



OPEN An improved transformer based traffic flow prediction model

Shipeng Liu & Xingjian Wang

Traffic flow prediction is a key challenge in intelligent transportation, and the ability to accurately forecast future traffic flow directly affects the efficiency of urban transportation systems. However, existing deep learning-based prediction models suffer from the following issues: First, CNN- or RNN-based models are limited by their architecture and unsuitable for modeling long-term sequences. Second, most Transformer-based methods focus solely on the traffic flow data itself during embedding, neglecting the implicit information behind the traffic data. This implicit information includes behavioral trends, community and surrounding traffic patterns, urban weather, semantic information, and temporal periodicity. Third, methods using the original multi-head self-attention mechanism calculate attention scores point by point in the temporal dimension without utilizing contextual information, which to some extent leads to less accurate attention computation. Fourth, existing methods struggle to capture long and short-range spatial dependencies simultaneously. To address these four issues, we propose an IEEAFormer technique (Implicit-information Embedding and Enhanced Spatial-Temporal Multi-Head Attention Transformer). First, it adopts a Transformer architecture and incorporates an embedding layer to capture implicit information in the input. Secondly, the method replaces the traditional multi-head self-attention with time-environment-aware self-attention in the temporal dimension, enabling each node to perceive the contextual environment. Additionally, the technique uses two unique graph mask matrices in the spatial dimension. It employs a novel parallel spatial self-attention architecture to capture both long-range and short-range dependencies in the data simultaneously. The results verified on four real-world traffic datasets show that the proposed IEEAFormer outperforms most existing models regarding prediction performance.

Keywords Traffic flow prediction, Deep learning method, Intelligent transportation systems

In recent years, many countries have focused on developing Intelligent Transportation Systems (ITS) to build smart cities. Traffic flow prediction, as an essential component of ITS, plays a vital role in optimizing traffic resource management. Due to its immense practical value, significant efforts have been made in the past few years to achieve accurate and long-term traffic forecasting. One of the key challenges in traffic flow prediction lies in effectively modeling traffic data in both the temporal and spatial dimensions. Despite certain advancements in existing traffic flow prediction methods, several issues remain to be addressed.

Early researchers attempted to model long sequence data using Convolutional Neural Networks (to process gridded data and capture spatial dependencies. CNNs) and Recurrent Neural Networks (were commonly used to capture temporal dependencies RNNs). However, RNNs suffer from the vanishing gradient problem in long sequence tasks¹, and CNNs, which apply a sliding window mechanism to convolve sequence data, are limited by their restricted receptive field, reducing their ability to model long-range dependencies effectively². Temporal Convolutional Networks (TCNs)³, a specialized form of 1D CNN, attempted to enhance long-term dependency capture by employing dilated convolutions in combination with stacked convolutional layers⁴. However, this approach requires information to pass through multiple layers to establish effective connections between any two points in the sequence. This process may weaken the model's ability to capture and learn long-range dependencies.

Furthermore, most Transformer-based methods primarily focus on the data representation itself⁵, neglecting the potentially valuable underlying information. This implicit information not only reveals the complex relationships between traffic flows but also more deeply reflects dynamic features such as behavioral trends, traffic periodicity, and traffic patterns both within and outside the region. Traditional approaches often tend to develop complex and large-scale models⁶. While some progress has been made, the complexity of these models frequently leads to a decline in performance.

College of Computer and Control Engineering, Northeast Forestry University, HeXing Road, Harbin, China. email: jianxingwang@126.com

Third, most self-attention-based methods^{7,8} calculate the attention between query and key on a point-by-point basis in the temporal dimension, without accounting for the inherent local contextual information present in sequential data. To illustrate, consider a scenario where a traffic accident occurs at time A at a certain point, causing the traffic flow to come to a near halt, with the traffic volume dropping to an extremely low level due to most vehicles being blocked at that point. At this time, the traffic volume at point A happens to be numerically similar to the traffic volume at point B, which corresponds to midnight when there is minimal traffic. In this case, traditional multi-head self-attention would mistakenly match time A and time B, assuming they share a higher similarity simply because their values are similar, when in fact, there is almost no connection between these two-time points. Therefore, applying traditional multi-head self-attention to the temporal dimension of traffic flow sequences may lead to unreasonable matching.

Finally, different locations within a city may exhibit similar traffic flow patterns, implying that spatial dependencies can be both long-range and short-range. Widely used Graph Neural Network (GNN) methods⁹ are limited by their architecture and tend to be over-smooth when capturing long-range dependencies. On the other hand, methods based on multi-head self-attention⁷ compute the similarity between all points in the spatial dimension, which significantly increases computational complexity.

To address the above-mentioned issues, this paper proposes a Transformer model that incorporates implicit information embedding and improves the spatial-temporal multi-head attention mechanism, referred to as IEEAFormer (Implicit-information Embedding and Enhanced Spatial-Temporal Multi-Head Attention Transformer). Compared to CNN- or RNN-based methods, the Transformer architecture effectively mitigates the vanishing gradient problem and is naturally suited for modeling long sequences. Additionally, IEEAFormer allows for the embedding of implicit information across multiple dimensions. It also employs a time-aware self-attention mechanism, which, while maintaining a global receptive field, emphasizes the inherent contextual information in sequential data. Furthermore, a parallel spatial self-attention mechanism based on long- and short-range masking is utilized to capture spatial information in traffic data. In summary, the contributions of this paper are as follows:

- (1) This paper proposes the IEEAFormer model, which is based on a spatial-temporal attention mechanism and effectively captures the spatial-temporal relationships to achieve accurate traffic flow prediction.
- (2) This paper introduces an embedding layer into the input data that captures implicit information, enabling the model to learn knowledge across multiple dimensions. This is a simple yet highly effective representation technique, often overlooked, but has proven its powerful potential in various fields, allowing the latent features of traffic flow data to be more comprehensively captured and expressed.
- (3) In the spatial-temporal Transformer encoder, a time-aware self-attention mechanism is used to replace the traditional multi-head self-attention, enabling the model to learn more accurate node-level contextual information. Additionally, a unique long-range and short-range masking method is employed, combined with parallel spatial attention, to separately model long-range and short-range spatial information.
- (4) We tested IEEAFormer on four real-world datasets. The model performed exceptionally well, demonstrating its unique advantages in traffic flow prediction.

Problem definition

In this section, we introduce the traffic flow prediction problem and the symbols used in IEEAFormer:

Definition 1 Road Network: We use the graph $G = \{v, \varepsilon, A\}$ to represent the road network. Here, $v = \{v_1 \dots v_n\}$ represents the set of nodes, with n being the number of nodes in the road network, $\varepsilon \subseteq v \times v$ represents the edges connecting the nodes, and A is the adjacency matrix of the graph.

Definition 2 Traffic Flow Tensor: The traffic flow tensor is represented by $X \in \mathbb{R}^{T \times N \times j}$, where T denotes T time steps, N denotes the number of nodes in the road graph, and j is the dimension of the input features. In this problem, the value of j is 1, representing the traffic flow.

