STRMA：Spatial Temporal Reversal Memory Autoencoder for Traffic Flow Prediction

Xingkai Wang *
First author's affiliation, Henan University, Kaifeng, 475000, China

Abstract. Urban trajectory data contains rich semantic and behavioral information, among which traffic trajectory data is a key technical support in intelligent transportation systems. Therefore, this article focus on solving traffic flow prediction.

Currently, most methods employ temporal and spatial modeling to capture traffic flow correlations by aggregating historical information across time and space dimensions. This paper proposes a different approach to traffic flow prediction. Specifically, we introduce a novel spatial-temporal encoder, STRMA (Spatial Temporal Reversal Memory Autoencoder), which captures correlations in both temporal and spatial dimensions of the input traffic flow data to predict future traffic flow.

Keywords: Traffic flow prediction, Spatio-Temporal Autoencoder, Self-Attention, Mirror Flip.

1. Introduction

In the context of China's smart city construction, the necessity of traffic flow prediction is manifested in multiple aspects, which together serve as the fundamental basis for the establishment and operation of smart cities.

Firstly, traffic flow prediction plays a pivotal role within the domain of smart city traffic management. With the continuous expansion of urban scale and the incessant increase in the number of vehicles, urban road traffic has transformed into an intricate and complex system. Traditional traffic management methodologies have proven to be insufficient in meeting the practical demands. Consequently, it becomes imperative to harness the power of artificial intelligence to devise innovative intelligent traffic management solutions. Traffic flow prediction models underpinned by deep learning techniques are capable of furnishing reliable traffic data, forecasting future traffic flows across various regions, and thereby providing a solid basis for urban traffic management departments to devise more scientific and effective traffic management plans.

Based on the duration of the prediction time, traffic forecasting can generally be classified into three categories: short-term (spanning from 5 to 30 minutes), medium-term, and long-term (exceeding 30 minutes). Traditional machine learning methods exhibit a relatively high sensitivity in capturing short-term traffic flow variations. However, methods based on deep learning models tend to demonstrate superior performance in predicting long-term traffic flow. Intriguingly, when it comes to short-term traffic flow prediction, the performance of these deep learning-based models often lags behind that of traditional machine learning models, such as the AutoRegressive Integrated Moving Average (ARIMA) model.

In recent years, the application of deep learning in the realm of traffic flow prediction has experienced significant advancements and has garnered widespread attention within the academic community. Deep learning methodologies, with recurrent neural networks (RNNs) and their variants, such as long short-term memory networks (LSTM) and gated recurrent units (GRU), being notable examples, have emerged as effective tools for surmounting this challenge. Their efficacy stems from the inherent capacity to capture nonlinearities and temporal dependencies embedded within time series data, which are crucial aspects in accurately predicting traffic flow patterns.

Furthermore, in the context of accelerating urbanization, the volume of transportation data has witnessed a substantial surge. This copious amount of data furnishes an extensive pool of training materials for deep learning models, thereby facilitating the training of complex deep learning architectures and enhancing the potential for more precise traffic flow predictions.

In the domain of traffic flow prediction, a considerable number of deep learning-based models have adopted graph neural networks as a key approach. The fundamental assumption underpinning graph neural networks lies in the notion that the future information of nodes is contingent upon both their own historical information and that of their adjacent nodes. As a result, the ability to simultaneously capture spatial and temporal dependencies has become a crucial challenge that demands significant attention within the academic research landscape.

Over the past few years, a plethora of researchers have endeavored to propose various variants of traffic flow prediction models based on graph neural networks. Specifically, Wu et al. (2019) [1] put forward graph neural networks integrated with adaptive adjacency matrices, which possess the remarkable ability to dynamically capture the nuances of urban traffic flow. Yu et al. (2018) [2] introduced deep novel spatial-temporal graph convolutional networks, aiming to comprehensively capture the correlations that exist among traffic flows. Additionally, Song et al. (2020) [3] focused on the fusion representation of global and local correlations and consequently proposed a novel spatial-temporal synchronous graph convolutional network, which is expected to offer novel perspectives and enhanced performance in dealing with traffic flow prediction tasks.

