http://dx.doi.org/10.62762/TETAI.2025.387543

RESEARCH ARTICLE

AST-GNNFormer: Adaptive Spatio-Temporal Graph Neural Network with Layer-Aware Preservation for Traffic Flow Prediction

Yusen Zheng[1] **and Xuebo Jin**[1,*]

[1] School of Computer Science and Artificial Intelligence, Beijing Technology and Business University, Beijing 100048, China

Abstract

Accurate traffic flow prediction plays a critical role in intelligent transportation systems, providing essential support for urban planning, traffic control, and congestion mitigation. To address the challenges of spatial heterogeneity and temporal dynamics inherent in traffic data, this paper proposes AST-GNNFormer, an adaptive spatio-temporal graph neural network that integrates graph attention mechanisms with temporal convolution. The model introduces three key components to enhance predictive accuracy and generalization: (1) a Layer-aware Information Preservation mechanism that mitigates over-smoothing in deep GNNs by retaining original node features across layers; (2) an Inter-Layer Attention Module that dynamically selects and weights informative layer-wise features to improve multi-layer fusion quality; and (3) an Adaptive Graph Learning Module that fuses prior adjacency knowledge with learnable structures, enabling dynamic topology adaptation. Additionally, a Temporal Convolution Module is incorporated to model multi-scale temporal dependencies efficiently. Extensive experiments on real-world benchmark datasets (PEMS04 and PEMS08) demonstrate that AST-GNNFormer significantly outperforms existing state-of-the-art methods in both short-term and long-term traffic forecasting tasks. Ablation studies further confirm the effectiveness of each proposed component.

Keywords: graph neural network, traffic flow prediction, adaptive graph learning, inter-layer attention mechanism.

1 Introduction

With the acceleration of urbanization, traffic congestion and unexpected incidents occur more frequently. Accurate traffic flow prediction can significantly enhance traffic scheduling efficiency, reduce the likelihood of accidents, lower economic and time costs, and alleviate travel stress [1]. As one of the core technologies of intelligent transportation systems, traffic flow prediction aims to forecast future traffic patterns based on historical observation data, playing a crucial role in modern urban traffic management. Moreover, prediction results can support route planning, vehicle scheduling, zone

Submitted: 07 July 2025
Accepted: 23 October 2025
Published: 26 October 2025

Vol. 2, **No.** 4, 2025.
10.62762/TETAI.2025.387543

***Corresponding author:**
✉ Xuebo Jin
jinxuebo@btbu.edu.cn

Citation
Zheng, Y., & Jin, X. (2025). AST-GNNFormer: Adaptive Spatio-Temporal Graph Neural Network with Layer-Aware Preservation for Traffic Flow Prediction. *ICCK Transactions on Emerging Topics in Artificial Intelligence*, 2(4), 203–219.

development, and road siting, providing data-driven insights to improve overall operational efficiency and safety [2].

In the field of traffic prediction, a major challenge lies in effectively modeling the complex spatiotemporal dependencies inherent in traffic data. With the rapid advancement of deep learning techniques, traffic flow prediction models have made remarkable progress. Early methods, for example, adopted convolutional neural networks (CNNs) to extract spatial features and combined them with long short-term memory (LSTM [3]) networks to capture temporal dependencies, forming hybrid architectures such as CNN-LSTM [4]. These methods laid a solid foundation for subsequent research in spatiotemporal modeling. In recent years, Graph Neural Networks (GNNs) have emerged as a dominant approach due to their ability to explicitly model the topological structure of road networks and overcome the limitations of traditional CNNs in handling non-Euclidean spatial data. GNNs are also capable of dynamically capturing spatiotemporal dependencies, making them particularly suitable for traffic flow prediction tasks.

Despite the promising performance of GNNs in traffic forecasting, their practical applications still face three major challenges:

First, deep GNNs tend to suffer from over-smoothing and information loss. When stacking multiple layers in a GNN, node representations become increasingly similar due to repeated neighbor aggregation, leading to reduced distinguishability between nodes—an issue known as over-smoothing. Meanwhile, information from distant nodes is gradually compressed through the layers, causing long-range dependencies to be inadequately propagated. This results in homogenized node representations and the loss of long-distance information, ultimately weakening the model's ability to represent complex graph structures.

Second, graph structures are often statically or manually defined, making them less adaptable to complex relational dynamics. Existing GNNs typically rely on predefined adjacency matrices or learnable parameters to build graph structures, which limits their ability to capture hidden or semantic spatial relationships among nodes, thereby restricting their expressiveness in modeling complex traffic networks.

Third, GNNs often exhibit limited capability in temporal modeling. Traditional GNNs are not well-equipped to capture dynamic patterns across the time dimension, making it difficult to model the temporal evolution of traffic flows. This insufficiency leads to error accumulation in multi-step forecasting, particularly when dealing with traffic data that exhibits strong temporal dependencies.

To address the above challenges, this paper proposes an Adaptive Spatio-Temporal Graph Neural Network with Layer-Aware Preservation model, referred to as AST-GNNFormer. This model integrates a cross-layer information preservation mechanism, an adaptive graph structure learning strategy, and a temporal convolution module to jointly enhance modeling accuracy and generalization ability for complex spatiotemporal prediction tasks. Firstly, as a core technical contribution, to mitigate the over-smoothing and feature degradation issues caused by deep GNN stacking, we introduce a Layer-aware Information Preservation (LIP) mechanism. This mechanism employs residual connections to explicitly preserve original node features during message passing. Additionally, a Cross-layer Dynamic Gating strategy based on attention weights is proposed to selectively transfer inter-layer information, thereby alleviating over-smoothing effectively. Secondly, to overcome the reliance on prior adjacency matrices, we further propose a hybrid approach that combines prior spatial knowledge with learnable parameters for adaptive graph structure generation. This method retains known spatial relationships while introducing a structure optimization module to model potential high-order dependencies and semantic correlations. It enables dynamic adjustment and adaptive representation of graph structures, thus improving the model's expressiveness and generalizability in complex traffic scenarios. Finally, to compensate for the limited temporal modeling capacity of GNNs, we incorporate a 1D convolution-based temporal modeling module that captures local dependencies and evolution patterns in the time series input. By modeling dynamic changes across time steps, this module enhances the model's sensitivity to temporal patterns, thereby boosting prediction accuracy in complex traffic flow forecasting tasks.

The structure of this paper is organized as follows: Section 2 summarizes recent advances in graph neural networks for temporal forecasting and traffic flow prediction, covering static and dynamic graph modeling, attention-based mechanisms, and multi-scale feature fusion. Section 3 presents the architecture of AST-GNNFormer in detail, including the Graph Multi-Head Attention Mechanism,

Information Retention Module, Inter-Layer Attention Module, Adaptive Graph Learning Module, and Temporal Convolution Module, highlighting their roles in capturing complex spatiotemporal dependencies. Section 4 introduces the datasets, evaluation metrics, and baseline methods. It reports performance comparisons and ablation studies to validate the effectiveness of each component in AST-GNNFormer. Section 5 concludes the contributions of the paper and discusses potential future directions, such as lightweight model deployment and robust traffic anomaly detection.

2 Related Work

2.1 Graph Neural Networks for Forecasting

In recent years, Graph Neural Networks (GNNs) have demonstrated powerful modeling capabilities in handling structured data, particularly in temporal prediction tasks that require the joint modeling of spatial dependencies and temporal dynamics. GNNs have emerged as a significant branch of deep learning research. Traditional temporal modeling methods, such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM [3]) networks, and 1D Convolutional Neural Networks (1D-CNNs [5]), are adept at capturing the dynamic evolution of time series data. However, they often struggle to effectively integrate complex spatial structural information. This limitation is especially evident in applications like traffic forecasting, power load prediction, environmental monitoring, and social network analysis, where the data exhibit not only strong temporal correlations but also pronounced spatial dependencies. These spatial relations are typically non-Euclidean and thus cannot be directly processed by conventional models, which limits the potential accuracy of predictions.

To address this challenge, Graph Neural Networks have been incorporated into temporal modeling frameworks. GNNs leverage graph structures to represent irregular relationships between entities and combine them with temporal modeling mechanisms to achieve joint spatiotemporal dependency learning. A representative model is the Diffusion Convolutional Recurrent Neural Network (DCRNN [6]), which introduced diffusion processes into graph structure modeling and embedded them into recurrent units to simulate traffic flow propagation over road networks. DCRNN [6] achieved superior performance compared to traditional temporal models and pioneered the integrated modeling of graph structures and time series.

