Journal of Computers Vol. 35 No. 4, August 2024, pp. 93-108
doi: 10.53106/199115992024083504007

Attention Mechanism Based Spatial-Temporal Graph Convolution Network for Traffic Prediction

Wenjuan Xiao[1,2*] and Xiaoming Wang[1]

[1] College of Electrical and Information Engineering, Lanzhou University of Technology,
Lanzhou 730050, Gansu, China
xwj@xbmu.edu.cn

[2] College of Electrical Engineering, Northwest Minzu University,
Lanzhou 730030, Gansu, China

Received 23 October 2023; Revised 30 January 2024; Accepted 14 March 2024

Abstract. Considering the complexity of traffic systems and the challenges brought by various factors in traffic prediction, we propose a spatial-temporal graph convolutional neural network based on attention mechanism (AMSTGCN) to adapt to these dynamic changes and improve prediction accuracy. The model combines the spatial feature extraction capability of graph attention network (GAT) and the dynamic correlation learning capability of attention mechanism. By introducing the attention mechanism, the network can adaptively focus on the dependencies between different time steps and different nodes, effectively mining the dynamic spatial-temporal relationships in the traffic data. Specifically, we adopt an improved version of graph attention network (GAT_v2) in the spatial dimension, which allows the model to capture more complex dynamic spatial correlations. Furthermore, in the temporal dimension, we combine gated recurrent unit (GRU) structure with an attention mechanism to enhance the model's ability to process sequential data and predict traffic flow changes over prolonged periods. To validate the effectiveness of the proposed method, extensive experiments were conducted on public traffic datasets, where AMSTGCN was compared with five different benchmark models. Experimental results demonstrate that AMSTGCN exhibits superior performance on both short-term and long-term prediction tasks and outperforms other models on multiple evaluation metrics, validating its potential and practical value in the field of traffic prediction.

Keywords: transportation system, attention mechanism, dynamic change, spatial-temporal dependency

1 Introduction

With the acceleration of urbanization, traffic congestion has become a common phenomenon in modern life. Accurate prediction of traffic conditions plays a very important role in improving the efficiency of transportation systems, reducing congestion, optimizing route selection, providing real-time navigation, and planning urban development.

As a key link in urban management and smart transportation systems, traffic prediction is becoming increasingly important. However, there are still many challenges in this field, in particular the need to accurately capture dynamically changing spatial correlations and complex temporal dependencies [1]. These factors significantly increase the complexity and difficulty of the prediction. To help the reader understand these challenges intuitively, we use Fig. 1 and Fig. 2 to provide specific examples for illustration. With these graphs, we show the spatial-temporal dynamics of traffic flows and the difficulty that traditional models have in capturing such dynamics.

(1) Dynamic Spatial Correlations. In previous studies, the spatial correlations between nodes are commonly represented by predefined static adjacency matrices, as mentioned in reference [2]. However, in real traffic environments, the spatial relationships between roads are dynamic systems subject to various factors such as traffic accidents and traffic regulations. Fig. 1 illustrates a schematic diagram of the road network in a certain urban area, where A, B, C, and D represent four intersections equipped with traffic detectors and treated as nodes in the network. For example, if the traffic authority implements a rule at A intersection that prohibits left turns from east to west, this will directly affect the traffic flow relationship between points A and B. Specifically, due to this restriction, the direct traffic flow from point A to point B will decrease, which means that the impact of traffic volume at point A on point B will correspondingly decrease, while the impact of point B on point A will rela-

[*] Corresponding Author

tively increase. Additionally, although point C and point B are not directly connected geographically, due to the left-turn restriction at point A, vehicles wishing to travel from east to west to reach point B must detour through points C and D, indirectly affecting the traffic conditions at point B. Furthermore, once the traffic rules at point A change, the spatial correlations of the entire A, B, C, and D node network will also adjust accordingly. With such an analysis, we can identify the dynamic nature of spatial relationships in real traffic scenarios and their impact on the design of predictive models.

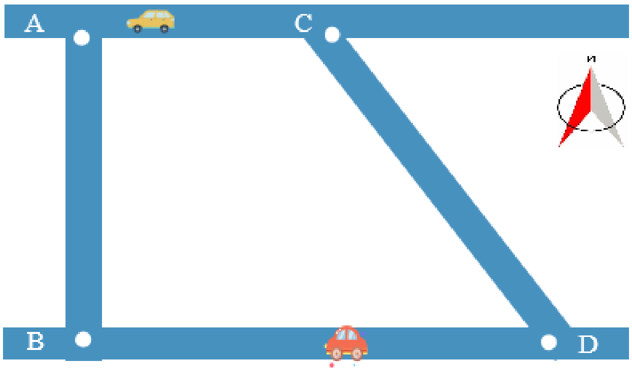


Fig. 1. Dynamic spatial-temporal relationship diagram of road node traffic flow

(2) Complex and Variable Temporal Dependencies. Traffic data inherently exhibit unique features on different time scales, including hours, days, weeks, and seasons. For example, morning and evening rush-hour traffic flows experience significant fluctuations, and weekday congestion patterns can be very different from those on weekends. In addition, temporal dependencies of traffic flow data can shift due to external events such as traffic accidents, weather fluctuations, or public gatherings. Fig. 2 illustrates the daily periodicity of traffic volume but also reveals an anomaly: a significant decrease in traffic volume during the period from 14:50 to 16:40 on Monday afternoons. The anomaly could be caused by congestion caused by factors such as road accidents or road construction. Therefore, when constructing traffic prediction models, it is necessary not only to take into account the regular temporal evolution of traffic data but also to capture and model temporal dependencies at different time scales, as well as potential anomalies that may arise. Such a comprehensive consideration is crucial for improving prediction accuracy.

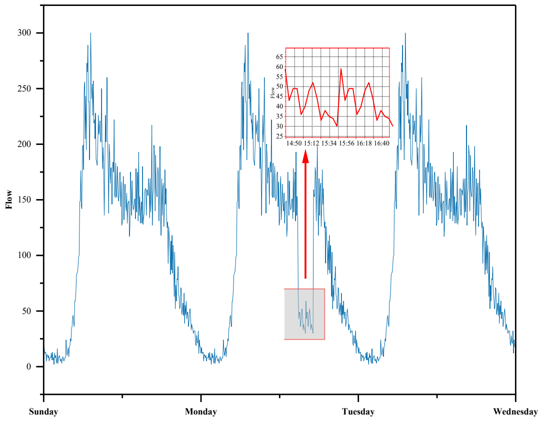


Fig. 2. The daily periodicity and dynamic variations of traffic flow volume

Through in-depth analysis, we have gained a new understanding of the importance of dynamic spatial-temporal relationships in traffic data. These relations include not only the periodic variations of traffic flow over time, but also sudden changes caused by unpredictable factors such as traffic accidents, weather conditions, and temporary road closures. Moreover, spatial correlations are continuously affected by factors such as traffic regulations, road network structure, and urban planning. These dynamic spatial-temporal dependencies require traffic prediction models to have a high level of adaptability and flexibility to accurately capture and respond to these complex traffic patterns.