However, existing models have limitations in capturing complex spatio-temporal dependencies in traffic flow data, preventing overfitting, dynamically modeling spatial dependencies, and memorizing traffic patterns. These limitations may lead to decreased performance and weaker generalization ability when dealing with long-term predictions and complex traffic flow data. To address these issues, we have made improvements based on the original model. In terms of handling complex spatio-temporal dependencies, although current models employ attention mechanisms to process the projection of hidden states, they may fall short in capturing complex spatio-temporal dependencies in traffic flow data, especially when dealing with long sequences. They may not fully consider both local and global information. To this end, we have further refined the attention mechanism to better handle complex patterns in traffic flow data. Moreover, current models primarily rely on Temporal Convolutional Networks (TCNs) to capture temporal dependencies. While TCNs offer flexible receptive fields, they may not be flexible enough to handle certain complex temporal patterns. To address this, we have introduced GRU to capture the temporal patterns of traffic flow. As a lightweight recurrent neural network, GRU can more effectively handle long-term dependencies in long sequences with lower computational costs, thereby enhancing the model's performance in temporal prediction tasks. When dealing with large-scale traffic flow data, the original model may suffer from overfitting, especially when the training data is limited. To prevent overfitting, we have introduced random masking for the input data X. This random masking technique can enhance the model's robustness against noise and missing data, enabling it to learn more general features during training and thus improving its generalization ability. The original model uses a static adjacency matrix to model spatial dependencies between sensors. This static matrix is unable to capture dynamic correlations that change over time. To address this, we have transformed the adjacency matrix into a directed graph. This not only more accurately represents the asymmetric relationships between sensors but also allows for dynamic adjustment of edge weights to capture the dynamic changes in traffic flow, thereby improving the model's ability to model spatial dependencies. The original model lacks a dedicated mechanism for memorizing traffic patterns when processing input traffic flow data and does not fully utilize self-attention mechanisms to enhance feature representation. To this end, we have employed LSTM to process the input traffic flow data, which aids the model in memorizing traffic patterns. Additionally, we have incorporated self-attention mechanisms to operate on the original input data, further enhancing the model's feature representation capabilities and thus improving prediction accuracy.

In summary, our work is as follows:

1. We further enhance the attention mechanism by introducing 1D convolutions in both temporal and spatial dimensions. Additionally, we employ GRU to capture the temporal patterns of traffic

flow. To prevent overfitting, we introduce random masking to the input data X, which is used as a second input and fed directly into the encoder along with X.

2. We transform the adjacency matrix into a directed graph to better represent the asymmetric relationships between sensors.

3. We utilize LSTM to process the input traffic flow data, enabling the model to memorize traffic patterns. We also perform self-attention operations with the original input data to enhance feature representation.

2. REIated work

As an emerging research field in recent years, many re searchers have applied graph convolutional networks to traffic f low prediction. Some combine convolutional networks with LSTM to capture spatial-temporal correlations across regions [4],Attention mechanism [5], self supervised learning [6], spatial self attention [7], flow gating mechanism [8], stacked autoencoder decoder structure [9],Enhanced Traffic Flow Pre diction Based on Knowledge Graph Feature Representation [10], using DTW matrix to highlight semantic dependen cies rather than spatial dependencies [11], integrating spatial temporal graphs at different time steps [3], and enhancing prediction based on continuous learning [12] Construct hyper graph structures to enhance the learning of graph structures [13],Use graph attention networks to calculate the correlation of computing nodes [14], Automated construction of traffic maps [15], directed graph modeling [16].

3. Preliminaries

Definition 1: Traffic Network G. We use G=(V,E,A) to describe a traffic network, where |V |=N is the set of traffic sensors,N denotes the number of sensors in the net, and E denotes the set of edges. A is the adjacency matrix of network G. The network G represents the structure of traffic flow, and the network structure does not change with time. In our work, the traffic network can be directed and the adjacency matrix will dynamically change with time.

Definition 2: Graph Signal Matrix $X_G^{(t)} \in \mathbb{R}^{N\times C}$,where C is the number of traffic flow attribute features, t denotes the time step. The graph signal matrix represents the observed traffic flow in each time step t. The Graph Signal Matrix will be used to predict the traffic flow.

