Towards Efficient and Effective Unimodal Trajectory Representation Learning: A Simple Yet Powerful Approach

Shaoxuan Gu[2] , **Xingyu Zhao**[2] , **Dongliang Chen**[2] , **Yuan Cao**[2] , **Yanwei Yu**[1,2*]

[1]State Key Laboratory of Physical Oceanography, Ocean University of China
[2]Faculty of Information Science and Engineering, Ocean University of China
{gushaoxuan, zhaoxingyu}@stu.ouc.edu.cn, {chendongliang, cy8661, yuyanwei}@ouc.edu.cn,

Abstract

Trajectory representation learning transforms trajectory data into low-dimensional embeddings for downstream analytics. Although trajectory data inherently contains rich spatiotemporal information that remains to be more deeply explored, recent approaches have increasingly favored integrating external multimodal information rather than deeply mining internal trajectory patterns, resulting in the underutilization of the significant potential inherent in spatiotemporal features. To address these issues, we propose **SimTRL**, a novel unimodal framework specifically designed for road network-based trajectory data. Specifically, we design a bi-directional mamba-traj encoder that leverages the linear computational efficiency of State Space Models to capture global non-causal topological dependencies, overcoming the limitations of traditional causal models. Furthermore, we introduce a context-aware time encoder mechanism to explicitly model dynamic time embeddings. Extensive experiments conducted on two real-world datasets demonstrate that our model achieves average performance gains of 5.19%, 13.91%, 7.2%, and 2.69% across four tasks, respectively. Compared to multimodal models, our approach achieves an average improvement of 13.39% across all evaluation metrics, with a pre-training speedup of 11.2×. The source code of our model is available at https://github.com/sxgu1/SimTRL.

1 Introduction

With the rapid advancement of GPS technology and the widespread adoption of mobile internet, massive volumes of vehicle trajectory data [Chen *et al.*, 2015; Ding *et al.*, 2018] are continuously generated within urban systems. Such data often contain rich mobility patterns and complex traffic state information. Today, the analysis and management of trajectory data play a critical role in various fields, including urban planning [Chen *et al.*, 2018; Chen *et al.*, 2022], intelligent transportation systems [Feng *et al.*, 2023; Wang *et al.*, 2018], and trajectory matching [Chen *et al.*, 2024; Zhang *et al.*, 2025]. Trajectory representation learning aims to transform raw discrete trajectory sequences into general-purpose low-dimensional vector representations. High-quality trajectory embeddings can support diverse downstream tasks and are essential for building intelligent transportation systems.

In recent years, mainstream research has primarily adopted Transformer architectures [Vaswani *et al.*, 2017] and Graph Neural Networks (GNNs) [Scarselli *et al.*, 2008] to capture spatio-temporal correlations. For instance, Toast [Chen *et al.*, 2021] and Trembr [Fu and Lee, 2020] leverage self-attention mechanisms to extract multi-scale spatio-temporal patterns from trajectory sequences [Chen *et al.*, 2021; Fu and Lee, 2020]. Building upon these studies, methods such as START [Jiang *et al.*, 2023] and TrajRL [Xia *et al.*, 2025] further incorporate contrastive learning to distinguish spatio-temporal features [Jiang *et al.*, 2023; Xia *et al.*, 2025]. Recently, inspired by the success of large language models (LLMs) and multimodal learning, current research has increasingly tended to introduce multimodal information. For example, studies like TrajCogn [Zhou *et al.*, 2025] and MM-Path [Xu *et al.*, 2025] integrate auxiliary modalities such as textual descriptions and remote sensing imagery into their models, aiming to improve semantic richness based on these inputs.

Despite these significant advances, existing methods still face challenges in two critical aspects: *First, reliance on multimodal data can lead to information redundancy and computational complexity.* Current research tends to integrate multimodal information such as images and texts to enrich trajectory representations. Although such external resources contain rich semantics, they also exhibit clear drawbacks. High-quality multimodal data is often difficult to acquire and process, while deep fusion of multimodal inputs imposes heavier computational burdens and leads to more complex model architectures. Furthermore, the semantic gap between multimodal data and road topology frequently introduces unavoidable hidden noise. Consequently, such methods remain insufficiently mature for application in large-scale real-time urban computing scenarios.

Second, it is challenging to balance computational efficiency with comprehensive spatiotemporal modeling capability. Existing research predominantly adopts Transformer architectures as the primary encoder. While Transformer-based

*Corresponding author: Yanwei Yu.

architectures can capture global dependencies, their quadratic computational complexity becomes a bottleneck when processing long sequences. Although current Mamba [Gu and Dao, 2023] architectures offer linear complexity, since standard SSMs are inherently causal, this architecture fails to capture the non-causal topological dependencies intrinsic to road networks. However, trajectory data exhibits strong topological dependencies in space, simple unidirectional causal scanning makes it difficult to capture the potential spatial information within the trajectory. Meanwhile, how to effectively integrate heterogeneous temporal signals (*e.g.*, continuous intervals and periodic patterns) into a linear architecture is also a problem that needs to be solved.

To address these challenges, we propose a simple yet efficient trajectory representation learning framework named SimTRL. Specifically, our approach does not rely on complex multimodal data and models only road network trajectories, aiming to demonstrate that effective trajectory representations do not require intricate multimodal inputs. To integrate heterogeneous temporal signals, we design a context-aware time encoder that dynamically fuses continuous time intervals with periodic contexts through a gating unit. Furthermore, to balance global spatio-temporal modeling with learning efficiency, we design a bi-directional mamba-traj encoder and a hybrid decoder, which captures global bidirectional dependencies via bidirectional state space models while incorporating an attention mechanism at the decoder layer for mask reconstruction. Experiments on two real-world datasets show that our model outperforms existing baseline methods across multiple downstream tasks while achieving superior training efficiency. In summary, the main contributions of this study are as follows:

- We propose a novel SimTRL framework that challenges the prevailing trend of over-relying on multimodal approaches. Our model learns general-purpose representations suitable for various downstream tasks by deeply mining the latent spatiotemporal information embedded within trajectories, aiming to demonstrate that a lightweight model based solely on road-network trajectories can outperform complex multimodal methods.
- We design a bi-directional mamba-traj encoder with a context-aware time encoder. This design aims to dynamically fuse spatiotemporal information and leverages a bi-directional mamba model with linear complexity to precisely model spatiotemporal patterns, ensuring robust extraction and representation of spatiotemporal features.
- We conduct extensive evaluations on two real-world datasets. Experimental results show that our model achieves SOTA performance across four downstream tasks while achieving an average training speed improvement of 4.4× compared to the fastest baseline across both datasets, our model also achieves maximum performance improvements of 6.88%, 41.81%, 11.96%, and 3.41% across the four tasks respectively.