In traditional traffic prediction methods, statistical models such as Autoregressive Integrated Moving Average (ARIMA), Exponential Smoothing models, and Regression models have dominated. However, these models have significant limitations when dealing with complex, nonlinear, and high-dimensional traffic data. They typically rely on manually extracted features and are built on linear assumptions, which limits their ability to capture the deep complexity of traffic patterns. With the introduction of machine learning, especially deep learning models, the field of traffic prediction has made significant advances. Deep learning models such as Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) [3] have shown superiority in multiple application scenarios by automatically learning features from the data. However, these models generally require a large amount of training data and have room for improvement in modelling the dynamics of spatial-temporal data.

To address these issues, we propose a spatial-temporal graph convolutional neural network (AMSTGCN) that incorporates an attention mechanism. This model not only automatically learns spatial-temporal features from complex traffic data, but also effectively adapts to spatial-temporal relationships with different time scales and dynamic changes. Through experimental validation on public datasets, AMSTGCN demonstrated its superiority in short-term and long-term traffic prediction tasks, demonstrating its effectiveness in capturing and predicting dynamic spatial-temporal traffic data. The main contributions of this study are as follows:

(1) Spatial-temporal relationship modelling. We propose an improved graph attention network (GAT_v2) [4] approach that dynamically extracts spatial relationships instead of relying on a static adjacency matrix used in traditional graph convolutional networks (GCN) [5]. This approach enables the model to adapt to dynamic changes caused by traffic rules and events, resulting in a more effective capture of spatial correlations in traffic data.

(2) Long-term dependency handling. To improve the accuracy of long-term traffic trend prediction, we combine gated recurrent units (GRU) and self-attention mechanisms. This enhances the model’s ability to explore complex dependencies within the time series and more effectively capture long-term dependencies, which are critical for accurate prediction of future traffic flows.

(3) Computational efficiency and performance. While incorporating advanced attention mechanisms and graph attention mechanisms, we prioritize computational efficiency. We simplify the model structure and optimize the algorithm to significantly reduce the computational resource requirements while still maintaining high-performance predictions. This makes the model more practical, feasible, and scalable for real-world applications.

(4) Empirical study. We conduct extensive experimental validation on public datasets, and the results demonstrate that our proposed model outperforms existing methods on both short-term and long-term traffic prediction tasks. Moreover, the model exhibits excellent generalization ability and robustness, further confirming the effectiveness and reliability of our approach.

The rest of the paper is organized as follows. Section 2 presents the research progress and related work. Section 3 provides definitions of the basic concepts. Section 4 presents the specific structure and implementation of the AMSTGCN model. Section 5 discusses the experimental validation and analysis of the results. Section 6 provides a summary and concluding remarks.

2 Related Work

With advances in science and technology, methods for traffic prediction have evolved from traditional statistical models to modern machine learning and deep learning models. In the early stages, statistical methods such as the Historical Average (HA) model and the ARIMA model were predominant. These models are typically applicable to linear time series data, but their predictive power is limited for complex, high-dimensional, externally influenced traffic data. In addition, the parameters of these methods often rely on expert knowledge for manual configuration, rather than being obtained through data-driven self-learning training.

In the wave of artificial intelligence, various machine learning methods, including K-Nearest Neighbors (K-NN) [6], Support Vector Regression (SVR) [7], Random Forest [8], and Bayesian Neural Networks [9] have been widely adopted in the field of traffic prediction. These techniques enhance prediction accuracy by exploring data features in-depth and typically optimize performance through the combination of different algorithms. However, they typically perform poorly in capturing long-term dependencies in traffic data, which is crucial for understanding traffic patterns and trends [10].

The rise of deep learning has brought revolutionary advances in traffic prediction. CNN benefiting from their remarkable achievements in image processing, has been applied in time series analysis. Researchers transform the spatial-temporal characteristics of traffic flow data into two-dimensional matrices (similar to images) and use CNN to extract features from these “spatial-temporal images” to achieve accurate prediction of speeds on extensive road networks [11]. However, CNN lacks mechanisms to handle long-term temporal dependencies, which may limit its effectiveness in predicting long-term trends.

To overcome the issue of long-term dependencies, Recurrent Neural Networks (RNN) have been introduced in traffic prediction, as they possess the ability to handle sequential data with memory [12]. However, inherent problems of RNN, such as vanishing gradients and exploding gradients, limit their performance in modelling long sequences. Therefore, improved versions of RNN, such as GRU and Long Short-Term Memory (LSTM) [13], have been more widely used in traffic prediction due to their structural advantages and stronger performance in addressing these challenges.

Despite extensive exploration of traffic prediction methods, traditional techniques have focused on temporal relationships, overlooking the significant impact of spatial dynamics on traffic patterns. To rectify this oversight, GCNs have been increasingly utilized for modelling and extracting spatial interrelations. GCNs have inherent strengths in representing non-Euclidean irregular graphs and capturing spatial correlations by aggregating information from nodes and their surroundings, rendering them particularly suitable for traffic network analysis. As a result, a surge of GCN-based traffic flow prediction models has surfaced, including T-GCN [14], STGCN [15], and STSGCN [16], among others. To further refine the capture of intricate spatial-temporal dependencies, models that incorporate attention mechanisms, such as GAT [17], have been developed and integrated into frameworks like ASTGCN [18], GAGCN [19], STN-GCN [20], along with other approaches [21, 22]. These sophisticated models combine Transformer, GCN, GRU, and additional architectures to achieve exceptional prediction capabilities. Nonetheless, as the complexity of these models escalates with the number of modules and depth, so does the computational intensity and the demand for resources.

Our study proposes an innovative model for traffic prediction that aims to combine the attention mechanism with GCN and GRU to intensively explore the relationships between data in both temporal and spatial dimensions. Our goal is to achieve prediction performance comparable to or even better than that of a complex model with a relatively simple model structure. The proposed model demonstrates significant advantages in three key aspects.

Firstly, in terms of spatial-temporal relationship modelling, we employ an improved Graph Attention Network (GAT_v2) instead of the traditional GCN approach based on a predefined adjacency matrix. GAT_v2 can adaptively learn dynamic relationships between nodes, which is particularly suitable for traffic prediction as it can accommodate spatial relation changes in the traffic network caused by regular variations or unexpected events. Secondly, to address the issue of long-term dependencies in time-series data, our model combines GRU with self-attention mechanisms. Compared to existing methods based on Long Short-Term Memory (LSTM), our combined approach is not only more concise but also performs equally well in handling lengthy sequences, providing an effective alternative. Finally, we carefully consider the computational efficiency during the design of our method, which is particularly important for resource-constrained scenarios. By optimizing the computational workflow, our model reduces the need for computational resources while maintaining high prediction performance. Compared to complex deep learning models, our approach demonstrates better practicality and scalability.

3 Problem Definition

Traffic flow refers to the movement and flow of vehicles, pedestrians, or goods in a transportation system. Traffic flow is characterized by its volume, speed, and density. These features reflect the congestion level, mobility, and efficiency of the transportation system. By managing and optimizing traffic flow, we can improve the operational efficiency and travel experience of the transportation system. Attention mechanisms have the potential to extract spatial-temporal relationships, so our study focuses on verifying their role in traffic flow prediction and evaluating the performance of designed models. To eliminate noise and perturbations caused by multiple input features, we specifically choose to predict traffic speeds and conduct experiments to visually illustrate the prediction results. Of course, our model is also applicable to predicting traffic volume and traffic density.

