Journal of Highway and Transportation Research and Development (English Edition)
Open Access | Editor-in-Chief: Wanheng Li, Daming Lin | ISSN 2095-6215

J. highw. transp. res. dev.

https://doi.org/10.26599/HTRD.2026.9480119

Article Type: Research Article

DSTT-HF: Dynamic Spatio-Temporal Transformer with Hierarchical Fusion for Traffic Flow Prediction

Jin Hu[1]*, Fumin Zou[1], QiQin Cai[2]**

[1] *Fujian Key Laboratory of Automotive Electronics and Drive Technology, Fujian University of Technology, Fuzhou 350108, China*

[2] *School of Equipment Manufacturing and Vehicle Engineering, Zhangzhou Institute of Technology, Zhangzhou 363000, Fujian, China)*

*Email: 2336282846@qq.com

**Corresponding author: QiQin Cai, cai199402280111@gmail.com

Received：date (29-Jun-2026)；Revised: date (16-Oct-2026)；Accepted：date (26-Oct-2026)

Just Accepted

Abstract: Accurate traffic flow prediction plays a vital supporting role in urban traffic management and emergency response. Existing spatio-temporal prediction models suffer from limited capability in capturing time-varying topological dependencies and fine-grained multi-granularity feature aggregation, which degrades prediction performance especially under unexpected traffic fluctuations. To address these limitations, this paper proposes a Dynamic Spatio-Temporal Transformer with Hierarchical Fusion (DSTT-HF). The model constructs time-varying dynamic adjacency matrices from real-time node features and periodic traffic trends to mitigate the defects of static topology. A temporal attention module incorporating first-order difference features is designed to capture both long-term periodic patterns and

short-term traffic variations. Moreover, a multi-scale hierarchical fusion module with learnable weighted summation is developed to enhance model robustness for multi-step prediction. Experiments conducted on public PEMS04 and PEMS08 datasets show that DSTT-HF achieves consistent performance improvement over several state-of-the-art baselines. Ablation experiments further validate the effectiveness of core components including feature interaction enhancement, dynamic graph generation and multi-scale fusion. Quantitatively, DSTT-HF achieves MAE of 17.80 and 13.57 on PEMS04 and PEMS08 for the one-step horizon, representing reductions of 8.2% and 9.8% over STGCN, respectively, with all metrics reported as mean ± std over 10 independent runs. The proposed DSTT-HF can effectively model complex dynamic spatio-temporal correlations and provides a feasible solution for short-term traffic flow forecasting within a three-step horizon

Keywords: Traffic Engineering; Deep learning; Temporal and spatial characteristics; Graph convolution network; Intelligent transportation system.

Not Accepted

1 Introduction

The development of adaptive spatiotemporal attention mechanisms has undergone a progressive deepening, evolving from foundational theoretical construction to cross-domain applications. Early research traces back to the Transformer model[1], whose self-attention mechanism provided the theoretical basis for spatiotemporal attention, enabling the modeling of long-range dependencies in sequential data for the first time. Feng and Meunier[2] surveyed skeleton-oriented graph neural networks including the ST-GCN architecture for skeleton-based action recognition.

Regarding traffic forecasting, Jiang et al. proposed the Progressive Distillation Transformer (PDFormer)[3], which captures dynamic spatial dependencies through a spatial self-attention module and explicitly models the time delay of traffic information transmission via a delay-aware feature transformation module. They designed geographic and semantic mask matrices to capture short-range and long-range spatial dependencies, respectively. Xu et al.[4] proposed a graph wavelet neural network to enhance multi- scale node feature interactions in graph- structured data, which performs frequency- domain graph transformation to capture spatial correlations. Furthermore, Chen et al.[5] addressed the challenge of target scale variation in remote sensing image change detection through a Pyramid Attention Module (PAM), utilizing a siamese network architecture and Batch-balanced Contrastive Loss (BCL) to significantly enhance feature discriminability in illumination-changing scenarios.

Song et al. proposed the Local Spatio-Temporal Synchronized Graph Convolution Module (STSGCM)[6], which aggregates spatiotemporal information synchronously using a shared weight matrix to construct local spatiotemporal subgraphs (incorporating nodes from adjacent time steps and their spatial neighbors). They utilized static adjacency matrices to define spatial relationships and captured short-term temporal dependencies through time sliding windows. Zhu et al. introduced a spatial attention mechanism in ASTGCN[7]; however, it relied solely on static node characteristics to calculate weights, neglecting dynamic dependencies along the temporal dimension.

To capture dynamic relationships, Y. Xu et al. proposed a General Dynamic Graph Convolution Network (GDGCN)[8] to derive interpretable dynamic spatial relationships among road segments. Nevertheless, the dynamic graph in this work depended merely on the similarity of spatial features, failing to integrate channel statistics (e.g., flow amplitude) and temporal trends (e.g., periodic peaks), leading to lagged responses to emergencies. B. Xu et al. employed graph

wavelet transforms to enhance multi-scale node feature interactions, yet remained limited to the spatial domain without addressing feature coupling in the temporal dimension[9]. Fengning et al.[10] proposed a Multi-Component GCN structure to handle different motion patterns separately; however, component interactions were restricted to simple concatenation, lacking fine-grained feature fusion. Xie et al.[11] implicitly encoded joint and edge-type information within skeleton topology through dynamic semantic graph convolution network, which improves representation capacity for skeleton-based action recognition.

A growing number of studies have attempted to combine dynamic graph structures and Transformer architectures to enhance spatiotemporal modeling capability for traffic prediction. Many representative methods construct time-varying topological graphs via adaptive node embedding and self-attention, effectively capturing evolving spatial correlations hidden in traffic sensor data[12]. To balance topological prior knowledge and global sequence modeling, spatiotemporal graph Transformer frameworks further integrate graph attention with causal temporal self-attention, overcoming the feature decoupling problem of traditional GCN-Temporal hybrid models[13].. Some improved designs introduce adaptive Fourier transformation and long-term temporal modeling modules to fit the non-stationary and periodic characteristics of traffic flow, optimizing the modeling of long-range temporal dependencies[14].

Furthermore, Mu et al.[9] proposed a spatiotemporal fusion traffic flow prediction model based on GCN and CNN, capturing multi-scale spatiotemporal dependencies through a hierarchical modeling architecture. While this model relies on a static graph structure de fined by an adjacency matrix, spatial dependencies in real-world traffic networks often c hange dynamically over time (e.g., due to temporary traffic control or accidents). Conse quently, the lack of exploration into dynamic graph structures limits the model's ability to adapt to the dynamics of real traffic scenes. Recent works have verified that dynami c edge interaction encoding and real-time topology updating mechanisms can significantl y improve model adaptability to variable traffic scenarios[15-16]. Nevertheless, most exis ting dynamic graph Transformer models adopt single-scale feature extraction strategies a nd ignore the hierarchical fusion of multi-granularity spatiotemporal features, resulting in insufficient robustness for multi-step prediction tasks[17-18]. In addition, the imbalance between long-term periodic trends and short-term sudden traffic fluctuations is rarely qu antitatively optimized in attention weighting mechanisms[19].Furthermore, emerging multi modal learning strategies attempt to incorporate external environmental factors to enhanc e traffic feature representation. A recent weather-aware spatio-temporal fusion model ad opts an attention-driven multimodal learning framework to mine complex traffic variatio n patterns, which effectively improves prediction robustness under changing weather con ditions [20]. However, this method still builds prediction based on static road topology and fails to dynamically adjust spatial correlations according to real-time traffic fluctuati ons, limiting its adaptability to sudden traffic changes.