2 Related Work

Unimodal Trajectory Representation Learning Trajectory representation learning aims to encode raw data into general-purpose latent vectors, offering superior universality over task-specific predictive models. Early research employed RNN-based Autoencoders [Li *et al.*, 2018; Liu *et al.*, 2020; Yao *et al.*, 2017] to reconstruct trajectory features. Recently, Transformer architectures have become dominant; methods like CTLE [Lin *et al.*, 2021] and Toast [Chen *et al.*, 2021] leverage Masked Language Modeling (MLM), treating trajectory points as tokens to capture long-range dependencies effectively. To further incorporate road network topology, Graph Neural Networks (GNNs) have been introduced; for instance, JCLRNT [Mao *et al.*, 2022] utilizes Graph Attention Networks to model topological relations, while Trembr [Fu and Lee, 2020] captures complex road segment associations via multi-task learning. Furthermore, to extract high-level semantics such as travel purposes, strategies like contrastive learning are adopted by PIM [Yang *et al.*, 2021], TrajCL [Chang *et al.*, 2023], and MMTEC [Lin *et al.*, 2023], while hybrid approaches like START [Jiang *et al.*, 2023] and LightPath [Yang *et al.*, 2023] enhance representation through synergistic reconstruction and rationale reasoning. JGRM [Ma *et al.*, 2024] is the first to jointly train the road trajectory view and the raw trajectory view. GTR [Wang *et al.*, 2025] adopts a multi-view perspective to explicitly fuse the Free Space View and Road Network View. TrajRL [Xia *et al.*, 2025] and TrajAgg [Zhang *et al.*, 2026] research has advanced towards explicitly mining multi-scale information within sequences. And MetaTRL [Yu *et al.*, 2026] further extends cross-city trajectory representation research using meta-learning.

Multi-Modal and LLM Trajectory Representation. Driven by the advancements in Multi-Modal Learning [Bao *et al.*, 2022; Menon and Vondrick, 2023] and Large Language Models (LLMs) [Zhou *et al.*, 2023; Jin *et al.*, 2024], recent studies have increasingly incorporated external modalities to augment trajectory representations. TrajCogn [Zhou *et al.*, 2025] explores a cross-modal paradigm by translating trajectories into linguistic descriptions and leveraging LLMs to derive embeddings. Similarly, TrajMamba [Liu *et al.*, 2026] converts POI and road attributes into text embeddings, employing a dual-view architecture to model both GPS and Road perspectives. Visual modalities are also widely integrated; UniTR [Zhao *et al.*, 2025] introduces street-view imagery atop complex graph interactions, while MM-Path [Xu *et al.*, 2025] incorporates remote sensing images to enrich the textural features of trajectories.

Overall, acquiring and processing multimodal data is often challenging, and coarse-grained fusion of multimodal inputs tends to introduce noise due to semantic gaps while imposing significant computational burdens. Therefore, this paper proposes a minimalist, highly efficient model focused solely on the road view.

3 Preliminaries

In this section, we formally define the road network, raw GPS trajectories, map-matched trajectories, and the problem statement.

Definition 1 (Road Network). A road network is represented as a directed graph $G = (V, E)$, where V represents the set

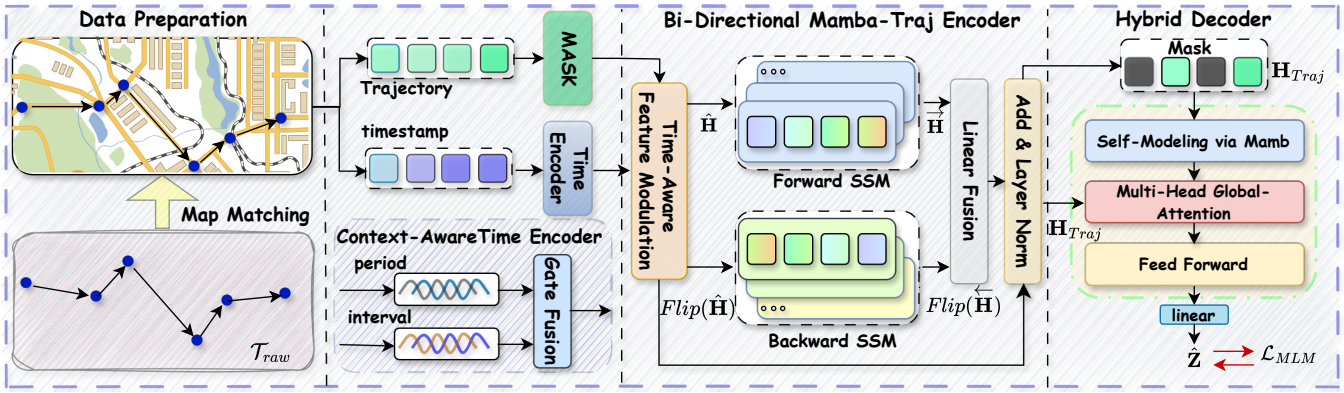

Figure 1: Overall framework of the proposed SimTRL.

of road intersections or endpoints, and E represents the set of road segments connecting these nodes.

Definition 2 (Raw GPS Trajectory). A raw GPS trajectory $\mathcal{T}_{raw} = \langle p_1, p_2, \ldots, p_n \rangle$ is a chronologically ordered sequence of spatio-temporal sample points, where each sample point $p_i = (lat_i, lon_i, t_i)$ is a triple containing latitude, longitude, and the sampling timestamp.