To capture the spatial correlation of traffic speed data, we define the road network as a graph structure. $\mathrm{G} = (\mathbf{V}, \mathbf{E}, \mathbf{A})$, Where G represents the road network graph, E represents the edge, $|\mathbf{V}| = N$ is the number of road nodes, and $\mathbf{A} \in R^{N \times N}$ is the adjacency matrix reflecting the connectivity relationship between nodes. The elements may be denoted by 0, 1, and may also be measured by distances. $\mathbf{X}$ represents the input characteristic matrix, X^{T_P} represents the feature at the P-th time step. $x_N^{T_P}$ denotes the input feature of the N-th node at the P-th time step. Here the features can be multi-dimensional, that is, speed, flow, density, etc. $\mathbf{X}$ can be expressed as Eq. (1):

$$\mathbf{X} = (X^1, X^2, ..., X^{T_p}) = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^{T_p} \\ x_2^1 & x_2^2 & \cdots & x_2^{T_p} \\ \vdots & \vdots & \ddots & \vdots \\ x_N^1 & x_N^2 & \cdots & x_N^{T_p} \end{bmatrix}. \quad \mathbf{(1)}$$

The traffic speed prediction problem becomes learning a function that can map the past P historical graphs to the future Q graphs given the known graph structure, which can be expressed as Eq. (2):

$$[X^{(t\text{-}T_p+1)}, \cdots, X^t; \mathbf{G}] \xrightarrow{f(\cdot)} [X^{(t+1)}, \cdots, X^{(t+T_Q)}]. \quad \mathbf{(2)}$$

4 Entire Structure

To effectively capture the spatial-temporal correlations in traffic data, this study proposes a spatial-temporal relationship extraction model called AMSTGCN. As shown in Fig. 3, this model consists of four core components: input module, spatial relation extraction module, temporal relation extraction module, and output module. In the input module, we perform a series of preprocessing operations on the raw traffic data to adapt to the requirements of the subsequent prediction task. These preprocessing steps include padding the missing data, removing outliers, normalizing the data, and partitioning the dataset to ensure data quality and efficient model training. The specific preprocessing method is detailed in Section 4.1 of the paper. For spatial relationship extraction, we utilize an upgraded version of GAT called GAT_v2. Compared to the original static attention mechanism in GAT, GAT_v2 can dynamically capture attention relationships, which is a significant advantage for complex and dynamic traffic road operation environments. The implementation details of this module are discussed further in Section 4.2. The temporal correlation extraction module uses gated units to capture the dependencies in the time series and combines the attention mechanism to compute the attention coefficients, thus revealing the temporal correlations between the data accurately. The specific implementation of this part is explained in detail in Section 4.3. Finally, in the output layer, we design a fully connected layer to generate multi-step prediction results. Through an organic combination of these four modules, the AMSTGCN model achieves high-precision traffic flow prediction.

4.1 Input Layer

In a traffic scenario, traffic features can include traffic speed, traffic flow, and lane occupancy. Any of these features can be chosen for traffic flow prediction. Typically, in short-term traffic flow prediction, equidistant sampling is done at intervals of 5 minutes, 10 minutes, or 15 minutes. However, traditional traffic data collectors are prone to faults, such as communication issues, power supply problems, and road maintenance, which can result in missing or abnormal data. To ensure the accuracy of subsequent predictions, the collected data needs to be reprocessed. For outliers and missing data, padding is done by computing the historical average. To make the input a feature representation that can participate in the computation of the GAT network, the form of the input data has been adjusted as Eq. (3):

$$\mathbf{X}=\left[X_{\mathbf{1}},X_{\mathbf{2}},\cdots X_{N}\right]=\begin{bmatrix} x_{\mathbf{1}}^{T_1} & x_{\mathbf{2}}^{T_1} & \cdots & x_{N}^{T_1} \\ x_{\mathbf{1}}^{T_2} & x_{\mathbf{2}}^{T_2} & \cdots & x_{N}^{T_1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{\mathbf{1}}^{T_p} & x_{\mathbf{2}}^{T_P} & \cdots & x_{N}^{T_P} \end{bmatrix}. \tag{3}$$

X_N represents the N-th node feature and $x_N^{T_P}$ represents the feature value of the N-th node at the P-th time step. $\mathbf{X}\in\mathbf{R}^{\mathbf{N}}$ is the node input feature that satisfies the GAT network operation and will be fed into the subsequent spatial relation extraction layer to obtain the spatial correlation and complete the node state update.

