SSL-STMFormer: Self-Supervised Learning Spatio-Temporal Entanglement Transformer for Traffic Flow Prediction

Zetao Li[1], Zheng Hu[1], Peng Han[1*], Yu Gu[1*], Shimin Cai[1*]

[1]University of Electronic Science and Technology of China, Chengdu 610054, China
{zetaoli99,huzheng}@std.uestc.edu.cn,{penghan,guyu,shimincai}@uestc.edu.cn

Abstract

Traffic flow prediction remains a critical issue in intelligent transport systems. Despite significant efforts in traffic flow modeling, existing approaches exhibit several notable limitations: (i) Most models fail to capture traffic flow similarities over long distances and extended periods; (ii) They struggle to account for spatio-temporal heterogeneity induced by varying traffic flow patterns; (iii) Due to their static modeling approach, they struggle to effectively capture the intricate spatio-temporal entanglement. To address these challenges, we propose a traffic flow prediction framework based on self-supervised learning spatio-temporal entanglement transformer(SSL-STMFormer). This framework adopts a self-supervised learning paradigm, leveraging a transformer architecture that captures richer spatio-temporal information to better represent traffic flow patterns. Specifically, a temporal attention module and a spatial attention module are employed to capture the spatio-temporal dependencies of traffic dynamics, respectively, and spatio-temporal entanglement-aware methods are introduced to allow the model to perceive spatio-temporal entanglement and thus better modelling of real traffic environments. Furthermore, to achieve adaptive spatio-temporal self-supervised learning, adaptive data augmentation is applied to the input traffic flow data, and the traffic flow prediction task is enhanced with temporal heterogeneity module and spatial heterogeneity module. Extensive experimental evaluations conducted on six publicly available real-world transportation datasets demonstrate that our method achieves substantial improvements across these datasets.

Code — https://github.com/ZetaoLiPhD/SSL-STMFomer

Introduction

The rapid urbanization and continuous growth of urban populations have led to significant challenges in urban traffic management. Traffic congestion, increased travel time, and air pollution are among the most pressing issues faced by modern cities(Djahel et al. 2014). Effective traffic flow prediction is crucial for mitigating these problems, enabling better traffic management, efficient route planning, and improved urban mobility(Wang et al. 2021).

For traffic flow prediction, the fundamental challenge lies in effectively capturing and modeling the spatio-temporal correlations within traffic data(Yin et al. 2021; Jiang and Luo 2022; Bui, Cho, and Yi 2022; Jin et al. 2023). Previous studies have employed various methods to address this challenge. Initially, convolutional neural networks (CNNs) were combined with recurrent neural networks (RNNs) to leverage CNNs' efficacy in handling spatial data and RNNs' proficiency in processing time series data, enabling the learning of spatio-temporal correlations(Zhang, Zheng, and Qi 2017; Yao et al. 2018; Vinayakumar, Soman, and Poornachandran 2017; Qiu et al. 2018). With the advent of graph neural networks (GNNs), the research focus shifted, as numerous studies demonstrated the superior capability of GNNs in modeling the graph structure of traffic data(Yu, Yin, and Zhu 2018; Wu et al. 2019, 2020; Song et al. 2020; Li and Zhu 2021; Fang et al. 2021; Choi et al. 2022). Recently, the trend has been towards developing generalized traffic flow prediction models that utilize pre-training strategies and unsupervised learning methods to enhance model performance and generalizability across diverse traffic scenarios(Ji et al. 2023; Zhang et al. 2023; Gao et al. 2024; Yuan et al. 2024).

Despite the promising outcomes of previous studies, these methods still present certain limitations. First, traffic systems exhibit spatiotemporal entanglement, where both spatiotemporal dependencies and heterogeneities dynamically evolve over time due to varying travel patterns. As illustrated in Figure 1, we selected the traffic flow data from three locations to elucidate this spatiotemporal entanglement: San Francisco International Airport (Node A), Oakland International Airport in the San Francisco Bay Area (Node B), and Bernal Heights (Node C). In Figure 1(b), we observe the traffic flow of these three locations from 2020/7/10 to 2020/7/17. From 7/10 to 7/13, the traffic flow trends of these locations exhibit strong similarities. However, this similarity sharply diminishes during the remaining period. Existing methods, which model this phenomenon in a static manner (pre-defined), fail to effectively capture these dynamics. Second, due to the functional segmentation of urban areas, even two distant locations may exhibit similar traffic patterns, though this similarity is also subject to dynamic changes. Furthermore, the inherent over-smoothing problem in GNNs hampers their ability to capture such dynamic long-term dependencies.

*Corresponding authors.

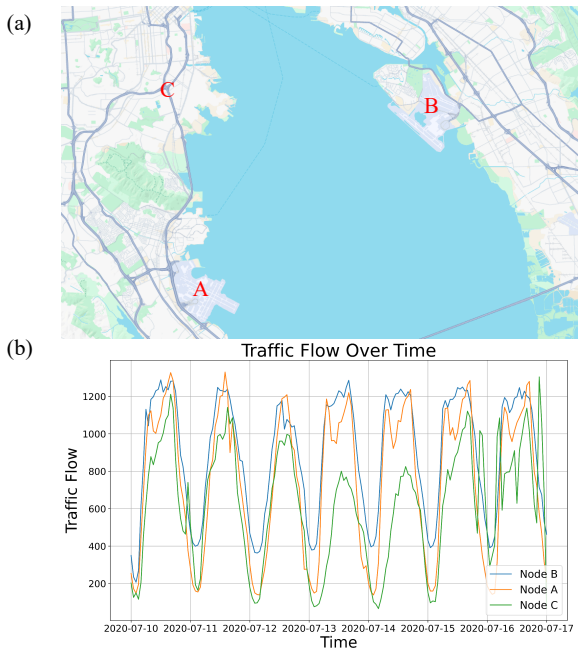

Figure 1: Research motivation statement. i.e., the long range long time entanglement of traffic flow data.

To address these issues, we propose a novel model, SSL-STMFormer, which leverages self-supervised learning and a spatio-temporal attention mechanism to enhance traffic flow prediction. Our model addresses the limitations of existing approaches by incorporating spatio-temporal entanglement-aware module to better capture the real-world traffic environment. The primary contributions of this paper are as follows:

- We introduce a spatio-temporal attention mechanism that dynamically captures the dependencies between different regions and time steps. This mechanism enhances the model's ability to understand complex traffic patterns and their evolution over time.
- To address the challenges of spatio-temporal heterogeneity, we design a self-supervised learning task that aids the model in capturing the underlying structure of traffic data. This task helps improve the model's generalization.
- We propose spatio-temporal entanglement-aware modules to capture the inherent spatio-temporal entanglement in traffic data. These modules enable the model to perceive the dynamic interactions within the transportation system, allowing it to accurately detect evolving traffic patterns and consequently improve prediction accuracy.
- We conduct extensive experiments on six real-world traffic datasets, including graph-based highway traffic data and grid-based citywide traffic data. Our results demonstrate the superiority of SSL-STMFormer over existing state-of-the-art models.