Definition 3 (Map-matched Trajectory). Through a map matching algorithm, we convert a raw trajectory $\mathcal{T}_{raw}$ into a road network-constrained sequence. Formally, a map-matched trajectory $\mathcal{T}$ is defined as a sequence of spatiotemporal tuples:

$$\mathcal{T} = \langle (v_1, t_1), (v_2, t_2), \ldots, (v_m, t_m) \rangle, \tag{1}$$

where $v_i \in V$ denotes the i-th road node visited by the object, and t_i is the corresponding timestamp.

Problem Statement. In this study, we primarily focus on the Map-matched Trajectory defined above. For the sake of brevity, we use the term "trajectory" to refer to the map-matched trajectory hereafter.

Next, we formally define the core problem of this study—Trajectory Representation Learning:

Problem 1. Given a road network G and a trajectory dataset $\mathcal{D} = \{\mathcal{T}_i\}_{i=1}^{|\mathcal{D}|}$, the goal of trajectory representation learning is to learn a general low-dimensional vector representation $\mathbf{h}_i \in \mathbb{R}^d$ for each trajectory $\mathcal{T}_i \in \mathcal{D}$. This vector is required to preserve the spatiotemporal correlations intrinsic to the trajectory.

In this work, our objective is to obtain robust and versatile representations for each trajectory, enabling them to achieve optimal performance across multiple downstream tasks.

4 Methodology

This section introduces the methodology of our unimodal trajectory representation learning framework, SimTRL. As illustrated in Figure 1, our architecture is designed to transform raw trajectory sequences into embedded representations.

4.1 Context-Aware Time Encoder

In urban traffic scenarios, trajectory movement patterns not only represent spatial transitions but also reveal underlying temporal regularities. To deeply mine the temporal information inherent in trajectories, we design a context-aware time encoder that explicitly performs fine-grained encoding of both periodic information and inter-node time intervals. To simultaneously capture both types of signals, we propose a temporal encoder consisting of two components: *periodic semantic extraction* and *time interval encoding*.

Given raw timestamps $\{t_i\}_{i=1}^N$ and interval values $\{\Delta t_i\}_{i=1}^N$, where N is the sequence length, the resulting temporal embedding is denoted as $T_{t_i} \in \mathbb{R}^d$, which is obtained by fusing the two branches.

Periodic Semantic Extraction

First, we design a periodic semantic extraction method to capture the temporal periodic patterns of urban traffic from trajectory time series. We extract the day of the week, the hour of the day, and the minute of the hour separately. The day of the week is treated as discrete categorical information and transformed into a discrete embedding $t_d^{(i)} \in \mathbb{R}^d$. For the fine-grained hour and minute information, we encode them as continuous sinusoidal features (sine and cosine values), concatenate them, and then apply a linear transformation to obtain the daily periodic feature $t_{cyc}^{(i)} \in \mathbb{R}^d$. The periodic temporal context $T_{t_i}^P \in \mathbb{R}^d$ is formulated as:

$$T_{t_i}^P = t_d^{(i)} + \text{Linear}(t_{cyc}^{(i)}). \tag{2}$$

Log-Fourier Interval Encoding

Furthermore, the time interval Δt_i between nodes reflects instantaneous traffic fluctuations and typically follows a long-tailed continuous distribution. To explicitly encode the time interval, we adopt Log-Fourier Interval Encoding. We compress its dynamic range using a logarithmic transformation:

$$\delta_i' = \log(1 + \Delta t_i). \tag{3}$$

Subsequently, we lift it into a multi-frequency Fourier feature space:

$$\Phi_{\text{freq}}(\delta_i') = [\cos(2\pi \mathbf{W}_f \delta_i'),\ \sin(2\pi \mathbf{W}_f \delta_i')], \tag{4}$$

where $\mathbf{W}_f \in \mathbb{R}^k$ is a learnable frequency vector. Through this mapping, we obtain the raw interval feature vector. Finally, we pass the frequency features through an MLP (mapping $\mathbb{R}^{2k} \to \mathbb{R}^d$) to obtain the interval embedding $T_{t_i}^I \in \mathbb{R}^d$.

Context-Aware Fusion

Intuitively, the time intervals between nodes often exhibit a high dependence on the periodic context. To model this dependency, we design a context-aware gating mechanism. We dynamically generate a gating vector $\mathcal{G}_i \in \mathbb{R}^d$ using the periodic context information to adaptively regulate the importance of the interval information:

$$\mathcal{G}_i = \sigma(\mathbf{W}_g T^P_{t_i} + b_g), \tag{5}$$

where $\sigma(\cdot)$ is the Sigmoid function, and $\mathbf{W}_g \in \mathbb{R}^{d\times d}$, $b_g \in \mathbb{R}^d$ are learnable parameters. Finally, we fuse the two branches through a gated residual combination, obtaining an embedding that incorporates both periodic information and time interval information:

$$T_{t_i} = T^P_{t_i} + \mathcal{G}_i \odot T^I_{t_i}. \tag{6}$$

This mechanism enables the model to adaptively adjust the feature distribution across the entire road network based on specific periodic time intervals and inter-node travel times.

4.2 Bi-Directional Mamba-Traj Encoder

In this section, we first fuse the previously obtained temporal embedding T_{t_i} with discrete road network features, and then encode the fused features using a bidirectional Mamba.

Time-Aware Feature Modulation

Previous methods typically concatenate temporal features with node embeddings in a straightforward manner. We argue that such simple concatenation fails to effectively capture the influence of time on traffic states. To address this issue, we employ an Adaptive Layer Normalization mechanism to dynamically fuse spatiotemporal features. We first extract the spatial embedding $h_i \in \mathbb{R}^d$ of the i-th road node from a learnable node embedding. Then, we treat the temporal embedding T_{t_i} as an environmental condition to dynamically modulate h_i. We utilize an MLP to project the temporal embedding T_{t_i} into a scaling factor $\gamma_i \in \mathbb{R}^d$ and a shift factor $\beta_i \in \mathbb{R}^d$:

$$[\gamma_i, \beta_i] = \text{MLP}(T_{t_i}). \tag{7}$$

Then, we apply an affine transformation to the normalized node features using these factors:

$$\hat{h}_i = \text{Norm}(h_i) \odot (1 + \gamma_i) + \beta_i. \tag{8}$$

The distribution of the modulated nodes becomes highly correlated with specific periodic time intervals and inter-node travel times. Subsequently, we stack the modulated features of the sequence to form the input matrix $\hat{\mathbf{H}} = [\hat{h}_1, \ldots, \hat{h}_N]^\top \in \mathbb{R}^{N\times d}$.