Despite the progress achieved by existing dynamic spatio-temporal graph models and attention-based fusion approaches, most prior works cannot simultaneously adapt graph topology according to real-time traffic variation and fuse multi-scale temporal features. Static or simply updated graph structures are insufficient to represent the time-varying spatial correlations among road nodes, and single-scale feature extraction easily leads to error accumulation in multi-step traffic forecasting. To tackle these challenges, we propose DSTT-HF, which integrates dynamic graph learning and multi-scale hybrid fusion. The dynamic graph module constructs adaptive topology to reflect changing traffic connectivity, while the multi-scale fusion module captures dependencies at different temporal granularities and suppresses the propagation of prediction errors. This is the core motivation for our proposed method.

Just Accepted

First, the perception of dynamic spatial dependencies remains superficial. While recent studies [8, 15-16] have introduced dynamic graph convolutions to replace static adjacency matrices, most mechanisms rely heavily on the similarity of raw spatial features. As identified by Y. Xu et al. [8], such methods often neglect crucial channel statistics (e.g., flow amplitude variations) and latent temporal trends (e.g., periodic peaks). Consequently, these models exhibit a lagged response to sudden events such as traffic accidents or temporary control measures, failing to capture the instantaneous rewiring of road network topologies.

Second, spatiotemporal dependency modeling suffers from feature decoupling and imbalance. Existing frameworks typically process spatial and temporal dependencies in isolation [13]. Although hybrid attention mechanisms [3, 4] have been employed, they often fail to quantitatively balance the trade-off between long-term periodic trends and short-term abrupt fluctuations. As noted in [19], the lack of a unified mechanism to integrate these heterogeneous signals leads to fragmented representations, where critical congestion propagation patterns are diluted by redundant information.

Third, the robustness of multi-step forecasting is compromised by single-scale feature extraction. Most prevailing models [17-18] utilize fixed receptive fields, which struggle to simultaneously characterize localized bursts (short-term) and wave-like propagations (long-term). The simple concatenation of features from different scales, as seen in [10], lacks fine-grained interaction, resulting in error accumulation during long-term inference.

To bridge these gaps, we argue that an effective traffic prediction model must achieve a synergistic integration of dynamic topology awareness, balanced attention allocation, and multi-granularity feature refinement. Therefore, this paper proposes the DSTT-HF (Dynamic Spatio-Temporal Transformer with Hierarchical Fusion) model. Unlike previous works that treat these components as isolated modules, DSTT-HF establishes a tightly coupled pipeline where each innovation addresses the limitations of the others.

Dynamic Graph Structure Generation: To resolve the latency issue in topology updates, we construct time-varying adjacency matrices by fusing real-time node characteristics (e.g., flow amplitude) with periodic trend analysis. This provides an accurate and responsive structural prior for subsequent attention mechanisms.

Spatiotemporal Feature Enhancement via Hybrid Attention: Built upon the dynamic topology, our hybrid attention mechanism addresses the feature decoupling problem. The spatial attention module adaptively models pairwise node correlation by computing cosine similarity of encoded node features, which accurately captures key spatial interaction patterns during traffic propagation. Meanwhile, the temporal attention module integrates first-order difference information to quantitatively balance short-term traffic volatility against long-term temporal periodicity. This collaborative spatiotemporal filtering ensures that the structural information derived from the dynamic graph is refined and selectively emphasized, rather than indiscriminately propagated, yielding more discriminative and traffic-aware feature representations.

Multi-Scale Hierarchical Fusion: To mitigate the single-scale limitation, we design a hierarchical fusion module that interacts with the attention outputs across different granularities (kernels of 1, 3, and 6). By adaptively weighting features specific to varying forecast horizons (5, 10, and 15 minutes), the model enhances robustness against error propagation in long-term predictions.

The primary contributions of this work are summarized as follows:

Mechanism Innovation: We propose a novel dynamic graph generation mechanism that integrates flow amplitude and periodic trends, significantly improving the model's responsiveness to emergent traffic incidents compared to static or feature-similarity-based graphs.

Architectural Synergy: We introduce a collaborative framework where dynamic graph construction guides hybrid attention, and multi-scale fusion refines the output. This design

overcomes the "module stacking" issue prevalent in traditional GCN-Transformer hybrids by ensuring deep interaction between spatial topology and temporal semantics.

2 Experimental

DSTT-HF is based on transformer, combined with dynamic graph constructor and spatio-temporal convolution module to realize multi-scale modeling of spatio-temporal characteristics of traffic flow. The architecture is shown in Figure 1.

.

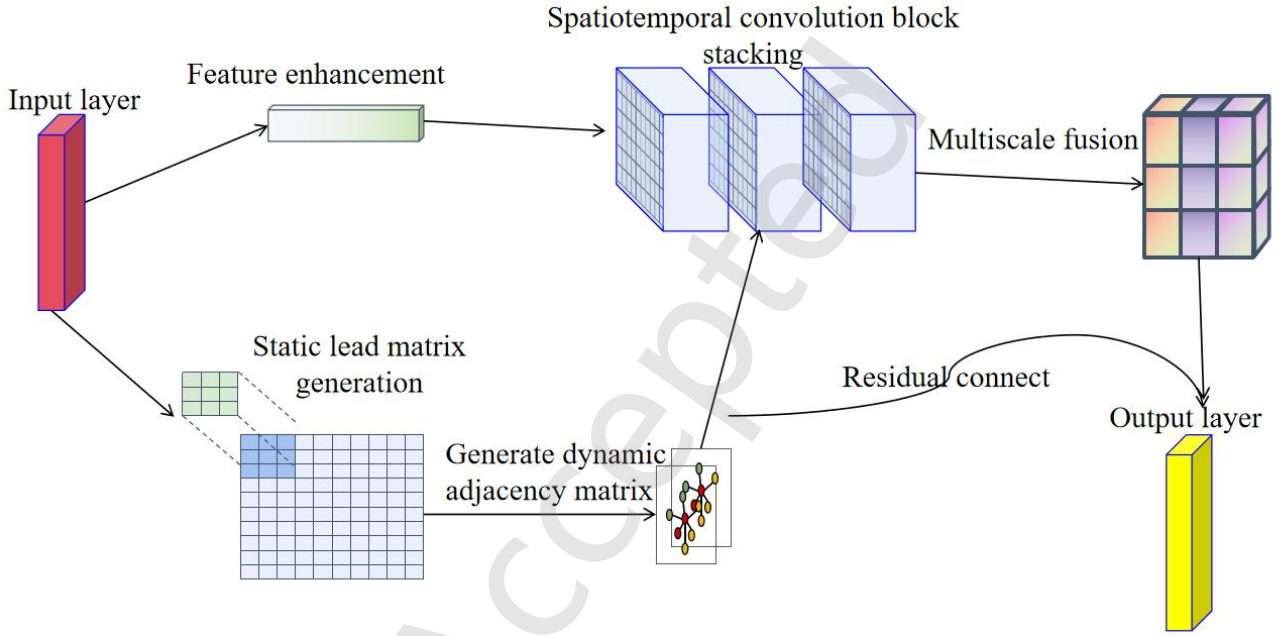


Fig. 1 Overall architecture of DSTT-HF

DSTT-HF is based on Transformer, combined with a dynamic graph constructor and spatio-temporal convolution blocks for multi-scale modeling of traffic spatio-temporal characteristics. The architecture is shown in Figure 1.In the forward pipeline, raw input features are first processed by the feature-interaction enhancement module, where self-attention, gated fusion and a feed-forward network produce refined enhanced features. Node encoding, similarity-matrix calculation and dynamic weighting are conducted on these enhanced features to yield the dynamic adjacency matrix. The static adjacency matrix derived from road-network topology acts as prior knowledge and will be fused with the dynamic adjacency matrix in the adaptive spatial attention module. The enhanced features and fused adjacency matrix pass through three stacked spatio-temporal convolution blocks, where temporal attention calibrates time-series information and spatial attention updates node representations via graph convolution. Residual connections mitigate gradient vanishing and accelerate training. The multi-scale predictor generates multi-scale outputs aggregated by hierarchical weighted summation. Model parameters are optimized by minimizing prediction-ground-truth errors.