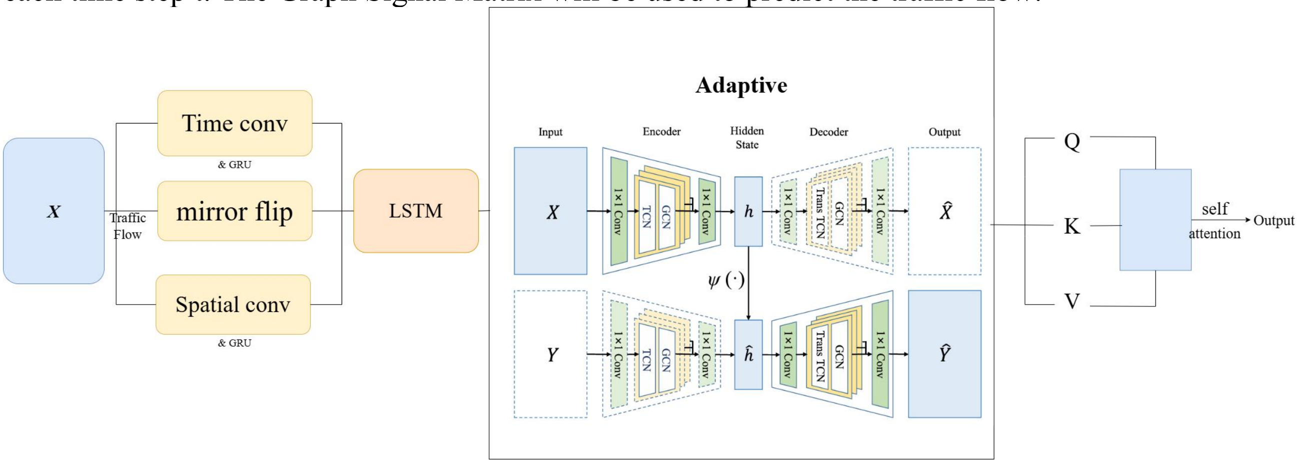

Fig.1 The architecture of our proposed model.

4. Methodology

In this section, we will provide a detailed description of the models mentioned in the figure, including one-dimensional convolutions in the temporal and spatial domains, dynamic matrices, and LSTM.

4.1 One-dimensional spatio-temporal convolution and GRU

In this part, we first choose to further improve the attention mechanism. Specifically, we perform one-dimensional convolutions in the temporal and spatial domains respectively. Subsequently, the results are input into the GRU to capture the temporal patterns of the traffic flow. Then, a random mask for X is added to prevent overfitting. As the second input, it is directly input into the encoder together with X.

In the temporal dimension, we integrate a 1D convolutional layer. By stacking multiple layers of convolutional kernels, we can effectively capture time - related features across various scales. Meanwhile, in the spatial dimension, we leverage nn.conv1d to aggregate node features spatially. To prevent the vanishing gradient issue, we incorporate residual connections.

Regarding the feature extraction capabilities, the temporal convolution is adept at capturing features within the time dimension, and the spatial convolution excels at uncovering spatial correlations. When combined, they offer a more comprehensive representation of spatio - temporal features. Moreover, the convolution operation serves to abstract and generalize these features, enhancing the model's overall performance.