Bi-Directional Mamba Backbone

Traditional methods mostly tend to employ powerful but computationally expensive Transformers for modeling trajectory sequences. Meanwhile, Mamba architectures with linear complexity are inherently designed for causal modeling, and this causality makes them difficult to apply to trajectory representation learning, which heavily relies on contextual information. To balance both linear encoding efficiency and the acquisition of contextual information, we design a bidirectional Mamba as an encoder for trajectory modeling.

To capture the contextual information of trajectories, we design two independent SSM branches that process the trajectory sequence from forward and backward directions respectively. The forward branch processes the modulated sequence matrix $\hat{\mathbf{H}}$ in its natural order, explicitly capturing the sequential transition patterns from the starting point:

$$\overrightarrow{\mathbf{H}} = \text{SSM}_{\text{fwd}}(\hat{\mathbf{H}}). \tag{9}$$

For the backward branch, we employ a "flip-process-flip" strategy. We first flip $\hat{\mathbf{H}}$ along the temporal dimension before feeding it into the SSM, allowing the SSM to treat future nodes as historical nodes and process the trajectory sequence in the reverse direction. After computation, we flip the output back to the original temporal order:

$$\overleftarrow{\mathbf{H}} = \text{flip}(\text{SSM}_{\text{bwd}}(\text{flip}(\hat{\mathbf{H}}))). \tag{10}$$

After processing through the bidirectional SSM, we obtain two complementary embeddings. In $\overrightarrow{\mathbf{H}} \in \mathbb{R}^{N\times d}$, each token position aggregates information from the start, while the corresponding position in $\overleftarrow{\mathbf{H}} \in \mathbb{R}^{N\times d}$ aggregates information from the end. To further fuse these complementary perspectives, we first concatenate the bidirectional representations and then integrate them via a learnable linear projection $\mathbf{W}_o \in \mathbb{R}^{2d\times d}$, ultimately generating the comprehensive trajectory representation $\mathbf{H}_{traj} \in \mathbb{R}^{N\times d}$:

$$\mathbf{H}_{traj} = \text{LN}\left(\hat{\mathbf{H}} + [\overrightarrow{\mathbf{H}}; \overleftarrow{\mathbf{H}}]\mathbf{W}_o\right). \tag{11}$$

4.3 Pre-training with Hybrid Decoder

Existing research has demonstrated the effectiveness and necessity of the masked language modeling (MLM) task for enhancing trajectory representation capabilities. In our pre-training phase, we also incorporate masked language modeling: given a trajectory to be masked, we randomly select a certain proportion of road nodes and replace them with mask tokens. To effectively recover these masked tokens, we design a hybrid state-space–attention decoder. Unlike traditional decoders that rely on independent inputs, our design adopts an encoder-refiner module that effectively integrates the efficiency of state-space models (SSMs) with the retrieval capability of attention mechanisms.

Hybrid Decoding Architecture

We construct the decoder by stacking multiple hybrid layers. Specifically, we denote $\mathbf{H}^{(l)}_{dec}$ as the input state of the l-th decoder layer (where $\mathbf{H}^{(0)}_{dec} = \mathbf{H}_{traj}$). Each layer progressively refines this state through a cascaded three-step mechanism.

First, we employ a unidirectional Mamba block to efficiently model the internal dependencies of the latent feature sequence, enabling the decoder to generate autoregressive refined features with only linear complexity $O(N)$. For the generated autoregressive refined features, we apply layer normalization and a residual connection to the input state $\mathbf{H}_{dec}$:

$$\mathbf{H}'_{dec} = \text{Mamba}(\text{Norm}(\mathbf{H}_{dec})) + \mathbf{H}_{dec}. \tag{12}$$

Secondly, since pure SSMs primarily focus on local sequential transitions, we introduce a Multi-Head Global Attention module (MHGA) to retrieve global context from the original trajectory representation for reconstructing the missing tokens. We use the Mamba-refined state as the query, while the fixed encoder output $\mathbf{H}_{traj}$ serves as both the key and value:

$$\mathbf{H}''_{dec} = \text{MHGA}(\text{Norm}(\mathbf{H}'_{dec}), \mathbf{H}_{traj}, \mathbf{H}_{traj}) + \mathbf{H}'_{dec}. \quad (13)$$

Finally, we employ a Feed-Forward Network (FFN) to further refine the features and produce the output for the next layer:

$$\mathbf{H}_{out} = \text{FFN}(\text{Norm}(\mathbf{H}''_{dec})) + \mathbf{H}''_{dec}. \quad (14)$$

This architecture effectively aggregates the bidirectional context via global attention while maintaining high inference efficiency for the decoder's self-processing.

4.4 Optimization Objective

In this section, we introduce the construction of our masked loss. To reconstruct the masked trajectories, we feed the decoder output $\mathbf{H}_{out}$ into a multilayer perceptron to generate the predicted logits $\hat{\mathbf{Z}} = \text{MLP}(\mathbf{H}_{out})$. Finally, we adopt the cross-entropy loss between the masked road segments and the predicted probabilities as the pre-training objective function.

$$\mathcal{L}_{MLM} = -\frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \log \frac{\exp(\hat{Z}_{i,v_i})}{\sum_{j=1}^{|\mathcal{V}|} \exp(\hat{Z}_{i,j})}, \quad (15)$$

where $\mathcal{M}$ denotes the set of indices corresponding to the masked road segments, and $|\mathcal{M}|$ is the number of masked tokens. v_i represents the ground-truth road node ID at position i, while $\hat{Z}_{i,\cdot} \in \mathbb{R}^{|\mathcal{V}|}$ denotes the predicted logit vector for the i-th segment derived from the corresponding position in $\mathbf{H}_{out}$. For a batch of trajectories, we average the losses to optimize the model parameters.