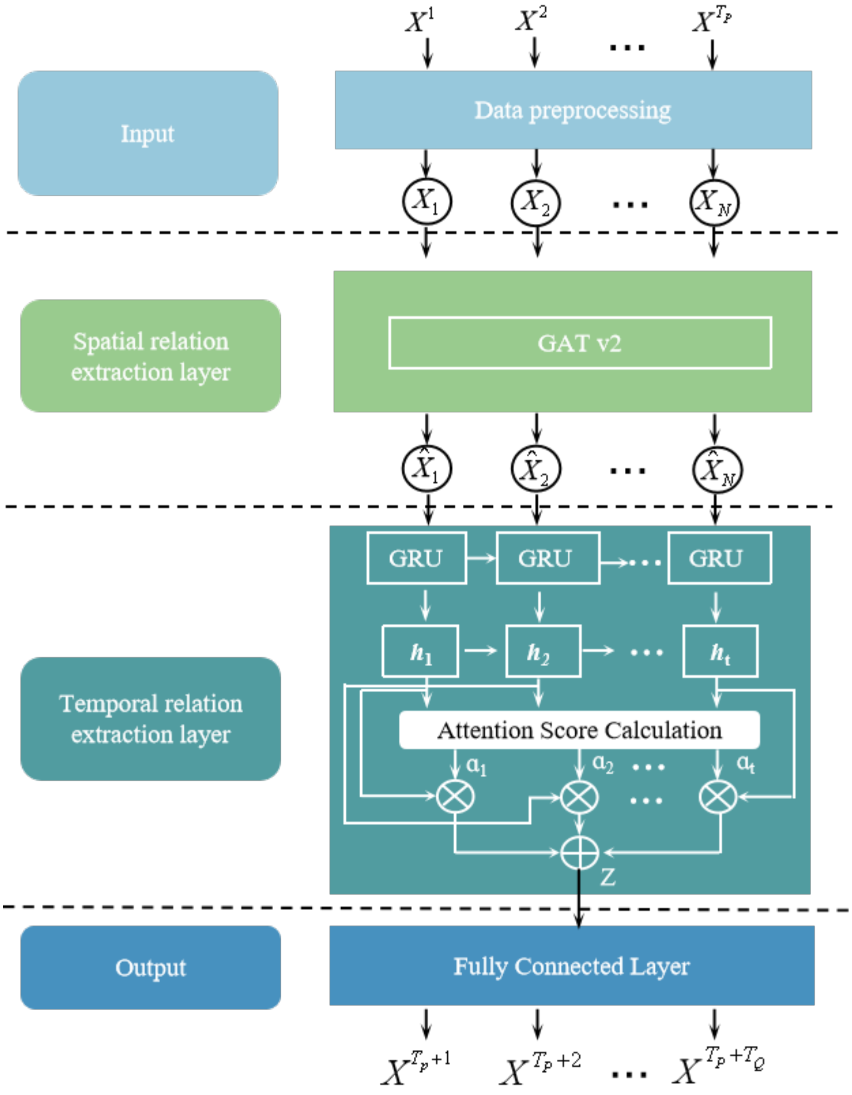


Fig. 3. Block diagram of the overall structure of AMSTGCN

4.2 Spatial Relation Extraction Layer

In the context of transportation, spatial relationships between road nodes are not only characterized by fixed spatial locations but also exhibit dynamic dependencies that shift over time. Therefore, it is crucial to obtain dynamic and adaptive relations that can account for scenario variations. GAT introduces an attenuation mechanism that allows each node to focus on its neighbours to varying degrees depending on their importance. This allows GAT to capture interactions between nodes more accurately, instead of merely averaging or weighting neighboring nodes as is done in GCN. In addition, GAT supports multi-head attention, meaning that multiple attention heads can be used simultaneously to learn the relationships between nodes. This approach enhances the expressive power of the model and enables a better capture of complex relations within the graph structure. However, reference [4] demonstrates that for a fixed set of GAT keys, the resulting attention coefficient remains relatively invariant if attention is computed using different queries on this set of keys. In other words, the ordering of attention coefficients is the same for all nodes in the graph and independent of the query node. This implies that the attention computation function is static and does not change with different queries. This is a problem with the GAT model, which significantly reduces the expressive power of GAT. To obtain a dynamic attention mechanism, a modified GAT model GAT_v2 is used in this paper. The improvement of GAT_v2 over GAT is shown in Eq. (4):

$$\begin{cases} \text{GAT} \quad \rightarrow e_{ij} = \text{LeakyRelu}\left(\alpha^{\mathrm{T}} \cdot \left[Wx_i \parallel Wx_j \right]\right). \\ \text{GATv2} \rightarrow e_{ij} = \alpha^{\mathrm{T}} \, \text{LeakyRelu}\left(W \cdot [x_i \parallel x_j]\right). \end{cases} \tag{4}$$

Observing Eq. (4), we find that GAT_v2 only modifies the order of the internal operations of GAT to play the role of repairing the attention function. Readers interested in the specific proof procedure for the GAT_v2 model can refer to reference [4], which we directly cite and apply in this paper. Fig. 4 shows the complete computational process of updating node features using GAT_v2 as an example of node *i*.

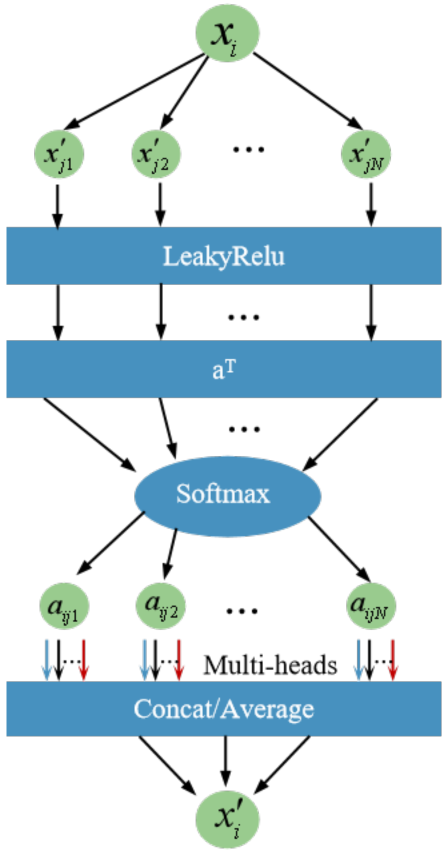


Fig. 4. Flowchart of GAT_v2 attention mechanism computation

GAT_v2 can be implemented by Eq. (5) ~ Eq. (7).

(1) Calculate the attention coefficient.

$$e_{ij} = \alpha^{\mathrm{T}} \text{ LeakyRelu}\left(W \cdot [x_i \,||\, x_j]\right) . \tag{5}$$

e_{ij} represent the attention value of node *i* relative to node *j*. α^{T} and W is shared learning parameters. || denotes the vector concatenation. This expression means that when computing the attention coefficients, the linear transformation is applied after concatenation, the nonlinear computation is done by the activation function, and finally, the transformation is applied. In this way, it can be conditioned on the query node and finally implement the computation of dynamic attention.

(2) The attention coefficients are normalized by softmax.

$$\alpha_{ij} = \text{softmax}(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in N_i} \exp(e_{ik})} . \tag{6}$$

(3) Node character updates.

$$x_i' = \sigma(\sum\nolimits_{j \in N_i} \alpha_{ij} \cdot W x_j) . \tag{7}$$

x_i' represents the current feature of node i after the fusion of neighbourhood information. σ is the activation function.

To enhance the ability to obtain spatial correlations, a multi-head attention mechanism is used. Since the final output is not the final result of our prediction, which is in the middle layer of the model, we employ a concatenation method such as Eq. (8). Of course, the sum-and-average approach can also be adopted depending on the different tasks, as shown in Eq. (9).

$$x_i' = \underset{K=\mathbf{1}}{\overset{K}{||}} \sigma\left(\sum_{j \in N_i} \alpha_{ij}^k \cdot W^k x_j\right) . \tag{8}$$

$$x_i' = \sigma\left(\frac{\mathbf{1}}{K} \sum_{K=\mathbf{1}}^{K} \sum_{j \in N_i} \alpha_{ij}^k \cdot W^k x_j\right) . \tag{9}$$

4.3 Temporal Relation Extraction Layer

In the time-related feature extraction module, we use a combination of GRU and self-attention mechanisms. Compared to RNN and LSTM, GRU has a simpler structure and a memory mechanism, making it suitable for long and short-term time series prediction. The self-attention mechanism can also extract correlations between each time step.

The structure of the GRU is shown in Fig. 5. In GRU, the update gate and reset gate are two essential gating mechanisms to control the flow and update of information. The role of the update is to determine the weight of the hidden state of the input at the current moment and the previous moment, and at the previous moment to decide whether the hidden state of the input needs to be updated. The role of the reset gate is to decide how the input information at the current moment interacts with the hidden state at the previous moment.

GRU is calculated as shown in Eq. (10):

$$\begin{cases} z_t = \sigma\left(W_Z \cdot \left[h_{t\text{-}1}, x_t\right] + b_z\right). \\ r_t = \sigma\left(W_r \cdot \left[h_{t\text{-}1}, x_t\right] + b_r\right). \\ \tilde{h}_t = \tanh\left(W_c \cdot \left[r_t \odot h_{t\text{-}1}, x_t\right] + b_c\right). \\ h_t = (1 - z_t) \odot h_{t\text{-}1} + z_t \odot \tilde{h}_t. \end{cases} \quad \textbf{(10)}$$

Where z_t is the update gate, r_t is the reset gate, $\tilde{h}_t$ is the candidate hidden state, $h_{t\text{-}1}$ is the hidden state at the previous time step, h_t is the hidden state at the last time step, $\odot$ is the Hadamard product, which stands for element-wise multiplication. σ and $\tanh$ are the activation function. W_z, W_c, and W_r are weight parameters. b_z, b_r, and b_c are the bias parameter.