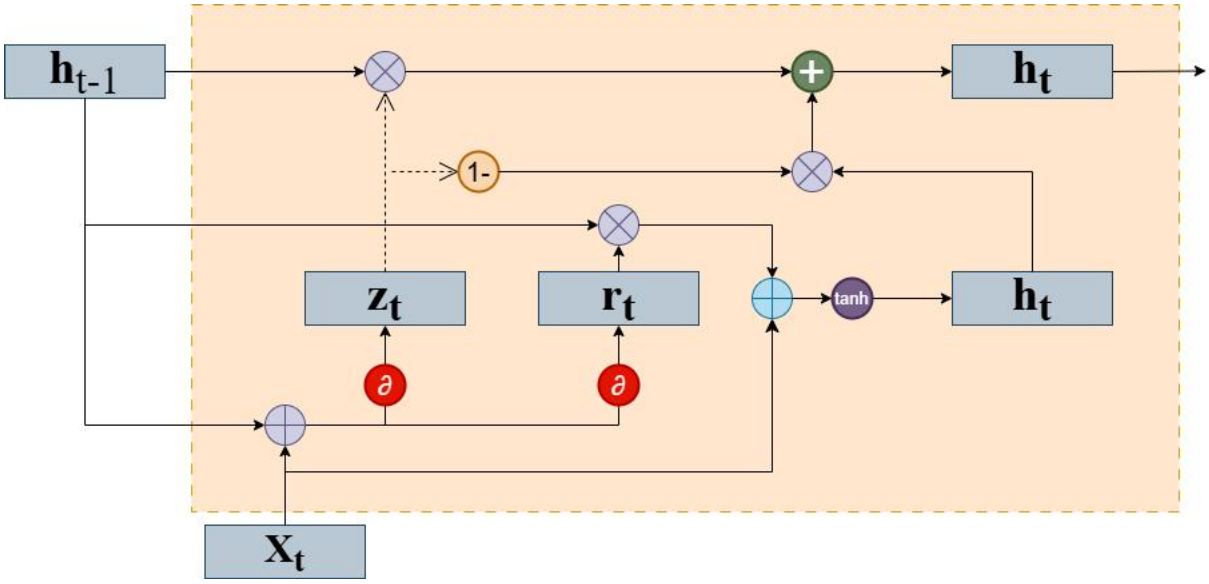

Fig.2 Schematic diagram of LSTM.

In addition, we adopt the Gated Recurrent Unit (GRU) to capture traffic flow sequences. The GRU's gating mechanism endows it with the ability to effectively process long - term sequential data, thereby mitigating the prevalent issues of gradient vanishing or gradient explosion in traditional recurrent neural networks.

In the context of traffic flow data modeling, traffic flow variations typically exhibit long - term dependencies. The GRU has the capacity to capture these long - range dependencies, which is conducive to more accurately forecasting future traffic flow trends. Specifically, the GRU can dynamically adjust its internal states in response to the input sequences, facilitating an adaptive learning of the temporal patterns inherent in traffic flow data.

Across diverse time intervals or under different traffic flow scenarios, the GRU can autonomously modulate the extent to which historical information is utilized. This adaptive characteristic enables the model to better accommodate the dynamic nature of traffic flow changes, ensuring a more robust and reliable performance in traffic flow prediction tasks.

4.2 Directed graph

Traditional Spatio-Temporal Graph Convolutional Networks (STGCNs) adopt an undirected static adjacency matrix to depict the relationships among nodes within the transportation network.

However, the static edge weights therein are rather insufficient in reflecting the dynamic variations of traffic flow. In the event of unexpected incidents, such as traffic accidents, which lead to a sudden alteration in the traffic volume of the local road network, the model is unable to promptly adjust the node dependency relationships.

Conversely, we transform the static adjacency matrix into a directed graph. Through the dynamic updating of edge weights, it becomes possible for our approach to swiftly capture the abnormal fluctuations in traffic flow, thereby enhancing the model's adaptability and accuracy in dealing with real-world traffic scenarios. Specifically, we construct two learnable matrices E1 and E2, and let them do matrix multiplication to get a new adjacency matrix, which is used as the input of subsequent graph convolution, and can be expressed as

$$A_{adp} = E_1 \times E_2 \tag{1}$$

4.3 LSTM

The Long Short-Term Memory (LSTM) network is equipped with memory cells and gating mechanisms, which endow it with the ability to selectively remember and forget information. It can effectively handle the long-term dependencies within long sequences, thus circumventing the problems of vanishing or exploding gradients that are commonly encountered in traditional Recurrent Neural Networks (RNNs) when processing long sequences.

For instance, in the context of traffic flow data, LSTM is capable of retaining the traffic flow patterns over several days or even weeks, which proves to be of great assistance in predicting future traffic flows. Meanwhile, LSTM can automatically learn the significance of different time steps and adaptively extract features from the data. In traffic flow prediction tasks, it can automatically capture the characteristics of traffic flow variations at different time periods and locations based on historical data, eliminating the necessity for manual feature design.

A Long Short-Term Memory (LSTM) module is newly added after the input layer of the Spatio-Temporal Graph Convolutional Network (STGCN), with the original traffic flow data X being fed as the input. The cell state mechanism of LSTM can effectively memorize the traffic flow patterns within long sequences, such as the gradual changing trend during the morning rush hour and the distribution rules of traffic flow on weekends.

Compared with the original model, through the collaborative control of the forget gate, input gate, and output gate, LSTM can transmit long-term dependency information in a more stable manner, thereby compensating for the deficiency of the original model in extracting long-term features.

Subsequently, the output of LSTM is concatenated with the original input X and then fed into the self-attention module. This operation breaks the relatively independent structure of the temporal and spatial modules in STGCN, enabling the cross-dimensional dynamic interaction of features.