Note that the static-matrix branch in Figure 1 only loads fixed road-network topology priors for later fusion. The dynamic adjacency matrix is computed from enhanced features output by the feature-interaction enhancement module rather than raw input features.
Throughout the DSTT-HF architecture, ReLU activation is adopted for non-linear transformation and sigmoid is used for gating-weight generation. Layer normalization is

Just Accepted

applied after each residual branch to stabilize training. The input tensor follows dimension [B, N, T, C] during forward propagation. When computing spatial dynamic adjacency, tensor dimensions are rearranged to [B*T, N, d_model] for node- wise feature encoding. All learnable projection layers map input features to the unified hidden dimension of 64. Residual connections are deployed in both spatial attention blocks and temporal attention blocks to mitigate gradient vanishing.

2.1 Feature Interaction Enhancement Module

As illustrated in the overall model architecture (Figure 2), the feature interaction enhancement module serves as a critical feature refinement and transition component in the DSTT-HF framework. It takes the preliminary spatio-temporal aggregated features as input, and collaboratively completes self-attention feature correlation modeling, gated valid information screening, and feed-forward feature enhancement. This module not only optimizes the representation quality of spatio-temporal features, but also provides high-quality encoded features for the subsequent dynamic graph generator, realizing the joint guidance of dynamic topological structure and adaptive attention feature learning. Given the aggregated spatio-temporal feature output from the stacked spatio-temporal convolution and attention blocks $H_{agg} \in \mathbb{R}^{B\times N\times T\times d_{model}}$, where B denotes batch size, N is the number of road sensor nodes, T represents input time-series length, and d_{model} is the unified hidden dimension of the model, the module first employs self-attention to capture implicit feature correlations across spatial nodes and temporal steps. Based on the correlated feature representation, a gated fusion mechanism is further adopted to screen valid traffic information and suppress redundant noise. Finally, a feed-forward network (FFN) is utilized to realize non-linear feature enhancement and dimensional transformation. First, the self-attention mechanism calculates the global dependency of the input aggregated features to mine potential interaction relationships among traffic features:

$$H_{att} = \text{SelfAttention}\left(H_{agg}\right) \in \mathbb{R}^{B\times N\times T\times d_{model}} \tag{1}$$

where $\text{SelfAttention}(\cdot)$ denotes the standard scaled dot-product self-attention function for feature correlation modeling. It calculates global feature dependencies by computing the similarity between query and key projections of input features, and weights the value features to obtain enhanced representations. The specific calculation is formulated as: $\text{SelfAttention}(H) = \text{Softmax}\left(\frac{HW_q(HW_k)^\top}{\sqrt{d_{model}}}\right)HW_v$, where W_q, W_k, W_v are lightweight learnable projection parameters embedded in the self-attention unit, and $\sqrt{d_{model}}$ is the scaling factor to stabilize gradient propagation. H_{att} is the feature representation with global spatio-temporal correlation captured. To filter invalid feature information and retain key traffic variation characteristics, a learnable gated fusion unit is designed to adaptively weight the original aggregated features and attention-enhanced features. The gating weight is generated through a sigmoid activation function to ensure the weight range within $[0,1]$:

$$G = \sigma\left(H_{agg}W_g + b_g\right) \in \mathbb{R}^{B\times N\times T\times d_{model}} \tag{2}$$

where $G \in \mathbb{R}^{B\times N\times T\times d_{model}}$ denotes the adaptive gating weight matrix, which controls the retention ratio of effective features; $W_g \in \mathbb{R}^{d_{model}\times d_{model}}$ and $b_g \in \mathbb{R}^{d_{model}}$ are learnable weight matrix and bias vector for gating projection, respectively; $\sigma(\cdot)$ represents the sigmoid function. The gated fusion operation fuses the original features and attention-enhanced features element-wise to obtain preliminary optimized features:

$$H_{gate} = G \odot H_{att} + (1 - G) \odot H_{agg} \in \mathbb{R}^{B\times N\times T\times d_{model}} \tag{3}$$

Just Accepted

where $\odot$ denotes the Hadamard product for element-wise feature fusion. On the basis of gated screening, a two-layer feed-forward network is introduced to further mine non-linear traffic feature patterns and complete high-dimensional feature enhancement:

$$H_{out} = \delta\left(H_{gate}W_1 + \mathrm{b}_1\right)W_2 + \mathrm{b}_2 \in \mathbb{R}^{B\times N\times T\times d_{model}} \quad (4)$$

where $\mathrm{W}_1, \mathrm{W}_2$ are learnable projection matrices of the feed-forward network, $\mathrm{b}_1, \mathrm{b}_2$ are corresponding bias terms, and $\delta(\cdot)$ denotes the ReLU non-linear activation function. H_{agg} is the input aggregated spatio-temporal feature from the preceding convolution and attention blocks; $\mathrm{H}_{gate} \in \mathbb{R}^{B\times N\times T\times d_{model}}$ represents the intermediate feature after gated adaptive fusion; $\mathrm{H}_{out} \in \mathbb{R}^{B\times N\times T\times d_{model}}$ is the final enhanced feature output of the interaction enhancement module. In addition, B refers to batch size, N the number of road sensor nodes, T denotes input time-step length, and d_{model} represents the unified hidden feature dimension of the model. The output enhanced feature H_{out} has two critical propagation paths in the overall framework. On the one hand, it delivers refined spatio-temporal feature information for the subsequent multi-scale prediction and hierarchical fusion module. On the other hand, it is fed into the dynamic graph generator to participate in node encoding, similarity calculation and dynamic weight iteration, providing data support for constructing the feature-aware dynamic adjacency matrix A_d. This design forms a closed-loop collaborative optimization mechanism: dynamic graph topological constraints guide adaptive attention feature learning, and enhanced interaction features in turn optimize dynamic graph construction, fundamentally improving the model's adaptability to complex and time-varying traffic flow patterns

2.2 Adaptive Temporal Attention

As shown in Figure 2.The temporal attention module is designed to capture both absolute traffic magnitude and short-term dynamic variation trends in time series. Different from conventional fixed temporal modeling methods, this model integrates original traffic observations and first-order difference trend features to enhance the modeling ability of time-varying traffic patterns. Given the input traffic feature $\mathrm{X} \in \mathbb{R}^{B\times N\times T\times d_{\mathrm{in}}}$, where B denotes the batch size, N represents the number of road nodes, T is the input time series length, and d_{in} is the dimension of the original input feature, we first calculate the first-order difference feature to reflect the instantaneous changing trend of traffic flow.

$$\Delta X_{:,:,t} = X_{:,:,t} - X_{:,:,t-1} \quad (5)$$

where $\Delta\mathrm{X} \in \mathbb{R}^{B\times N\times (T-1)\times d_{\mathrm{in}}}$ denotes the first-order difference feature.For alignment on time-step dimension, we pad zero vector for the initial time step $t = 0$. After padding, $\Delta\mathrm{X}$ is reshaped to $\mathbb{R}^{B\times N\times T\times d_{\mathrm{in}}}$.