Preliminaries

Notations and Definitions

Definition 1 (Road Network) *Road Network is represented as a graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \boldsymbol{A})$*, where,* $\mathcal{V} = \{v_1, \cdots, v_N\}$ *is a set of N nodes* ($|V| = N$)*,* $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ *is a set of edges, and A is the adjacency matrix of network* $\mathcal{G}$*. Here, N indicates the number of nodes in the graph.*

Definition 2 (Traffic Flow Tensor) $\boldsymbol{X}_t \in \mathbb{R}^{N \times D}$ *is employed to denote the traffic flow at time t of N nodes in the road network, where D is the dimension of the traffic flow. We use* $\mathcal{X} = (\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots, \boldsymbol{X}_T) \in \mathbb{R}^{T \times N \times D}$ *to denote the traffic flow tensor of all nodes across total T time slices.*

Problem Formalization

Traffic flow prediction seeks to forecast future traffic patterns based on historical data. Formally, given a traffic flow tensor $\mathcal{X}$ observed within a traffic system, our objective is to learn a mapping function f that utilizes traffic observations from the preceding T time steps to predict the traffic flow for the subsequent T' time steps,

$$\left[\boldsymbol{X}_{(t-T+1)}, \cdots, \boldsymbol{X}_t; \mathcal{G}\right] \xrightarrow{f} \left[\boldsymbol{X}_{(t+1)}, \cdots, \boldsymbol{X}_{(t+T')}\right]. \quad (1)$$

Methodology

This section provides a detailed exposition of the technical aspects of our SSL-STMFormer model, with the overall architecture depicted in Fig. 2. We describe each module in detail below.

Data Embedding Layer

The purpose of establishing a data embedding layer is to transform the input data into a high-dimensional representation, thereby facilitating more effective learning and modeling of complex patterns(Jiang et al. 2023). First, the raw input $\mathcal{X}$ is converted to $\boldsymbol{X}_{data} \in \mathbb{R}^{T \times N \times d}$ through a fully connected layer, d is the embedding dimension. In our study, spatio-temporal information encompasses road network structure data (spatial information) and urban traffic flow data (temporal information). For spatial information, Graph Laplacian eigenvectors is used to represent the road network structure(Belkin and Niyogi 2003). This method accurately describes the distance information between nodes in the graph. A linear projection onto the k smallest non-trivial eigenvectors generates the spatial graph Laplacian embedding $\boldsymbol{X}_{spe} \in \mathbb{R}^{N \times d}$.

For temporal information, the time-periodic embeddings $\boldsymbol{X}_w, \boldsymbol{X}_d \in \mathbb{R}^{T \times d}$ are obtained by concatenating the embeddings ($\boldsymbol{t}_{w(t)}$ and $\boldsymbol{t}_{d(t)} \in \mathbb{R}^d$) of all T time slices to capture daily and weekly periodicity in travel patterns(Ding et al. 2023). Here, $w(t)$ and $d(t)$ convert time t into a weekly index (1 to 7) and a minute index (1 to 1440), respectively. Finally, we also utilize a temporal position encoding $\boldsymbol{X}_{tpe} \in \mathbb{R}^{T \times d}$ to incorporate the positional information of the input sequence(Vaswani et al. 2017). The output of the data embedding layer is then obtained by summing the aforementioned embedding matrices(matrix addition is implemented through dimensional replication), as follows:

$$\boldsymbol{X}_{emb} = \boldsymbol{X}_{data} + \boldsymbol{X}_{spe} + \boldsymbol{X}_w + \boldsymbol{X}_d + \boldsymbol{X}_{tpe}. \quad (2)$$

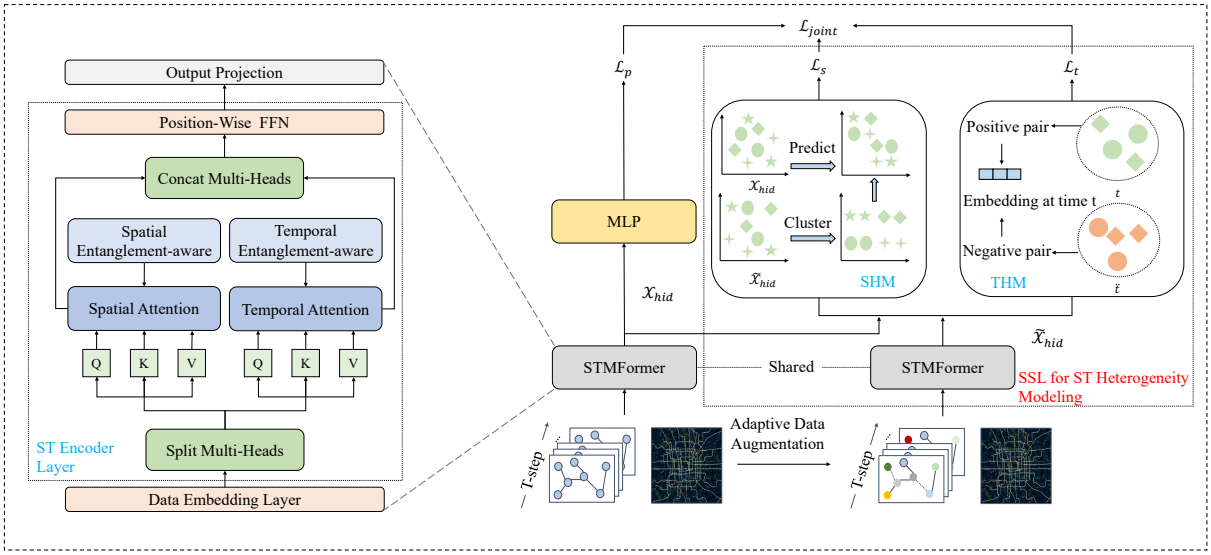

Figure 2: The overall architecture of SSL-STMFormer.

$\boldsymbol{X}_{emb}$ will be fed into the following models, and we use $\boldsymbol{X}$ to replace $\boldsymbol{X}_{emb}$ for convenience.

Spatial-Temporal Encoder Layer

We propose a spatio-temporal encoder layer, grounded in a attention mechanism and entanglement-aware methods, to effectively model the complex and dynamic dependencies within spatio-temporal data. The encoder layer is composed of four key components: (1) a spatial attention module designed to capture both short- and long-range dynamic spatial dependencies, (2) a temporal attention module that similarly captures dynamic patterns over varying temporal scales, (3) a spatial entanglement-aware module, and (4) a temporal entanglement-aware module. These latter two components are critical for enhancing the model's ability to grasp intricate spatio-temporal relationships and interwoven dynamics. The attention operations within this framework are formalized using the following slice notation. For a tensor $\boldsymbol{X} \in \mathbb{R}^{T\times N\times D}$, the t-th slice is represented by the matrix $\boldsymbol{X}_{t::} \in \mathbb{R}^{N\times D}$, and the n-th slice is denoted as $\boldsymbol{X}_{:n:} \in \mathbb{R}^{T\times D}$.