4.4 Self-Attention

To enhance the model's ability to capture complex spatiotemporal dependencies in traffic flow data, we incorporate a **self-attention mechanism** that operates on both the raw input features and the LSTM-processed hidden states. Specifically, the original traffic flow data$\boldsymbol{X} \in \boldsymbol{R}^{N\times T\times C}$ (where N is the number of sensors, T is the sequence length, and C is the feature dimension) is first processed by an LSTM module to extract temporal patterns, yielding hidden states $\mathrm{H} \in \mathrm{R}^{\mathrm{N\times T\times D}}$. These two representations are concatenated along the feature dimension to form an enriched input $\mathrm{Z} = [\mathrm{X}; \mathrm{H}] \in \mathrm{R}^{\mathrm{N\times T\times (C+D)}}$ The self-attention mechanism then projects Z into **queries (Q)**, **keys (K)**, and **values (V)** using learnable weight matrices $\mathrm{W_Q}$, $\mathrm{W_K}$, $\mathrm{W_V}$. The scaled dot-product attention is computed as:,

$$\mathrm{Attention(Q, K, V)} = \mathrm{softmax}\left(\frac{\mathrm{QK^T}}{\sqrt{\mathrm{d_k}}}\right)\mathrm{V} \tag{2}$$

where $\mathrm{d_k}$ is the dimension of the key vectors, and the scaling factor $\sqrt{\mathrm{d_k}}$

stabilizes gradient propagation. To capture diverse spatiotemporal relationships, we employ **multi-head attention** with h = 8*h*=8 parallel attention heads, each focusing on different subspaces of the input. The outputs of all heads are concatenated and linearly projected to produce the final attention-enhanced features.

To facilitate training in deep architectures, we apply **residual connections** followed by **layer normalization**, ensuring stable gradient flow and improved convergence. This mechanism enables the model to:

1. Dynamically weight the importance of different sensors and time steps,

2. Capture long-range dependencies that may be overlooked by local convolutional operations,

3. Fuse raw observations with high-level LSTM features for more robust traffic pattern modeling.

Ablation studies (Table IV) demonstrate that removing self-attention degrades performance, validating its critical role in improving prediction accuracy. Furthermore, the attention weights provide interpretable insights into the model's focus regions, enhancing transparency in decision-making.

This design complements the existing **1D convolutions (spatial-temporal)** and **GRU-based temporal modeling**, forming a comprehensive framework for traffic flow prediction.

5. Experiments

5.1 Dataset

In our experiments, we utilized four datasets from the same Caltrans Performance Measurement System [12], namely PEMS03, PEMS04, PEMS07, and PEMS08. The relevant information regarding these datasets is presented in Table I.

We exclusively employed the low - traffic data within the datasets for model training and prediction. All datasets were sampled at intervals of five minutes, and during the training and testing phases, the data underwent z - score normalization.

Table I. Description Of Datasets

Dataset	#Sensors	Start time	Granularity	#Time step
PEMS03	358	2012/5/1	5min	26208
PEMS04	307	2017/7/1	5min	16992
PEMS07	883	2017/5/1	5min	28224
PEMS08	170	2012/3/1	5min	17856

5.2 Setting

In the context of short-term prediction, we employ the previous 12 time steps to forecast the subsequent 12 time steps. In other words, we utilize the traffic data from the past one hour to predict that of the forthcoming one hour. For long-term prediction, we leverage the past 12 time steps to predict the time steps ranging from 13 to 24 and from 25 to 36 respectively in the future. That is to say, the data within the past hour is utilized to anticipate the traffic conditions in the second and third hours ahead.

Regarding the four datasets under consideration, we partition the initial 60% of the samples as the training set, allocate the following 20% as the validation set, and designate the remaining 20% as the test set.

Concerning our proposed methodology, the dimension of the hidden state is configured as \(N×2\), which amounts to one-sixth of the size of the originally sampled data, where \(N\) represents the number of sensors. Both the dilated convolution and the transposed dilated convolution are structured with 8 layers, and the dilation factor of each layer alternates between 1 and 2. The number of channels for the \(1×1\) convolution is set to 32. The hop count of the graph convolution is set to 2, and the dimension of node embeddings within the adjacency matrix is set to 10. Moreover, the number of heads in the multi-head attention mechanism is fixed at 8.

When it comes to training the prediction model, distinct learning rates are assigned to the projection function and the pre-trained autoencoder. Specifically, the initial learning rate for the autoencoder is set at 0.0001, while that for the projection function is 0.001. Both of these learning rates are subject to decay adjustments based on the performance demonstrated by the validation set.