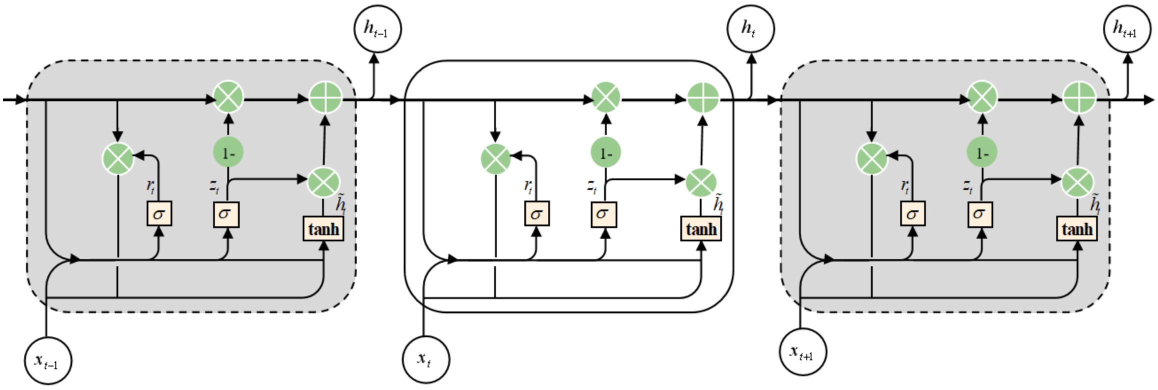


Fig. 5. The basic structure diagram of GRU

After the GRU calculation, we can obtain the hidden states at all-time steps. To further obtain the long-range dependence, we perform an attention calculation. The attention score calculation process is shown in Fig. 6, and the calculation steps are shown in Eq. (11):

$$\begin{cases} e_t = \tanh(W_e h_t + b_e). \\ \alpha_t = \text{softmax}(e_t). \\ z = \sum_{i=t-P+1}^{t} \alpha_i h_i. \end{cases} \quad \textbf{(11)}$$

Where h_t is the hidden state output of the GRU, W_e and b_e are the learnable weight and bias parameters, respectively.

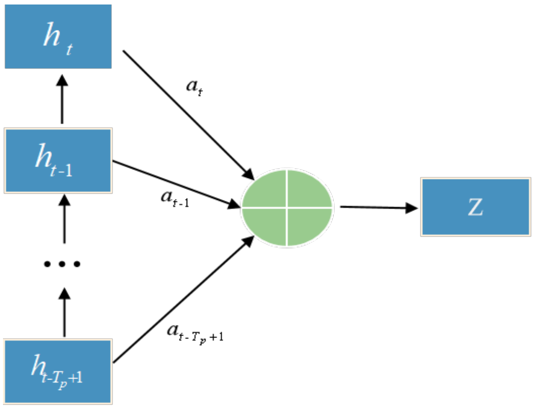


Fig. 6. Schematic of the attention mechanism followed by GRU

4.4 Output Layer

Our goal is to predict the traffic flow for Q steps into the future. Therefore, a fully connected layer is used in the output layer to complete the dimensional transformation.

$$Y = \text{Relu}(W_o Z + b_o) \,. \tag{12}$$

The fully connected input is the attention value Z obtained by the time extraction layer, $W_O \in R^{N\times Q}$ is the learnable parameter and the output of the output layer is the traffic feature of each node in the future Q time steps.

4.5 Loss Function

During the model training process, the primary objective is to minimize the discrepancy between the observed traffic speed and the predicted values generated by the model. We denote the true traffic speed Y and the predicted traffic speed by $\hat{Y}$. The loss function as Eq. (13):

$$Loss = \left\| Y - \hat{Y} \right\| + \lambda L_{reg} \,. \tag{13}$$

The first term in Eq. (13) is used to calculate the difference between the actual traffic velocity and the expected velocity. The next component is the L2 regularization component, which is used to control the complexity of the model and λ is a hyperparameter.

5 Experiments

5.1 Datasets and Experimental Settings

To evaluate the performance of our model, we conduct experiments using the publicly available Loop Seattle dataset. This dataset was collected by the Seattle Department of Transportation and consists of traffic speed data from 323 sensor stations located on highways in the Seattle area (I-5, I-405, I-90, and SR-520). The data spans the entire year 2015 and is collected at a resolution of 5 minutes [23].

In the experiments, we split the datasets using a ratio of 0.7:0.1:0.2, which means the Train dataset: Valid dataset: Test dataset = 0.7: 0.1: 0.2. The data is normalized after partitioning and we use z-score normalization method for normalization: $\overline{x} = (x - \mu)/\sigma$, x_{max} is the maximum and x_{min} is the minimum of the sample data.

The model is developed using the PyTorch 1.9.0 deep learning framework. The specific configuration information is as follows: CPU: Intel(R) Core(TM) i7-7800X，24GB Graphics Card: GeForce RTX 3090, CUDA version: 11.3.

5.2 Evaluation Metrics

To evaluate the performance of the AMSTGCN model, we use two evaluation metrics, namely Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|Y - \hat{Y}\right| . \tag{14}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y - \hat{Y})^2} . \tag{15}$$

Y represents the true traffic speed, $\hat{Y}$ represents the predicted traffic speed. The smaller MAE and RMSE demonstrate the better prediction performance of the model.

5.3 Experiment and Result Analysis

To evaluate the performance of the model, we performed a series of experiments, including comparing the predictive power of the model to the baseline model, analyzing the effect of different components on the model performance, and measuring the computational time cost.

(1) Comparison experiments with baseline models.

Our task is to predict future velocities at the 3rd, 9th, and 12th time points using the known velocity values at the past 12 sampling points. Given that the raw data is sampled at 5-minute intervals, this amounts to predicting the next 15, 45, and 60-minute velocities based on historical velocity data from the past hour. To evaluate the model performance, we compare AMSTGCN with five baseline models. The comparison of the prediction performance of the AMSTGCN model on the LOOP_SEATTLE dataset with the five baseline methods is presented in Table 1.

Table 1. The prediction performance of the AMSTGCN model and other baseline methods on the LOOP_SEATTLE dataset

Methods	LOOP_SEATTLE					
	15min		45min		60min	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
HA	5.32	8.96	5.32	8.96	5.32	8.96
FNN	3.17	5.99	4.45	8.13	4.99	9.05
GRU	4.27	7.67	4.40	7.93	4.52	8.14
T-GCN	3.65	5.95	4.86	7.84	5.32	8.64
DCRNN	2.94	5.96	4.07	7.33	4.40	8.15
AMSTGCN	3.73	5.94	4.06	6.62	4.21	6.91

HA: History Average Model. It predicts future observations based on the average value of past observations over a certain period.

FNN: Fully connected Neural Network. We constructed the simplest three-layer fully connected neural network to verify the prediction performance of the simple model.