Spatial Attention (SA). The spatial attention module is designed to capture complex dynamic spatial dependencies within traffic data. Formally, at time t, the query, key, and value matrices for the self-attention operations are derived to as follows:

$$\boldsymbol{Q}_t^{(S)} = \boldsymbol{X}_{t::}\boldsymbol{W}_Q^S, \boldsymbol{K}_t^{(S)} = \boldsymbol{X}_{t::}\boldsymbol{W}_K^S, \boldsymbol{V}_t^{(S)} = \boldsymbol{X}_{t::}\boldsymbol{W}_V^S, \tag{3}$$

where $\boldsymbol{W}_Q^S, \boldsymbol{W}_K^S, \boldsymbol{W}_V^S \in \mathbb{R}^{d\times d'}$ are learnable parameters and d' is the dimension of the query, key, and value matrix in this work. Subsequently, attention operations in the spatial dimension are applied to model the interactions between nodes and to derive the spatial dependencies (attention scores) among all nodes at time t as follows:

$$\boldsymbol{A}_t^{(S)} = \frac{(\boldsymbol{Q}_t^{(S)})(\boldsymbol{K}_t^{(S)})^\top}{\sqrt{d'}}. \tag{4}$$

It is evident that the spatial dependencies between nodes vary dynamically across different time slices. Therefore, the SA module is employed to capture these dynamic spatial dependencies, and F stands for the $softmax$ function. The final output of the spatial attention module is then obtained as follows:

$$SA(\boldsymbol{Q}_t^{(S)}, \boldsymbol{K}_t^{(S)}, \boldsymbol{V}_t^{(S)}) = F(\boldsymbol{A}_t^{(S)})\boldsymbol{V}_t^{(S)}. \tag{5}$$

Spatial Entanglement-Aware. It is important to note that Equation(5) implies that each node interacts with all other nodes, which does not accurately reflect real transport systems. To more closely simulate the actual traffic environment and to enhance the model's ability to capture complex spatio-temporal entanglement relationships and interwoven dynamics, we design spatio-temporal entanglement-aware module based on multiple masking strategies inspired by pre-trained linguistic models and visual models.

- **Random Masking.** This strategy is analogous to the approach utilized in Masked Autoencoders (MAE), where spatio-temporal patches are randomly masked(Feichtenhofer et al. 2022). The primary objective is to capture fine-grained spatio-temporal relationships.
- **Tube Masking.** This strategy simulates scenarios in which data for specific spatial units is completely missing across all time instances, reflecting real-world situations where certain sensors may be nonfunctional—a common occurrence(Yuan et al. 2024). The primary objective is to enhance the model's capability for spatial extrapolation.
- **Block Masking.** A more challenging variant of tube masking, block masking involves the complete absence

of an entire block of spatial units across all time instances. This approach makes the reconstruction task more complex due to the reduced context information, with the goal of enhancing spatial transferability(Yuan et al. 2024).

In addition to the three aforementioned masking strategies, it is important to acknowledge that interactions between only a subset of node pairs are crucial in a traffic system. This includes both nearby node pairs and those that are geographically distant but exhibit similar traffic flow patterns. Therefore, two graph masking matrices are constructed to simultaneously capture the spatial dependencies within the traffic system. Specifically, the short-range masking matrix, denoted as $\boldsymbol{M}_{short}$, is assigned a weight of 1 if the distance (i.e., the number of hops in the graph) between two nodes is less than a threshold λ, and 0 otherwise. The long-range masking matrix, denoted as $\boldsymbol{M}_{long}$, is constructed using the Dynamic Time Warping (DTW) algorithm to compute the similarity of historical traffic flows between nodes(Berndt and Clifford 1994). For each node, we identify the K nodes with the highest similarity as its long-range neighbors, and then construct $\boldsymbol{M}_{long}$ by setting the weights between the current node and its long-range neighbors to 1, and to 0 otherwise. Finally, we apply these masking strategies to the spatial attention module, which can be defined as:

$$\begin{aligned}
RandSA(\boldsymbol{Q}_t^{(S)}, \boldsymbol{K}_t^{(S)}, \boldsymbol{V}_t^{(S)}) &= F(\boldsymbol{A}_t^{(S)} \odot \boldsymbol{M}_{rand}^{(S)})\boldsymbol{V}_t^{(S)}, \\
TubeSA(\boldsymbol{Q}_t^{(S)}, \boldsymbol{K}_t^{(S)}, \boldsymbol{V}_t^{(S)}) &= F(\boldsymbol{A}_t^{(S)} \odot \boldsymbol{M}_{tube}^{(S)})\boldsymbol{V}_t^{(S)}, \\
BlockSA(\boldsymbol{Q}_t^{(S)}, \boldsymbol{K}_t^{(S)}, \boldsymbol{V}_t^{(S)}) &= F(\boldsymbol{A}_t^{(S)} \odot \boldsymbol{M}_{block}^{(S)})\boldsymbol{V}_t^{(S)}, \\
ShortSA(\boldsymbol{Q}_t^{(S)}, \boldsymbol{K}_t^{(S)}, \boldsymbol{V}_t^{(S)}) &= F(\boldsymbol{A}_t^{(S)} \odot \boldsymbol{M}_{short})\boldsymbol{V}_t^{(S)}, \\
LongSA(\boldsymbol{Q}_t^{(S)}, \boldsymbol{K}_t^{(S)}, \boldsymbol{V}_t^{(S)}) &= F(\boldsymbol{A}_t^{(S)} \odot \boldsymbol{M}_{long})\boldsymbol{V}_t^{(S)},
\end{aligned} \tag{6}$$

where $\odot$ indicates the Hadamard product. In this manner, the spatial entanglement-aware module is enabled to capture more comprehensive spatial information.

Temporal Attention (TA). Phenomena such as the periodicity and transient nature of traffic flows suggest that dependencies exist between flows across different time slices. Consequently, the temporal attention module is introduced to model these dynamic dependencies. Formally, for node n, the query, key, and value matrices are computed as follows:

$$\boldsymbol{Q}_n^{(T)} = \boldsymbol{X}_{:n:}\boldsymbol{W}_Q^T, \boldsymbol{K}_n^{(T)} = \boldsymbol{X}_{:n:}\boldsymbol{W}_K^T, \boldsymbol{V}_n^{(T)} = \boldsymbol{X}_{:n:}\boldsymbol{W}_V^T, \tag{7}$$

where $\boldsymbol{W}_Q^T, \boldsymbol{W}_K^T, \boldsymbol{W}_V^T \in \mathbb{R}^{d \times d'}$ are learnable parameters. Similarly, the attention mechanism is applied along the time dimension to capture the temporal dependencies across all time slices for node n:

$$\boldsymbol{A}_n^{(T)} = \frac{(\boldsymbol{Q}_n^{(T)})(\boldsymbol{K}_n^{(T)})^\top}{\sqrt{d'}}. \tag{8}$$

Temporal Entanglement-Aware. Clearly, temporal attention is capable of revealing diverse temporal patterns across different nodes in the traffic data. However, this capability assumes data completeness, which is often compromised in real-world traffic environments due to data loss. To address this issue, we apply the first three masking strategies used in the spatial entanglement-aware module to the temporal entanglement-aware module. Additionally, we introduce a temporal masking strategy that involves masking future data, thereby compelling the model to reconstruct future traffic conditions based solely on historical information. The objective is to enhance the model's ability to capture temporal dependencies from the past to the future, which in turn makes the model obtain ability to perceive temporal entanglement.

$$\begin{aligned}
TA(\boldsymbol{Q}_n^{(T)}, \boldsymbol{K}_n^{(T)}, \boldsymbol{V}_n^{(T)}) &= F(\boldsymbol{A}_n^{(T)})\boldsymbol{V}_n^{(T)}, \\
TempTA(\boldsymbol{Q}_t^{(S)}, \boldsymbol{K}_t^{(S)}, \boldsymbol{V}_t^{(S)}) &= F(\boldsymbol{A}_t^{(S)} \odot \boldsymbol{M}_{temp})\boldsymbol{V}_t^{(S)},
\end{aligned} \tag{9}$$

where $\odot$ indicates the Hadamard product. In this way, the temporal attention module is enabled to capture more comprehensive temporal information.