Definition 3 Traffic Flow Prediction: Traffic flow prediction aims to predict the traffic flow in future time intervals based on the traffic flow from a past period. Assuming we have trained an IEEAFormer, this problem can be formulated as

$$\{X_1 \dots X_T\} \xrightarrow{\text{IEEAFormer}} \{X_{T+1} \dots X_{T'}\} \quad (1)$$

where T represents the time steps.

Methods

Our model diagram, as shown in Figure 1, includes an embedding layer, a Temporal-Spatial Transformer encoder layer, and a regression layer. As depicted in the figure, we will introduce each module step by step.

Embedding layer

The embedding layer includes implicit information embedding of traffic data and graph Laplacian embedding of the road network structure. Specifically, implicit information embedding first involves feature embedding, which transforms the raw data into a high-dimensional representation. This converts the raw input $E \in \mathbb{R}^{T \times N \times j}$ to $E_d \in \mathbb{R}^{T \times N \times d}$, where d is the dimension of the feature embedding. Considering that traffic flow data exhibits significant periodicity within days and weeks, such as rush hours and differences between weekdays

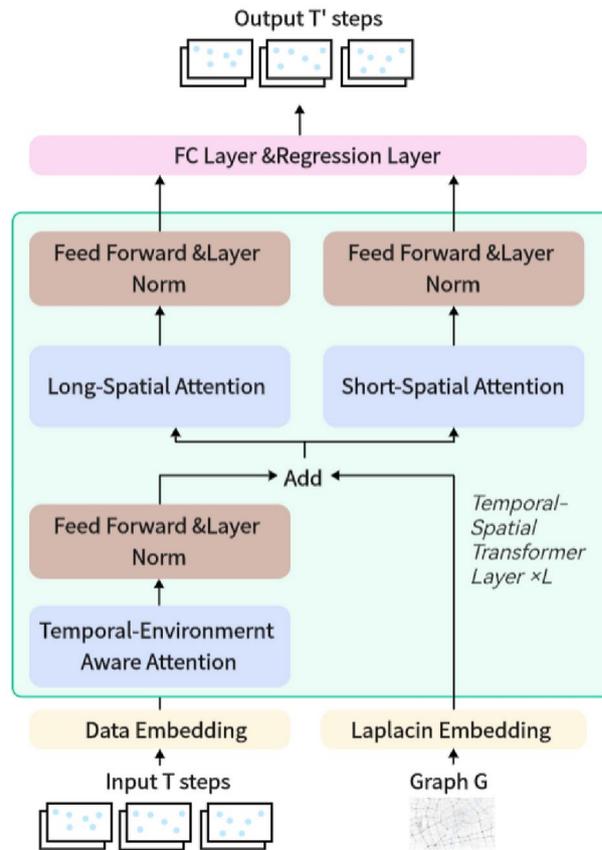


Figure 1. IEEAFormer.

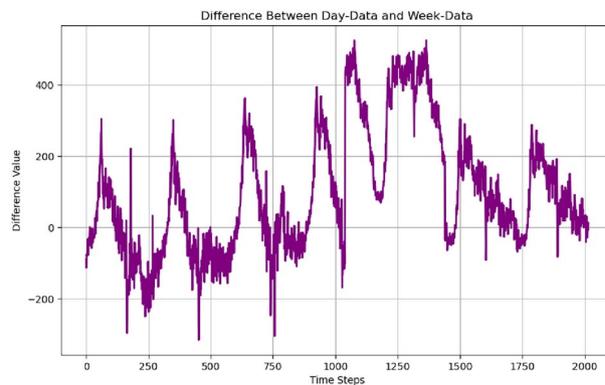


Figure 2. Difference between day-data and week-data.

and weekends, we introduce periodic time embeddings. These embeddings include daily time embeddings and weekly time embeddings to capture traffic variation patterns across different periods. We implement two transformation functions, $tod(t)$ and $dow(t)$, which convert time t into a week index (1 to 7) and a minute index (1 to 1440). These are then transformed into day-of-week embeddings and timestamp embeddings. After broadcasting and concatenation, they are finally converted into periodic time embeddings $E_f \in \mathbb{R}^{T \times N \times f}$, where f is the dimension of the periodic time embeddings.

As shown in Fig. 2, this is a comparison between the traffic flow data recorded by a detector over one day and one week. The larger the fluctuation, the greater the difference between the two datasets. It is evident that simply using feature embedding and time-periodic embedding is insufficient for accurately modeling traffic flow over time. Similarly, Fig. 3 shows the significant differences in traffic flow data recorded by different detectors within a single day. This is because, in real-world road traffic, congestion often occurs for various reasons and is transitive-i.e., congestion at one point can affect surrounding points. Furthermore, because of the different

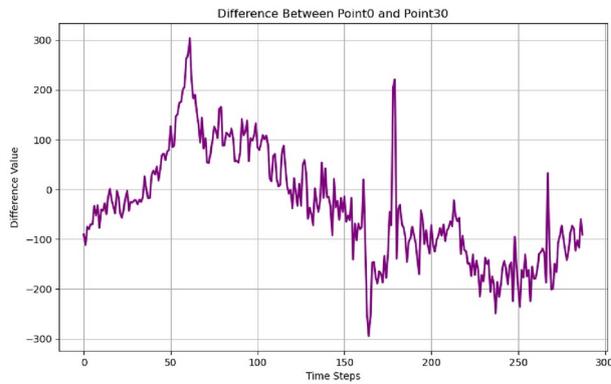


Figure 3. Difference between points.

geographical locations of the sensors, such as those at intersections, which typically capture more information than sensors located on one-way streets, the traffic patterns captured by the sensors may vary.

This spatial-temporal semantic information also needs to be modeled in a targeted manner. Therefore, we use the Xavier initialization method to design an embedding $E_s \in \mathbb{R}^{T \times N \times s}$ to uniformly capture this information, where s represents the dimension of the semantic information. The semantic information embedding can be shared across each time sequence.

Finally, we concatenate the feature embedding, time embedding, and implicit semantic information embedding mentioned above to obtain the output of the implicit information embedding of traffic data in the embedding layer $E_i \in \mathbb{R}^{T \times N \times i}$

$$E_i = E_d || E_f || E_s \tag{2}$$

where $i = d + f + s$.

For the graph Laplacian embedding of the road network structure, we first obtain the normalized Laplacian matrix by

$$\Delta = I - D^{-1/2} A D^{-1/2} \tag{3}$$

where A is the adjacency matrix, D is the degree matrix, and I is the identity matrix. Then, we perform eigenvalue decomposition $\Delta = U^T \Lambda U$ to obtain the eigenvalue matrix Λ and the eigenvector matrix U . We then linearly project the k smallest non-trivial eigenvectors to generate the spatial graph Laplacian embedding $X_{lap} \in \mathbb{R}^{N \times l}$, where l is the dimension of the Laplacian embedding. The Laplacian eigenvectors embed the graph into Euclidean space, preserving the global graph structure information. This representation is more precise than the traditional adjacency matrix and better reflects the distances between nodes in the road network.