Building upon this, Spatio-Temporal Graph Convolutional Networks (ST-GCN [7]) further enhanced the capability of modeling dynamic sequences by incorporating temporal convolutions into graph convolutional networks. Though originally designed for action recognition, the spatiotemporal decoupling strategy of ST-GCN [8] has been widely adopted in various domains, including traffic, meteorology, and energy systems.

Building on this foundation, researchers started incorporating attention mechanisms into GNNs to assign different levels of importance to nodes and time steps, enhancing the model's ability to adapt to dynamic environments. The Attention-based Spatio-Temporal Graph Convolutional Network (ASTGCN [9]) added spatial and temporal attention modules to dynamically modify the weight relationships between nodes at each time step, effectively overcoming the limitations of static graph structures in time-varying scenarios. Similarly, the Graph Multi-Attention Network (GMAN [10]) used multiple attention mechanisms to combine information across various nodes and time steps, allowing for flexible modeling of diverse spatiotemporal relationships. Additionally, Graph WaveNet (GWN [11]) employed a gated temporal convolutional network for sequence modeling and a hybrid graph convolution framework to manage both static and adaptive graph structures, greatly improving stability and generalization in multi-step forecasting tasks.

Another notable direction is the learning of implicit graph structures and dynamic graph learning. For instance, the Multivariate Time-Series Graph Neural Network (MTGNN [12]) eliminates the need for predefined graph structures by automatically learning latent graph connections from multivariate time series data. It integrates graph convolution with gated convolution to enable end-to-end training, dynamically optimizing node relationships during training rather than relying on manually constructed adjacency matrices. This substantially enhances the model's ability to adapt to complex dynamic systems. Furthermore, the Adaptive Graph Convolutional Recurrent Network (AGCRN [13]) introduced a node-specific adaptive graph construction mechanism, allowing each node to learn a personalized graph structure and thereby overcoming the performance bottlenecks caused by using a shared graph for

heterogeneous nodes.

Despite substantial progress, the application of GNNs in temporal prediction still faces several challenges. First, most models rely on fixed topologies, which limits their ability to capture evolving inter-node relationships in dynamic systems. Second, deep GNNs often suffer from the over-smoothing problem, where node representations become indistinguishable as more convolution layers are stacked, thus reducing discriminative power. Additional challenges include maintaining computational efficiency in large-scale graphs, effectively integrating multi-scale temporal features, and improving model robustness against anomalous fluctuations.

To tackle these issues, various innovative approaches have emerged in recent years. Dynamic Graph Neural Networks (DGNNs [14]) have been introduced into temporal prediction tasks by modeling temporal graph evolution through sliding windows, dynamic adjacency matrix learning, and structure evolution mechanisms. Meanwhile, some studies have integrated self-supervised learning with graph modeling by designing pretext tasks such as adjacency prediction and masked reconstruction, thereby enhancing the model's structural understanding and generalization in low-resource or long-horizon prediction scenarios. Techniques such as residual connections, skip-layer aggregation, and graph pooling have also been widely employed to alleviate over-smoothing and vanishing gradients, thereby improving the stability and representational power of deep GNNs.

2.2 Applications of Graph Neural Networks in Traffic Flow

With the rapid development of urban transportation systems, accurately predicting future traffic flow states has become a critical issue in intelligent transportation systems (ITS). Traditional traffic flow forecasting methods mainly rely on time series models (e.g., ARIMA), machine learning algorithms (e.g., SVR, RF), or deep learning architectures such as RNNs and CNNs. Although these methods have achieved success in modeling temporal dependencies, they often overlook the inherent non-Euclidean spatial relationships in road networks and fail to effectively capture complex topological dependencies between nodes, resulting in limited prediction performance.

Graph Neural Networks (GNNs), as powerful tools for modeling graph-structured data, have been widely introduced into traffic forecasting tasks in recent years. GNNs naturally represent the connections among traffic sensing nodes and model spatial dependencies through information propagation over graph structures. DCRNN [6] was the first to integrate diffusion convolution with recurrent neural networks to simulate the dynamic propagation of traffic flow on graphs. It demonstrated significant performance gains over traditional methods on datasets such as METR-LA and PEMS-BAY. The success of DCRNN [6] marked the establishment of a new prediction paradigm combining graph structure with temporal modeling.

To further enhance modeling flexibility, GWN [11], employed a Gated Temporal Convolutional Network (GTCN) for temporal modeling and introduced adaptive adjacency matrices for learning graph structures, thereby removing reliance on static prior graphs. GMAN [10] introduced spatio-temporal multi-head attention mechanisms to finely model heterogeneous spatiotemporal dependencies, significantly improving the model's generalization ability in complex scenarios.

Recently, dynamic graph modeling has become a research focus. HTVGNN [15] (Hybrid Time-Varying Graph Neural Network), utilized a time-varying masked attention mechanism and coupled graph structure learning to jointly model static and dynamic graphs, enhancing the robustness of long-term prediction. EG-NODE, introduced neural ordinary differential equations (Neural ODEs) to enable continuous-time modeling of graph topology, thereby improving the representation of complex structural changes. These models no longer rely on static graphs but dynamically learn semantic associations between nodes during prediction, making them more suitable for the high-frequency variability of real-world road networks.

Multi-scale spatiotemporal feature fusion has also emerged as a key research direction in recent years. MSTIFNet [16] (Multi-Scale Temporal Information Fusion Network) enhances sensitivity to traffic patterns at different temporal granularities through multi-scale modeling. GFAGNN [17] (Gated Fusion Adaptive Graph Neural Network) combines gated convolution and graph attention mechanisms to adaptively aggregate features of different road nodes across time periods, making it well-suited for highly dynamic urban scenarios.

Meanwhile, as sensor deployments expand and

privacy concerns rise, self-supervised learning and federated learning have become active research frontiers for GNN-based forecasting. FLAGCN [18] (Federated Learning and Asynchronous Graph Convolutional Network) integrates federated learning with asynchronous graph convolution, safeguarding user privacy while improving training efficiency. This represents a promising direction for distributed traffic forecasting systems. In the realm of self-supervised graph learning, several studies have designed auxiliary tasks such as graph reconstruction and adjacency prediction to enhance the model's understanding of graph structure and alleviate the problem of limited labeled traffic data.

Recently, Jiang [19] conducted a comprehensive survey on the application of GNNs in traffic flow forecasting and identified several ongoing challenges, including dynamic graph representation learning, modeling anomalous traffic conditions, and heterogeneous graph fusion. In particular, existing models often exhibit limited robustness in the presence of unexpected events such as extreme weather or traffic accidents. Furthermore, the computational bottlenecks associated with large-scale graph processing pose significant barriers to real-world deployment.

In conclusion, although existing studies have made remarkable progress in graph structure modeling, spatiotemporal feature fusion, and federated learning, they still fall short in handling sudden and abnormal traffic events (e.g., extreme weather, traffic disruptions). Current models often lack effective dynamic response mechanisms and robust forecasting capabilities. Additionally, the computational inefficiency associated with large-scale graph modeling severely restricts the deployment and practical application of these models. These challenges form the core motivation of this study, which aims to enhance the generalization and applicability of GNNs in dynamic and noisy traffic environments by introducing integrated techniques including dynamic graph learning, multi-scale temporal modeling, and multi-head graph attention mechanisms.

3 Methodology

To address the challenges of spatial heterogeneity and temporal dynamics in graph-structured data for traffic flow prediction, this paper proposes a novel spatiotemporal graph neural network model named AST-GNNFormer. The overall architecture of AST-GNNFormer is illustrated in Figure 1. The model is designed to enhance the capability of capturing both short-term and long-term dependencies as well as spatiotemporal coupling patterns in time series data.

In terms of architecture, AST-GNNFormer innovatively integrates GNNs with temporal convolutional networks (TCNs [20]). Along the spatial dimension, the model adopts a multi-layer graph attention structure to capture heterogeneous dependencies among nodes. Along the temporal dimension, convolutional operations are introduced to improve sensitivity to long-term trends and periodic fluctuations, thereby enhancing the model's generalization ability in complex spatiotemporal environments. By combining a spatial multi-head attention mechanism with a temporal feature extraction module, the model adaptively captures relationships among different nodes in the graph while simultaneously modeling the evolution.

3.1 Graph Multi-Head Attention Mechanism

To more effectively capture the spatial dependencies and contextual interactions between nodes in the graph, this paper introduces a Graph Multi-Head Attention (GMHA) mechanism to replace traditional graph convolution methods based on fixed adjacency structures. This mechanism learns attention weights between nodes and dynamically adjusts the aggregation ratio of neighboring node information, thereby enhancing the model's capability to represent spatially heterogeneous structures and improving its expressive flexibility. Figure 2a and Figure 2b illustrate the structure of the graph attention layer.