Multiple Attention Fusion. Multi-attention fusion module is employed to fuse temporal and spatial attention, thereby reducing the computational complexity of the model while fully integrating spatio-temporal information. In the aforementioned process, we obtain multiple heads, whose results are concatenated and projected to yield the outputs. This process enables the model to merge the distinct temporal and spatial information into a comprehensive spatio-temporal representation. Formally, this can be described by the following equation:

$$\begin{aligned}
STA = cat(&\boldsymbol{Z}_{1\cdots h_{SRand}}^{SRand}, \boldsymbol{Z}_{1\cdots h_{STube}}^{STube}, \boldsymbol{Z}_{1\cdots h_{SBlock}}^{SBlock}, \\
&\boldsymbol{Z}_{1\cdots h_{short}}^{short}, \boldsymbol{Z}_{1\cdots h_{long}}^{long}, \boldsymbol{Z}_{1\cdots h_{TRand}}^{TRand}, \boldsymbol{Z}_{1\cdots h_{TTube}}^{TTube}, \\
&\boldsymbol{Z}_{1\cdots h_{TBlock}}^{TBlock}, \boldsymbol{Z}_{1\cdots h_{temp}}^{temp}, \boldsymbol{Z}_{1\cdots h_t}^{t})\boldsymbol{W}^l,
\end{aligned} \tag{10}$$

where cat represents concatenation, all of Z in the Equation stand for output concatenations, and all of h stand for the corresponding number of attention heads, respectively, and $\boldsymbol{W}^l$ is a learnable projection matrix. Additionally, we apply a position-wise fully connected feed-forward network to the output of the Multiple Attention Fusion to obtain the outputs $\mathcal{X}_o \in \mathbb{R}^{T \times N \times d}$. Consistent with the original Transformer architecture, layer normalization and residual connections are incorporated in this step.

Output Projection

We employ an output projection, consisting of 1×1 convolution, after each spatial-temporal encoder layer to convert the outputs $\mathcal{X}_o$ into a temporary dimension $\mathcal{X}_{temp} \in \mathbb{R}^{T \times N \times d_{temp}}$, where d_{temp} is the temporary dimension. Finally, the final hidden state $\mathcal{X}_{hid} \in \mathbb{R}^{T \times N \times d_{temp}}$ is obtained by summing the outputs of each output projection. We use $\boldsymbol{H}_i \in \mathbb{R}^{N \times d_{temp}}$ to represent the embedding matrix of the i-th time slice. And $\boldsymbol{h}_{i,n} \in \mathbb{R}^{d_{temp}}$ denotes the n-th row in $\boldsymbol{H}_i$, i.e., the embedding of n-th region(node) r_n.

Adaptive Data Augmentation

Regarding the traffic patterns of the transportation system, an adaptive data augmentation scheme has been designed for

the traffic flow tensor and road network data to improve the model's perception of heterogeneous regional dependencies.

Traffic Flow Augmentation. Inspired by data augmentation strategies(Zhu et al. 2021; Ji et al. 2023), an augmentation operator is introduced for the constructed traffic tensor $\boldsymbol{X}_{t-T+1} \cdots \boldsymbol{X}_t$, which adapts to the learned time-aware traffic pattern dependencies of each region. Specifically, our goal is to mask less relevant traffic volume at the τ-th time step of the region r_n to mitigate noise perturbation. This is based on a derived mask probability $\rho_{\tau,n}$ drawn from a Bernoulli distribution, $\rho_{\tau,n} \sim \text{Bern}(1-p_{\tau,n})$. A higher $\rho_{\tau,n}$ value indicates that the traffic volume $x_{\tau,n}$ at the τ-th time step for region r_n is more likely to be masked, due to its lower relevance to the overall traffic regularities of the region r_n. The augmented data with this traffic-level augmentation is denoted as $\widetilde{\boldsymbol{X}}_{t-T+1} \cdots \widetilde{\boldsymbol{X}}_t$.

Road Network Augmentation. In addition to traffic flow augmentation, road network augmentation is applied to the road network G(Ji et al. 2023; Hu et al. 2024). This approach allows our model to not only mitigate biases in region connections with low inter-correlated traffic patterns but also to capture long-range regional dependencies within the global urban context.

Finally, augmented flow data and road network data $\left[\widetilde{\boldsymbol{X}}_{(t-T+1)}, \cdots, \widetilde{\boldsymbol{X}}_t; \widetilde{\mathcal{G}}\right]$ are obtained.

Spatial Heterogeneity Modeling (SHM)

To enable region embeddings to efficiently capture spatial heterogeneity through auxiliary self-supervised signals, we introduce a spatial heterogeneity modeling (SHM) module based on self-supervised learning with soft clustering(Ji et al. 2023).

Initially, regions are projected into multiple latent representation spaces, each corresponding to distinct urban functions, such as residential zones and commercial districts. Subsequently, we generate a set of K cluster embeddings, denoted as $\{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_K\}$, where each $\boldsymbol{c}_i$ represents a latent factor used for clustering regions based on their functional characteristics. Formally, the clustering process is performed using the following equation:

$$\widetilde{\boldsymbol{z}}_{i,n,k} = \boldsymbol{c}_k^\top \widetilde{\boldsymbol{h}}_{i,n} \tag{11}$$

Here, $\widetilde{\boldsymbol{h}}_{i,n} \in \mathbb{R}^{d_{temp}}$ is the region embedding of region r_n encoded from the augmented traffic flow and road network of i-th time slice. The term $\widetilde{\boldsymbol{z}}_{i,n,k}$ represents the estimated relevance score between the embedding of region r_n and the embedding $\boldsymbol{c}_k$ of the k-th cluster. Subsequently, the cluster assignment of region r_n is generated as $\widetilde{\boldsymbol{z}}_{i,n} = (\widetilde{\boldsymbol{z}}_{i,n,1}, \ldots, \widetilde{\boldsymbol{z}}_{i,n,K})^\top$, capturing the degree of association with each cluster.