Temporal-spatial transformer encoder layer

This paper proposes targeted improvements to the attention mechanism of the original Transformer encoder layer, designing a Temporal-Spatial Transformer encoder layer to capture traffic relationships in both temporal and spatial dimensions.

Temporal-environment-aware self attention

In the temporal dimension, this study introduces a Temporal-Environment-Aware self-attention mechanism in place of the traditional multi-head self-attention. Traditional multi-head self-attention relies solely on node values to determine relevance. However, in traffic scenarios, road points exhibit interdependence, with impacts spreading to neighboring nodes, indicating local temporal context interactions. Furthermore, distant points in time, despite having numerically similar flow data, often show weak correlations. The traditional mechanism disregards local temporal interactions and mistakenly matches numerically similar but temporally distant points, resulting in erroneous sequence representations and unrealistic matching issues that adversely affect predictions.

To mitigate these challenges, this study proposes a Temporal-Environment-Aware self-attention mechanism. This approach incorporates convolution operations to capture local temporal context, enabling the model to discern contextual information. Linear projections of Q and K are replaced with 1D convolutions, facilitating the detection of local trends in traffic flow data and accurate matching of similar traffic flow points. This mechanism enhances sequence representation accuracy and improves prediction performance. Formally, the temporal context-aware multi-head self-attention mechanism is defined as follows:

$$\text{TEAselfAttention}(\mathbf{Q}^{(\text{te})}, \mathbf{K}^{(\text{te})}, \mathbf{V}^{(\text{te})}) = \oplus(\text{TEAhead}1, \dots, \text{TEAhead}h)W^O \tag{4}$$

$$\text{TEAhead}j = \text{Attention}(\Phi_j^Q \star \mathbf{Q}^{(\text{te})}, \Phi_j^K \star \mathbf{K}^{(\text{te})}, \mathbf{V}^{(\text{te})}W_j^V) \tag{5}$$

where \star denotes the convolution operation, and Φ_j^Q and Φ_j^K denote the parameters of the convolution kernels. Given the input $X \in \mathbb{R}^{T \times N \times d}$, where T is the number of time steps, N is the number of nodes, and d is the dimension of the input features. First, two 1D causal convolution layers with a kernel size of 1k extract the contextual representation of each time step, generating Q and K for the attention mechanism. V is generated through a 1×1 convolution layer. The 1D causal convolution ensures that each time step can only attend to information from previous time steps, and padding operations maintain the output shape consistent with the input shape. This allows the contextual information to be shared among different nodes. Here, the time-environment information is computed and adjusted in real time based on the current input data. That is, the model flexibly captures the temporal environment trends by dynamically responding to data variations at each time step. In contrast, traditional models based on RNNs and CNNs assume that the relationships between time steps are fixed, meaning that the influence of each time step remains constant and is not adjusted according to changes in the input data. Therefore, compared to CNN or RNN-based methods, our module can flexibly adapt to actual input changes and capture varying temporal trends and patterns. Then, we calculate the self-attention scores:

$$A^{(te)} = \text{Softmax} \left(\frac{Q^{(te)} K^{(te)\top}}{\sqrt{d_h}} \right) \quad (6)$$

Here, $A^{(te)} \in \mathbb{R}^{N \times T \times T}$ learns the temporal context information among different spatial nodes. Finally, we obtain the output of this module $X^{(te)} \in \mathbb{R}^{N \times T \times m}$:

$$X^{(te)} = A^{(te)} V^{(te)}. \quad (7)$$

After applying the temporal-environment-aware self-attention mechanism, as shown in Fig. 4, the model correctly learns the relationships between different times (points B and C), matching the correct time points. The data, after passing through this attention mechanism, can be considered to have integrated contextual information along the temporal axis.

Spatial transformer encoder

After obtaining the output from the time-environment-aware self-attention mechanism, this paper adds the spatial embedding vector to the output, which serves as the input to the spatial Transformer encoder. The encoder consists of two self-attention modules: short-range spatial self-attention and long-range spatial self-attention. In traditional attention-based traffic flow prediction models, the standard self-attention mechanism connects each node to all other nodes, essentially treating the road network as a fully connected graph. However, in reality, the relationships between nodes are complex and varied. Some nodes are only closely related to their neighboring locations, while nodes that are far apart may still exhibit similar traffic patterns. For example, a traffic jam at an intersection may cause congestion at surrounding intersections, whereas schools and government institutions may share similar peak traffic patterns in the morning and evening. In such cases, the interactions between road nodes cannot be simply represented by a fully connected graph.

To better capture the complex relationships between nodes, we innovatively utilize a parallel attention mechanism in the spatial Transformer encoder to simultaneously capture both long-range and short-range dependencies. Additionally, two graph mask matrices, M_s and M_l , are introduced, which are applied to the short-range and long-range self-attention mechanisms, respectively, to represent short-range and long-range

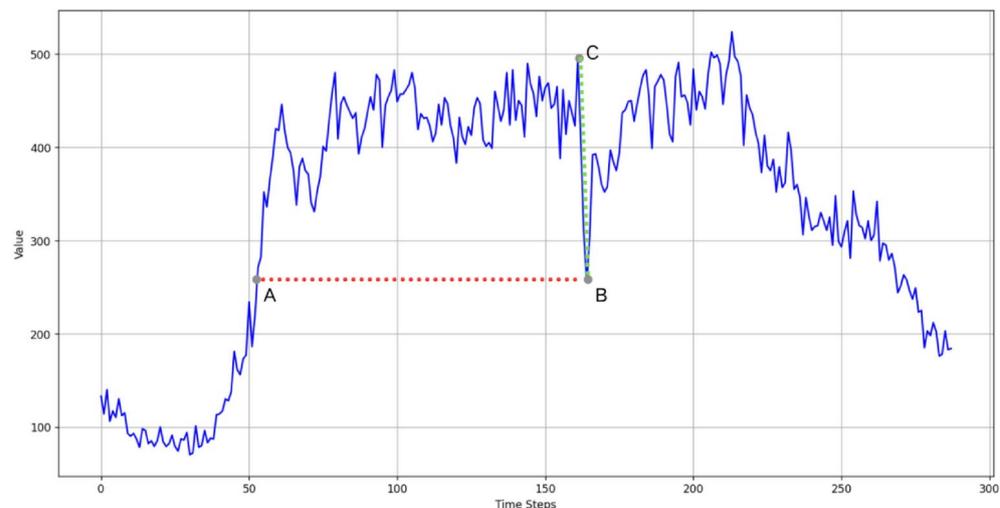


Figure 4. Temporal environment aware attention.