5 Experiments

In this section, we first introduce the datasets, baselines, and experimental settings. Then, we evaluate our model with extensive experiments to answer the following research questions:

- **(RQ1)** How does SimTRL 's performance compare to other comparison methods on four downstream tasks?
- **(RQ2)** How does each module of SimTRL contribute to its overall performance?
- **(RQ3)** How do parameters affect model performance?
- **(RQ4)** How does the efficiency of SimTRL compare to other SOTA baseline methods?

5.1 Datasets

Dataset Description. We evaluate our method on two real-world datasets: Chengdu and Xi'an. The raw GPS data is provided by JGRM and sourced from the publicly available dataset released by DiDi Chuxing[1]. The corresponding road network data is collected from OpenStreetMap[2]. Using a map-matching algorithm[3], we map all GPS records to the road network to generate the map-matched Trajectory dataset. Finally, we divide the dataset into training, validation, and test sets in an 8:1:1 ratio.

Dataset	Chengdu	Xi'an
Region Sizes (km^2)	68.26	65.62
Nodes	6,250	6002
Trajectories	209889	156736
Avg travel time of traj.(s)	436.12	516.24
Avg. Trajectory Length (m)	2857.81	2976.52
Time span	2018/11/01 -2018/11/15	2018/11/01 -2018/11/15

Table 1: Details of the Datasets.

5.2 Baselines

We compare SimTRL with SOTA baseline methods: PIM [Yang *et al.*, 2021], Mtrajrec [Ren *et al.*, 2021], JCLRNT [Mao *et al.*, 2022], Trajbert [Si *et al.*, 2023],LightPath [Yang *et al.*, 2023], TrajCL [Chang *et al.*, 2023], START [Jiang *et al.*, 2023], JGRM [Ma *et al.*, 2024], UniTR [Zhao *et al.*, 2025], MM-Path [Xu *et al.*, 2025]. For baseline details, please see the supplementary Material.

5.3 Experiment Settings

Our experiments are conducted on a machine running Ubuntu 20.04 with an NVIDIA GeForce 4090 GPU. The embedding dimension d is set to 64, the mask ratio to 40%, the training batch size to 256, and the training epochs to 10.

5.4 Experimental Results

Overall Performance Comparison (RQ1)

Our experimental results are summarized in Table 2 and Table 3, where the best results are highlighted in **bold** and the second-best results are underlined. In all tasks, our model is significantly superior to existing methods. For details of tasks, please see the supplementary Material.

First, compared to existing state-of-the-art methods, our model achieves significant performance improvements across four downstream tasks.

In the travel time estimation task, our model achieved an average improvement of 5.19% compared to the second-best model. Although JGRM and START also model temporal information, they only perform coarse-grained processing of time features, while multimodal models like MM-Path and UniTR do not explicitly model temporal information, resulting in even poorer performance. This indicates that fine-grained processing and dynamic fusion of temporal information are highly effective and essential for time estimation tasks.

In the path ranking [Yang *et al.*, 2020; Yang and Yang, 2020] task, our model achieved an average improvement of over 18% in ranking metrics compared to the second-best

[1]https://outreach.didichuxing.com/

[2]https://www.openstreetmap.org/

[3]https://gotrackit.readthedocs.io/

Task		Travel Time Estimation			Path Ranking			Destination Prediction		
Datasets	Methods	MAE↓	MARE↓	MAPE↓	MAE↓	τ↑	ρ↑	ACC@1↑	ACC@5↑	MRR↑
Xi'an	PIM(21)	95.371	0.226	24.328	0.113	0.421	0.450	0.592	0.670	0.601
	JCLRNT(22)	99.321	0.237	25.367	0.107	0.431	0.470	0.540	0.635	0.573
	TrajCL(23)	96.324	0.225	25.681	0.085	0.511	0.535	0.623	0.729	0.636
	LightPath(23)	93.623	0.209	23.321	0.089	0.501	0.521	0.530	0.650	0.583
	START(23)	90.631	0.190	20.320	0.081	0.533	0.571	0.627	0.731	0.649
	JGRM(24)	87.673	0.189	20.548	0.090	0.503	0.531	0.677	0.769	0.736
	UniTR(25)	93.506	0.205	22.013	0.099	0.457	0.517	0.627	0.720	0.651
	MM-Path(25)	88.789	0.196	21.279	0.078	0.598	0.635	0.662	0.746	0.717
	SimTRL(ours)	**83.673**	**0.176**	**19.086**	**0.062**	**0.682**	**0.720**	**0.719**	**0.861**	**0.785**
	Improvement	4.56%	6.88%	6.07%	20.51%	14.04%	13.39%	6.20%	11.96%	6.65%
Task		Travel Time Estimation			Path Ranking			Destination Prediction		
Datasets	Methods	MAE↓	MARE↓	MAPE↓	MAE↓	τ↑	ρ↑	ACC@1↑	ACC@5↑	MRR↑
Chengdu	PIM(21)	91.371	0.229	24.692	0.109	0.690	0.736	0.563	0.689	0.603
	JCLRNT(22)	95.321	0.235	25.753	0.097	0.706	0.779	0.520	0.651	0.593
	TrajCL(23)	92.300	0.223	24.901	0.069	0.763	0.831	0.635	0.730	0.685
	LightPath(23)	85.623	0.215	23.856	0.095	0.740	0.803	0.442	0.619	0.523
	START(23)	83.198	0.191	20.527	0.055	0.790	0.862	0.633	0.726	0.675
	JGRM(24)	81.361	0.195	20.694	0.106	0.659	0.760	0.691	0.772	0.734
	UniTR(25)	86.992	0.209	22.554	0.058	0.834	0.885	0.621	0.723	0.624
	MM-Path(25)	82.611	0.199	21.210	0.065	0.775	0.851	0.680	0.763	0.691
	SimTRL(ours)	**77.804**	**0.181**	**19.699**	**0.032**	**0.933**	**0.954**	**0.715**	**0.860**	**0.783**
	Improvement	4.37%	5.23%	4.03%	41.81%	10.03%	11.87%	3.40%	11.40%	6.68%

Table 2: Three tasks' performance of methods on Xi'an and Chengdu datasets.