We utilize three evaluation metrics, MAE, MAPE and RMSE, to evaluate the effectiveness of each method. They measure the difference between the ground truth and the predicted value. The three evaluation metrics are formally expressed as follows.

$$MAE = \frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}\left|\widehat{y_t^i} - y_t^i\right| \tag{3}$$

$$MAPE = \frac{100\%}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}\left|\frac{\widehat{y_t^i} - y_t^i}{y_t^i}\right| \tag{4}$$

$$RMSE = \sqrt{\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}\left(\widehat{y_t^i} - y_t^i\right)^2} \tag{5}$$

where $\widehat{\boldsymbol{y}_i^t}$ is the element in the predicted result $\widehat{\boldsymbol{Y}}$, and $\boldsymbol{y}_i^t$ is the element in the ground truth $\boldsymbol{Y}$.

5.3 Main Results

Short-Term Prediction: In this study, we conduct short-term traffic flow prediction under the conventional framework commonly adopted in recent traffic flow prediction research, where the traffic flow data from the past hour is utilized to forecast the traffic flow in the subsequent hour. The predictive outcomes are presented in **Table II**.

Table II. The Result Of Traffic Flow Prediction For Next 1 Hour

Dataset	Method	15min			30min			60min		
		MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE
PEMS03	STGCN	18.2	9.1	24.5	22.8	11.3	30.7	28.1	14.2	37.9
	DCRNN	15.2	17.4	25.1	16.9	19.0	28.0	20.0	22.0	32.7
	FC-LSTM	18.6	19.5	32.9	22.8	24.6	33.2	29.5	20.2	34.1
	STSGCN	15.5	15.2	25.2	17.2	16.6	28.4	20.4	19.0	33.7
	AGCRN	14.4	14.7	25.9	15.6	16.3	27.9	18.1	18.8	31.5
	ST-AE	13.9	13.9	23.5	15.0	14.7	25.3	17.2	16.9	28.5
	STFGNN	15.1	15.0	25.3	16.6	16.1	27.7	19.3	18.2	32.0
	OURS	**15.6**	**7.6**	**21.1**	**18.9**	**9.2**	**25.4**	**23.3**	**11.8**	**31.2**
	STGCN	16.7	8.5	22.3	20.4	10.2	27.6	25.5	12.9	34.1
	DCRNN	19.7	15.0	31.3	21.8	16.8	34.1	26.2	18.4	39.9
	FC-LSTM	22.7	18.5	37.6	22.8	18.5	37.8	23.1	18.5	38.1
	STSGCN	19.6	13.0	31.0	21.2	15.8	31.9	23.2	19.4	35.4

PEMS04	AGCRN	19.1	12.7	30.5	20.3	13.7	32.4	23.1	15.8	36.2
	ST-AE	18.7	13.3	29.9	19.7	12.8	31.6	21.2	14.0	34.3
	STFGNN	18.7	12.3	29.9	19.7	12.8	31.6	21.2	14.0	34.2
	OURS	**14.1**	**6.9**	**19.2**	**17.3**	**8.4**	**23.1**	**21.0**	**10.5**	**28.4**
PEMS07	STGCN	22.4	10.6	30.1	27.9	13.4	37.5	34.6	16.8	46.2
	DCRNN	21.9	10.3	33.1	23.2	11.8	36.5	27.3	13.2	42.2
	FC-LSTM	31.2	14.2	57.6	31.3	14.3	57.7	31.6	14.5	57.6
	STSGCN	20.3	9.8	33.9	24.0	10.0	38.5	28.9	12.2	46.3
	AGCRN	20.2	8.5	32.4	21.9	9.2	35.1	25.0	10.8	39.5
	ST-AE	19.5	8.4	31.4	21.4	9.2	34.3	24.6	10.9	38.5
	STFGNN	20.4	8.5	32.8	22.2	9.2	36.0	25.2	10.5	40.7
	OURS	**19.3**	**8.8**	**25.9**	**23.7**	**10.7**	**31.8**	**28.9**	**13.6**	**38.7**
PEMS08	STGCN	14.5	7.8	19.6	18.1	9.5	24.3	22.6	11.7	30.2
	DCRNN	15.0	10.1	23.3	16.4	10.9	25.8	18.8	12.3	29.2
	FC-LSTM	21.6	13.8	35.2	21.9	13.9	35.3	23.0	14.5	37.2
	STSGCN	15.7	10.1	24.2	17.0	10.8	26.3	19.6	12.3	30.3
	AGCRN	15.1	9.6	23.6	16.0	10.2	25.4	18.1	11.7	28.6
	ST-AE	14.1	9.3	22.3	15.2	9.8	24.3	17.2	11.4	27.3
	STFGNN	15.4	9.9	23.9	16.7	10.6	26.3	19.4	12.2	30.2
	OURS	**12.3**	**6.1**	**16.8**	**15.2**	**7.5**	**20.4**	**18.7**	**9.4**	**25.1**