In the single-head graph attention mechanism, the model first applies a linear transformation to the input features of each node and then computes attention coefficients between the node and its neighbors. These coefficients reflect the relative importance of each neighboring node to the target node. Specifically, the attention score between node i and its neighboring node j can be expressed as:

$$e_{ij} = a\left(Wh_i \parallel Wh_j\right) \cdot A \tag{1}$$

where W is the learnable weight matrix, $\parallel$ denotes feature concatenation, $a(\cdot)$ is the feed-forward neural network function, and A is the adjacency matrix of the graph, which controls the scope of attention computation. This score is further normalized via the softmax function to obtain standardized attention weights:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(e_{ij}))}{\sum_{k \in \mathcal{N}_i} \exp(\text{LeakyReLU}(e_{ik}))} \tag{2}$$

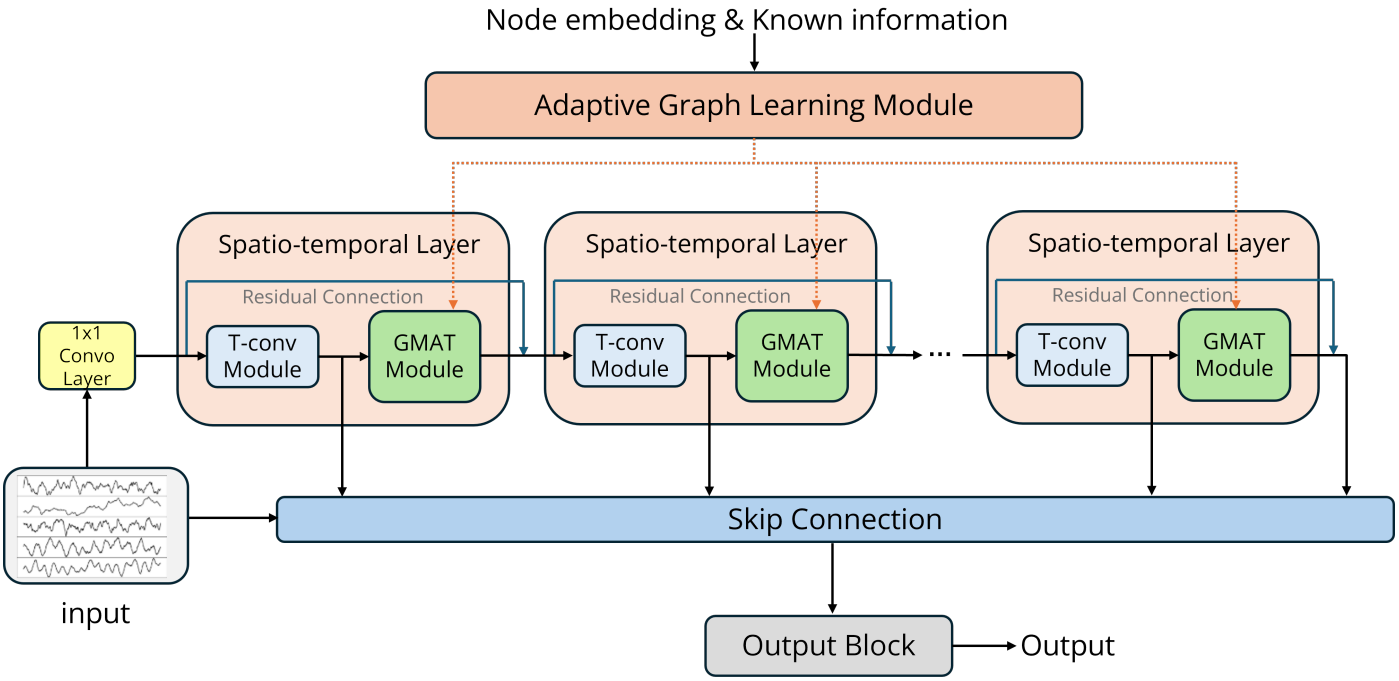


Figure 1. The overall framework of AST-GNNFormer, which integrates an Adaptive Graph Learning Module with multiple spatio-temporal layers, each consisting of temporal convolution (T-conv) and graph multi-attention (GMAT) modules, to capture both spatial and temporal dependencies.

Using the normalized attention weights, the feature representation of node i is updated as the weighted sum of its neighboring nodes' features:

$$h_i' = \sigma\left(\sum_{j \in N_i} \alpha_{ij} W h_j\right) \tag{3}$$

Formally, let $\sigma(\cdot)$ denote a nonlinear activation function, and let $\mathcal{N}_i$ represent the neighborhood of node i.

To enhance both the representational capacity and training stability of the model, we extend the single-head attention mechanism to a multi-head attention framework. Specifically, we employ K independent attention heads operating in parallel, enabling the model to jointly attend to information from different representation subspaces and capture diverse relational patterns. The final node representation is obtained by concatenating the outputs from all attention heads:

$$h_i' = \|_{k=1}^{K} \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij}^k W^k h_j\right) \tag{4}$$

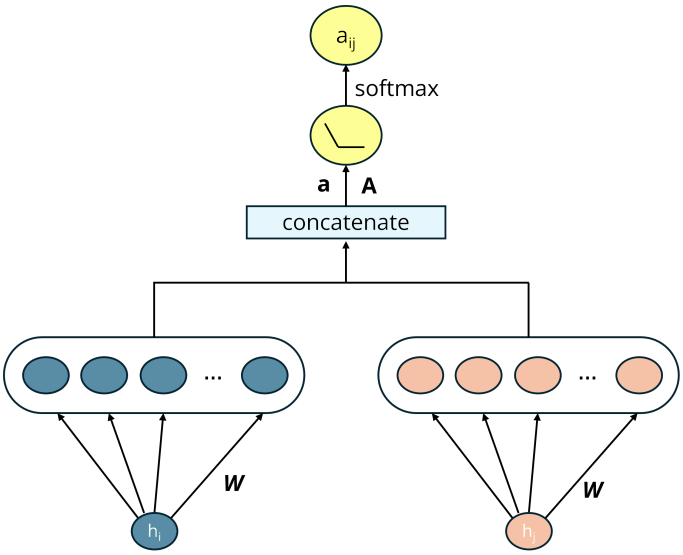
To reduce model complexity and streamline computational processes, particularly during the prediction phase, this study further modifies the output of multi-head attention by replacing concatenation with averaging. This approach yields a more stable and compact representation:

$$h_i' = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j \in \mathcal{N}_i} \alpha_{ij}^k W^k h_j\right) \tag{5}$$

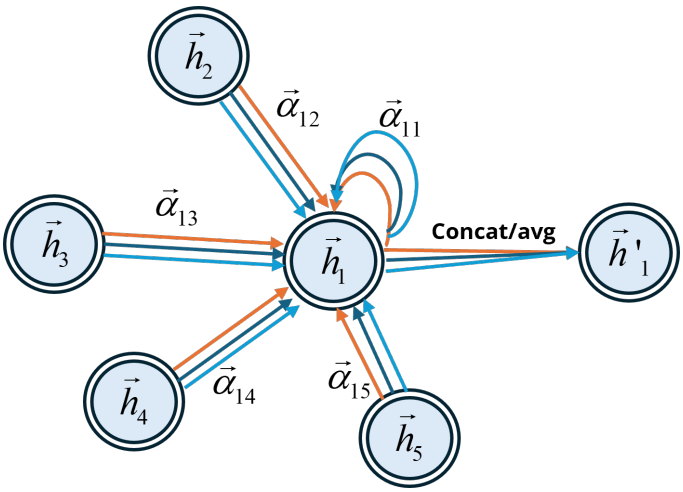
This design not only simplifies the model architecture but also enhances generalization capability and prediction stability, making it particularly suitable for traffic flow forecasting scenarios where sensitivity to mid- and short-term graph structural features is crucial.

3.2 Information Retention Module and Inter-layer Attention Module

Graph Neural Networks (GNNs) effectively capture spatial correlations between nodes by propagating node features and aggregating neighbor information within the graph structure. However, as the number of network layers increases, node representations gradually become more similar during layer-wise propagation, leading to degraded feature representation capability - a phenomenon known as "over-smoothing." Furthermore, information captured at different graph layers exhibits distinct characteristics, and directly stacking them without differential processing may introduce redundant or even invalid information. Therefore, to enhance the



(a)Architecture of the Spatial Multi-Head Attention Aggregation mechanism.



(b)Graph Attention Mechanism for Node-Level Dependency Modeling.