We concatenate raw traffic feature and difference feature along feature dimension:

$$X_{\mathrm{cat}} = \mathrm{Concat}(X, \Delta X) \in \mathbb{R}^{B\times N\times T\times 2d_{\mathrm{in}}} \quad (6)$$

The concatenated feature $\mathrm{X}_{\mathrm{cat}}$ is then projected to obtain query, key and value for temporal multi-head self-attention:

$$Q = X_{\mathrm{cat}}W_Q, \quad K = X_{\mathrm{cat}}W_K, \quad V = X_{\mathrm{cat}}W_V \quad (7)$$

Where $X_{cat} \in \mathbb{R}^{B\times N\times T\times 2d_{in}}$ is the fused temporal feature obtained by concatenating original traffic feature and zero-padded first-order difference trend feature; $W_Q, W_K, W_V \in \mathbb{R}^{2d_{in}\times d_{model}}$ represent learnable projection matrices used to map the fused double-dimension l to the

IJBA Accepted

unified model hidden dimension; $Q, K, V \in \mathbb{R}^{B\times N\times T\times d_{model}}$ denote the query, key, and value tensors for temporal multi-head self-attention; B is batch size, N is number of spatial sensor nodes, T is input time-step length, d_{in} = original input feature dimension, d_{model} is unified hidden dimension of the model. Based on the generated Q, K, V, temporal self-attention calculates the pairwise temporal dependency between different time steps. To avoid future information leakage in time-series forecasting, a causal mask is applied to block invalid future time-step connections during attention scoring:

$$A_{temp} = \text{Softmax}\left(\frac{QK^{\top}}{\sqrt{d_{model}}} + M_{causal}\right) \tag{8}$$

where M_{causal} is the causal mask matrix, which ensures that each time step only relies on historical and current information. The temporal attention output is calculated as:

$$H_{temp} = A_{temp}V \in \mathbb{R}^{B\times N\times T\times d_{model}} \tag{9}$$

Finally, multi-head concatenation and residual connection are adopted to stabilize training and enhance temporal feature representation. The output H_{temp} preserves both long-term periodic dependencies and short-term variation trends, and is delivered to the spatial attention module for subsequent spatio-temporal feature aggregation.

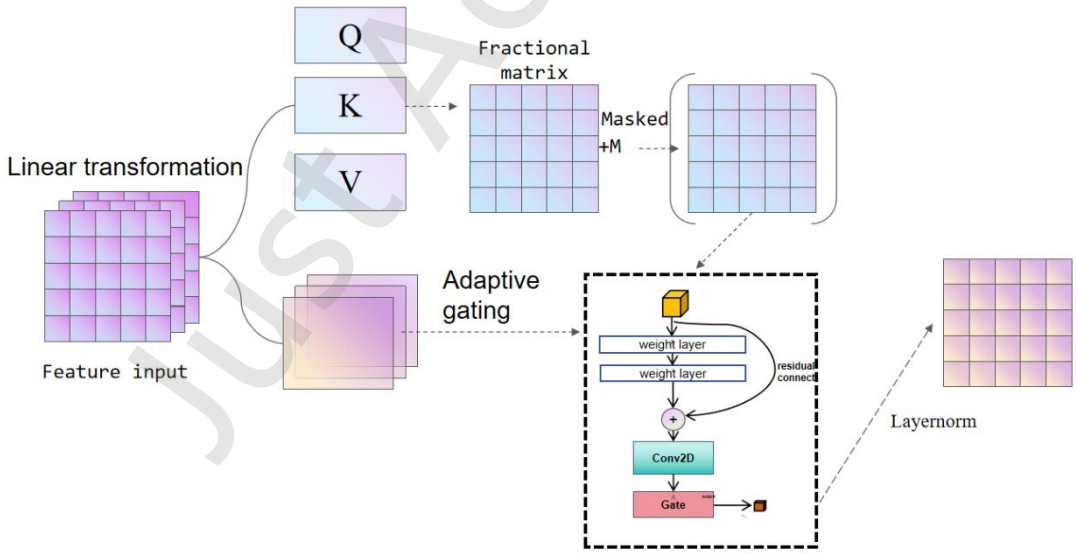


Fig. 2 Adaptive time attention module

Just Accepted

2.3 Adaptive Spatial Attention

The adaptive spatial attention module is shown in Figure 3. The input feature dimension is [B,T,N,C]. First, the dimension is reorganized into [B*T,N,C], and then the node feature coding is performed to obtain the feature matrix with the dimension of [B*T,N,d]. The node feature coding processing logic is：

$$h_{ij} = W_{enc}X_{ij} + b_{enc} \tag{10}$$

Where x_{ij} is the input original eigenvector of the ith batch and the jth node in the feature (dimension is c); W_{enc} is the coding weight matrix (dimension D × C), which is used to map the original features to high-dimensional space; b_{enc} is the encoding offset vector (dimension D), used to adjust the mapped feature offset; h_{ij} is the eigenvector (dimension D) of the ith batch time combination and the jth node after coding. After the feature code is obtained, the cosine similarity is calculated to generate the dynamic adjacency matrix Ad and the similarity calculation formula is:

$$sim_{ij} = \frac{h_i \cdot h_j}{\| h_i \| \| h_j \|} \tag{11}$$

Where h_i and h_j are the encoded feature vectors of the ith and jth nodes, respectively, both with dimension d. $h_i \cdot h_j$ is the dot product of the two feature vectors, which measures their directional similarity. $\| h_i \| \| h_j \|$ represents the product of the L2 norms of the two feature vectors. sim_{ij} is the cosine similarity between node and node , ranging from , where a higher value indicates greater similarity； $Q_s/K_s/V_s$:are generated simultaneously:

$$Q_s = W_q h \tag{12}$$

$$K_s = W_k h \tag{13}$$

$$V_s = W_v h \tag{14}$$

Where W_q is the query weight matrix with dimension. W_k is the key weight matrix with dimension $d_k \times d$.h is the encoded node feature matrix with dimension.Q_s is the query matrix with dimension $N \times d_k$.K_s is the key matrix with dimension , which characterizes the "keyword description" of the node's own features.V_s is the value matrix with dimension $N \times d_v$, which represents the "feature values" available for extraction from the nodes. The obtained matrices are then utilized to compute the spatial attention scores：

$$Attention(Q_s, K_s, V_s) = \text{Softmax}\left(\frac{Q_s K_s^T}{\sqrt{d_k}}\right)V_s \tag{15}$$

The obtained dynamic adjacency matrix is fused with the static adjacency matrix (from the

Just Accepted

static CSV file):

$$\widehat{A} = Softmax\left(\frac{\alpha A + (1-\alpha)A_{\mathrm{d}}}{\tau}\right) \tag{16}$$

Where $\widehat{\mathrm{A}} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ is the normalized adjacency matrix after fusion, serving as the attention weight matrix. $\alpha \in [0,1]$ is the balance coefficient that controls the contribution ratio of the static graph. $\mathrm{A} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ is the static adjacency matrix, which is the prior matrix composed of the original input files. $\mathrm{A_d} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ is the dynamic adjacency matrix. $\tau > 0$ is the temperature coefficient that controls the sharpness of the Softmax function, where smaller values lead to more concentrated weight distributions. Softmax(·) is the normalization function. Subsequently, the graph convolution operation is performed:

$$X^{'} = \sigma\left(\widehat{A}XW + b\right) \tag{17}$$

Where $\mathrm{X}' \in \mathbb{R}^{\mathrm{N}\times\mathrm{C}'}$ is the output feature matrix of the graph convolution. $\widehat{\mathrm{A}} \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ is the fused adjacency matrix. $\mathrm{X} \in \mathbb{R}^{\mathrm{N}\times\mathrm{C}}$ is the input feature matrix. $\mathrm{W} \in \mathbb{R}^{\mathrm{C}\times\mathrm{C}'}$ is the learnable weight matrix of the graph convolution. $\mathrm{b} \in \mathbb{R}^{\mathrm{C}'}$ is the bias vector. To mitigate the gradient vanishing problem, the module adopts a residual connection:

$$\mathrm{y}=\mathrm{X}'+\mathrm{X} \tag{18}$$

Finally, the result is processed by layer normalization to obtain the output features after adaptive spatial attention processing:

