing points are uniformly distributed across the entire trajectory), and block missing (where the missing points form a continuous sub-trajectory segment within the trajectory). Additionally, our experiments adopt two missing rates: 7/8 or 15/16 in line with the rates used in RNTrajRec [Chen et al., 2023].

Baselines. To evaluate the effectiveness of GRTFrajRec, we implement in total seven baselines. i) HMM + Linear [Hoteit et al., 2014] uses linear interpolation to obtain a high sample trajectory, and then uses HMM algorithm to obtain a map-matched ϵ-sampling rate trajectory. ii) t2vec [Li et al., 2018] proposes a deep learning network for trajectory similarity learning with a BiLSTM [Graves and Graves, 2012] model. iii) ST2vec [Fang et al., 2022] encodes spatial and temporal information of trajectories. iv) Transformer [Vaswani et al., 2017] learns the representation with temporal dependency. v) MTrajRec [Ren et al., 2021] solve the map-constrained trajectory recovery problem via seq2seq multi-task learning. vi) RNTrajRec [Chen et al., 2023] is the state-of-the-art method for trajectory recovery task. vii) Start [Jiang et al., 2023] is the state-of-the-art method for trajectory representation in spatial networks which is applicable for many trajectory-based downstream tasks.

Parameter Settings. We implemented all the baseline models on a machine equipped with an Intel(R) Xeon(R) 32-core CPU E5-2620 v4 @ 2.10GHz and a 12GB NVIDIA TITAN V GPU. All models were trained using the Adam optimizer for at most 50 epochs, with a batch size of 128 and a learning rate of 1e-3. For the Porto dataset, we configured the hidden-state size hyperparameter d to be 512. For the YanCheng and NanJing datasets, we set it to 128. We explore (alpha ∈ [1,10,100,1000], beta ∈ [1,10,100,1000], lambda ∈ [10,20,30,40,50]). The optimal model parameters are Porto (100, 100, 10), YanCheng (1, 1, 10) and NanJing (1, 1, 10) while the worst model parameters are Porto (1, 1, 20), YanCheng (321, 1000, 20) and NanJing (1000,1000,20).

Metrics. Following RNTrajRec [Chen et al., 2023], we adopt the accuracy, recall, precision, f1 score of the road segments recovered and the RN MAE, RN RMSE of distance on road network, as well as MAE and RMSE of the euclidean distance between predicted and actual points to evaluate the performances of different models.

objective function is a weighted sum of the two functions:

\[ L = L_1(\lambda) + \lambda L_2(\lambda). \]
5.2 Overall Performance

We conducted a comprehensive performance evaluation in the scenario of random missing data using the Porto and YanCheng datasets, as summarized in Table 2. The figures highlighted in bold signify the top-performing models. Notably, \textbf{GRFTrajRec} consistently outperforms the baseline models across all datasets, demonstrating significant advantages across most metrics. Meanwhile, Linear+HMM consistently underperforms across all datasets. Notably, both TFreqRec and MRec demonstrate superior performance on ID prediction metrics (accuracy, f1score), showcasing their ability to capture spatiotemporal information in low-sample trajectories. However, they fall short in segment prediction error metrics (RN\_MAE, RN\_RMSE) due to their lack of consideration for the road network’s topological aspects. In contrast, RNTrajRec and StartRec incorporate the topological characteristics of the road network, resulting in improved performance in segment prediction error metrics. Remarkably, their ID prediction performance is on par with the previous two models. Our framework, \textbf{GRFTrajRec}, emerges as the standout performer, excelling in both ID prediction metrics and road network segment prediction error metrics.

5.3 Further Experiments

We extend our experimentation to include the Nanjing dataset and introduce two additional missing types: uniform missing and block missing. This broader scope allows us to assess the generalizability and robustness of our model across various scenarios and datasets with differing distributions and complexities. The results, as depicted in Figure 3 alongside Table 2, consistently reaffirm the superior performance of \textbf{GRFTrajRec} across all missing types and datasets. This robust performance underscores the adaptability, versatility, and reliability of GRFTrajRec in diverse real-world settings. These findings bolster confidence in the effectiveness of GRFTrajRec and highlight its potential for real-world applications across different contexts and real data distributions.

5.4 Ablation Study

To further verify the effectiveness of different modules in our model, we have devised five variants of \textbf{GRFTrajRec}: i) \textbf{GRF-ATF} removes the spatiotemporal interval consideration in the attention-enhanced transformer encoder and uses the traditional transformer. ii) \textbf{GRF-FD} removes the spatiotemporal interval consideration in the feature differences-aware decoder and uses the traditional GRU. iii) \textbf{GRF-GR} removes the trajectory-aware graph representation. iv) \textbf{GRF-CG+} removes CandiGNN but add the traditional GCN [Kipf and Welling, 2017] to the trajectory-aware graph representation. v) \textbf{GRF-TR} removes the spatiotemporal trajectory representation. The experimental results are detailed in Table 3. It’s evident from the data that \textbf{GRFTrajRec} consistently outperforms all its variants across the majority of settings, highlighting the importance and effectiveness of these modules.