Figure 2. Schematic illustration of the Graph Multi-Head Attention mechanism architecture.

model's robustness and expressive power in deep architectures, this paper introduces two key modules into the graph attention network: the Information Retention Module and the Inter-Layer Attention Module.

3.2.1 *Information Preservation Module*

In conventional graph neural networks, node embeddings undergo iterative updates through multi-layer propagation. While this updating mechanism progressively aggregates neighborhood information, it inevitably dilutes the original node features—particularly in deep networks, where node representations often suffer from vanishing gradients or loss of discriminative power, ultimately degrading model performance.

To mitigate these issues, we propose an Information Preservation Module (as illustrated in Figure 3) that selectively retains partial input features during each propagation layer. The core innovation lies in introducing a controllable "residual connection" during node updates: by adaptively fusing a node's initial features with the current layer's output through learnable weighting, the module achieves dynamic information equilibrium and residual propagation.

Specifically, in the S-th layer of the graph neural network, the feature update for node i is defined as follows:

$$h_i^{(S)'} = \beta h_i^{\text{in}} + (1-\beta) h_i^{(S)} \tag{6}$$

where h_i^{in} denotes the original input feature of node i, and h_i^S represents the output after propagation through the S-th layer. The parameter $\beta \in [0,1]$ is a hyperparameter that controls the proportion of the original information to be retained. A larger β helps preserve the individual characteristics of the node and mitigates over-smoothing, while a smaller β allows more information from neighboring nodes to be integrated, enhancing contextual fusion.

The introduction of this module offers three key advantages: (1) Enhanced Feature Stability, which ensures the preservation of original structural features even in deep networks, thereby mitigating information degradation risks; (2) Improved Gradient Flow, facilitated by a residual structure that optimizes gradient propagation efficiency during training; and (3) Balanced Local-Global Information Fusion, enabling the simultaneous retention of node-specific features and structural context, which significantly benefits downstream tasks such as classification and regression.

3.2.2 *Inter-Layer Attention Module*

Although deep graph networks can extract multi-level spatial information, there exist significant differences in feature representations across layers: shallow layers focus more on local neighborhoods, while deeper layers integrate more global information from the entire graph. If features from different layers are simply concatenated in the model, this may introduce excessive noise and redundant information, potentially hindering the learning of critical node features. To address this issue, we propose an attention-based inter-layer feature selection module, namely the **Inter-Layer Attention Module**.

The core objective of this module is to leverage an

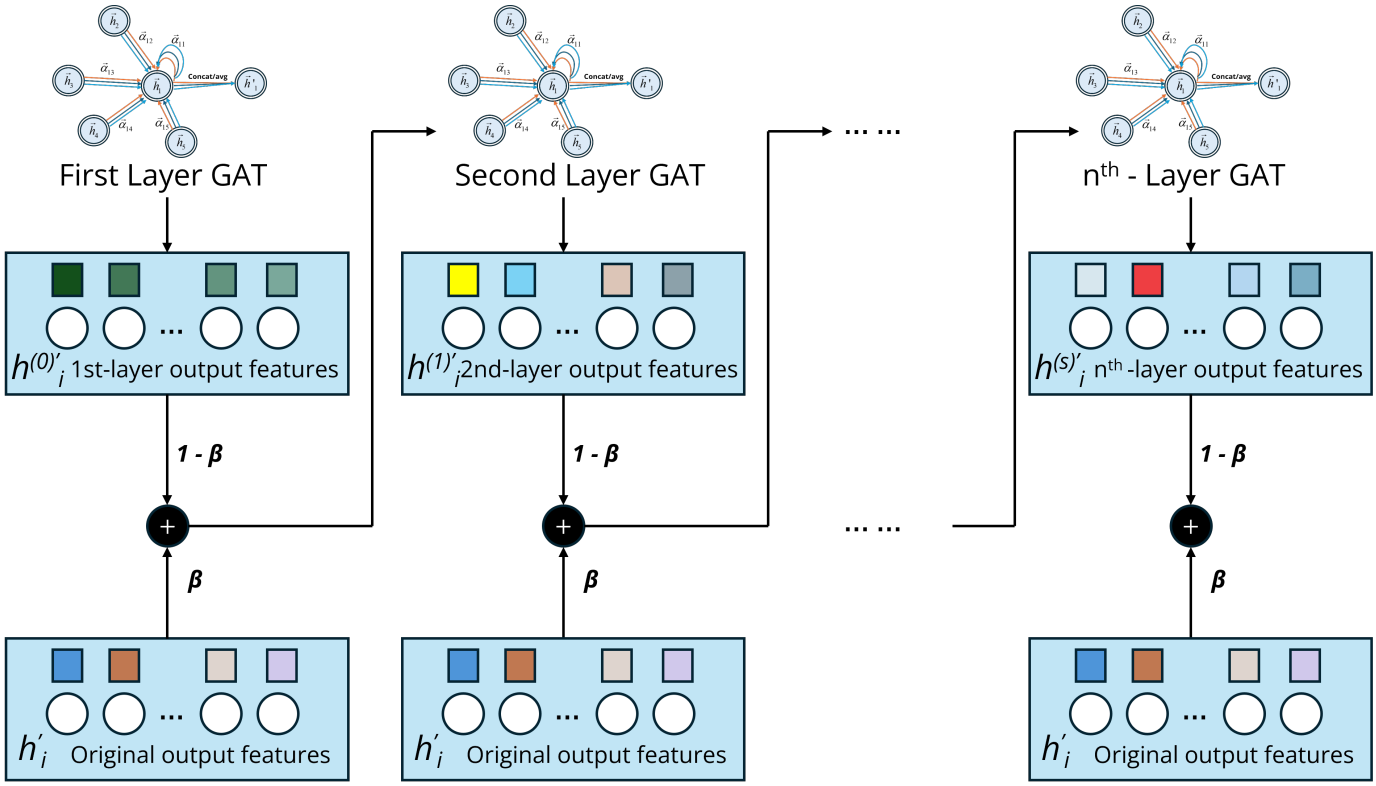


Figure 3. Illustration of the information retention and residual propagation mechanism in Multi-layer GAT.

attention mechanism to dynamically identify the most task-relevant features from multiple GAT layers and assign higher weights accordingly, thereby improving the quality and efficiency of multi-layer feature fusion. Specifically, the representations of each node from different layers are concatenated and integrated to form a cross-layer node representation:

$$h_i' = \|_{s=1}^{s} \sigma\left(h_i^{(s)}\right) \tag{7}$$

where S denotes the total number of layers, $h_i^{(s)}$ the embedding of node i at the s-th layer, $\sigma(\cdot)$ represents an activation function, and $\|$ denotes concatenation. This representation retains the structural features from each layer but does not differentiate their relative importance.

To select effective information from these multi-layer features, an inter-layer attention mechanism is introduced. The attention coefficients are computed based on the similarity between features across different layers:

$$C_{ij} = (Wh_i)(Wh_j)^T A \tag{8}$$

where W is a learnable weight matrix, and A is the adjacency matrix used to constrain the computation range. To enhance discriminability, we apply a softmax normalization to the coefficients:

$$C_{ij} = \frac{\exp(\text{LeakyReLU}(C_{ij}))}{\sum_{k\in\mathcal{N}_i} \exp(\text{LeakyReLU}(C_{ik}))} \tag{9}$$

Finally, the fused features are aggregated using the attention weights and passed through a linear transformation to obtain the final node representation:

$$h_i^{\text{out}} = \sigma(C_{ij} W h_j) \tag{10}$$

By incorporating this module, the model can explicitly model inter-layer dependencies during fusion and adaptively assign different attention weights to different layers, thereby avoiding the feature contamination caused by naive equal-weight concatenation.

3.3 Adaptive Graph Learning Module

The construction of graph structures plays a crucial role in Graph Neural Networks (GNNs). Traditional methods typically rely on predefined adjacency matrices, which are often based on spatial distance, static correlations, or domain knowledge. However, such static graph structures often fail to accurately capture the potential dependencies or complex semantic relationships between nodes, especially in scenarios with heterogeneity or dynamic changes.