spatial dependencies. M_s is the short-range mask matrix, which filters out nodes that are too distant by applying a threshold λ

$$Mask_{short} = \begin{cases} 1, & distance \leq \lambda \\ 0, & otherwise \end{cases} \quad (8)$$

M_l is the long-range mask matrix, which uses the Dynamic Time Warping (DTW) algorithm to compute the historical traffic similarity between nodes. This algorithm is based on dynamic programming, and it constructs the correspondence between elements of two sequences of different lengths by the principle of nearest distance, to evaluate the similarity between the two time series. Using this method, the top K most similar nodes are selected as neighbors, with the weights set to 1.

$$Mask_{long} = \begin{cases} 1, & D(X, Y) \leq \varepsilon \\ 0, & otherwise \end{cases} \quad (9)$$

For two time series of different lengths, the DTW algorithm first aligns the sequences, and then calculates the total distance of the aligned sequences, as defined below:

$$D(x_i, y_j) = d(x_i, y_j) + \min \{D(x_{i-1}, y_j), D(x_i, y_{j-1}), D(x_{i-1}, y_{j-1})\} \quad (10)$$

$D(x_i, y_j)$ represents the shortest distance between the subsequences $X = (x_1, x_2, \dots, x_i)$ and $Y = (y_1, y_2, \dots, y_i)$. $d(x_i, y_i)$ denotes the absolute distance between x_i and y_i . In this way, the model can identify nodes that are spatially distant but exhibit similar traffic patterns.

The heatmaps of the long-range and short-range masks, shown in Figs. 5 and 6, reveal that the long-range mask selects the most similar nodes, while the short-range mask naturally forms clusters, selecting nodes that are closer in distance.

The model first adds the output of the temporal-environment-aware self attention, $X^{(te)} \in \mathbb{R}^{N \times T \times m}$, to the Laplacian embedding of the map in the data embedding layer. $X \in \mathbb{R}^{T \times N \times d}$ (after broadcasting):

$$X = X^{(te)} + X^{(lap)} \quad (11)$$

$d = m$ Here, the model receives the input to the spatial self-attention module. The input is then fed in parallel into the long-range spatial self-attention and short-range spatial self-attention modules. Take the short-range spatial self-attention module as an example.

$$Q^{(s)} = XW_Q^{(s)} \quad (12)$$

$$K^{(s)} = XW_K^{(s)} \quad (13)$$

$$V^{(s)} = XW_V^{(s)} \quad (14)$$

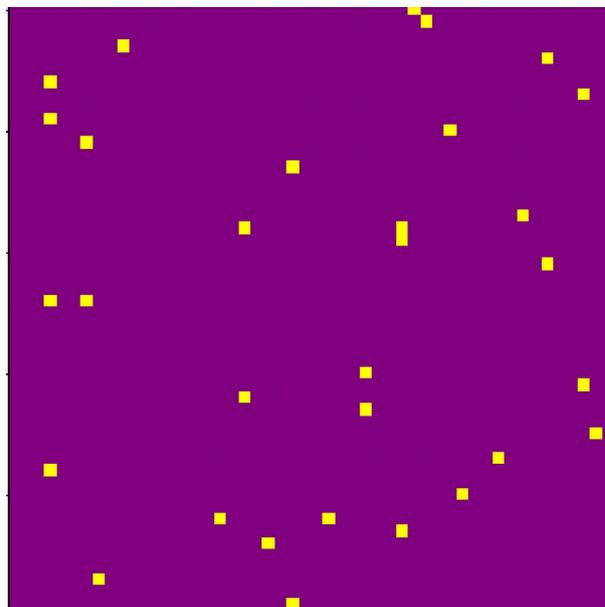


Figure 5. Long range.

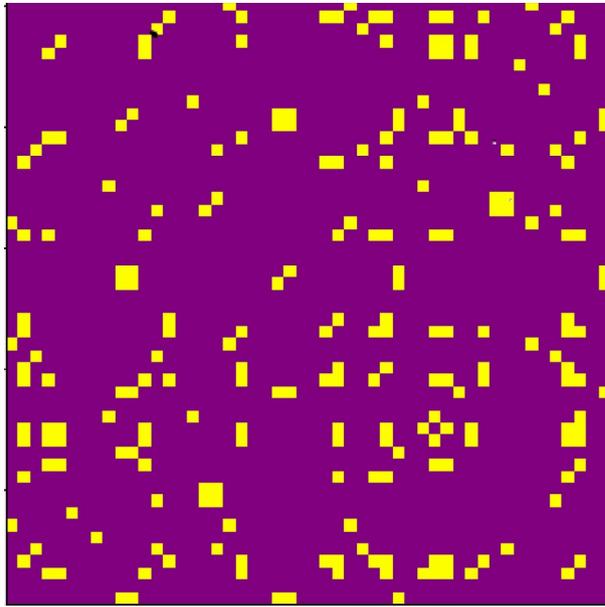


Figure 6. Short range.

DataSets	Number of nodes	Time range
Pems03	358	09/01/2018–11/30/2018
Pems04	307	01/01/2018–02/28/2018
Pems07	883	05/01/2017–08/31/2017
Pems08	170	07/01/2016–08/31/2016

Table 1. DataSets.

where $W_Q^{(s)}$, $W_K^{(s)}$, and $W_V^{(s)}$ are learnable parameters. Using the earlier Eq. (6), we obtain the attention scores A^s , which are then Hadamard-multiplied with the short-range mask matrix M_s :

$$X^{(s)} \in \mathbb{R}^{T \times N \times m} = \text{softmax}(A^{(s)} \odot M_s) V^{(s)} \quad (15)$$

This results in the output of the short-range spatial self-attention, $X^{(s)} \in \mathbb{R}^{T \times N \times m}$. Similarly, the output of the long-range spatial self-attention is $X^{(l)} \in \mathbb{R}^{T \times N \times m}$. Following the original Transformer methodology, we use residual connections and layer normalization. Finally, we concatenate the outputs of the two modules:

$$X^{(spa)} = X^{(s)} \parallel X^{(l)} \quad (16)$$

and apply a fully connected layer for dimensionality transformation:

$$X^{spa} = FC(X^{spa}) \quad (17)$$

Ultimately, we obtain the output of the spatial-temporal Transformer encoder, $X^{spa} \in \mathbb{R}^{T \times N \times m}$. This data can be considered to have accurately learned the long-range and short-range relationships in the spatial domain.

Experiment

DataSets

This study validates the IEEAFormer using real-world traffic datasets: PEMS03, PEMS04, PEMS07, and PEMS08. STSGCN introduced these datasets¹⁰. Each dataset consists of data collected every 5 min by multiple sensors distributed across different locations over a certain period. For instance, Pems04 comprises traffic flow data collected by 307 sensors every 5 min over 59 days, while Pems08 includes data collected by 170 sensors every 5 min over 62 days. Detailed information about these datasets is provided in Table 1. The spatial network for each dataset is constructed based on the actual road network. If two observation points are located on the same road, they are considered connected in the spatial network. During the preprocessing stage, we detect and remove missing values in the data, and standardize the data by subtracting the mean and scaling it to unit variance.

$$X' = \frac{X - \text{mean}(X)}{\text{std}(X)}$$

Here, $\text{mean}(X)$ and $\text{std}(X)$ represent the mean and standard deviation of the time series, respectively.