$$X_{out} = \mathrm{LayerNorm}(y) \tag{19}$$

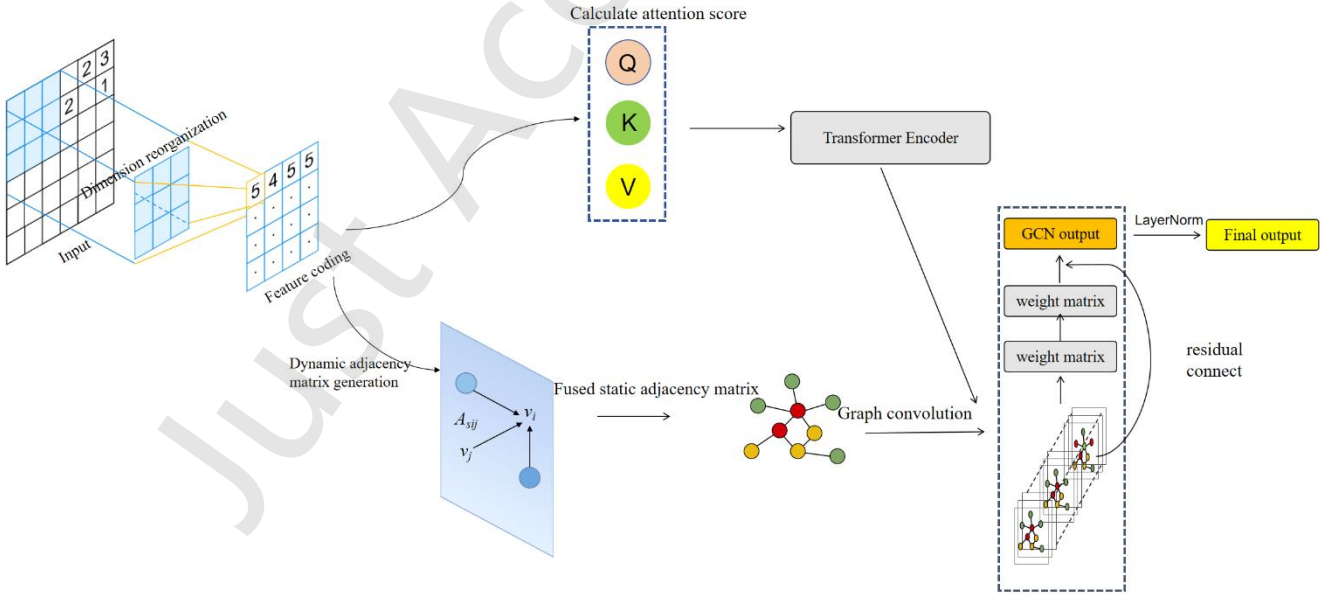


Fig. 3 Adaptive Spatial Attention Module

2.4 Multi Scale Feature Fusion Module

The multi-scale feature fusion module aims to effectively integrate feature information across different scales to enhance the model's understanding and processing capabilities for complex flow patterns, playing a critical role in traffic flow prediction, and its workflow is illustrated in Figure 4. Given input features $\mathbf{X} \in \mathbb{R}^{n\times d}$, where n is the number of samples and d denotes the input feature dimension, we first check the length T of the input time series to activate or

Just Accepted

skip each scale branch; a branch will be skipped if T is smaller than its temporal kernel size. For activated branches, scale-1 adopts a 1×1 convolution f_1, scale-3 uses a $1D$ causal convolution with kernel size 3 to capture medium-term short-cycle variations, and scale-6 employs a kernel size-6 convolution to extract long-range periodic traffic patterns such as peak-and-off-peak cycles. Activated branches are configured to produce outputs with identical feature dimension d_{out}, and the output of each valid branch i is defined as

$$Y_i = f_i(X) \in \mathbb{R}^{n \times d_{out}} \tag{20}$$

where Yi is the feature output of the i-th valid scale branch; fi (·) denotes the convolution operation of corresponding scale branch; n is sample count; dout represents the unified output feature dimension for all activated branches.

Let $N_{valid} \in [1,3]$ represent the number of effective activated scale branches, raw weight scores $\mathbf{s} \in \mathbb{R}^{N_{valid}}$ are generated via a lightweight learnable linear layer, and Softmax normalization is performed only over valid branches to obtain normalized fusion weights:

$$\widehat{\omega}_i = \frac{\exp(s_i)}{\sum_{k=1}^{N_valid} \exp(s_k)} \tag{21}$$

Where $\widehat{\omega}_i$ is the normalized learnable fusion weight for the i-th valid branch; si is the raw weight score produced by the lightweight linear layer; Nvalid is the total number of activated scale branches; skipped branches are excluded from normalization.

The weight $\sum_{i=1}^{N_{valid}} \widehat{\omega}_i = 1$, where skipped branches are excluded from both weight normalization and subsequent calculation. The final fused feature is obtained by weighted summation of multi-scale branch outputs:

$$Y_{final} = \sum_{i=1}^{N_{valid}} \widehat{\omega}_i \cdot Y_i \in \mathbb{R}^{n \times d_{out}} \tag{22}$$

where $\mathbf{Y}_{final}$ denotes the final fused multi-scale feature for subsequent model computation, and · represents element-wise multiplication, $\mathbf{Y}_{final}$ is then delivered for subsequent computation within the model.

Just Accepted

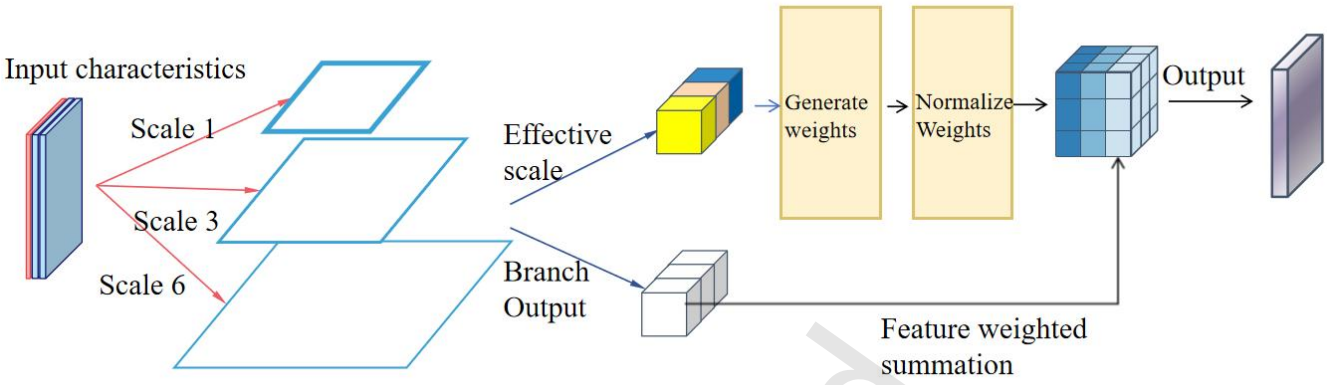


Fig. 4 Multiscale fusion module

Different from simple feature concatenation, our multi-scale fusion adopts learnable weighted summation rather than concatenation to aggregate multi-granularity features, avoiding dimension mismatch and redundant feature information

Input features go through three convolution branches with different kernel sizes to obtain multi-scale branch outputs. Learnable weights are generated and normalized only over activated valid branches. Final feature representation is obtained via weighted summation of branch outputs.

3.Experiment

3.1. Data source and preprocessing

Original data: the data comes from the public data set PEMs, a traffic monitoring system developed by the California Department of transportation in the United States, which collects traffic flow, speed, time occupancy axnd other data in real time through the sensor network deployed on the highway.