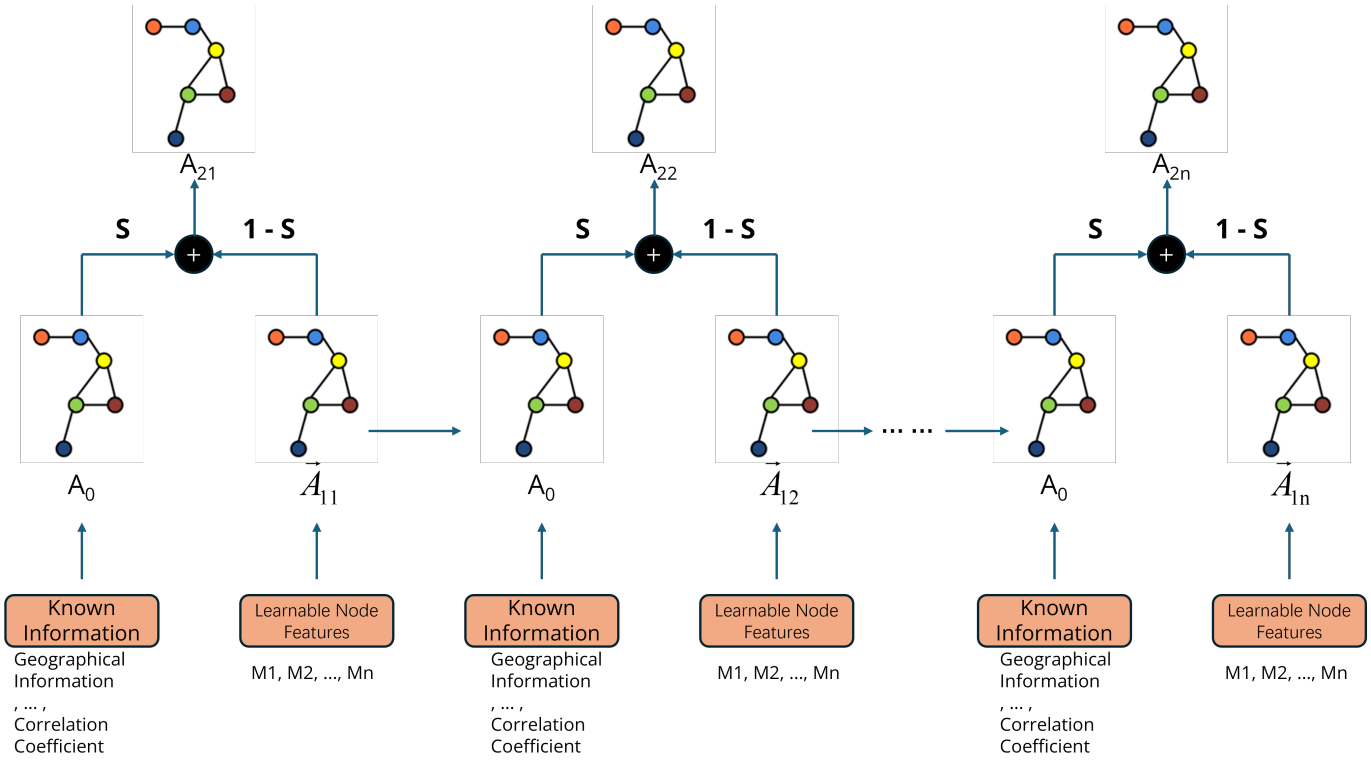


Figure 4. Adaptive Graph Learning Module for dynamic topology construction.

To enhance the model's capability of constructing graph structures, we propose an Adaptive Graph Learning Module (as illustrated in Figure 4), which integrates explicit structural information with learnable feature relationships, enabling the construction of graph topologies with stronger representational power and generalization ability.

This module mainly consists of three substructures, each modeling the graph structure from a different perspective.

3.3.1 *Graph Structure Construction Based on Prior Knowledge*

First, we consider prior spatial location information, topological structures, or other domain-specific knowledge available in real-world scenarios to construct a predefined graph structure as the initial adjacency matrix A_0. This structure can be derived from distance or correlation functions and then normalized or transformed. A typical construction approach uses a Gaussian kernel function based on Euclidean distance:

$$A_0(i,j) = \begin{cases} \exp\left(\frac{-D_{ij}^{\text{geo}}}{\sigma^2}\right), & D_{ij}^{\text{geo}} \geq \varepsilon \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

where D_{ij}^{geo} is the spatial distance between i and j, σ is a scaling hyperparameter, and ε is the distance threshold.

In addition to distance-based information, we can also use correlation metrics between node attributes, such as Pearson correlation or covariance, to build a graph structure based on attribute similarity:

$$A_0(i,j) = \begin{cases} \frac{\text{Cov}(i,j)}{\sqrt{D(i)D(j)}}, & \frac{\text{Cov}(i,j)}{\sqrt{D(i)D(j)}} \geq \nu \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

This approach is particularly suitable for scenarios with clear physical or spatial priors and serves as an initialization guide for subsequent adaptive graph learning.

3.3.2 *Graph Structure Construction Based on Trainable Node Embeddings*

Considering the lack of reliable prior information in some applications, this paper further proposes a method for constructing graph structures based on node embedding features. This method inputs node embeddings M_1 and M_2 into a neural network to learn structural correlations between nodes. The typical structures are as follows:

$$\overline{A}_n = \text{Softmax}\left(\text{ReLU}(M_1 M_2^{\text{T}}) + \text{Diag}(\mathcal{E})\right) \tag{13}$$

or

$$\overline{A}_n = \text{Softmax}\left(\text{ReLU}(M_1 M_2^{\text{T}} - M_2 M_1^{\text{T}} + \text{Diag}(\mathcal{E}))\right) \tag{14}$$

where M_1, M_2 are node embeddings and E is the identity matrix. By performing nonlinear normalization on the structural similarity, data-driven neighborhood relationships can be learned.

3.3.3 *Graph Structure Construction by Fusing Prior Knowledge and Embedding Information*

To simultaneously leverage explicit prior structural information and implicit feature relationships learned from data, this paper proposes a fused graph learning mechanism. This mechanism introduces a gating strategy to balance the predefined graph A_0 and the self-learned graph $\overline{A_n}$, forming the final fused graph structure A_n:

$$S = \text{Sigmoid}\left(h(A_0, \overline{A}_n)\right) \tag{15}$$

$$A_n = S \otimes A_0 + (1 - S) \otimes \overline{A}_n \tag{16}$$

where λ is a selection weight learned via lightweight convolutional networks or attention mechanisms to control the contribution ratio between the two.

To enhance model robustness, we further introduce sparsity regularization and weighted normalization mechanisms into A_n, enhancing the graph's stability under dynamic scenarios. The final structure between nodes is calculated as:

$$D = \sum_{j=1}^{N} A_n^{(i,j)} \quad A_{sn} = \text{ReLU}\left(D^{-\frac{1}{2}} A_n D^{-\frac{1}{2}} - \boldsymbol{\varepsilon}\right) \quad A_f = D^{\frac{1}{2}} A_{sn} D^{\frac{1}{2}} \tag{17}$$

This design ensures unified and regularized processing of the graph structure, maintaining its expressive power while suppressing redundancy. It improves the adaptability to both local and global structural contexts. In summary, the self-supervised graph learning module effectively fuses static prior and dynamic features, providing a flexible, expressive, and robust structure learning framework for graph neural networks.

3.4 Temporal Convolution Module

To fully capture temporal evolution patterns in sequential data such as traffic, this paper introduces a temporal convolutional network (TCN [20]) module. As a key component in the overall architecture, the TCN module effectively captures both short- and long-term temporal dependencies to improve forecasting accuracy.

Originally proposed by Bai et al. [20], TCN leverages a combination of causal convolution and dilated convolution to construct deep temporal feature extractors. Unlike RNN-based models, TCNs avoid issues such as gradient vanishing and sequential dependency, allowing for efficient parallel processing and improved stability.

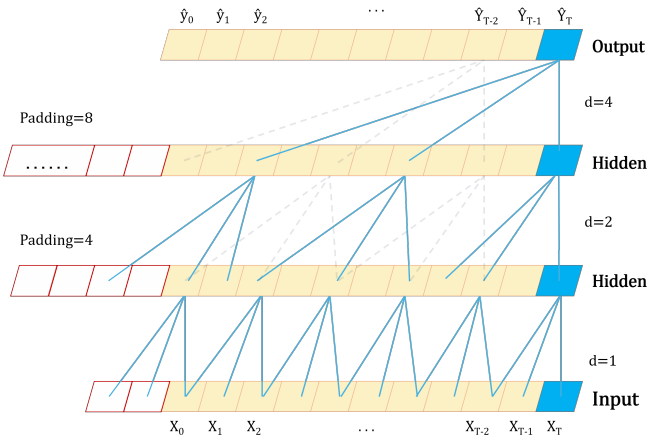


Figure 5. Schematic illustration of the Temporal Convolutional Network architecture.