GRU: Gated Recurrent Unit employs gate mechanisms to regulate information flow, mitigating gradient vanishing and exploding issues prevalent in traditional RNNs.

T-GCN [14]: Temporal Graph Convolutional Network exploits graph convolutions to discern node interactions and GRUs to apprehend temporal dynamics.

DCRNN [24]: Diffusion Convolutional Recurrent Neural Network synergizes diffusion convolution with recurrent networks to model the spatial-temporal dynamics inherent to traffic networks.

Based on the experimental data, the AMSTGCN model shows significant advantages in spatial-temporal prediction tasks. By combining the spatial correlation acquisition capability of GAT_v2 with the temporal feature extraction advantage of the attention mechanism, this model significantly improves the predictive performance across different time scales. Specifically, AMSTGCN does not outperform DCRNN, FNN, and T-GCN when predicting 15 minutes, but its MAE of 3.73 and RMSE of 5.94 are still quite impressive. This indicates that AMSTGCN can provide competitive results even within a relatively short prediction window.

The advantage of AMSTGCN becomes apparent as the prediction horizon extends to 45 minutes. MAE was reduced to 4.06 and RMSE was further reduced to 6.62, outperforming all compared models. This suggests that AMSTGCN has a stronger ability to capture and exploit long-term dependencies in the data.

The advantage of AMSTGCN becomes even more prominent at the 60-minute prediction point. It achieves an MAE of 4.21 and RMSE of 6.91, again showing the lowest error rate among all models. This significant performance improvement is attributed to the deep spatial relationship mining capability of GAT_v2 and the flexibility provided by the attention mechanism in handling temporal information.

In summary, the AMSTGCN model not only maintains good performance in short-term prediction but also demonstrates excellent capabilities in long-term prediction. This is driven by the carefully designed model structure, in particular the efficient integration of GAT_v2 and attention mechanisms, which enables AMSTGCN to accurately capture the crucial spatial-temporal dynamics in complex data. As a result, it achieves higher accuracy and reliability in future predictions. This has practical implications and applications in domains that require accurate spatial-temporal predictions, such as traffic management, weather forecasting, and urban planning.

The predictions for node 10 and node 320 in the LOOP_SETTLE dataset are displayed in Fig. 7 and Fig. 8, which help to make the prediction results more understandable.

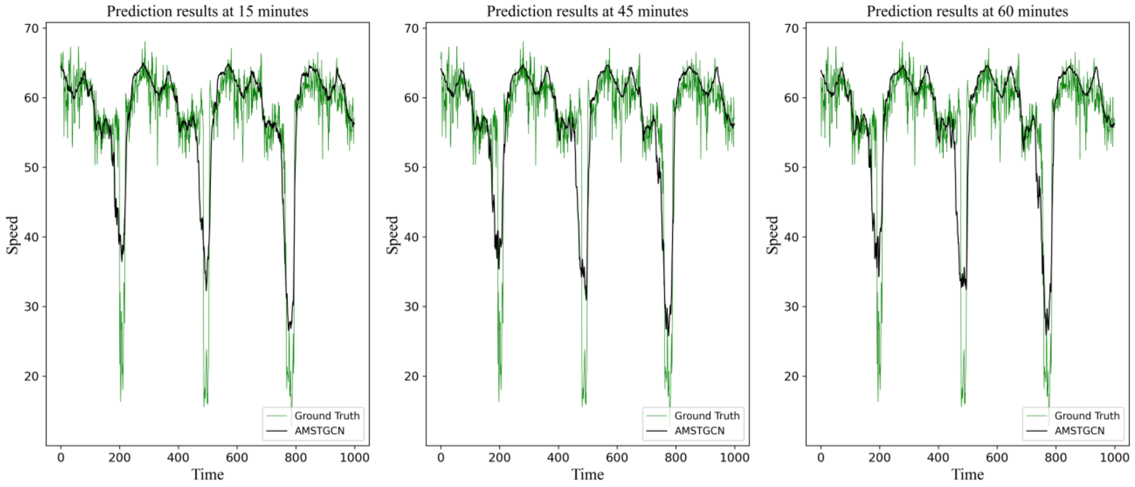


Fig. 7. Visualization of prediction results for node 10 in the LOOP_SEATTLE dataset

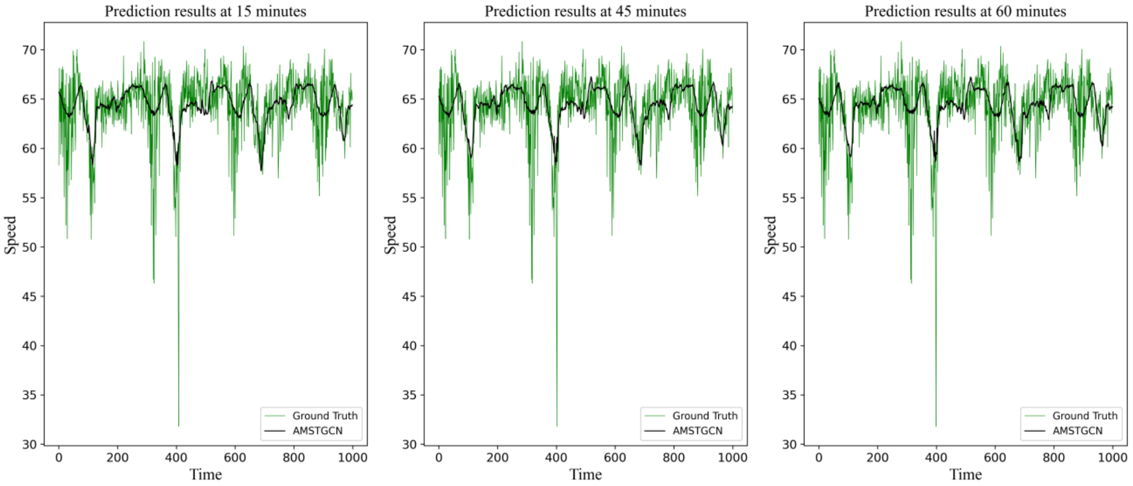


Fig. 8. Visualization of prediction results for node 320 in the LOOP_SEATTLE dataset

(2) Performance testing of models with different components.

To deeply investigate the specific role of GAT_v2 and the attention mechanism in spatial-temporal prediction models, we conduct a series of comparative experiments. These experiments aim to assess the contribution of each component in capturing spatial-temporal correlations through different combinations of models. Specifically, we integrated and combined baseline models such as GAT, GAT_v2, GRU, and Attention Mechanisms to construct various hybrid models. These hybrid models are designed to reveal the unique value and synergistic impact of each module in integrating spatial-temporal information. For ease of comparison and understanding, we present the structure of different model combinations in Table 2. In addition, to simplify the exposition and aid the reader's understanding, we refer to the AMSTGCN model as G2GA.

Table 2. Models and naming of different combinations

Model name	GG	GGA	G2G	G2GA (AMSTGCN)
Combination	GAT+GRU	GAT+GRU+Attention	GAT_v2+GRU	GAT_v2+GRU+Attention

With these comprehensive tests, we aim to demonstrate the advantage of GAT_v2 in spatial relation mining and the effectiveness of the attention mechanism in extracting temporal sequence features. We also compare the MAE and RMSE values predicted for different time points. The performance tests for the models with different components are shown in Table 3.