As discussed in subsection 4.1, the importance of trajectory-aware graph representation (GR) becomes evident in extracting advanced road embeddings. Removing GR from \textbf{GRFTrajRec} leads to a significant decline in overall performance. Notably, \textbf{GRFTrajRec} consistently outperforms \textbf{GRF-CG+}, only when replacing CandiGNN with the traditional GCN, highlighting the necessity of a dedicated graph neural network for trajectory recovery. Moreover, the omission of the spatiotemporal interval consideration in the feature differences-aware decoder results in a notable drop of over 0.4 across all datasets. This highlights the effectiveness of considering spatiotemporal interval in the decoder, reaffirming its importance in enhancing recovery accuracy. Across various metrics, \textbf{GRFTrajRec} consistently outperforms both \textbf{GRF-ATF} and \textbf{GRF-TR}, demonstrating the contributions of the attention-enhanced transformer and spatiotemporal representation learning in improving recovery accuracy. Overall, the trajectory-aware graph representation and the feature differences-aware decoder emerge as the two most crucial components within \textbf{GRFTrajRec}, underscoring their indispensable roles in achieving superior recovery performance.

Table 3: Performance evaluation of ablation experiments (random missing).

<table>
<thead>
<tr>
<th>Method</th>
<th>Porto (missing rate = 7/8)</th>
<th>YanCheng (missing rate = 7/8)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy</td>
<td>Recall</td>
</tr>
<tr>
<td>GRF-ATF</td>
<td>0.6033</td>
<td>0.6904</td>
</tr>
<tr>
<td>GRF-FD</td>
<td>0.5726</td>
<td>0.6721</td>
</tr>
<tr>
<td>GRF-GR</td>
<td>0.5866</td>
<td>0.6752</td>
</tr>
<tr>
<td>GRF-CG+</td>
<td>0.5713</td>
<td>0.6700</td>
</tr>
<tr>
<td>GRF-TR</td>
<td>0.6047</td>
<td>0.6900</td>
</tr>
<tr>
<td>GRFTrajRec</td>
<td>0.6413</td>
<td>0.6993</td>
</tr>
</tbody>
</table>

Figure 3: Further performance results for different methods (b: block missing, u: uniform missing, r: random missing, missing rate = 7/8).
5.5 Case Study

We conduct the visualization comparison experiment on the trajectory recovery task. The input is a low-sampling trajectory, while the ground truth is the actual trajectory at a higher sampling rate. We provide the trajectory recovery results of our model, GRFTrajRec, as well as two of the current best-performing models, RNTrajRec and StartRec.

As illustrated in Figure 4, while all three models can recover the approximate high-sampling trajectory, our model’s trajectory recovery closely aligns with the ground truth. In contrast, RNTrajRec and StartRec exhibit noticeable deviations from the ground truth, particularly at the trajectories’ ends. Our model’s precision in achieving seamless recovery from start to finish is bolstered by its holistic consideration of the interaction between road networks and trajectories, along with the spatiotemporal intervals between trajectory points. The dynamic interaction effectively incorporates missing information into the representation of low-sampling trajectories. Additionally, the spatiotemporal intervals between points facilitate a nuanced understanding of the specific spatiotemporal relationships among missing points and other data points. This comprehensive analysis enhances our ability to accurately pinpoint the spatiotemporal positions of each missing point, reducing overall recovery errors and notably enhancing trajectory recovery accuracy.

5.6 Efficiency Study

In addition to evaluating method effectiveness, we also assess efficiency from two perspectives: inference time for recovering a trajectory and the number of model parameters.

As shown in Figure 5, when compared to RNTrajRec and StartRec, which share similar inference times and model parameters, GRFTrajRec demonstrates superior performance in both accuracy (ACC) and mean absolute error (MAE) metrics. Many other models, despite having relatively fewer parameters or shorter runtimes, often compromise accuracy (ACC) or substantially increase mean absolute error (MAE).

Our model’s longer inference time is due to the consideration of spatiotemporal intervals in seq2seq modelling, which significantly improves prediction accuracy (ACC) and reduces prediction error (MAE), as revealed in our ablation experiments in subsection 5.4. Thus, we believe that the little increase in the inference time is a worthwhile trade-off, considering it leads to enhanced ACC and reduced MAE.

5.7 Assessment of Trajectory Recovery Necessity

Assessing the necessity of trajectory recovery is paramount across various tasks, scenarios and different data densities, to effectively optimize predictive models based on trajectory data and ultimately save costs. For instance, by evaluating the impact of trajectory data of varying densities on model performance, we can determine when trajectory recovery is advantageous. We predict driver IDs using trajectory representation from the model in subsection 4.2. We employ 10,000 trajectories (sample one point per second), adjusting the missing rate to control trajectory data density. We observe a significant drop in model performance when the missing rate exceeds 0.75 (sample 0.25 points per second). (1) In scenarios with several points sampled per second below 0.25 due to data collection limitations, indicating excessively sparse sampling, trajectory recovery may enhance model performance. (2) Conversely, if the number of points sampled per second is above 0.25, and the data density is relatively high, model optimization takes precedence over trajectory recovery.

6 Discussion and Conclusion

In this paper, we introduce a new graph-based framework for trajectory recovery with spatiotemporal interval-informed seq2seq. GRFTrajRec. Experiments demonstrate that GRFTrajRec outperforms current methods by better integrating trajectory-road network interactions and considering spatiotemporal intervals. For future work, we aim to focus on advancing trajectory-network interaction analysis beyond only using road segment candidate probability modelling and explore calculating spatiotemporal intervals for selected points rather than all to speed up the computational speed.
Acknowledgments
This work is supported by the MOE Project of Key Research Institute of Humanities and Social Sciences (No.22JD110001), the National Natural Science Foundation of China (No.72171229) and the Outstanding Innovative Talents Cultivation Funded Programs 2023 of Renmin University of China. This work is also partly supported by the Marsden Fund Council from Government funding (MFP-UOA2123) administered by the Royal Society of New Zealand, and NSF IIS award #2153369.

References