As illustrated in Figure 5, the temporal convolution process begins with a one-dimensional input sequence, which is passed through multiple stacked 1D convolutional layers (as shown in the Figure 6). Each layer consists of several channels and is capable of capturing temporal dependencies at different scales by adjusting the kernel size and stride. To further extend the receptive field without increasing parameter count, dilated convolution is employed—sampling the input in a skipping manner. This enables the model to capture long-range patterns efficiently across time steps.

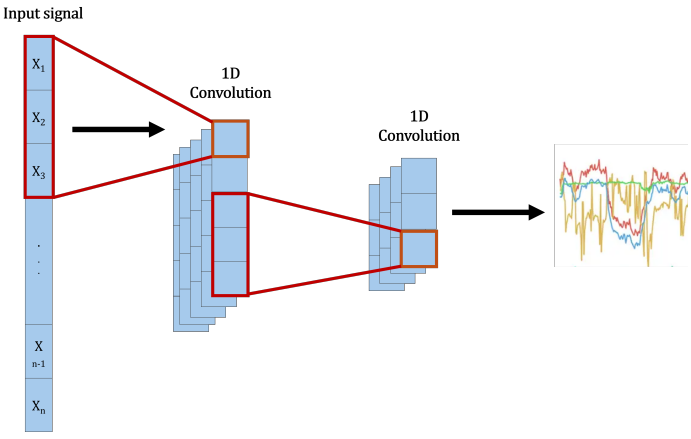


Figure 6. Illustration of the 1D temporal convolution process used in the TCN [20] module.

The standard convolution operation suffers from a fixed receptive field, making it difficult to capture long-range temporal dependencies. To address this issue, this paper introduces a dilated convolution mechanism. Dilated convolutions sample the

input sequence in a jump-like manner, significantly expanding the convolution kernel's receptive field without explicitly increasing the number of model parameters. The dilation rates are set to grow exponentially, such as $1, 3, 6, 7$ allowing the model to capture multi-scale temporal information from short-term to long-term dependencies at different layers.

In addition, to enhance the model's ability to integrate multi-scale temporal features, multiple convolution paths with different dilation rates are executed in parallel, and a gating mechanism is used to control the fusion of information across these paths. The gating mechanism consists of two parallel TCN layers with different activation functions (tanh and sigmoid), which together form a gate unit (see Figure 7). The final output is obtained by combining through point-wise multiplication, enhancing the model's selective capability for important temporal features.

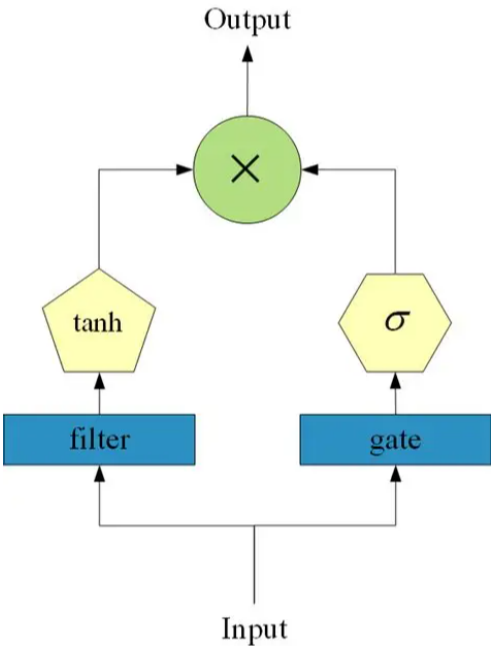


Figure 7. Gated Fusion Unit.

For dilated convolution, the effective receptive field R is related to the kernel size k, the number of layers q, and the dilation rate α. The calculation formula is:

$$R = l + \frac{(c-1)q(k-1)}{(q-1)} \tag{18}$$

This design enables the model to reasonably configure the receptive field size even under shallow structures, thereby improving its ability to capture long-range temporal dependencies.

By combining the TCN module with the GAT module, the model not only captures structural dependencies in the spatial dimension but also models multi-scale evolution patterns in the temporal dimension. The combination of dilated convolution and gating mechanism further enhances the model's ability to selectively represent temporal features with different granularities, enabling the model to perform more stably and predictively in tasks involving long sequences, instability, or localized abrupt changes.

4 Experiments

4.1 Dataset

This paper conducts experimental evaluations using commonly used traffic sensor datasets, specifically **PEMS04** and **PEMS08**. These two datasets originate from real highway traffic monitoring data provided by the California Department of Transportation (Caltrans) and were published by the UC Irvine (University of California, Irvine) Transportation Data Research Center. They are widely used to validate the performance and generalization capabilities of graph neural network-based traffic prediction models.

PEMS04

The PEMS04 dataset was collected from District 4 of the California highway network, covering 307 traffic monitoring stations (nodes). The data spans from January 2018 to February 2018, with each node collecting traffic information every 5 minutes, including average speed, flow, and occupancy rate. The final time series data is structured as a 3D tensor $\mathbb{R}^{T\times N\times C}$, where T denotes the number of time steps, $N = 307$ represents the number of nodes, and $C = 3$ indicates traffic state variables (such as average speed, flow, and occupancy rate).

PEMS08

The PEMS08 dataset is larger in scale, covering 170 monitoring stations, and spans a longer time range, from July 2016 to August 2016. The data is also sampled every 5 minutes to build a time series with a total duration of approximately 2 months. Like the previous dataset, PEMS08 uses traffic state values as prediction targets, which are crucial for evaluating the model's performance under different spatial structures and data sparsity scenarios. Detailed information about the datasets is summarized in Table 1.

4.2 Evaluation Metrics

In this study, to comprehensively evaluate the prediction performance of the model, we selected three commonly used error evaluation metrics: Mean

Table 1. Detail information of PEMS04 & PEMS08.

Dataset	Num of samples	Num of nodes	Num of features	Sampling frequency	Input length	Output length
PEMS04	16992	307	3	5mins	12	12
PEMS08	17856	170	3	5mins	12	12

Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE). The formulas for these three metrics are as follows:

1. Mean Absolute Error (MAE):

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{19}$$

MAE measures the average absolute difference between the predicted value and the actual value, regardless of direction. A smaller value indicates better model performance. MAE is suitable for scenarios where all errors are equally weighted, and it shares the same unit as the original data.

2. Root Mean Square Error (RMSE):

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{20}$$

RMSE computes the square of the errors and then takes the square root of their average. It amplifies the impact of larger errors and is highly sensitive to outliers. It is suitable for scenarios where penalizing large errors is important. A lower RMSE indicates higher model prediction accuracy.

3. Mean Absolute Percentage Error (MAPE):

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100 \tag{21}$$

MAPE expresses the prediction error as a percentage of the actual value. A lower MAPE indicates that the prediction is closer to the actual value. Its advantage lies in the comparability across different datasets and domains. However, when the actual value approaches zero, MAPE may produce large errors.

4.3 Baseline

1. HA [21] (Historical Average)

   HA is the most basic forecasting method. Its principle is to directly use the average value of historical data at the same time point to predict the target time point. Although this method does not possess learning ability and cannot capture the dynamic changes in data, it still provides certain reference value in traffic data with clear periodicity and regularity, serving as a basic baseline model.

2. ARIMA [22] (AutoRegressive Integrated Moving Average)

   ARIMA is a classical statistical model for time series forecasting, combining autoregressive, integrated, and moving average components. It excels at capturing linear trends and seasonality in univariate data but struggles with multivariate spatial dependencies in traffic networks.

3. LSTM [3] (Long Short-Term Memory)

   LSTM [3] is a variant of recurrent neural networks (RNNs) designed to capture long-term dependencies in sequential data by incorporating memory cells and gating mechanisms. It effectively addresses the vanishing gradient problem in traditional RNNs. LSTM [3] is widely used for time series forecasting but lacks the capability to model spatial correlations between traffic nodes.

4. GCN [23] (Graph Convolutional Network)

   GCN [23] is a foundational graph neural network that performs convolutional operations on graph-structured data by aggregating node and neighborhood features. While powerful for static graph representation learning,GCN [23] lacks temporal modeling components, making it less suitable for dynamic spatiotemporal forecasting tasks.

5. STGCN [7] (Spatio-Temporal Graph Convolutional Network)

   STGCN [7] is one of the first deep learning models to jointly consider spatial and temporal dependencies in traffic networks. It integrates spatial graph convolution with temporal convolution, enabling effective modeling of spatial relationships between road segments and time-series dynamics. It is often used as a strong baseline in spatiotemporal prediction.