We further examine the performance of different models at prediction intervals of 15, 30, 45, and 60 minutes. At the 15-minute point, the G2G model performs slightly better than the others, but the AMSTGCN model is also very close in performance. However, as the prediction duration increased, especially in the 60-minute forecast task, we observed that the AMSTGCN model has lower MAE and RMSE values compared to the other three models, with respective values of 4.21 and 6.91. This indicates that the AMSTGCN model is more effective in capturing and exploiting complex patterns within spatial-temporal data, especially for long-term prediction. This can be attributed to the integration of GAT_v2 and attention mechanisms in the AMSTGCN model, which are better equipped to capture spatial relationships and temporal sequential features in spatial-temporal data. GAT_v2 has the advantage of mining spatial relationships to effectively capture correlations between geographic locations, while the attention mechanism can weight information across different time steps in the temporal dimension to extract salient features from the time series. With this combination, the AMSTGCN model can predict future spatial-temporal changes with greater accuracy. This has significant practical implications for applications in various fields such as traffic flow prediction, weather prediction, and human motion prediction.

Overall, through a comprehensive comparison and analysis of different models, we have validated the specific role of GAT_v2 and the attention mechanism in spatial-temporal prediction models. As a hybrid model integrating these two components, the AMSTGCN model demonstrates superior performance in the long-term spatial-temporal prediction task. These findings provide valuable references and guidance for further research and applications of spatial-temporal prediction models. The visual comparison results for MAE and RMSE are shown in Fig. 9.

Table 3. Performance testing of models with different components

Methods	LOOP_SEATTLE							
	15min		30min		45min		60min	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
GG	3.76	5.92	3.96	6.37	4.12	6.72	4.31	7.10
G2G	3.67	5.83	3.90	6.36	4.09	6.77	4.27	7.10
GGA	3.72	5.95	3.89	6.33	4.07	6.68	4.24	6.98
G2GA (AMSTGCN)	3.73	5.94	3.90	6.29	4.06	6.62	4.21	6.91

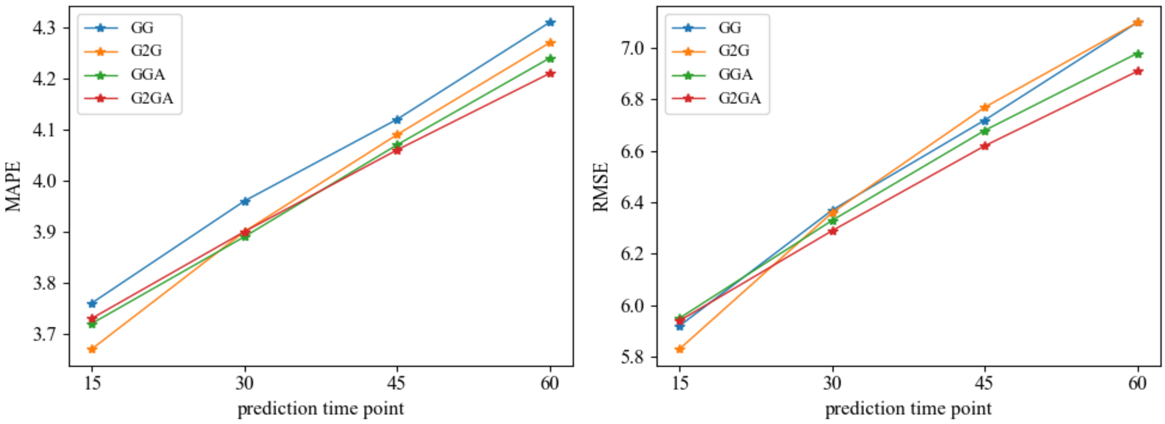


Fig. 9. Changes in performance metrics of different models for prediction tasks of different time lengths

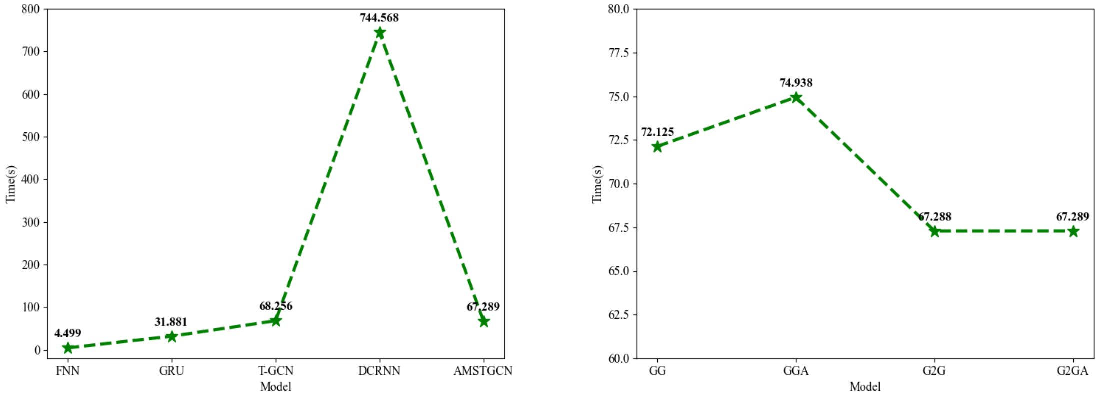
(3) Model training time measurements.

To demonstrate the computational performance advantage of AMSTGCN, we compare its training time with different models. Fig. 10(a) illustrates the comparison of AMSTGCN's training time with four baseline models. We can see that AMSTGCN has a relatively short training time of 67.289 seconds. By comparing the results, we can observe that while the training time of AMSTGCN is longer than FNN and GRU, it is significantly shorter than DCRNN, which requires the most training time. In addition, AMSTGCN also has a slightly shorter training time compared to T-GCN. This indicates that even though AMSTGCN is more complex than some simple models like FNN, it is more efficient in handling complex graph data.

Fig. 10(b) illustrates the training time with different components. When comparing the number of training epochs of different models, we find that the G2G model has a significant computational efficiency advantage over the GG model. Specifically, the training time of the G2G model is 67.288s, while the training time of the GG model is 72.125s. This indicates that the GAT_v2 version is more efficient than the original GAT version, saving approximately 4.837s of training time without introducing the Attention mechanism. When we introduce the attention mechanism into both GG and G2G models, we observe an increase in training time. This is because the attention mechanism adds complexity and computational overhead to the model. However, even after incorporating the Attention mechanism, the training time of the AMSTGN model remains highly close to that of the G2G model with only a 0.001s increase. This minor increase is negligible and suggests that the AMSTGN model maintains strong computational efficiency while improving model performance with the Attention mechanism. In contrast, the training time of the GGA model is significantly increased to 74.938s, which is an increase of 2.813s compared to the GG model. This increase in time may be attributed to the lower computational efficiency of the

GAT version when dealing with the Attention mechanism.

Therefore, we can conclude that the GAT_v2 version provides higher computational efficiency under the same conditions, while the AMSTGN model maintains acceptable computational efficiency while incorporating the Attention mechanism.