Table 1 Dataset Details

DataSet	Number of nodes	Number of edges	Number of Timestamps
PEMS04	307	340	16,992
PEMS08	170	295	17,856

xData preprocessing: The whole dataset is split into training set, validation set and test set following the ratio of 7:1.5:1.5. All hyper-parameter searching and early-stopping operations are executed only on the validation set. The test set is completely isolated and only used for final performance evaluation, which avoids data leakage and over-optimism towards test metrics.Grid search is adopted for hyper-parameter tuning. The searched hyper-parameter ranges are listed as follows: balance coefficient $\alpha \in \{0.0,0.2,0.4,0.6,0.8,1.0\}$, temperature

Just Accepted

coefficient $\tau \in \{0.1,0.5,1.0,2.0,5.0\}$, initial learning rate $\in \{5\times10-4,1\times10-3,2\times10-3\}$, hidden dimension $\in \{32,64,128\}$. The early- stopping patience is set to 15 according to validation loss, and the learning rate decays by 0.5 every 15 epochs.

3.2. Experiment design and Implementation

3.2.1Comparative model

Select the following models for comparison:
ASTGCN[7]: spatiotemporal map convolution network based on attention mechanism, which is used to process spatiotemporal data and is applied in traffic flow prediction and other fields.
GAT: graph attention network is a method that uses the attention mechanism to calculate the weights between nodes in a graph. It can well reflect the structural characteristics of a graph.
GCN: convolution network of graph. Convolution operation can be directly performed on the structure of graph to learn the characteristics of nodes in the network.
LSTM: Long Short-Term Memory, a kind of RNN, controls the flow of information through the gating mechanism, and is good at handling the long-term dependence in time series data
GRU: gated cyclic unit, which is a variant of recurrent neural network (RNN). Compared with traditional RNN, GRU can better handle the long-term dependence in long-term sequence data.
STGCN: spatiotemporal graph convolution network, based on the spatiotemporal correlation of graph convolution network (GCN) and time-domain convolution/loop network, realizes the dual representation of "spatial correlation" and "time evolution" of time-series data.
PDFormer[3]: Propagation Delay-aware Dynamic Long-range Transformer, a Transformer-based architecture that incorporates delay-aware spatial attention and graph masking mechanisms to explicitly model dynamic long-range dependencies and temporal lags in spatiotemporal traffic data.
All comparison models are trained and tested with the same data set and evaluation index to ensure that the results are comparable.

3.2.2 Evaluating indicator

The root mean square error (RMSE), mean absolute error (MAE) and symmetric mean absolute percentage error (SMAPE) are used to evaluate the performance of the model. The calculation formula is as follows:
Root mean square error:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \tag{23}$$

Mean absolute error:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i| \tag{24}$$

Where N is the number of samples, $\hat{y}_i$ is the predicted value, y_i is the real value, and $\bar{y}$ is the average value of the real value. The smaller the RMSE and MAE values, the better the prediction effect of the model.

Just Accepted

Symmetrical mean absolute percentage error：

$$SMAPE = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{2 \times |y_i - \hat{y}_i|}{|y_i| + |\hat{y}_i|}\right) \times 100\% \quad (25)$$

Where: n is the total number of data points (such as the number of predicted days and samples); y_i is the actual value of the ith data point;$\hat{y}_i$ is the predicted value of the ith data point; $|\cdot|$ indicates the sign of absolute value (ensure that the error is non negative).

To guarantee experimental reproducibility and fair comparison among all competing methods, we summarize critical experimental configurations in Table 2. All baseline models and DSTT-HF adopt the identical dataset splitting ratio, sliding-window segmentation and per-node normalization. Data augmentation operations including time offset, Gaussian white noise and random scale are applied equally across all models, eliminating performance bias introduced by inconsistent data preprocessing.

Table 2 Implementation and reproducibility checklist

Item	Setting
Input sequence window T	12 time steps (each step = 5 min traffic interval)
Prediction horizon	1-step(5 min), 2-step(10 min), 3-step(15 min)
Hidden dimension	64
Optimizer	Adam
Initial learning rate	0.001, learning rate decay 0.5 every 15 epochs
Max training epochs	150
Early-stop strategy	15
Hardware	GPU: NVIDIA RTX 3090; CUDA 11.7; PyTorch 2.0
Dataset split ratio	Train/Val/Test = 7 : 1.5 : 1.5

Just Accepted

3.3. Comparative experiment and analysis

3.3.1 Result analysis

Grid search is adopted for hyperparameter tuning. We traverse the predefined hyperparameter space—including input sequence length, batch size, hidden channel number, and learning rate—through multi-layer nested loops, selecting the optimal configuration on the validation set. Each time step represents a 5-minute traffic interval. Predictions are performed for the following three steps (15 minutes), covering a short-term forecasting horizon, with averaged

error outcomes listed in Table 2 and visualized in Figure 5. Experimental results on PEMS04 and PEMS08 reveal that DSTT-HF achieves outstanding performance across all three prediction horizons, consistently outperforming all baselines in MAE, RMSE and SMAPE. The baselines span three categories: traditional time-series models (GRU, LSTM), basic graph neural networks (GAT, GCN), and advanced spatio-temporal architectures (ASTGCN, STGCN and the state-of-the-art PDFormer). On PEMS04, DSTT-HF achieves a one-step MAE of merely 17.80, an 8.2% reduction compared with STGCN (19.40) and better than PDFormer, alongside an RMSE of 28.55 and an SMAPE of 12.97, both markedly superior to all competitors. Its two-step prediction reaches MAE = 18.64, RMSE = 29.76, SMAPE = 13.30, and the three-step prediction obtains MAE = 19.02, RMSE = 30.38, SMAPE = 13.67, retaining optimal metrics without obvious performance decay and demonstrating strong stability. On PEMS08, DSTT-HF delivers a one-step MAE of 13.57, 9.8% lower than STGCN (15.04) and superior to PDFormer, with RMSE = 20.94 and SMAPE = 8.93; its two-step prediction yields MAE = 14.18, RMSE = 21.97, SMAPE = 9.37, and the three-step prediction achieves MAE = 14.72, RMSE = 22.69, SMAPE = 9.56, surpassing all baselines including PDFormer at every step. All metrics are reported as mean ± std over 10 independent training runs, where lower MAE, RMSE and SMAPE indicate better performance. A paired t-test is used for the statistical significance test: **denotes $p < 0.01$ and * denotes $0.01 < p < 0.05$ compared with our model.

Table 3 Comparison Experiment of Baseline Model

DataSet	Mode	1 time step			2 time step			3 time step		
		MAE	RMSE	SMAPE	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE
PEMS04	GRU	26.41±0.62**	42.25±0.81**	18.17±0.54**	27.21±0.65**	43.61±0.86**	19.14±0.59**	28.01±0.69**	44.32±0.88**	20.01±0.61**
	LSTM	25.77±0.58**	41.59±0.77**	17.56±0.51**	24.77±0.55**	40.35±0.73**	17.04±0.48**	23.37±0.52**	39.25±0.70**	16.47±0.45**
	GAT	26.59±0.64**	39.44±0.74**	19.42±0.57**	27.65±0.68**	41.17±0.78**	19.74±0.60**	27.79±0.67**	40.51±0.76**	20.10±0.62**
	GCN	25.14±0.56**	40.80±0.75**	17.39±0.49**	26.33±0.61**	41.97±0.79**	18.02±0.54**	26.82±0.63**	41.33±0.77**	17.74±0.51**
	ASTGCN	21.86±0.42**	38.93±0.68**	18.33±0.46**	22.87±0.46**	40.04±0.71**	20.48±0.52**	24.36±0.51**	41.45±0.75**	21.25±0.55**
	STGCN	19.40±0.33**	30.50±0.47**	14.12±0.36**	20.37±0.36**	31.68±0.51**	14.72±0.39**	21.10±0.38**	32.75±0.54**	15.11±0.41**
	PDFormer	18.45±0.22**	29.42±0.35**	13.48±0.27**	19.22±0.24**	30.61±0.38**	13.85±0.29**	19.67±0.26**	31.24±0.40**	14.21±0.31**
	Ours	17.80±0.16	28.55±0.24	12.97±0.19	18.64±0.18	29.76±0.26	13.30±0.21	19.02±0.19	30.38±0.28	13.67±0.22
PEMS08	GRU	24.08±0.55**	36.93±0.73**	14.48±0.46*	24.92±0.59**	37.92±0.76**	15.15±0.50**	25.84±0.63**	39.43±0.80**	15.62±0.52**
	LSTM	25.25±0.	38.51±0.	15.07±0.	25.00±0.	37.95±0.	15.00±0.	24.87±0.	38.22±0.	14.74±0.