Dataset	Methods	Recall@3↑	Recall@5↑	MAP↑
Xi'an	Trajbert	0.903	0.933	0.889
	MtrajRec	0.955	0.970	0.939
	MM-Path	0.854	0.890	0.801
	SimTRL(ours)	**0.984**	**0.994**	**0.971**
	Improvement	3.03%	2.47%	3.41%
Chengdu	Trajbert	0.891	0.927	0.869
	MtrajRec	0.956	0.974	0.938
	MM-Path	0.830	0.878	0.774
	SimTRL(ours)	**0.982**	**0.990**	**0.970**
	Improvement	2.72%	1.62%	3.41%

Table 3: Performance on Trajectory Imputation task.

multimodal model. Although multimodal methods such as UniTR and MM-Path enhance representation capabilities by incorporating multimodal information, thereby outperforming traditional unimodal baselines, the inherent long-range dependency modeling ability and global receptive field of the mamba architecture in our bi-directional mamba-traj encoder enable precise capture of topological structures, leading to significantly high accuracy in path ranking.

In the destination prediction task, our model achieves state-of-the-art performance, with ACC@5 improving by approximately 11.96% compared to the strongest baseline and by 15.41% compared to the strongest multimodal baseline. Trajectories inherently contain rich travel semantics, and deeply mining this semantic information is far more pragmatic than introducing complex multimodal data.

Finally, regarding the Trajectory Imputation task, we also demonstrate superior performance. This success is driven by our bidirectional encoder's ability to capture global trajectory information and the hybrid decoder tailored for the pre-training phase. During the pre-training phase, we thoroughly learned the mask reconstruction task, enabling accurate reconstruction of missing trajectory structures based solely on partial observations.

Ablation Study (RQ2)

To check the contribution of each component in SimTRL, we conduct ablation studies on the Xi'an and Chengdu datasets. We compare the full SimTRL against three variants:

- ***w/o Transformer***: We replace the Bi-Directional Traj-Mamba Encoder with a standard Transformer encoder.
- ***w/o Mamba***: We replace the Bi-Directional Mamba backbone with a standard, unidirectional Mamba block.
- ***w/o TE***: We only keep the Periodic Semantic Extraction module and remove the other parts.

As seen in Table 4, ***w/o Mamba*** has the biggest performance drop; except for the time prediction task, it fails in all other tasks. This is because the unidirectional Mamba has linear complexity, but its one-way nature makes it hard to learn trajectory tasks that need rich context. Second, ***w/o Transformer*** also drops in performance across different tasks, with

Methods	Travel Time Estimation			Path Ranking			Destination Prediction			Trajectory Imputation		
	MAE↓	MARE↓	MAPE↓	MAE↓	τ↑	ρ↑	ACC@1↑	ACC@5↑	MRR↑	Recall@3↑	Recall@5↑	MAP↑
w/o Transformer	85.195	0.183	19.561	0.073	0.582	0.630	0.559	0.702	0.626	0.938	0.951	0.925
w/o Mamba	102.873	0.221	26.277	0.162	0.220	0.268	0.121	0.277	0.204	0.145	0.176	0.139
w/o TE	87.691	0.192	20.506	0.067	0.635	0.677	0.714	0.851	0.780	0.985	0.989	0.969
SimTRL(full)	**83.721**	**0.177**	**19.086**	**0.067**	**0.635**	**0.678**	**0.719**	**0.861**	**0.785**	**0.984**	**0.990**	**0.971**
Methods	**Travel Time Estimation**			**Path Ranking**			**Destination Prediction**			**Trajectory Imputation**		
	MAE↓	MARE↓	MAPE↓	MAE↓	τ↑	ρ↑	ACC@1↑	ACC@5↑	MRR↑	Recall@3↑	Recall@5↑	MAP↑
w/o Transformer	79.168	0.189	20.373	0.039	0.909	0.938	0.573	0.706	0.636	0.946	0.965	0.913
w/o Mamba	102.873	0.221	26.277	0.125	0.696	0.793	0.121	0.277	0.204	0.145	0.176	0.139
w/o TE	82.643	0.195	20.734	0.033	0.934	0.956	0.709	0.861	0.782	0.982	0.992	0.973
SimTRL(full)	**77.804**	**0.181**	**19.699**	**0.032**	**0.933**	**0.954**	**0.715**	**0.860**	**0.783**	**0.982**	**0.990**	**0.970**

Table 4: Performance of variants of SimTRL on Xi'an and Chengdu.

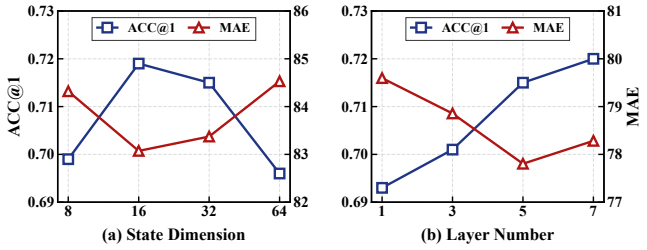

Figure 2: Hyperparameter impact w.r.t. the State Dimension N and number of Layers L.

Method	#Epoch	Time per Epoch (s)		Total Time (s)	
		Xi'an	Chengdu	Xi'an	Chengdu
TrajCL (23)	20	112	215	2,240	4,300
LightPath (23)	50	65	99	3,250	4,950
START (23)	30	221	402	6,630	12,060
JGRM (24)	20	960	1,397	19,200	27,940
UniTR (25)	1,000	30	36	30,000	36,000
MM-Path (25)	60	110	160	6,600	9,600
SimTRL (ours)	**10**	**59**	**86**	**590**	**860**

Table 5: Comparison of Pre-training Efficiency on two datasets.

average drops of 7.04% and 5.56% on the Xi'an and Chengdu datasets. This shows that our bidirectional trajectory-Mamba encoder is better than the traditional transformer at capturing dynamic movement and bidirectional topology in trajectories. Finally, the ***w/o TE*** variant mainly affects the travel time estimation task, causing average decreases of 6.88% and 6.40% on the Xi'an and Chengdu datasets. This means that our time encoding strategy successfully captures the dynamic time information in the trajectories.