Experimental setup

Following standard practice, the four datasets were split into training, validation, and test sets with a ratio of 6:2:2. The prediction task involves using the past one hour (12 time steps) of traffic data to forecast the next hour (12 time steps), known as a multi-step prediction.

The model was implemented using the PyTorch framework on an online NVIDIA GeForce 2080Ti server provided by Autodl (<https://www.autodl.com/>). The dimension of the implicit information embedding is 152, and the depth of the encoder layers is 4. The number of attention heads is 4. We selected Adam as the optimizer, with an initial learning rate of 0.001 that decays over time, and a batch size of 16. The training ran for 200 epochs, and early stopping was applied.

Three metrics were used to evaluate the traffic prediction tasks: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Squared Error (RMSE). Consistent with previous studies, the average performance over the 12 forecasted time steps on the PEMS03, PEMS04, PEMS07, and PEMS08 datasets was used to assess the model.

We compared IEEAFormer with several widely used benchmark methods, including a typical traditional model, HI¹¹. Graph neural network models: DCRNN¹², AGCRN¹³, STGCN¹⁴, GTS¹⁵, ASTGNN¹⁶, and MTGNN¹⁷, all of which use simple feature embeddings. We also included STNorm¹⁸, which focuses on decomposing traffic time series. Additionally, we selected short-term time series prediction methods based on Transformer:GMAN¹⁹, PDFormer²⁰, and DRFormer²¹, which are Transformer models designed for the same task as ours. Furthermore, we included MFSTN²² and SGRU²³, which are relatively new models in this field.

Performance evaluation

As shown in Table 2 below, we can draw the following conclusion:

- (1) Our proposed method significantly outperforms the baseline methods in most metrics and datasets, demonstrating strong performance and generalizability.
- (2) The IEEAFormer deep learning model excels in spatial-temporal data prediction tasks compared to traditional time-series forecasting models (e.g., HI). This is mainly because traditional models overlook spatial dependencies in traffic data.
- (3) IEEAFormer performs better than STGCN, which also models the map. This is because our approach incorporates a unique long- and short-range masking mechanism that better captures geographic relationships.
- (4) Among self-attention-based models, ASTGNN and PDFormer stand out: ASTGNN aggregates neighbor information by combining GCN and self-attention modules, while PDFormer employs a more complex architecture. Compared to ASTGNN, our method utilizes two matrices in conjunction with spatial self-attention to capture both short-range and long-range spatial dependencies simultaneously, achieving superior performance. Compared to PDFormer, our network architecture is simpler yet more effective in capturing latent information behind the data. Compared to the latest models, our IEEAFormer also demonstrates competitive performance.

	PEMS03			PEMS04			PEMS07			PEMS08		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
AGCRN	15.25	26.65	15.88%	19.44	31.25	13.40%	20.56	34.41	8.77	15.32	24.99	10.03%
STGCN	15.83	27.51	16.13%	19.57	31.39	13.44%	21.77	35.31	9.24%	16.08	25.39	10.61%
GTS	15.41	26.16	25.40%	20.99	32.95	14.66%	22.14	35.15	9.42%	16.49	26.08	10.54%
ASTGNN	15.04	26.99	18.90%	18.62	31.12	12.66%	20.61	34.00	8.59%	14.94	23.55	9.09%
MTGNN	16.07	25.77	14.66%	19.07	31.56	12.96%	21.01	34.17	9.09%	15.93	24.89	10.18%
STNorm	15.32	25.90	14.40%	18.96	30.98	12.77%	20.60	34.69	8.88%	15.15	24.19	10.13%
HI	32.62	49.89	30.60%	42.35	61.66	29.92%	49.03	71.18	22.75%	36.66	50.44	21.63%
GMAN	16.87	27.92	18.23%	19.11	31.58	13.16%	20.97	34.10	9.05%	15.31	24.92	10.12%
DCRNN	15.64	27.22	15.62%	19.61	31.26	13.60%	21.16	34.14	9.01%	15.30	24.19	10.26%
PDFormer	14.97	25.41	15.88%	18.36	30.03	12.01%	19.97	32.97	8.55%	13.63	23.41	9.11%
DRFormer	15.01	26.14	16.98%	19.47	32.46	13.13%	21.11	33.47	9.95%	14.73	25.04	10.38%
SGRU	15.22	26.23	15.64%	19.96	31.98	13.22%	21.60	34.58	9.16%	15.96	25.13	10.02%
MFSTN	18.63	27.77	16.30%	22.08	33.49	15.49%	23.02	35.98	10.29%	17.26	26.11	13.01%
IEEAformer	15.00	25.37	15.17%	18.22	30.31	11.99%	19.11	32.60	8.01%	13.49	23.20	8.89%

Table 2. LULC change transition matrix showcasing percentage wise area changes (AC) in each class. Bold values (no change).

The impact of hyperparameters on experimental results

To further investigate the impact of hyperparameter settings, as well as the effects of residual connections and layer normalization on model performance, we conducted experiments using the Pems08 dataset under different hyperparameters and configurations. Except for the factors under investigation, all models followed the same settings as described in “Experimental setup”. The results are shown in Fig. 7. In general, IEEAFormer is not sensitive to hyperparameter settings. Increasing the dimension of the implicit information embedding (modifying the semantic embedding E_s in the implicit information embedding), the depth of the model, and the number of attention heads can slightly improve performance, but the differences are not statistically significant. Additionally, we observed that the normalization layer and residual connections contribute significantly to the model’s results. When neither residual connections nor normalization layers were included, the model was difficult to train and the loss did not converge. The model that combined both residual connections and normalization layers achieved the best performance.

Time and space complexity

In the multi-head self-attention module informed by time environment, the computational complexity of the multi-head self-attention and convolution operations for each node is $O(T_h^2 d_{\text{model}})$ and $O(kT_h d_{\text{model}}^2)$, respectively. The time complexity of spatial self-attention is also $O(T_h^2 d_{\text{model}})$. The spatial complexity of the model is approximately $O(N^2 d + Nd)$.

Ablation study

To further investigate the effectiveness of each component in IEEAFormer, this study employed a controlled variable method to design the following experiments:

- (1) Removal of Time Embedding E_f and Semantic Embedding E_s : This means that the model does not learn periodic information about time or semantic information.
- (2) Replacement of Time-Environment-Aware Self-Attention with Traditional Multi-Head Self-Attention: Traditional multi-head self-attention does not effectively learn contextual environmental information or correctly match similar nodes.
- (3) Removal of Masks for Long-Range and Short-Range Spatial Self-Attention: In this case, each node attends to all other nodes without restriction.