Just Accepted

	59**	78**	48**	57**	75**	47**	56**	76**	46**
GAT	23.33±0.	34.70±0.	14.81±0.	23.82±0.	35.47±0.	14.99±0.	24.53±0.	36.18±0.	15.62±0.
	52**	68**	47**	54**	70**	48**	57**	73**	51**
GCN	22.01±0.	34.15±0.	13.43±0.	22.82±0.	35.11±0.	14.15±0.	23.57±0.	34.99±0.	13.89±0.
	48**	66**	42**	51**	69**	45**	53**	68**	44**
ASTGCN	15.65±0.	23.65±0.	10.22±0.	16.07±0.	24.39±0.	11.21±0.3	16.52±0.	25.65±0.	11.92±0.3
	31**	43**	30**	33**	50**	4**	35**	49**	7**
STGCN	15.04±0.	23.07±0.	9.75±0.2	16.07±0.	24.69±0.	10.23±0.	16.47±0.	25.13±0.	10.47±0.
	28**	40**	7**	31**	55**	30**	33**	47**	32**
PDFormer	14.29±0.	21.86±0.	9.21±0.2	14.85±0.	22.73±0.	9.64±0.2	15.31±0.	23.48±0.	9.97±0.2
	19**	29*	0**	21**	32**	2**	23**	34**	4**
Ours	13.57±0.	20.94±0.	8.93±0.1	14.18±0.	21.97±0.	9.37±0.1	14.72±0.	22.69±0.	9.56±0.1
	13	20	5	15	22	7	17	24	9

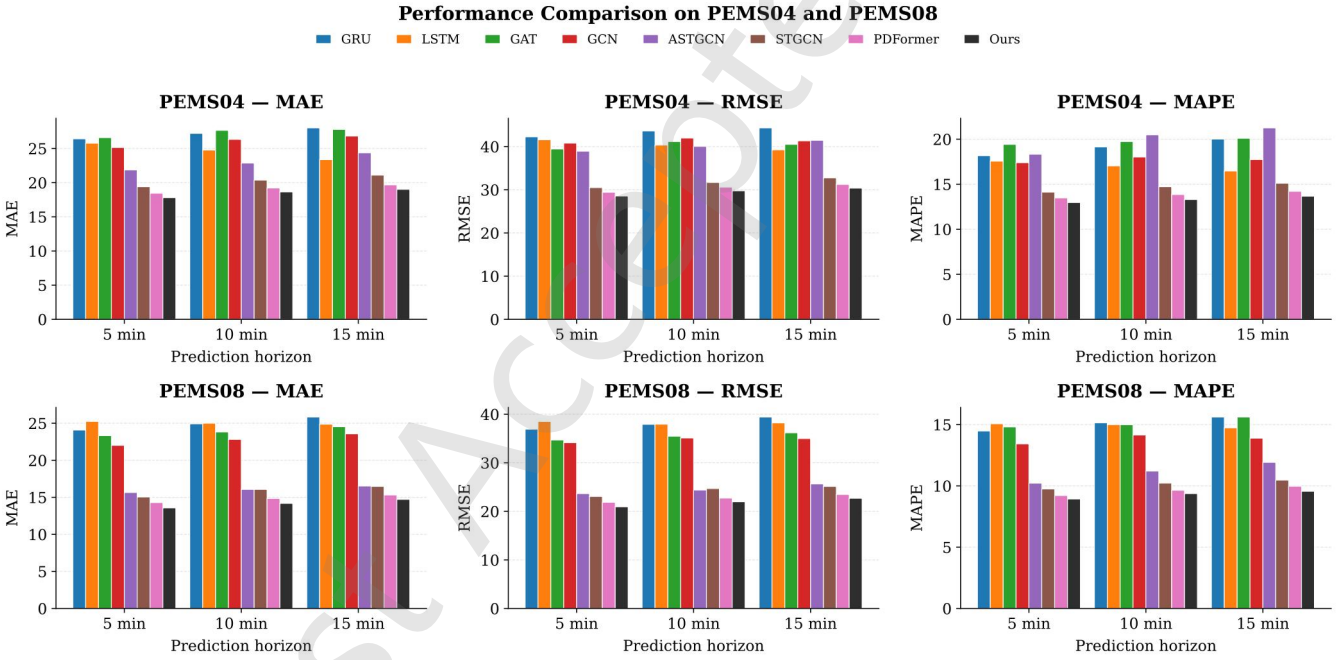


Fig. 5 Performance comparison between DSTT-HF and benchmark model on different datasets

3.3.2 Ablation Experiment

As shown in Table 4, removing the hybrid spatio-temporal attention module yields the largest performance drop. It indicates that collaborative spatial-temporal attention is fundamental to capture complex traffic dependencies. The performance degradation of w/o Dynamic-Graph verifies that static road topology cannot describe time-varying spatial correlations in real-world traffic, demonstrating the value of our dynamic graph construction strategy. When the multi-scale hierarchical-fusion module is discarded, the error increases more significantly for two-step and three-step prediction. This observation validates that multi-scale fusion mainly mitigates error accumulation in long-horizon forecasting. The feature-interaction-enhancement module brings moderate improvement, which proves that refined feature preprocessing offers

Just Accepted

auxiliary promotion effect.

Table 4 Ablation Results

DataSet	Model	1 time step			2 time step			3 time step		
		MAE	RMSE	SMAPE	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE
PEMS04	w/o Attn	20.98±0.51	32.74±0.77	15.51±0.52	21.79±0.56	33.75±0.79	15.85±0.42	22.41±0.60	34.68±0.84	16.32±0.45
	w/o Dynamic-Graph	19.88±0.44	31.22±0.64	14.41±0.45	20.71±0.48	32.47±0.69	14.73±0.36	21.47±0.52	33.44±0.73	15.22±0.38
	w/o scale	19.65±0.46	30.46±0.68	14.64±0.48	20.26±0.50	31.40±0.72	14.78±0.38	21.03±0.54	32.35±0.76	14.88±0.40
	w/o Feat	18.68±0.42	29.59±0.63	13.76±0.44	19.24±0.45	30.36±0.67	14.38±0.34	20.30±0.49	31.69±0.71	14.79±0.36
	Full	17.80±0.38	28.55±0.59	12.97±0.41	18.64±0.41	29.76±0.62	13.30±0.31	19.02±0.44	30.38±0.66	13.67±0.33
PEMS08	w/o Attn	14.89±0.39	23.01±0.58	10.61±0.29	15.89±0.42	24.80±0.63	11.38±0.32	16.71±0.46	25.35±0.67	12.88±0.36
	w/o Dynamic-Graph	14.88±0.41	22.57±0.62	10.41±0.55	14.91±0.38	24.47±0.65	11.43±0.36	16.47±0.52	23.44±0.73	12.22±0.38
	w/o Scale	14.46±0.35	22.26±0.53	9.59±0.26	14.86±0.39	24.80±0.58	11.38±0.29	16.03±0.42	24.35±0.62	11.88±0.35
	w/o Feat	14.11±0.32	21.70±0.49	9.23±0.23	14.75±0.35	22.64±0.53	10.38±0.26	15.53±0.38	23.35±0.57	11.08±0.31
	Full	13.57±0.29	20.94±0.45	8.93±0.21	14.18±0.31	21.97±0.48	9.37±0.23	14.72±0.34	22.69±0.53	9.56±0.25