Parameter Sensitivity Analysis (RQ3)

We also do a sensitivity analysis on key hyperparameters, specifically the State Dimension N and the number of encoder layers L. Figure 2 shows the performance changes on the Xi'an dataset for Travel Time Estimation (measured by MAE) and Destination Prediction (measured by ACC@1).

For State Dimension N, as shown in Figure 2(a), increasing the State Dimension N helps the model performance at first. But when the dimension gets too big, the results drop sharply. This suggests that we need a large state space to capture complex features, but if it is too big, it causes redundancy or optimization problems, which hurts performance. For Encoder Layers L, Figure 2(b) shows the effect of layer depth. For the TTE task, the model works best (lowest MAE) at $L = 5$, and after that, the error rate goes up as we add more layers. But for the DP task, performance keeps improving slightly with deeper layers. To balance the multiple tasks, we choose $L = 5$ as the best setting for our model.

Efficiency Comparison (RQ4)

As shown in Table 5, SimTRL shows high efficiency. First, regarding single-epoch training time, compared to the classic Transformer-based baseline START, our model is about 3.7× speedup on the Xi'an dataset. Even compared to the lightweight LightPath, our model has faster per-epoch training speeds. Second, our SimTRL converges very quickly. SimTRL only needs 10 epochs to reach its best performance, while baselines usually need 20 to 60 epochs. It is worth noting that while UniTR has a short single-epoch training time, it needs 1000 epochs to converge, so the total training time is extremely high.

In contrast, for both multimodal models and traditional Transformer-based models, the total training cost of SimTRL is very low. This shows that our method allows the model to capture core spatio-temporal features efficiently without unnecessary iterations, ensuring it is scalable for real-world large-scale use.

6 Conclusion

In this paper, we proposed SimTRL, A streamlined and efficient unimodal model focused on road-network constrained trajectory learning. By integrating a Bi-Directional Traj-Mamba Encoder with Context-Aware Gated Time Fusion, SimTRL captures global non-causal topology and dynamic traffic fluctuations with linear efficiency. Experiments on real-world datasets demonstrate that SimTRL significantly outperforms complex multi-modal and unimodal baselines, achieving notable accuracy gains while substantially reducing training cost.

Acknowledgments

This work is partially supported by the Key R&D Program of Shandong Province, China under Grant No. 2025CXPT185, the Fundamental Research Funds for the Central Universities under Grant No. 202442005, the National Natural Science Foundation of China under Grant Nos. 62176243 and 62302120, and the Natural Science Foundation of Shandong Province Grant No. ZR2024MF128.

References

[Bao *et al.*, 2022] Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *Advances in neural information processing systems*, 35:32897–32912, 2022.

[Chang *et al.*, 2023] Yanchuan Chang, Jianzhong Qi, Yuxuan Liang, and Egemen Tanin. Contrastive trajectory similarity learning with dual-feature attention. In *2023 IEEE 39th International conference on data engineering (ICDE)*, pages 2933–2945. IEEE, 2023.

[Chen *et al.*, 2015] Lu Chen, Yunjun Gao, Xinhan Li, Christian S Jensen, and Gang Chen. Efficient metric indexing for similarity search and similarity joins. *IEEE Transactions on Knowledge and Data Engineering*, 29(3):556–571, 2015.

[Chen *et al.*, 2018] Lu Chen, Qilu Zhong, Xiaokui Xiao, Yunjun Gao, Pengfei Jin, and Christian S Jensen. Price-and-time-aware dynamic ridesharing. In *2018 IEEE 34th international conference on data engineering (ICDE)*, pages 1061–1072. IEEE, 2018.

[Chen *et al.*, 2021] Yile Chen, Xiucheng Li, Gao Cong, Zhifeng Bao, Cheng Long, Yiding Liu, Arun Kumar Chandran, and Richard Ellison. Robust road network representation learning: When traffic patterns meet traveling semantics. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 211–220, 2021.

[Chen *et al.*, 2022] W Chen, S Li, C Huang, Y Yu, Y Jiang, and Y Dong. Mutual distillation learning network for trajectory-user linking. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pages 1974–1979. IJCAI, 2022.

[Chen *et al.*, 2024] Wei Chen, Chao Huang, Yanwei Yu, Yongguo Jiang, and Junyu Dong. Trajectory-user linking via hierarchical spatio-temporal attention networks. *ACM Transactions on Knowledge Discovery from Data*, 18(4):1–22, 2024.

[Ding *et al.*, 2018] Xin Ding, Lu Chen, Yunjun Gao, Christian S Jensen, and Hujun Bao. Ultraman: A unified platform for big trajectory data management and analytics. *Proceedings of the VLDB Endowment*, 11(7):787–799, 2018.

[Feng *et al.*, 2023] Tao Feng, Huan Yan, Huandong Wang, Wenzhen Huang, Yuyang Han, Hongsen Liao, Jinghua Hao, and Yong Li. Ilroute: A graph-based imitation learning method to unveil riders' routing strategies in food delivery service. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4024–4034, 2023.

[Fu and Lee, 2020] Tao-Yang Fu and Wang-Chien Lee. Trembr: Exploring road networks for trajectory representation learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(1):1–25, 2020.

[Gu and Dao, 2023] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[Jiang *et al.*, 2023] Jiawei Jiang, Dayan Pan, Houxing Ren, Xiaohan Jiang, Chao Li, and Jingyuan Wang. Self-supervised trajectory representation learning with temporal regularities and travel semantics. In *2023 IEEE 39th international conference on data engineering (ICDE)*, pages 843–855. IEEE, 2023.

[Jin *et al.*, 2024] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. In *International conference on learning representations*, volume 2024, pages 23857–23880, 2024.

[Li *et al.*, 2018] Yaguang Li, Kun Fu, Zheng Wang, Cyrus Shahabi, Jieping Ye, and Yan Liu. Multi-task representation learning for travel time estimation. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1695–1704, 2018.