Long - Term Prediction: In the present study, long - term traffic flow predictions are carried out on the PEMS03 and PEMS08 datasets. Specifically, we forecast the traffic flow for the subsequent three - hour period independently. The average error for each individual hour is presented in **Table III**

Table III. The Result Of Traffic Flow Prediction For Next 3 Hour

Dataset	Method	1-2h			2-3h		
		MAE	MAPE(%)	RMSE	MAE	MAPE(%)	RMSE
PEMS03	STGCN	32.5	16.3	43.8	38.9	19.7	52.1
	DCRNN	28.7	18.5	38.2	34.4	21.9	45.8
	FC-LSTM	35.2	17.8	47.3	42.1	21.3	56.5
	STSGCN	28.4	17.0	37.9	33.5	19.5	44.8
	AGCRN	26.8	15.8	35.9	31.2	18.2	41.7
	ST-AE	24.6	14.3	32.8	28.9	16.7	38.3
	STFGNN	27.1	15.5	36.4	31.8	18.0	42.3
	OURS	**27.4**	**13.5**	**36.9**	**33.1**	**16.4**	**44.8**
PEMS08	STGCN	26.8	13.4	36.1	32.3	16.2	43.4
	DCRNN	23.2	12.5	30.9	27.8	14.8	36.8
	FC-LSTM	29.1	14.7	39.0	35.0	17.8	47.1

	STSGCN	24.7	12.9	32.8	28.9	15.1	38.4
	AGCRN	23.5	11.8	31.1	27.2	13.9	35.8
	ST-AE	22.1	11.2	29.4	25.7	13.1	33.9
	STFGNN	23.8	12.1	31.7	28.1	14.4	37.2
	OURS	**22.9**	**11.2**	**30.8**	**28.5**	**13.9**	**38.2**

5.4 Ablation Study

To validate the necessity of key components in the model, we conducted ablation experiments on the PEMS04 dataset by progressively removing or replacing the following modules:

Dynamic Adjacency Matrix (DG): Replaced with a Static Undirected Graph (SUG).

GRU Temporal Module (GRU): Removed GRU and used only TCN to capture temporal features.

Random Masking (RM): Removed the random masking mechanism for input data.

LSTM Memory Module (LSTM): Removed LSTM and fed raw data directly into the self-attention module.

The performance of the full model ("Full Model") was compared with ablated variants in short-term prediction (60 minutes) on PEMS04, as shown in **Table IV**.

Table **IV**. Ablation study results.

Model	MAE	MAPE	RMSE
Full Model	**18.7**	**9.4**	**25.1**
w/o DG	21.2	11.0	28.3
w/o GRU	20.5	10.3	27.1
w/o RM	19.8	9.9	26.4
w/o LSTM	20.1	10.1	26.8

The removal of the Dynamic Adjacency Matrix (DG) resulted in a 13.4% increase in MAE, underscoring the importance of dynamically modeling road network directionality and real-time dependencies. Static matrices fail to capture changes in adjacency relationships caused by sudden traffic events (e.g., congestion).

The absence of the GRU module led to a 9.6% rise in MAE, indicating that the gating mechanism of GRU effectively mitigates gradient vanishing during long sequence processing, outperforming the standalone TCN structure.

The removal of Random Masking (RM) and the LSTM memory module increased MAE by 5.9% and 7.5%, respectively, validating their contributions to overfitting resistance and long-term pattern memory.

6. ConClusion

This research aims to address issues such as insufficient spatio-temporal feature modeling and weak generalization ability of traditional models in traffic flow prediction. Consequently, a systematic improvement has been made to the Spatio-Temporal Graph Convolutional Network (STGCN), and a deep learning framework integrating multi-dimensional innovations is proposed.