6. DCRNN [6] (Diffusion Convolutional Recurrent Neural Network)

   DCRNN [6] introduces diffusion convolution to model spatial dependencies in road networks and uses a GRU-based encoder-decoder structure to model temporal dynamics. This model captures both spatial and temporal correlations effectively and is widely applied in large-scale traffic forecasting tasks.

7. ASTGCN [9] (Attention-Based Spatio-Temporal Graph Convolutional Network)

   ASTGCN [9] builds upon STGCN [7] by further introducing attention mechanisms in both spatial and temporal dimensions, allowing the model to assign different weights to nodes and time steps. This helps emphasize critical structures and stable time dependencies while suppressing less important ones, thus improving model prediction accuracy and robustness.

8. AGCRN [13] (Adaptive Graph Convolutional Recurrent Network)

   AGCRN [13] addresses the challenge of acquiring fixed adjacency matrices in real-world traffic networks by learning dynamic node-specific graph structures through adaptive learning. It combines GRU for temporal modeling and dynamically updates spatial structures with time. AGCRN [13] allows the model to adapt to both spatial and temporal structural variations, and is a major representative of dynamic graph-based prediction models.

9. GMAN [10] (Graph Multi-Attention Network)

   GMAN [10] utilizes multiple attention mechanisms for dynamic spatiotemporal modeling. It employs spatial and temporal attention modules within an encoder-decoder framework, dynamically learning graph structures and time dependencies. GMAN [10] demonstrates strong performance across various traffic datasets.

10. MTGNN[12] (Multivariate Time Series Graph Neural Network)

    MTGNN [12] combines graph convolution with Causal Convolution and automatically learns a graph structure to capture hidden dependencies among multivariate time series. Unlike models that rely on real-world geographic positions, MTGNN [12] learns purely from data-driven correlations, enabling good generalization and flexibility.

11. STFGNN [24] (Spatiotemporal Fusion Graph Neural Network)

    STFGNN [24] fuses spatial graph convolutions and temporal filter modules and extracts features through multiple channels. It excels in modeling hierarchical spatial structures and selective long-term temporal dependencies. As a result, it is a strong representative among recent advanced spatiotemporal prediction models in the field of traffic forecasting.

4.4 Hyperparameter Optimization Strategy

To enhance the model's performance, this study To improve model performance, this study employs Bayesian Hyperparameter Optimization to fine-tune key hyperparameters within the model. This method, grounded in probabilistic inference theory, establishes a mapping between hyperparameters and the target objective function. By approximating the objective function distribution using a surrogate model, it guides the search process efficiently.

At each iteration, an acquisition function is utilized to strategically select the next sampling point, aiming to identify the near-optimal solution with relatively few function evaluations. Compared with traditional methods such as grid search and random search, Bayesian optimization offers clear advantages in terms of both search efficiency and result quality.

The hyperparameters optimized in this study include: batch size, dilation factor, dropout rate, number of training epochs, number of GMAT layers, number of ST layers, learning rate, number of attention heads, and the information retention coefficient. The final hyperparameter configuration adopted in subsequent experiments corresponds to the best result obtained through the optimization process, as shown in Table 2.

4.5 Performance Comparison

To comprehensively evaluate the practical value and generalization ability of the AST-GNNFormer model in traffic forecasting tasks, we conduct systematic experiments on two real-world urban traffic volume datasets—PEMS04 and PEMS08. This section verifies AST-GNNFormer's performance in short-term, mid-term, and long-term prediction scenarios, and compares it against mainstream spatiotemporal

Table 2. Optimized Hyperparameters from Bayesian Optimization.

Hyperparameter	Optimized Value
Batch size	32
Dilation factor size	2
Dropout ratio of neurons	0.4
Number of training epochs	180
Number of GAT layers	8
Number of spatiotemporal layers	3
Learning rate	0.001
Number of attention heads	2
Original information retention ratio	0.05

modeling methods to validate the effectiveness of the proposed model.

All models are trained under the same data splits and hardware environment with consistent training parameters (such as batch size, learning rate, optimizer, etc.). We use a sliding window mechanism to construct input sequences, and each model is trained 5 times to reduce randomness. The final results are reported as the average of the runs.

In addition, to better simulate real-world traffic forecasting application requirements, we set prediction lengths of 15, 30, and 60 minutes to evaluate the model's ability to handle both short-term and long-term dependencies.

The comparison covers traditional statistical methods, sequence-based models, graph-based neural networks, and various attention-based hybrid structures. Specifically:

- HA [21] (Historical Average) and ARIMA [22]: Traditional statistical models serving as baseline references;
- LSTM [3]: A standard deep sequence model;
- GCN [23], STGCN [7], DCRNN [6]: Graph-based neural network models for traffic forecasting;
- ASTGCN [9], GMAN [10], AGCRN [13], MTGNN [12], STFGNN [24]: Recently proposed high-performance deep learning models integrating attention mechanisms, gating structures, graph topology learning, etc.;
- AST-GNNFormer (ours): The proposed model integrates multi-head temporal-spatial attention and gated skip connection mechanisms for spatiotemporal modeling.

These models represent the mainstream research directions and modeling paradigms in the field, showcasing strong representativeness and comparative value.

The following table reports the MAE, RMSE, and MAPE metrics of each model on the PEMS-04 dataset under 15/30/60-minute prediction scenarios.

From Table 3, we can find that:

1. Overall Performance Superiority:

   AST-GNNFormer achieves the best results across all prediction time points and evaluation metrics. Especially under long-term forecasting scenarios (60 minutes), it reaches an RMSE of 41.92, significantly outperforming the current best model STFGNN [24] (RMSE of 43.95), demonstrating its stronger capability in capturing long-range spatiotemporal dependencies.

2. Significant Improvement over Dynamic Models:

   Although models such as GMAN [10] and MTGNN [12] already consider spatial attention and structural adaptiveness, AST-GNNFormer further enhances prediction accuracy in complex networks by simultaneously constructing multi-head spatio-temporal attention and combining it with a gating mechanism to optimize information transmission paths.

3. Robust and Stable Performance:

   As indicated by the MAPE metric, AST-GNNFormer maintains stable prediction accuracy even under scenarios with sharply fluctuating traffic volumes or road segments with abrupt changes. This is of practical value for traffic management during urban peak hours.

4. Balanced Short- and Long-term Performance:

   Traditional models (e.g., DCRNN [6]) often perform better in the short term but exhibit notable degradation over longer horizons. AST-GNNFormer, on the other hand, performs consistently well across all time intervals, indicating that its architecture supports robust modeling across multiple temporal scales.

4.6 Ablation Study

To further validate the contribution of each core module in the AST-GNNFormer model to its overall performance, we designed a series of ablation experiments. Specifically, we selectively removed or replaced key components in the model and conducted

Table 3. Performance Comparison of Different Models on PEMS-04 Dataset (15/30/60-minute Forecasting).

Model	MAE (15/30/60 min)	RMSE (15/30/60 min)	MAPE (15/30/60 min)
HA [21]	38.12 / 41.92 / 45.87	58.22 / 63.51 / 68.91	27.63% / 30.71% / 34.89%
ARIMA [22]	34.78 / 38.93 / 43.27	52.64 / 57.31 / 62.84	23.54% / 26.88% / 30.62%
LSTM [3]	28.41 / 31.25 / 34.84	44.21 / 48.36 / 52.79	18.02% / 20.64% / 23.97%
GCN [23]	26.97 / 29.92 / 33.45	42.36 / 46.15 / 50.41	16.58% / 19.21% / 22.45%
STGCN [7]	25.68 / 28.43 / 31.76	40.94 / 44.52 / 48.87	15.74% / 17.83% / 20.39%
DCRNN [6]	24.91 / 27.65 / 31.02	39.85 / 43.62 / 47.91	14.93% / 17.05% / 19.64%
ASTGCN [9]	23.74 / 26.54 / 29.61	38.67 / 41.53 / 45.38	14.31% / 16.22% / 18.88%
GMAN [10]	23.08 / 25.96 / 29.12	37.92 / 40.78 / 44.92	13.76% / 15.63% / 18.25%
AGCRN [13]	22.87 / 25.70 / 28.93	37.56 / 40.43 / 44.62	13.58% / 15.45% / 18.06%
MTGNN [12]	22.63 / 25.43 / 28.62	37.33 / 40.21 / 44.27	13.42% / 15.28% / 17.88%
STFGNN [24]	22.41 / 25.17 / 28.35	37.01 / 39.94 / 43.95	13.26% / 15.14% / 17.62%
AST-GNNFormer (ours)	21.68 / 24.39 / 27.51	33.97 / 37.84 / 41.92	12.83% / 14.63% / 16.97%

Table 4. Ablation Study Results on PEMS04 and PEMS08 Datasets for 15-Minute Forecasting.