We applied the above experiments to the Pems04 and Pems08 datasets. The experimental results shown in Figs. 8 and 9 demonstrate that the traffic flow data inherently contains implicit information such as periodicity and traffic patterns. The implicit information embedding layer used in this study effectively captures these features

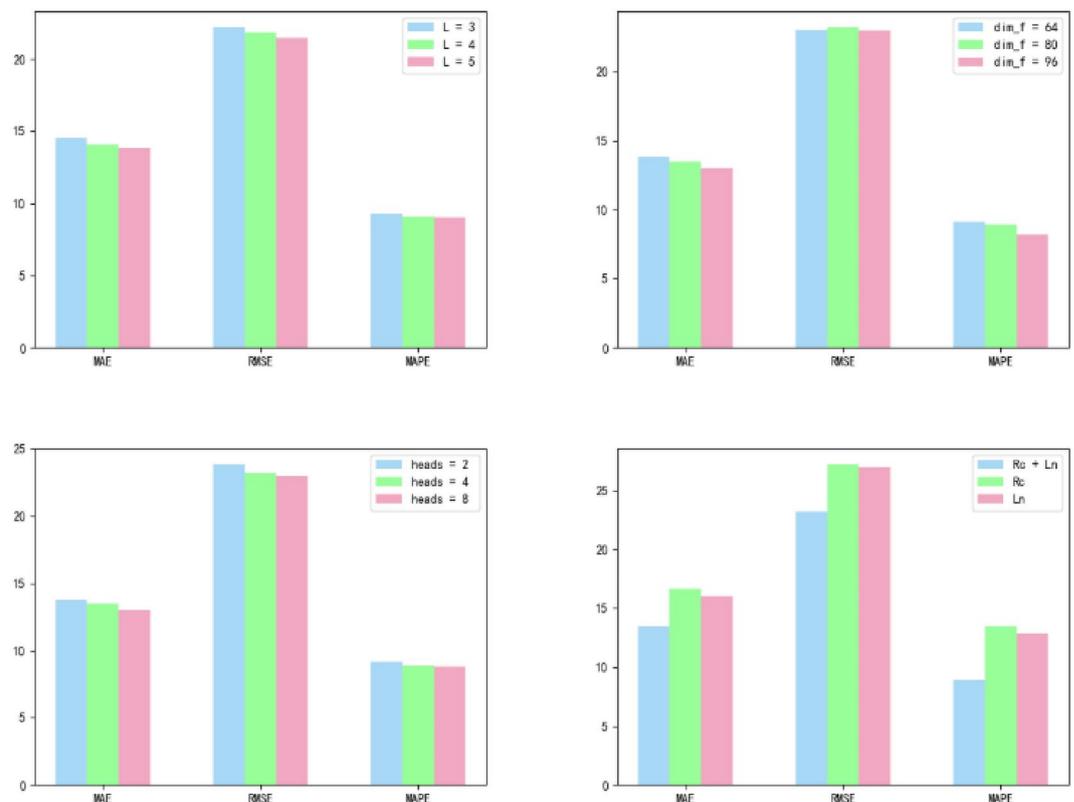


Figure 7. Effect of different network configurations.

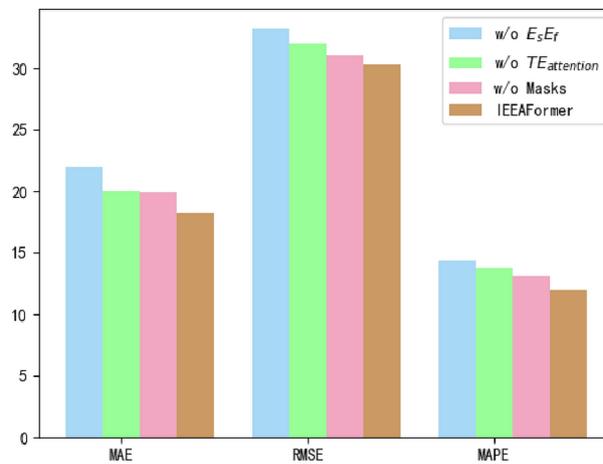


Figure 8. Ablation on Pems04.

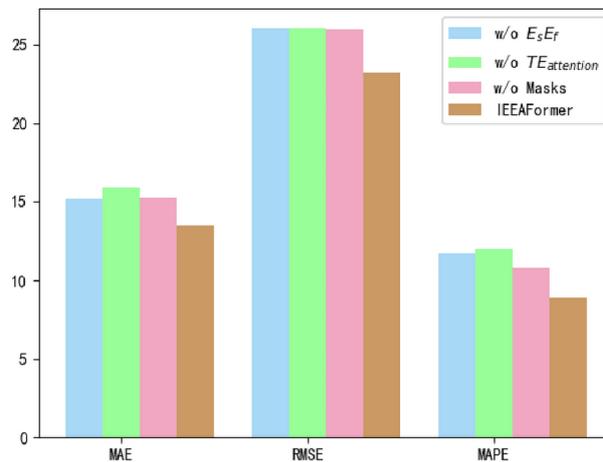


Figure 9. Ablation on Pems08.

. Furthermore, the application of the time-environment-aware self-attention mechanism along the time axis effectively models temporal relationships, significantly outperforming the traditional multi-head self-attention mechanism. In addition, the spatial self-attention mechanism, combined with the long-range and short-range masks, highlights important spatial features and effectively suppresses

the influence of noise. In summary, the proposed method can effectively extract spatial-temporal features and accurately predict traffic data.

Case study

To further analyze the effectiveness of IEEAFormer's spatiotemporal Transformer encoder in capturing local temporal trends and modeling both long- and short-range spatial information, we referred to methods in^{24,25}. Specifically, we shuffled the original input along the temporal axis T , significantly weakening the temporal environmental information in the data, as illustrated in the figure. On the dataset with shuffled time order, IEEAFormer failed to capture the temporal environmental information, which led to a degradation in the model's performance. This result indicates that our model is more sensitive to the implicit temporal environmental information in the data. Additionally, we visualized the original self-attention scores and the time-environment-aware self-attention scores (Fig. 10).

It can be observed that the original self-attention mechanism tends to focus on points that are temporally distant, whereas the time-environment-aware self-attention correctly focuses on the information of a given time step and its surrounding context. Additionally, we gradually increased the prediction horizon, and Fig. 11 shows the impact of this increase on the predictive performance of different models. Generally, as the prediction horizon expands, the task becomes more challenging, leading to a performance decrease across all models. However, in most cases, the performance of IEEAFormer decreases the least. This means that even as the temporal correlation weakens, IEEAFormer is still able to make relatively accurate predictions based on the learned spatial correlation information.