To incorporate self-supervised signals within a heterogeneity-aware soft clustering framework, we introduce an auxiliary learning task designed to predict cluster assignments based on region embeddings $\boldsymbol{h}_{i,n}$, which are encoded from the original road network $\mathcal{G}$. The predicted cluster assignment score $\hat{\boldsymbol{z}}_{i,n,k}$ for $\widetilde{\boldsymbol{z}}_{i,n,k}$ is computed as: $\hat{\boldsymbol{z}}_{i,n,k} = \boldsymbol{c}_k^\top \boldsymbol{h}_{i,n}$,where $\hat{\boldsymbol{z}}_{i,n,k}$ represents the predicted assignment score for $\widetilde{\boldsymbol{z}}_{i,n,k}$. The self-supervised enhancement task is then optimized using the following objective function:

$$\ell\left(\boldsymbol{h}_{i,n}, \tilde{\boldsymbol{z}}_{i,n}\right) = -\sum_k \tilde{z}_{i,n,k} \log \frac{\exp\left(\hat{z}_{i,n,k}/\gamma\right)}{\sum_j \exp\left(\hat{z}_{i,n,j}/\gamma\right)}, \tag{12}$$

where γ is a parameter that controls the smoothing degree of the softmax output. The overall self-supervised objective across all regions is defined as follows:

$$\mathcal{L}_S = \sum_{i=1}^{t-T+1} \sum_{n=1}^{N} \ell\left(\boldsymbol{h}_{i,n}, \tilde{\boldsymbol{z}}_{i,n}\right). \tag{13}$$

To reflect the true distribution of regional features in urban space, we employ a distribution regularization strategy for regional clustering(Ji et al. 2023). SHM module significantly enhances the regional embedding $\boldsymbol{h}_n$ to capture the spatial heterogeneity within the urban environment. The integration of the SHM module enables the model to capture complex spatial dependencies, enhancing the representation of urban spatial structures. This improved embedding is essential for accurately modeling traffic patterns by reflecting the unique characteristics and interactions of urban areas.

Temporal Heterogeneity Modeling (THM)

A Temporal Heterogeneity Modeling (THM) module is introduced to systematically capture and analyze variations in traffic patterns at specific time steps.(Ji et al. 2023).

Initially, we combine the time-dimension embeddings encoded from both the original and augmented data as follows: $\boldsymbol{v}_{t,n} = \boldsymbol{w_1} \odot \boldsymbol{h}_{t,n} + \boldsymbol{w_2} \odot \tilde{\boldsymbol{h}}_{t,n}$, where $\odot$ denotes the Hadamard product, and $\boldsymbol{w_1}$ and $\boldsymbol{w_2}$ are learnable parameters. Subsequently, the urban-level representation $\boldsymbol{s}_t$ at time step t is generated by aggregating embeddings from all regions (σ represents the sigmoid function): $\boldsymbol{s}_t = \sigma\left(\frac{1}{N}\sum_{n=1}^{N} \boldsymbol{v}_{t,n}\right)$. To enhance the discriminatory ability of representations across different time steps, district-level and urban-level embeddings $(\boldsymbol{v}_{t,n}, \boldsymbol{s}_t)$ corresponding to the same time step are treated as positive pairs within the self-supervised learning (SSL) task, while embeddings from different time steps are considered as negative pairs. This approach ensures that positive pairs align city-wide traffic patterns—such as rush hours and weather impacts—within a specific time step, thereby reinforcing temporal consistency. Conversely, negative pairs enhance the model's ability to capture temporal heterogeneity by effectively differentiating between distinct time steps. Formally, the time-heterogeneity-enhanced SSL task is optimized by minimizing the following loss function using the cross-entropy metric:

$$\mathcal{L}_T = -\sum_{n=1}^{N} \log g\left(\boldsymbol{v}_{t,n}, \boldsymbol{s}_t\right) - \sum_{n=1}^{N} \log\left(1 - g\left(\boldsymbol{v}_{t',n}, \boldsymbol{s}_t\right)\right), \tag{14}$$

where g is an abbreviation for the function $g\left(\boldsymbol{v}_{t,n}, \boldsymbol{s}_t\right) = \sigma\left(\boldsymbol{v}_{t,n}^\top \boldsymbol{W}_3 \boldsymbol{s}_t\right)$. $\boldsymbol{W}_3 \in \mathbb{R}^{N \times N}$ is a learnable transformation matrix.

Datasets	Nodes	Edges	Timesteps	Interval	Range
PeMS04	307	340	16992	5min	01/01/18–02/28/18
PeMS07	883	866	28224	5min	05/01/17–08/31/17
PeMS08	170	295	17856	5min	07/01/16–08/31/16
NYCTaxi	75 (15×5)	484	17520	30min	01/01/14–12/31/14
CHIBike	270 (15×18)	1966	4416	30min	07/01/20–09/30/20
T-Drive	1024 (32×32)	7812	3600	60min	02/01/15–06/30/15

Table 1: Data Description

Model	PeMS04			PeMS07			PeMS08		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
DCRNN	22.73	14.75	36.57	23.63	12.28	36.51	18.18	11.23	28.17
STGCN	21.75	13.87	34.76	22.89	11.98	35.44	17.83	11.21	27.12
GWNET	19.35	13.30	31.71	21.22	9.07	34.11	15.06	9.51	24.85
MTGNN	19.07	12.96	31.56	20.82	9.03	34.08	15.39	10.17	24.93
STSGCN	21.18	13.88	33.64	24.26	10.20	39.03	17.13	10.96	26.78
STFGNN	19.83	13.02	31.87	22.07	9.21	35.80	16.63	10.54	26.20
STGODE	20.84	13.78	32.82	22.97	10.14	36.19	16.81	10.62	26.24
STGNCDE	19.21	12.77	31.08	20.62	8.86	34.03	15.45	9.92	24.81
STTN	19.47	13.63	31.91	21.34	9.93	34.58	15.48	10.34	24.96
GMAN	19.13	13.19	31.60	20.96	9.05	34.09	15.30	10.13	24.91
TFormer	18.91	12.71	31.34	20.75	8.97	34.06	15.19	9.92	24.88
ASTGNN	18.60	12.63	31.02	20.61	8.86	34.01	14.97	9.48	24.71
PDFormer	18.32	12.10	29.96	19.83	8.52	32.87	13.58	9.04	23.50
ST-SSL	18.56	12.74	31.01	20.36	9.64	33.98	14.73	10.05	24.49
GraphST	19.10	12.91	31.57	20.87	10.31	33.97	15.13	10.61	24.84
STD-MAE	*17.80	*12.07	*29.25	*18.65	*8.47	*31.71	*13.44	*8.98	*22.47
Ours	**17.06**	**11.32**	**28.11**	**18.53**	**8.41**	**31.44**	**12.05**	**8.11**	**20.15**
Improve	**4.15%**	**6.21%**	**3.89%**	**0.64%**	**0.70%**	**0.85%**	**10.34%**	**9.68%**	**10.32%**

Table 2: Performance on Graph-based Datasets.