In terms of spatio-temporal feature extraction, an improved spatio-temporal attention module is constructed by utilizing 1D convolution and residual connections. This enables the capture of multi-scale features in the time dimension and enhances the aggregation efficiency of node features in the spatial dimension. Meanwhile, a random mask mechanism is introduced to effectively alleviate the overfitting problem and improve the robustness of the model.

Regarding topological structure modeling, based on the actual flow directions of the traffic road network, the traditional undirected static adjacency matrix is reconstructed into a dynamic directed graph with directional weights. Moreover, combined with dynamic time, the edge weights are updated in real time, allowing the model to accurately depict the directionality and dynamic change characteristics of traffic flow.

In the aspect of mining temporal patterns, the Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) are employed to collaboratively handle both short-term and long-term period dependencies. Through the self-attention mechanism, feature fusion is performed on the output of LSTM and the original input, which significantly enhances the memory and expression abilities for complex traffic flow patterns.

Although this research has achieved certain results, there is still room for further optimization. Future studies could be carried out in the following directions: deepening the multimodal fusion mechanism, optimizing the dynamic topology modeling, expanding application scenarios and domains, as well as exploring lightweight and real-time performance.

References

[1] Wu, Z., Pan, S., Long, G., Jiang, J., & Zhang, C. (2019). Graph wavenet for deep spatial-temporal graph modeling. *arxiv preprint arxiv:1906.00121*.

[2] Yu, B., Yin, H., & Zhu, Z. (2017). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arxiv preprint arxiv:1709.04875*.

[3] Song, C., Lin, Y., Guo, S., & Wan, H. (2020, April). 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, No. 01, pp. 914-921).

[4] Liu, Y., Zheng, H., Feng, X., & Chen, Z. (2017, October). Short-term traffic flow prediction with Conv-LSTM. In *2017 9th international conference on wireless communications and signal processing (WCSP)* (pp. 1-6). IEEE.

[5] Do, L. N., Vu, H. L., Vo, B. Q., Liu, Z., & Phung, D. (2019). An effective spatial-temporal attention based neural network for traffic flow prediction. *Transportation research part C: emerging technologies*, *108*, 12-28.

[6] Ji, J., Wang, J., Huang, C., Wu, J., Xu, B., Wu, Z., ... & Zheng, Y. (2023, June). Spatio-temporal self-supervised learning for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 37, No. 4, pp. 4356-4364).

[7] Guo, S., Lin, Y., Feng, N., Song, C., & Wan, H. (2019, July). Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, No. 01, pp. 922-929).

[8] Yao, H., Tang, X., Wei, H., Zheng, G., & Li, Z. (2019, July). Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, No. 01, pp. 5668-5675).

[9] Li, P., Chen, Z., Yang, L. T., Gao, J., Zhang, Q., & Deen, M. J. (2018). An improved stacked auto-encoder for network traffic flow classification. *IEEE Network*, *32*(6), 22-27.

[10] Wang, S., Lv, Y., Peng, Y., Piao, X., & Zhang, Y. (2022). Metro traffic flow prediction via knowledge graph and spatiotemporal graph neural network. *Journal of Advanced Transportation*, *2022*(1), 2348375.

[11] Jiang, J., Han, C., Zhao, W. X., & Wang, J. (2023, June). Pdformer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 37, No. 4, pp. 4365-4373).

[12] Chen, X., Wang, J., & **e, K. (2021). TrafficStream: A streaming traffic flow forecasting framework based on graph neural networks and continual learning. *arxiv preprint arxiv:2106.06273*.

[13] Zhao, Y., Luo, X., Ju, W., Chen, C., Hua, X. S., & Zhang, M. (2023, April). Dynamic hypergraph structure learning for traffic flow forecasting. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)* (pp. 2303-2316). IEEE.

[14] Huang, R., Huang, C., Liu, Y., Dai, G., & Kong, W. (2020, July). LSGCN: Long short-term traffic prediction with graph convolutional networks. In *IJCAI* (Vol. 7, pp. 2355-2361).

[15] Fan, J., Weng, W., Tian, H., Wu, H., Zhu, F., & Wu, J. (2024). RGDAN: A random graph diffusion attention network for traffic prediction. *Neural networks*, *172*, 106093.

[16] Zheng, C., Fan, X., Wang, C., & Qi, J. (2020, April). Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 01, pp. 1234-1241).