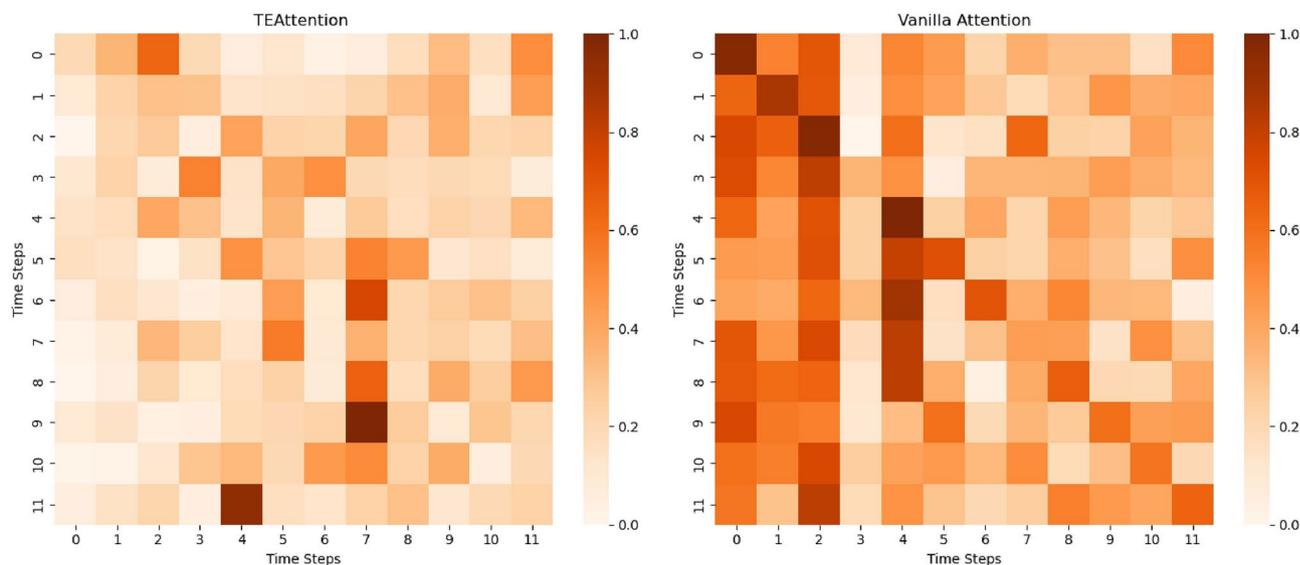


Figure 10. Temporal environment aware attention.

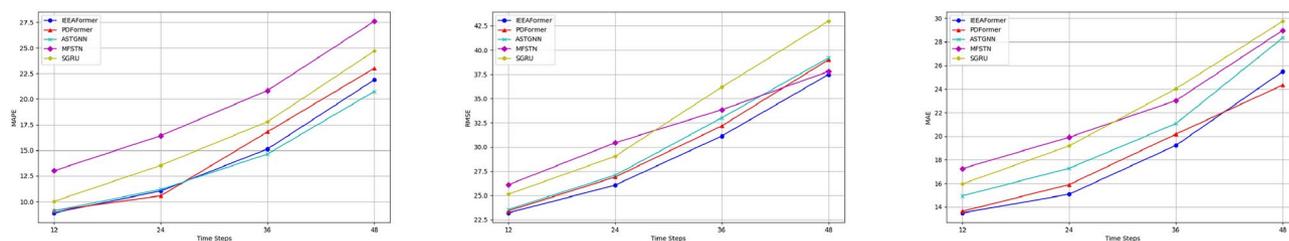


Figure 11. Performance changes of different methods on Pmes08.

Conclusion

In this study, we enhance the traditional Transformer model by addressing three key aspects: input embedding, temporal attention, and the modeling of both long-range and short-range spatial relationships. We propose a novel IEEFormer architecture, which includes: the addition of latent information embeddings in the input, the use of time-environment-aware self-attention mechanisms for modeling temporal dependencies, and the introduction of long-range and short-range mask matrices combined with parallel self-attention to capture both long-range and short-range spatial features. Extensive experiments on four real-world datasets demonstrate the superior performance of our proposed model. In the future, we will continue to optimize the model to adapt to more complex and dynamic traffic scenarios.

Related work

In recent years, an increasing number of researchers have begun to adopt deep learning models for traffic prediction. Due to the inherent temporal and spatial information in traffic flow data, early studies predominantly used Convolutional Neural Networks (CNNs) to model the temporal and spatial features of traffic data. However, CNNs were originally designed to handle Euclidean data such as images and videos, whereas real-world road data are irregular and graph-structured. As a result, CNNs struggle to accurately extract spatial features. On the other hand, Graph Neural Networks (GNNs), due to their intrinsic capability to handle non-Euclidean data, address the limitations of CNNs in spatial feature modeling, as demonstrated by works such as²⁶. The Spatial-Temporal Graph Convolutional Network (STGCN)¹⁴ combines both CNN and GNN approaches and has shown promising results. However, it relies on a predefined graph structure, limiting its ability to capture spatial features dynamically. In contrast, models like GraphWaveNet²⁷ and AGCRN¹⁵ can adaptively learn the graph structure of the road network from the data. Recently, attention mechanisms have gained attention due to their effectiveness in modeling the dynamic dependencies in traffic data. Transformer-based traffic flow prediction models have also emerged^{28,29}. Among them, STAEFormer²⁴ uses embedding techniques similar to those presented in this paper, while PDFormer²⁰ introduces a similar masking strategy^{30,31} propose a unified replay-based continuous learning framework. With the rapid development of large models, many methods based on large language models have also been proposed such as^{32,33}. Moreover,^{34–37} are notable contributions in this field. However, unlike the aforementioned works, we not only improve the embedding layer to capture implicit

information but also adopt an enhanced multi-head self-attention mechanism in both the temporal and spatial dimensions to better handle complex spatial-temporal relationships.

Data availability

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Code availability

Code is available from the <https://gitee.com/xanax9203/ieeaformer>.