Model Trainning

During the training process of SSL-STMFormer, the hidden embeddings $\mathcal{X}_{\text{hid}} \in \mathbb{R}^{T\times N\times d_{\text{temp}}}$ are fed into MLP to predict future traffic flow $\hat{\boldsymbol{x}}_{t+1,t+T'} = \left[\boldsymbol{X}_{(t+1)}, \cdots, \boldsymbol{X}_{(t+T')}\right]$. The model is optimized by minimizing the loss function below:

$$\mathcal{L}_P = \text{MSEloss}(\boldsymbol{x}_{t+1,t+T'}, \hat{\boldsymbol{x}}_{t+1,t+T'}), \tag{15}$$

Finally, the overall loss function is formulated by integrating the spatial and temporal heterogeneity modeling losses, as described in Equations (13) and (14), into the joint learning objective $\mathcal{L}_{\text{joint}} = \mathcal{L}_p + \mathcal{L}_s + \mathcal{L}_t$.

Experiments

Datasets. We evaluate the performance of SSL-STMFormer using six real-world public traffic datasets. These include three graph-based highway traffic datasets (PeMS04, PeMS07, PeMS08)(Song et al. 2020) and three grid-based citywide traffic datasets (NYCTaxi(Liu et al. 2020), CHIBike(Wang et al. 2021), T-Drive(Pan et al. 2019)). Detailed descriptions of these datasets are provided in Tab. 1.

Baselines. We selected 19 baselines for comparison with our proposed model, categorizing them into three distinct groups: (1) grid-based dataset models, including: STResNet(Zhang, Zheng, and Qi 2017), DMVSTNet(Yao et al. 2018) and DSAN(Lin et al. 2020). (2) graph neural network-based models, including: DCRNN(Li et al. 2018), STGCN(Yu, Yin, and Zhu 2018), GWNET(Wu et al. 2019), MTGNN(Wu et al. 2020), STSGCN(Song et al. 2020), STFGNN(Li and Zhu 2021), STGODE(Fang et al. 2021) and STGNCDE(Choi et al. 2022). (3) self-attention-based models, including: STTN(Xu et al. 2020), GMAN(Zheng et al. 2020), TFormer(Yan, Ma, and Pu 2021), PDFormer(Jiang et al. 2023) and ASTGNN(Guo et al. 2021). (4) Self-supervised learning-based and pre-training-based models, including: ST-SSL(Ji et al. 2023), GraphST(Zhang et al. 2023), STD-MAE(Gao et al. 2024). This classification allows for a comprehensive evaluation of our model's performance against a diverse set of established approaches.

Dataset Processing and Evaluation Metrics. To ensure consistency with most studies, the three graph-based datasets were divided into training, validation, and test sets in a 6:2:2 ratio. Data from the past hour (12 steps) was employed to predict the traffic flow for the next hour (12 steps). Additionally, the three grid-based datasets were split in a 7:1:2 ratio, utilizing traffic inflow and outflow data from the past six steps to predict the inflow and outflow for the next single step. In our experiments, three metrics are employed for evaluation: Mean Average Error (MAE), Mean Average Percentage Error (MAPE%) and Root Mean Squared Error (RMSE).

Performance Comparison

The results for the graph-based and grid-based datasets, compared to the baselines, are presented in Tab. 2 and Tab. 3, respectively. The best results are highlighted in bold, and the second-best results are marked with an asterisk (*). The analysis of the results presented in Tab. 2 and Tab. 3 leads to the following conclusions: (1) On the graph-based dataset, SSL-STMFormer achieves superior performance across all metrics, demonstrating average improvements of 5.04%, 5.53%, and 5.02% in MAE, MAPE, and RMSE, respectively, compared to the next best results. (2) On the grid-based datasets, SSL-STMFormer significantly outperforms all baseline models across all metrics and datasets, achieving average improvements of 10.91%, 9.86%, and 7.18% in MAE, MAPE, and RMSE, respectively, over the next best results.

Ablation Study

To evaluate the effectiveness of the different components in SSL-STMFormer, we compared SSL-STMFormer with the following variants: **w/o SSL:** This variant omits the self-supervised learning component, thereby impairing the model's ability to capture spatial and temporal heterogeneity. **w/o EA:** This variant excludes temporal and spatial entanglement-aware modules, limiting the ability of the model to perceive spatio-temporal entanglement. **w/o SEA:** This variant removes the spatial entanglement-aware module, resulting in every node being connected to all others. **w/o TEA:** This variant excludes the temporal entanglement-aware module. **w/ STEA:** This variant replaces the separate temporal and spatial entanglement-aware methods with a hybrid entanglement-aware method.

Tab.4 presents a comparative analysis of these variants on the PeMS08 and T-Drive datasets. The ablation experiment results reveal the following insights: (1) The performance of the hybrid temporal-spatial masking strategy is inferior to that of SSL-STMFormer. Moreover, removing either the temporal or spatial masking individually leads to a degradation in model performance. This underscores the

Model	NYCTaxi						T-Drive						CHIBike					
	Inflow			outflow			Inflow			outflow			Inflow			outflow		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
STResNet	14.49	14.54	24.05	12.79	14.36	20.63	19.63	17.83	34.89	19.61	18.50	34.59	4.76	31.38	6.70	4.62	30.57	6.55
DMVSTNet	14.37	14.31	23.73	12.56	14.31	20.40	19.59	17.68	34.47	19.53	17.62	34.30	4.68	32.11	6.63	4.59	31.31	6.45
DSAN	14.28	14.20	23.58	12.46	14.27	20.29	19.38	17.46	34.31	19.29	17.37	34.26	4.61	31.62	6.69	4.49	31.25	6.36
DCRNN	14.42	14.35	23.87	12.82	14.34	20.06	22.12	17.75	38.65	21.75	17.38	38.16	4.23	31.26	5.99	4.21	30.82	5.82
STGCN	14.37	14.21	23.86	12.54	14.09	19.96	21.37	17.53	38.05	20.91	16.98	37.61	4.21	31.22	5.95	4.14	30.78	5.77
GWNET	14.31	14.19	23.79	12.28	13.68	19.61	19.55	17.18	36.15	19.55	15.93	36.19	4.15	31.15	5.91	4.10	30.69	5.69
MTGNN	14.19	13.98	23.66	12.27	13.65	19.56	18.98	17.05	35.38	18.92	15.76	35.99	4.11	31.14	5.80	4.08	30.56	5.66
STSGCN	15.60	15.20	26.19	13.23	14.69	21.65	23.82	18.54	41.18	24.28	19.04	42.25	4.25	32.99	5.94	4.26	32.61	5.87
STFGNN	15.33	14.86	26.11	13.17	14.58	21.62	22.14	18.09	40.07	22.87	18.98	41.03	4.23	32.22	5.93	4.26	32.32	5.87
STGODE	14.62	14.79	25.44	12.83	14.39	20.20	21.51	17.57	38.21	22.70	18.50	40.28	4.16	31.16	5.92	4.12	30.72	5.69
STGNCDE	14.28	14.17	23.74	12.27	13.68	19.60	19.34	17.13	36.09	19.23	15.87	36.14	4.12	31.15	5.91	4.09	30.59	5.67
STTN	14.35	14.20	23.84	12.37	13.76	19.82	20.58	17.32	37.22	20.44	15.99	37.06	4.16	31.20	5.93	4.11	30.70	5.72
GMAN	14.26	14.11	23.72	12.27	13.67	19.59	19.24	17.11	35.98	18.96	15.78	36.12	4.11	31.15	5.91	4.09	30.66	5.67
TFormer	13.99	13.91	23.48	12.21	13.61	19.52	18.82	16.91	34.47	18.88	15.67	35.21	4.07	31.14	5.87	4.03	30.64	5.63
ASTGNN	13.84	13.69	23.17	12.11	13.60	19.20	18.79	16.10	33.87	18.79	15.58	33.99	4.06	31.13	5.81	3.98	30.61	5.60
PDFormer	13.15	12.74	21.95	11.57	12.82	18.39	17.83	14.71	31.60	17.74	14.64	31.50	3.95	*30.21	5.55	*3.83	*29.91	*5.40
ST-SSL	12.97	12.39	21.74	9.78	11.93	16.86	16.21	13.11	31.39	16.69	13.72	31.04	4.03	31.07	5.76	3.94	30.48	5.49
GraphST	11.67	12.07	19.32	10.79	12.14	18.05	15.97	13.04	31.06	15.84	12.97	30.81	4.01	31.06	5.59	3.86	30.11	5.46
STD-MAE	*10.71	*11.12	*18.94	*9.53	*11.29	*16.49	*15.42	*12.71	*30.53	*14.93	*11.80	*30.03	*3.91	30.93	*5.41	3.89	30.17	5.52
Ours	**9.49**	**10.06**	**18.12**	**8.98**	**10.33**	**15.84**	**12.85**	**9.68**	**29.10**	**12.79**	**9.60**	**29.00**	**3.49**	**29.81**	**5.09**	**3.44**	**29.34**	**5.04**
Improve	**11.39%**	**9.53%**	**5.77%**	**8.50%**	**3.94%**	**16.66%**	**10.34%**	**23.83%**	**4.68%**	**14.33%**	**18.64%**	**3.42%**	**10.74%**	**1.32%**	**5.91%**	**10.18%**	**1.90%**	**6.66%**