[Lin *et al.*, 2021] Yan Lin, Huaiyu Wan, Shengnan Guo, and Youfang Lin. Pre-training context and time aware location embeddings from spatial-temporal trajectories for user next location prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 4241–4248, 2021.

[Lin *et al.*, 2023] Yan Lin, Huaiyu Wan, Shengnan Guo, Jilin Hu, Christian S Jensen, and Youfang Lin. Pre-training general trajectory embeddings with maximum multi-view entropy coding. *IEEE Transactions on Knowledge and Data Engineering*, 36(12):9037–9050, 2023.

[Liu *et al.*, 2020] Yiding Liu, Kaiqi Zhao, Gao Cong, and Zhifeng Bao. Online anomalous trajectory detection with deep generative sequence modeling. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 949–960. IEEE, 2020.

[Liu *et al.*, 2026] Yichen Liu, Yan Lin, Shengnan Guo, Zeyu Zhou, Youfang Lin, and Huaiyu Wan. Trajmamba: An efficient and semantic-rich vehicle trajectory pre-training model. *Advances in Neural Information Processing Systems*, 38:21779–21804, 2026.

[Ma *et al.*, 2024] Zhipeng Ma, Zheyan Tu, Xinhai Chen, Yan Zhang, Deguo Xia, Guyue Zhou, Yilun Chen, Yu Zheng, and Jiangtao Gong. More than routing: Joint gps and

route modeling for refine trajectory representation learning. In *Proceedings of the ACM Web Conference 2024*, pages 3064–3075, 2024.

[Mao *et al.*, 2022] Zhenyu Mao, Ziyue Li, Dedong Li, Lei Bai, and Rui Zhao. Jointly contrastive representation learning on road network and trajectory. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 1501–1510, 2022.

[Menon and Vondrick, 2023] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. *ICLR*, page 11733, 2023.

[Ren *et al.*, 2021] Huimin Ren, Sijie Ruan, Yanhua Li, Jie Bao, Chuishi Meng, Ruiyuan Li, and Yu Zheng. Mtrajrec: Map-constrained trajectory recovery via seq2seq multi-task learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1410–1419, 2021.

[Scarselli *et al.*, 2008] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[Si *et al.*, 2023] Junjun Si, Jin Yang, Yang Xiang, Hanqiu Wang, Li Li, Rongqing Zhang, Bo Tu, and Xiangqun Chen. Trajbert: Bert-based trajectory recovery with spatial-temporal refinement for implicit sparse trajectories. *IEEE Transactions on Mobile Computing*, 23(5):4849–4860, 2023.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Wang *et al.*, 2018] Zheng Wang, Kun Fu, and Jieping Ye. Learning to estimate the travel time. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 858–866, 2018.

[Wang *et al.*, 2025] Xiangheng Wang, Ziquan Fang, Chenglong Huang, Danlei Hu, Lu Chen, and Yunjun Gao. Gtr: A general, multi-view, and dynamic framework for trajectory representation learning. In *International Conference on Machine Learning*, pages 62825–62844. PMLR, 2025.

[Xia *et al.*, 2025] Hong Xia, Xiao Zhang, Yuan Cao, Lei Cao, Yanwei Yu, and Junyu Dong. Self-supervised trajectory representation learning with multi-scale spatio-temporal feature exploration. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, pages 779–792. IEEE, 2025.

[Xu *et al.*, 2025] Ronghui Xu, Hanyin Cheng, Chenjuan Guo, Hongfan Gao, Jilin Hu, Sean Bin Yang, and Bin Yang. Mm-path: Multi-modal, multi-granularity path representation learning. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pages 1703–1714, 2025.

[Yang and Yang, 2020] Sean Bin Yang and Bin Yang. Learning to rank paths in spatial networks. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 2006–2009. IEEE, 2020.

[Yang *et al.*, 2020] Sean Bin Yang, Chenjuan Guo, and Bin Yang. Context-aware path ranking in road networks. *IEEE Transactions on Knowledge and Data Engineering*, 34(7):3153–3168, 2020.

[Yang *et al.*, 2021] Sean Bin Yang, Chenjuan Guo, Jilin Hu, Jian Tang, and Bin Yang. Unsupervised path representation learning with curriculum negative sampling. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 3286–3292, 2021.

[Yang *et al.*, 2023] Sean Bin Yang, Jilin Hu, Chenjuan Guo, Bin Yang, and Christian S Jensen. Lightpath: Lightweight and scalable path representation learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2999–3010, 2023.

[Yao *et al.*, 2017] Di Yao, Chao Zhang, Zhihua Zhu, Jianhui Huang, and Jingping Bi. Trajectory clustering via deep representation learning. In *2017 international joint conference on neural networks (IJCNN)*, pages 3880–3887. IEEE, 2017.

[Yu *et al.*, 2026] Yanwei Yu, Hong Xia, Shaoxuan Gu, Xingyu Zhao, Dongliang Chen, and Yuan Cao. Self-supervised cross-city trajectory representation learning based on meta-learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 16217–16225, 2026.

[Zhang *et al.*, 2025] Hao Zhang, Wei Chen, Xingyu Zhao, Jianpeng Qi, Guiyuan Jiang, and Yanwei Yu. Scalable trajectory-user linking with dual-stream representation networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pages 13224–13232, 2025.

[Zhang *et al.*, 2026] Xiao Zhang, Xingyu Zhao, Yuan Cao, Bin Wang, Guiyuan Jiang, and Yanwei Yu. Trajagg: Dual-scale feature aggregation with hybrid training for trajectory similarity computation in free space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 16370–16378, 2026.

[Zhao *et al.*, 2025] Jie Zhao, Chao Chen, Yuanshao Zhu, Mingyu Deng, and Yuxuan Liang. Unitr: A unified framework for joint representation learning of trajectories and road networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 13348–13356, 2025.

[Zhou *et al.*, 2023] Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.

[Zhou *et al.*, 2025] Zeyu Zhou, Yan Lin, Haomin Wen, Shengnan Guo, Jilin Hu, Youfang Lin, and Huaiyu Wan. Trajcogn: leveraging llms for cognizing movement patterns and travel purposes from trajectories. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pages 3698–3706, 2025.