Model Variant	PEMS04 MAE	PEMS04 RMSE	PEMS08 MAE	PEMS08 RMSE
AST-GNNFormer (Full)	21.68	33.97	18.15	29.31
w/o LIP	22.41	35.12	18.74	30.19
w/o ILA	22.79	35.88	19.05	30.74
w/o AGL	23.14	36.72	19.62	31.45

comparative analyses on both PEMS04 and PEMS08 datasets to assess how different modules impact the model's predictive performance.

We focus on three key modules: (1) Layer-aware Information Preservation (LIP), which mitigates over-smoothing; (2) Inter-Layer Attention (ILA), for dynamic layer fusion; and (3) Adaptive Graph Learning (AGL), for dynamic topology. The ablation variants are:

1. w/o LIP: Removes the information preservation module;
2. w/o ILA: Removes the inter-layer attention module;
3. w/o AGL: Removes the adaptive graph learning module;
4. AST-GNNFormer (Full Model): Includes all modules.

The results shown in Table 4 reveal the critical importance of each module in the model's architecture. Removing the AGL module causes the most severe performance degradation, with the PEMS04 dataset showing an MAE increase of approximately 1.46. This demonstrates that traffic flow prediction heavily relies on capturing spatial dependencies, and failing to model these relationships significantly impairs the model's predictive ability. Similarly, the inter-layer attention (ILA) module proves essential, particularly in datasets like PEMS08 where multi-layer fusion is crucial. Its absence weakens the model's capacity to integrate features across layers, leading to notably higher prediction errors and underscoring the value of dynamic layer-wise processing. While the layer-aware information preservation (LIP) module's removal results in comparatively milder performance drops, its contribution remains vital. Though not a primary spatial extractor, LIP enhances training stability by addressing over-smoothing issues, thereby improving both training controllability and the model's eventual generalization performance. Together, these findings emphasize how each component—whether directly responsible for feature extraction or supporting training dynamics—plays an indispensable role in the model's overall effectiveness.

5 Conclusion

This paper addresses the problem of time series prediction in complex spatial structures and proposes a deep prediction model called AST-GNNFormer, which is based on adaptive spatial multi-attention graph networks. The motivation behind this work is to tackle issues in traditional GNN-based

spatial modeling, such as inaccurate spatial structure modeling, over-smoothing, and information loss in deep stacking, and insufficient temporal modeling capabilities. The goal is to improve both the predictive performance and the generalization ability of the model.

To this end, we design the Graph Multi-Attention Network (GMAT), which incorporates information retention modules and spatial attention modules across layers. This effectively alleviates the information over-smoothing and loss problems commonly seen in traditional deep neural networks, thereby enhancing the model's expressive capacity for spatial structures.

The proposed model makes significant contributions through three pivotal innovations. First, an adaptive graph structure learning mechanism dynamically captures spatial dependencies by synergizing predefined topological knowledge with data-driven node relationships, overcoming limitations of static graph constructions. Second, the integration of temporal convolution modules enables precise modeling of nonlinear temporal variations, forming a core component of the AST-GNNFormer framework's time-aware processing capability. These technical advances are rigorously validated through comprehensive experiments on real-world traffic and meteorological datasets, where the model consistently demonstrates performance advantages across various metrics. Notably, systematic ablation studies provide empirical evidence for each component's contribution, confirming the graph learning module's effectiveness in spatial modeling, the temporal modules' role in sequence processing, and the overall architecture's design coherence. The combined results establish a new state-of-the-art in spatio-temporal forecasting while offering practical insights for intelligent transportation and environmental monitoring applications.

Data Availability Statement

Data will be made available on request.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant 62173007, Grant 62203020, Grant 62473008, Grant 62433002, and Grant 62476014; in part by the Beijing Nova Program under Grant 20240484710; in part by the Project of Humanities and Social Sciences (Ministry of Education in China, MOC) under Grant 22YJCZH006; in part by the Beijing Scholars Program under Grant 099; in part by the Project of ALL China Federation of Supply and Marketing Cooperatives under Grant 202407; in part by the Project of Beijing Municipal University Teacher Team Construction Support Plan under Grant BPHR20220104.

Conflicts of Interest

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Lv, Y., Duan, Y., Kang, W., Li, Z., & Wang, F. Y. (2014). Traffic flow prediction with big data: A deep learning approach. *IEEE transactions on intelligent transportation systems, 16*(2), 865-873. [Crossref]

[2] Tedjopurnomo, D. A., Bao, Z., Zheng, B., Choudhury, F. M., & Qin, A. K. (2020). A survey on modern deep neural network for traffic prediction: Trends, methods and challenges. *IEEE Transactions on Knowledge and Data Engineering, 34*(4), 1544-1561. [Crossref]

[3] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation, 9*(8), 1735-1780. [Crossref]

[4] Ma, X., Tao, Z., Wang, Y., Yu, H., & Wang, Y. (2015). Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transportation Research Part C: Emerging Technologies, 54*, 187-197. [Crossref]

[5] Ke, J., Zheng, H., Yang, H., & Chen, X. M. (2017). Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach. *Transportation research part C: Emerging technologies, 85*, 591-608. [Crossref]

[6] Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*.

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

[8] Yan, S., Xiong, Y., & Lin, D. (2018, April). Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 32, No. 1). [Crossref]

[9] 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). [Crossref]

[10] 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). [Crossref]

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

[12] Wu, Z., Pan, S., Long, G., Jiang, J., Chang, X., & Zhang, C. (2020, August). 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* (pp. 753-763). [Crossref]

[13] Bai, L., Yao, L., Li, C., Wang, X., & Wang, C. (2020). Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in neural information processing systems, 33*, 17804-17815.

[14] Zheng, Y., Yi, L., & Wei, Z. (2025). A survey of dynamic graph neural networks. *Frontiers of Computer Science, 19*(6), 196323. [Crossref]

[15] Dai, B. A., Ye, B. L., & Li, L. (2024). A novel hybrid time-varying graph neural network for traffic flow forecasting. *arXiv preprint arXiv:2401.10155*.

[16] Lu, S., Chen, H., & Teng, Y. (2024). Multi-scale non-local spatio-temporal information fusion networks for multi-step traffic flow forecasting. *ISPRS International Journal of Geo-Information, 13*(3), 71. [Crossref]

[17] Xiong, L., Yuan, X., Hu, Z., Huang, X., & Huang, P. (2024). Gated fusion adaptive graph neural network for urban road traffic flow prediction. *Neural Processing Letters, 56*(1), 9. [Crossref]

[18] Yaqub, M., Ahmad, S., Manan, M. A., Pathan, M. S., & He, L. (2025). Predicting traffic flow with federated learning and graph neural with asynchronous computations network. *Array*, 100411. [Crossref]

[19] Jiang, W., & Luo, J. (2022). Graph neural network for traffic forecasting: A survey. *Expert systems with applications, 207*, 117921. [Crossref]

[20] Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*.

[21] Smith, B. L., Williams, B. M., & Oswald, R. K. (2002). Comparison of parametric and nonparametric models for traffic flow forecasting. *Transportation Research Part C: Emerging Technologies, 10*(4), 303-321. [Crossref]

[22] Williams, B. M., & Hoel, L. A. (2003). Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. *Journal of transportation engineering, 129*(6), 664-672. [Crossref]

[23] Kipf, T. N. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

[24] Li, M., & Zhu, Z. (2021, May). Spatial-temporal fusion graph neural networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 35, No. 5, pp. 4189-4196). [Crossref]

Yusen Zheng graduated from Shenyang Jianzhu University in 2020 with a bachelor's degree in Electrical Engineering. He is currently a master's candidate in Control Engineering at Beijing Technology and Business University. His main research interests include pattern recognition and information fusion, remote sensing image segmentation, deep learning, and related fields. (Email: 2330602081@st.btbu.edu.cn)

Xuebo Jin received the B.S. and M.S. degrees in control theory and control engineering from Jilin University, Changchun, China, in 1994 and 1997, and the Ph.D. degree in control theory and control engineering from the University of Zhejiang, Zhejiang, China, in 2004.,She was a Senior Visiting Scholar with the University of Illinois at Chicago, Chicago, IL, USA, in2007. From 2009 to 2012, she was an Assistant Professor with Zhejiang Sci-tech University. Since 2012, she has been a Professor with Beijing Technology and Business University, Beijing, China. Her research includes a variety of areas in information fusion, big data analysis, condition estimation, and video tracking. (Email: jinxuebo@btbu.edu.cn)