Table 3: Performance on Grid-based Datasets.

Model	PeMS08			T-Drive(inflow)			T-Drive(outflow)		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
w/o SSL	13.70	9.16	23.30	14.14	10.76	30.90	14.15	10.69	30.92
w/o EA	13.59	9.14	23.52	14.20	11.04	30.79	14.21	11.00	30.78
w/o SEA	13.46	9.00	23.22	13.51	10.60	29.83	13.47	10.64	29.80
w/o TEA	13.40	9.12	23.32	13.47	10.16	29.61	13.38	10.06	29.50
w/ STEA	13.86	9.31	23.61	13.65	10.89	29.90	13.60	10.82	29.83
SSL-STMFormer	**12.05**	**8.11**	**20.15**	**12.85**	**9.68**	**29.10**	**12.79**	**9.60**	**29.00**

Table 4: Performance of Ablation Experiment.

importance of employing masking strategies to identify critical node pairs, thereby demonstrating the necessity of these strategies in accurately modeling real traffic environments. (2) The reduction in model performance when the SSL component is removed can be attributed to the variant's inability to account for complex spatio-temporal heterogeneity. This omission prevents the model from recognizing underlying spatio-temporal features, which are crucial for accurate traffic flow prediction.

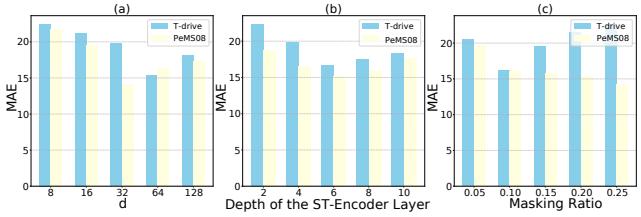

Figure 3: Effect of hyperparameters on PeMS08 and T-Drive.

Effect of Different Parameters

In this section, we conduct a comprehensive investigation into the impact of various parameters on the model's performance. Specifically, we examine the hidden dimension d across the set $\{8, 16, 32, 64, 128\}$, assessing how different dimensionalities influence the model's capacity to capture and represent the underlying traffic patterns. Additionally, we explore the depth of the spatio-temporal encoder layer, over the set $\{2, 4, 6, 8, 10\}$, to determine the optimal depth for accurately modeling the intricate spatio-temporal dependencies. Furthermore, we evaluate the masking probability over the set $\{0.05, 0.1, 0.15, 0.2, 0.25\}$, analyzing how varying degrees of masking affect the model's ability to generalize and handle data with missing or noisy information. These parameter searches are crucial for fine-tuning the model to achieve the best performance across different datasets and traffic scenarios. Based on our extensive experimentation, we present the results in Fig. 3. These results inform our selection of optimal parameters for the model.

Related Work

Transformer. Large-scale pre-trained models based on the Transformer, such as BERT, have achieved considerable success in the NLP community(Devlin et al. 2019). More recently, with the advent of models like GPT, it has become evident that the Transformer architecture excels in representation learning(Liu et al. 2023; Kalyan 2023; Yenduri et al. 2024).

Self-Supervised Learning for Representation Learning. Self-supervised learning seeks to extract valuable information from input data to enhance the quality of its representation(Ji et al. 2022). This approach typically involves augmenting the input data and devising auxiliary tasks that serve as pseudo-labels for representation learning (Ren et al. 2021).

Conclusion

In this study, we propose SSL-STMFormer, a novel model for traffic flow prediction that combines self-supervised learning with spatio-temporal attention and entanglement-aware methods. This approach effectively captures dynamic traffic patterns and complex spatio-temporal dependencies, while enhancing the model's ability to interpret real-world traffic complexities. Extensive experiments on real-world datasets validate its superior performance. Future work will extend SSL-STMFormer to other spatio-temporal prediction tasks, such as crime rate forecasting (Zhou et al. 2023).

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (Nos.T2293771 and 42361144718), the Sichuan Provincial Science and Technology Program (Grant No.2023NSFSC1919), and the Sichuan Provincial Natural Science Foundation (No.2024NSFTD0042 and 2024NSFSC0506). The funders had no role in the study design, data collection, analysis, decision to publish, or preparation of the manuscript.

References

Belkin, M.; and Niyogi, P. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6): 1373–1396.

Berndt, D. J.; and Clifford, J. 1994. Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd international conference on knowledge discovery and data mining*, 359–370.

Bui, K.-H. N.; Cho, J.; and Yi, H. 2022. Spatial-temporal graph neural network for traffic forecasting: An overview and open research issues. *Applied Intelligence*, 52(3): 2763–2774.

Choi, J.; Choi, H.; Hwang, J.; and Park, N. 2022. Graph neural controlled differential equations for traffic forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 6367–6374.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Ding, J.; Yang, C.; Wang, Y.; Li, P.; Wang, F.; Kang, Y.; Wang, H.; Liang, Z.; Zhang, J.; Han, P.; et al. 2023. Influential factors of intercity patient mobility and its network structure in China. *Cities*, 132: 103975.