Received: 21 July 2024; Accepted: 27 February 2025

Published online: 10 March 2025

References

- Pascanu, R. On the difficulty of training recurrent neural networks. arXiv preprint [arXiv:1211.5063](https://arxiv.org/abs/1211.5063) (2013)
- Li, D., Zhang, J., Zhang, Q., & Wei, X. Classification of ECG signals based on 1D convolution neural network. In *2017 IEEE 19th International Conference on E-health Networking, Applications and Services (Healthcom)*. 1–6 (IEEE, 2017).
- Yu, F. Multi-scale context aggregation by dilated convolutions. arXiv preprint [arXiv:1511.07122](https://arxiv.org/abs/1511.07122) (2015)
- Hochreiter, S. et al. *Gradient Flow in Recurrent Nets: The Difficulty of Learning Long-Term Dependencies* (IEEE Press, A field guide to dynamical recurrent neural networks, 2001).
- Xu, M., Dai, W., Liu, C., Gao, X., Lin, W., Qi, G.-J., & Xiong, H. Spatial-temporal transformer networks for traffic flow forecasting. arXiv preprint [arXiv:2001.02908](https://arxiv.org/abs/2001.02908) (2020)
- Cai, L., Janowicz, K., Mai, G., Yan, B. & Zhu, R. Traffic transformer: Capturing the continuity and periodicity of time series for traffic forecasting. *Trans. GIS* **24**(3), 736–755 (2020).
- Yan, H., Ma, X. & Pu, Z. Learning dynamic and hierarchical traffic spatiotemporal features with transformer. *IEEE Trans. Intell. Transport. Syst.* **23**(11), 22386–22399 (2021).
- Zhang, H., Zou, Y., Yang, X. & Yang, H. A temporal fusion transformer for short-term freeway traffic speed multistep prediction. *Neurocomputing* **500**, 329–340 (2022).
- Roy, A., Roy, K.K., Ahsan Ali, A., Amin, M.A., & Rahman, A.M. Sst-gnn: Simplified spatio-temporal traffic forecasting model using graph neural network. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. 90–102 (Springer, 2021).
- Song, C., Lin, Y., Guo, S., & Wan, H. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 914–921 (2020).
- Cui, Y., Xie, J., & Zheng, K. Historical inertia: A neglected but powerful baseline for long sequence time-series forecasting. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2965–2969 (2021)
- Li, Y., Yu, R., Shahabi, C., & Liu, Y. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. arXiv preprint [arXiv:1707.01926](https://arxiv.org/abs/1707.01926) (2017)
- Bai, L., Yao, L., Li, C., Wang, X. & Wang, C. Adaptive graph convolutional recurrent network for traffic forecasting. *Adv. Neural Inf. Process. Syst.* **33**, 17804–17815 (2020).
- Yu, B., Yin, H., & Zhu, Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. arXiv preprint [arXiv:1709.04875](https://arxiv.org/abs/1709.04875) (2017)
- Shang, C., Chen, J., & Bi, J. Discrete graph structure learning for forecasting multiple time series. arXiv preprint [arXiv:2101.06861](https://arxiv.org/abs/2101.06861) (2021)
- Guo, S., Lin, Y., Wan, H., Li, X. & Cong, G. Learning dynamics and heterogeneity of spatial-temporal graph data for traffic forecasting. *IEEE Trans. Knowl. Data Eng.* **34**(11), 5415–5428 (2021).
- Wu, Z., Pan, S., Long, G., Jiang, J., Chang, X., & Zhang, C. 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 (2020)
- Deng, J., Chen, X., Jiang, R., Song, X., & Tsang, I.W. St-norm: Spatial and temporal normalization for multi-variate time series forecasting. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 269–278 (2021)
- Zheng, C., Fan, X., Wang, C., & Qi, J. Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 1234–1241 (2020)
- Jiang, J., Han, C., Zhao, W.X., & Wang, J. Pdformer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. 4365–4373 (2023)
- Ding, R., Chen, Y., Lan, Y.-T., & Zhang, W. Drformer: Multi-scale transformer utilizing diverse receptive fields for long time-series forecasting. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 446–456 (2024)
- Yan, J. et al. A multi-feature spatial-temporal fusion network for traffic flow prediction. *Sci. Rep.* **14**(1), 14264 (2024).
- Zhang, W., Li, X., Li, A., Huang, X., Wang, T., & Gao, H. Sgru: A high-performance structured gated recurrent unit for traffic flow prediction. In *2023 IEEE 29th International Conference on Parallel and Distributed Systems (ICPADS)*. 467–473 (IEEE, 2023).
- Liu, H., Dong, Z., Jiang, R., Deng, J., Deng, J., Chen, Q., & Song, X. Spatio-temporal adaptive embedding makes vanilla transformer SOTA for traffic forecasting. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 4125–4129 (2023)
- Zeng, A., Chen, M., Zhang, L., & Xu, Q. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. 11121–11128 (2023)
- Jin, G., Liang, Y., Fang, Y., Shao, Z., Huang, J., Zhang, J., & Zheng, Y. Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. In *IEEE Transactions on Knowledge and Data Engineering* (2023)
- Wu, Z., Pan, S., Long, G., Jiang, J., & Zhang, C. Graph wavenet for deep spatial-temporal graph modeling. arXiv preprint [arXiv:1906.00121](https://arxiv.org/abs/1906.00121) (2019)
- Liu, J., Kang, Y., Li, H., Wang, H. & Yang, X. STGHTN: Spatial-temporal gated hybrid transformer network for traffic flow forecasting. *Appl. Intell.* **53**(10), 12472–12488 (2023).
- Zhu, W., Sun, Y., Yi, X., Wang, Y. & Liu, Z. A correlation information-based spatiotemporal network for traffic flow forecasting. *Neural Comput. Appl.* **35**(28), 21181–21199 (2023).
- Miao, H., Zhao, Y., Guo, C., Yang, B., Zheng, K., & Jensen, C.S. Spatio-temporal prediction on streaming data: A unified federated continuous learning framework. In *IEEE Transactions on Knowledge and Data Engineering* (2025)
- Miao, H., Zhao, Y., Guo, C., Yang, B., Zheng, K., Huang, F., Xie, J., & Jensen, C.S. A unified replay-based continuous learning framework for spatio-temporal prediction on streaming data. arXiv preprint [arXiv:2404.14999](https://arxiv.org/abs/2404.14999) (2024)
- Liu, C., Yang, S., Xu, Q., Li, Z., Long, C., Li, Z., & Zhao, R. Spatial-temporal large language model for traffic prediction. arXiv preprint [arXiv:2401.10134](https://arxiv.org/abs/2401.10134) (2024)

33. Liu, C., Xu, Q., Miao, H., Yang, S., Zhang, L., Long, C., Li, Z., & Zhao, R. Timecma: Towards llm-empowered time series forecasting via cross-modality alignment. arXiv preprint [arXiv:2406.01638](https://arxiv.org/abs/2406.01638) (2024)
34. Miao, H., Shen, J., Cao, J., Xia, J. & Wang, S. Mba-stnet: Bayes-enhanced discriminative multi-task learning for flow prediction. *IEEE Trans. Knowl. Data Eng.* **35**(7), 7164–7177 (2022).
35. Jin, G., Liu, L., Li, F., & Huang, J. Spatio-temporal graph neural point process for traffic congestion event prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. 14268–14276 (2023)
36. Li, F., Yan, H., Jin, G., Liu, Y., Li, Y., & Jin, D. Automated spatio-temporal synchronous modeling with multiple graphs for traffic prediction. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 1084–1093 (2022)
37. Jin, G., Li, F., Zhang, J., Wang, M. & Huang, J. Automated dilated spatio-temporal synchronous graph modeling for traffic prediction. *IEEE Trans. Intell. Transport. Syst.* **24**(8), 8820–8830 (2022).

Author contributions

Shipeng Liu wrote the main manuscript text. Xingjian Wang gave the idea and funding. All authors reviewed the manuscript.

Funding

Natural Science Foundation of Heilongjiang Province under Grant(LH2020C048).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to X.W.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025