(a) Training time with baseline models (b) Training time with different components

Fig. 10. Changes in performance metrics of different models for prediction tasks of different time lengths

We can conclude that the advantage of AMSTGN lies in its ability to capture temporal dependencies and graph structural features, which allows it to maintain relatively high accuracy while effectively controlling the computational cost. Therefore, AMSTGCN should be a relatively good choice if the application scenario requires taking into account the dynamic nature of the graph and the complex interactions between nodes.

6 Conclusions

This study presents a novel model that integrates GAT_v2 and GRU to address the core issues in traffic prediction. This model goes beyond the limitations of traditional GCN in modelling spatial-temporal relationships. By adaptively learning the dynamic relationships between nodes, it effectively handles spatial relation changes caused by regular variations or unexpected events in the traffic network. Moreover, the model combines the self-attention mechanism and GRU to elegantly address the problem of long-term dependencies in time series data. Moreover, the model is designed with a focus on computational efficiency, optimizing the computational process to adapt to resource-constrained real-world applications while maintaining strong predictive performance. Experimental results on public datasets validate the superior performance of the proposed model compared to existing methods on short-term and long-term traffic prediction tasks and also demonstrate its excellent generalization ability and robustness.

Future work can further expand and deepen the achievements of this study in several directions. First, explore the integration of this model with different types of spatial-temporal data modelling approaches, such as introducing multi-scale analysis or considering more complex spatial-temporal relationships. Second, given the diversity of real-world traffic scenarios, the adaptability and robustness of models remain crucial research topics that can be tested and improved by introducing more diverse datasets and scenarios. Finally, as computational resources continue to evolve, exploring how to leverage parallel computing and distributed systems to tackle larger-scale traffic prediction problems is also an essential direction for future research.

7 Acknowledgement

This research was funded by the Natural Science Foundation of Gansu Province under Grant (22JR5RA186), the

Fundamental Research Funds for the Central Universities under Grant (31920230139), and the National Natural Science Foundation of China (62366048).

References

[1] D.A. Tedjopurnomo, Z.F. Bao, B.H. Zheng, F. Choudhury, A.K. Qin, A Survey on Modern Deep Neural Network for Traffic Prediction: Trends, Methods and Challenges, IEEE Transactions on Knowledge and Data Engineering 34(4) (2022) 1544-1561.

[2] J. Ye, J. Zhao, K. Ye, C. Xu, How to Build a Graph-Based Deep Learning Architecture in Traffic Domain: A Survey, IEEE transactions on intelligent transportation systems 23(5)(2022) 3904-3924.

[3] Y. Ma, S. Cheng , Y. Ma , Y. Ma, Convolutional Neural Networks and their applications in Intelligent Transportation Systems: A Survey, Journal of Transportation Engineering 21(4)(2021) 24.

[4] S. Brody, U. Alon, E. Yahav, How Attentive are Graph Attention Networks?, in: Proc. 2022 International Conference on Learning Representations(ICLR), 2022.

[5] T.N. Kipf, M. Welling, Semi-Supervised Classification with Graph Convolutional Networks, in: Proc. 2017 International Conference on Learning Representations(ICLR), 2017.

[6] T.H.H. Aldhyani, M.R. Joshi, Enhancement of Single Moving Average Time Series Model Using Rough k-Means for Prediction of Network Traffic, International Journal of Engineering Research and Applications 7(3)(2017) 45-51.

[7] Z.E. Abou Elassad, H. Mousannif, H. Al Moatassime, A proactive decision support system for predicting traffic crash events: A critical analysis of imbalanced class distribution, Knowledge-Based Systems 205(2020) 14.

[8] Y. Wen, R. Wu, Z. Zhou, S. Zhang, S. Yang, T.J. Wallington, W. Shen, Q. Tan, Y. Deng, Y. Wu, A data-driven method of traffic emissions mapping with land use random forest models, Applied Energy 305(2022) 117916.

[9] D. Wang, C. Wang, J. Xiao, Z. Xiao, W. Chen, V. Havyarimana, Bayesian optimization of support vector machine for regression prediction of short-term traffic flow, Intelligent data analysis 23(2)(2019) 481-497.

[10] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, B. Yin, Deep Learning on Traffic Prediction: Methods, Analysis, and Future Directions, IEEE Transactions on Intelligent Transportation Systems 23(6)(2022) 4927-4943.

[11] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, Learning Traffic as Images: A Deep Convolutional Neural Network for Large-Scale Transportation Network Speed Prediction, Sensors 17(4)(2017) 818.

[12] R. Bikmukhamedov, A. Nadeev, G. Maione, D. Striccoli, Comparison of HMM and RNN models for network traffic modeling, Internet Technology Letters 3(2)(2020) e147.

[13] X. Zhang, Q. Zhang, Short-Term Traffic Flow Prediction Based on LSTM-XGBoost Combination Model, Computer Modeling in Engineering & Sciences 125(1)(2020) 95-109.

[14] L. Zhao, Y. Song, C. Zhang, Y. Liu, H. Li, T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction, IEEE Transactions on Intelligent Transportation Systems (99)(2019) 1-11.

[15] B. Yu, H.T. Yin, Z.X. Zhu, Spatial-temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting, in: Proc. 27th International Joint Conference on Artificial Intelligence (IJCAI), 2018.

[16] C. Song, Y. Lin, S. Guo, H. Wan, Spatial-Temporal Synchronous Graph Convolutional Networks: A New Framework for Spatial-Temporal Network Data Forecasting, in: Proc. 2020 Association for the Advancement of Artificial Intelligence(AAAI), 2020.

[17] P. Velikovi, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph Attention Networks, in: Proc. 2018 International Conference on Learning Representations(ICLR), 2018.

[18] S. Guo, Y. Lin, N. Feng, C. Song, H. Wan, Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting, in: Proc . 2019 Association for the Advancement of Artificial Intelligence(AAAI), 2019.

[19] C. Tang, J. Sun, Y. Sun, Dynamic Spatial-Temporal Graph Attention Graph Convolutional Network for Short-Term Traffic Flow Forecasting, in: Proc. 2020 IEEE International Symposium on Circuits and Systems (ISCAS), 2020.

[20] C. Wang, L. Wang, S. Wei, Y. Sun, B. Liu, L. Yan, STN-GCN: Spatial and Temporal Normalization Graph Convolutional Neural Networks for Traffic Flow Forecasting, Electronics 12(14)(2023) 3158.

[21] Y. Chen, J. Huang, H. Xu, J. Guo, L. Su, Road traffic flow prediction based on dynamic spatiotemporal graph attention network, Scientific Reports 13(1)(2023) 14729.

[22] W. Fang, W. Zhuo, J. Yan, Y. Song, D. Jiang, T. Zhou, Attention meets long short-term memory: A deep learning network for traffic flow forecasting, Physica A: Statistical Mechanics and its Applications 587(2022) 126485.

[23] Z.Y. Cui, K. Henrickson, R.M. Ke, Y.H. Wang, Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting, IEEE Transactions on Intelligent Transportation Systems 21(11)(2020) 4883-4894.

[24] Y. Li, R. Yu, C. Shahabi, Y. Liu, Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting, in: Proc. 2018 International Conference on Learning Representations (ICLR), 2018.