Djahel, S.; Doolan, R.; Muntean, G.-M.; and Murphy, J. 2014. A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches. *IEEE Communications Surveys & Tutorials*, 17(1): 125–151.

Fang, Z.; Long, Q.; Song, G.; and Xie, K. 2021. Spatial-temporal graph ode networks for traffic flow forecasting. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 364–373.

Feichtenhofer, C.; Li, Y.; He, K.; et al. 2022. Masked autoencoders as spatiotemporal learners. *Advances in Neural Information Processing Systems*, 35: 35946–35958.

Gao, H.; Jiang, R.; Dong, Z.; Deng, J.; Ma, Y.; and Song, X. 2024. Spatial-Temporal-Decoupled Masked Pre-training for Spatiotemporal Forecasting. In Larson, K., ed., *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, 3998–4006. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Guo, S.; Lin, Y.; Wan, H.; Li, X.; and Cong, G. 2021. Learning dynamics and heterogeneity of spatial-temporal graph data for traffic forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 34(11): 5415–5428.

Hu, Z.; Nakagawa, S.; Zhuang, Y.; Deng, J.; Cai, S.; Zhou, T.; and Ren, F. 2024. Hierarchical Denoising for Robust Social Recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 1–14.

Ji, J.; Wang, J.; Huang, C.; Wu, J.; Xu, B.; Wu, Z.; Zhang, J.; and Zheng, Y. 2023. Spatio-temporal self-supervised learning for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 4356–4364.

Ji, J.; Wang, J.; Wu, J.; Han, B.; Zhang, J.; and Zheng, Y. 2022. Precision CityShield Against Hazardous Chemicals Threats via Location Mining and Self-Supervised Learning. KDD '22, 3072–3080. New York, NY, USA: Association for Computing Machinery. ISBN 9781450393850.

Jiang, J.; Han, C.; Zhao, W. X.; and Wang, J. 2023. Pdformer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 4365–4373.

Jiang, W.; and Luo, J. 2022. Graph neural network for traffic forecasting: A survey. *Expert Systems with Applications*, 207: 117921.

Jin, G.; Liang, Y.; Fang, Y.; Shao, Z.; Huang, J.; Zhang, J.; and Zheng, Y. 2023. Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

Kalyan, K. S. 2023. A survey of GPT-3 family large language models including ChatGPT and GPT-4. *Natural Language Processing Journal*, 100048.

Li, M.; and Zhu, Z. 2021. Spatial-temporal fusion graph neural networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 4189–4196.

Li, Y.; Yu, R.; Shahabi, C.; and Liu, Y. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations (ICLR '18)*.

Lin, H.; Bai, R.; Jia, W.; Yang, X.; and You, Y. 2020. Preserving dynamic attention for long-term spatial-temporal prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 36–46.

Liu, L.; Zhen, J.; Li, G.; Zhan, G.; He, Z.; Du, B.; and Lin, L. 2020. Dynamic spatial-temporal representation learning for traffic flow prediction. *IEEE Transactions on Intelligent Transportation Systems*, 22(11): 7169–7183.

Liu, X.; Zheng, Y.; Du, Z.; Ding, M.; Qian, Y.; Yang, Z.; and Tang, J. 2023. GPT understands, too. *AI Open*.

Pan, Z.; Liang, Y.; Wang, W.; Yu, Y.; Zheng, Y.; and Zhang, J. 2019. Urban traffic prediction from spatio-temporal data using deep meta learning. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 1720–1730.

Qiu, C.; Zhang, Y.; Feng, Z.; Zhang, P.; and Cui, S. 2018. Spatio-temporal wireless traffic prediction with recurrent neural network. *IEEE Wireless Communications Letters*, 7(4): 554–557.

Ren, H.; Wang, J.; Zhao, W. X.; and Wu, N. 2021. RAPT: Pre-training of Time-Aware Transformer for Learning Robust Healthcare Representation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, 3503–3511. New York, NY, USA: Association for Computing Machinery. ISBN 9781450383325.

Song, C.; Lin, Y.; Guo, S.; and Wan, H. 2020. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 914–921.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Vinayakumar, R.; Soman, K.; and Poornachandran, P. 2017. Applying deep learning approaches for network traffic prediction. In *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2353–2358. IEEE.

Wang, J.; Jiang, J.; Jiang, W.; Li, C.; and Zhao, W. X. 2021. Libcity: An open library for traffic prediction. In *Proceedings of the 29th international conference on advances in geographic information systems*, 145–148.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; and Zhang, C. 2020. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 753–763.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; and Zhang, C. 2019. Graph WaveNet for Deep Spatial-Temporal Graph Modeling. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, 1907–1913. International Joint Conferences on Artificial Intelligence Organization.

Xu, M.; Dai, W.; Liu, C.; Gao, X.; Lin, W.; Qi, G.-J.; and Xiong, H. 2020. Spatial-temporal transformer networks for traffic flow forecasting. *arXiv preprint arXiv:2001.02908*.

Yan, H.; Ma, X.; and Pu, Z. 2021. Learning dynamic and hierarchical traffic spatiotemporal features with transformer. *IEEE Transactions on Intelligent Transportation Systems*, 23(11): 22386–22399.

Yao, H.; Wu, F.; Ke, J.; Tang, X.; Jia, Y.; Lu, S.; Gong, P.; Ye, J.; and Li, Z. 2018. Deep multi-view spatial-temporal network for taxi demand prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Yenduri, G.; Ramalingam, M.; Selvi, G. C.; Supriya, Y.; Srivastava, G.; Maddikunta, P. K. R.; Raj, G. D.; Jhaveri, R. H.; Prabadevi, B.; Wang, W.; et al. 2024. Gpt (generative pre-trained transformer)–a comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions. *IEEE Access*.

Yin, X.; Wu, G.; Wei, J.; Shen, Y.; Qi, H.; and Yin, B. 2021. Deep learning on traffic prediction: Methods, analysis, and future directions. *IEEE Transactions on Intelligent Transportation Systems*, 23(6): 4927–4943.

Yu, B.; Yin, H.; and Zhu, Z. 2018. Spatio-temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*.

Yuan, Y.; Ding, J.; Feng, J.; Jin, D.; and Li, Y. 2024. UniST: A Prompt-Empowered Universal Model for Urban Spatio-Temporal Prediction. *arXiv preprint arXiv:2402.11838*.

Zhang, J.; Zheng, Y.; and Qi, D. 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Zhang, Q.; Huang, C.; Xia, L.; Wang, Z.; Yiu, S. M.; and Han, R. 2023. Spatial-Temporal Graph Learning with Adversarial Contrastive Adaptation. In *International Conference on Machine Learning*, 41151–41163. PMLR.

Zheng, C.; Fan, X.; Wang, C.; and Qi, J. 2020. Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 1234–1241.

Zhou, S.; He, D.; Chen, L.; Shang, S.; and Han, P. 2023. Heterogeneous region embedding with prompt learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4981–4989.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2021. Graph contrastive learning with adaptive augmentation. In *Proceedings of the web conference 2021*, 2069–2080.