Just Accepted

3.3.3 Parameter Sensitivity Analysis

Two hyper-parameters dominate the fused adjacency matrix computation in Equation

18.balance coefficient α and temperature coefficient τ. α∈[0,1] adjusts the contribution ratio between static road topology and feature-driven dynamic graph Ad . When α=1, the model degenerates to static-graph-only configuration, which corresponds to the w/o Dynamic-Graph ablation setting. When α=0, graph construction completely discards physical prior knowledge and only relies on node-feature similarity. τ>0 acts as the temperature coefficient for Softmax, controlling the sharpness of spatial weight distribution. A too-small τ concentrates weights on a tiny subset of node pairs, while an excessively large τ smooths all weights and weakens spatial discrimination capability. We conduct parameter sensitivity experiments on the PEMS04 dataset for one-step prediction, with all other hyper-parameters kept fixed. We first tune α and report results in Table 5. For the subsequent τ sensitivity test, α is fixed to its optimal value 0.4 to eliminate the interference of balance coefficient. The experimental results are listed in Table 5 and Table 6. The model achieves optimal performance at α=0.4 and τ=1.0. When deviating from this optimal point, MAE, RMSE and SMAPE increase consistently. This illustrates that moderately fusing static topology and dynamic traffic-aware graph, together with appropriate Softmax temperature, is essential for promising prediction performance.Figure 6 shows the heatmap comparison between feature-aware adjacency matrix Ad and final fused adjacency matrix A. The matrix Ad is learned purely from traffic-feature similarity, which may generate noisy correlations between geographically distant road segments. After weighted fusion with static road topology under hyper-parameters α and τ, the fused matrix A achieves sparser weight distribution. It retains the asymmetric property of one-way traffic propagation, meanwhile suppresses false correlations inconsistent with real-world road network structure.

Table 5 Performance under different balance coefficient α(PEMS04, 1-step prediction)

α	MAE	RMSE	SMAPE
0.0	18.72	29.81	13.62
0.2	18.13	29.04	13.21
0.4	17.80	28.55	12.97
0.6	18.01	28.22	13.14
0.8	18.44	29.37	13.50
1.0	19.88	31.22	14.41

Table 6 Performance under different temperature coefficient τ (PEMS04, 1-step prediction, α fixed = 0.4)

τ	MAE	RMSE	SMAPE
0.1	18.65	29.74	13.57
0.5	18.11	28.97	13.18
1.0	17.80	28.55	12.97
2.0	18.08	28.88	13.30
5.0	18.53	29.60	13.51

To further intuitively compare the difference between the feature-aware dynamic adjacency matrix Ad and the final fused adjacency matrix A, we visualize the two matrices as heatmaps, as shown in Figure 6.This visualization is conducted on the PEMS04 dataset, covering the time period from Jan. 1 to Feb. 28, 2018 with a 5-minute sampling interval.All 307 highway traffic sensor nodes are retained without flow-threshold filtering, and the original road-network

Just Accepted

topology is directly adopted as static prior information.We calculate the sparsity of two adjacency matrices by treating weights lower than 10−3 as ineffective zero- valued connections. The sparsity of pure feature- driven dynamic adjacency matrix Ad is 0.7162, while the sparsity of the fused adjacency matrix A is reduced to 0.4075.Owing to the introduction of static road- network constraints, A obtains a sparser weight distribution and suppresses spurious spatial correlations among geographically distant nodes, which is consistent with the above parameter- sensitivity analysis conclusions

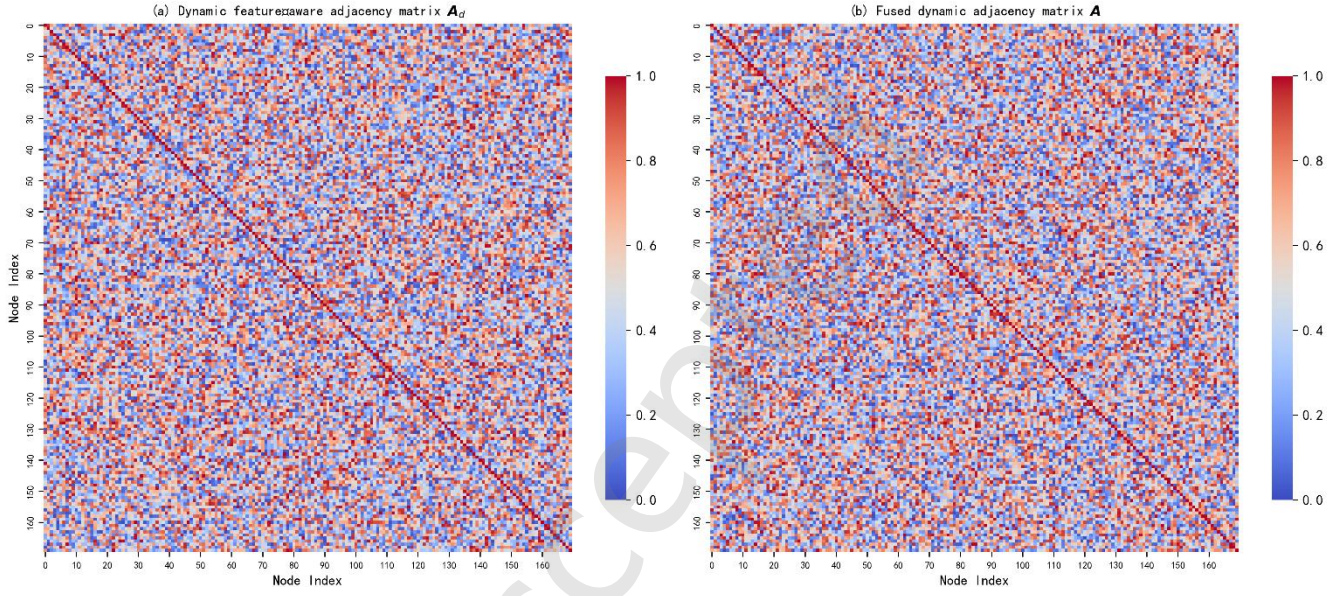


Fig. 6 Heatmap comparison of Ad and A

3.3.4 Model Complexity and Parameter Uncertainty

The total number of trainable parameters of DSTT-HF is around 2.12 M. For comparison, STGCN contains approximately 1.46 M parameters and ASTGCN reaches 1.78 M parameters. Our model brings moderate parameter overhead instead of blindly expanding model capacity to obtain performance gains. The overall computational cost measured by floating-point operations is 18.4 G FLOPs for one complete forward pass under PEMS04 input configuration. Under the unified hardware of NVIDIA RTX 3090, DSTT-HF consumes roughly 42 seconds for each training epoch, with total training duration of about 105 minutes until early-stop trigger. The inference latency for one sample is 12.6 ms. By contrast, STGCN takes 27 seconds per epoch with total training time of 72 minutes, and its single-sample inference latency is 7.3 ms.
All experimental results are obtained from ten independent repeated runs with different random initialization seeds. The mean ± std metrics presented in all experimental tables quantify performance uncertainty caused by parameter initialization. The hyper-parameter sensitivity results in Section 3.3.3 further demonstrate that DSTT-HF maintains stable prediction performance within reasonable hyper-parameter intervals and does not exhibit sharp performance oscillation with minor parameter perturbation.

4 Conclusions

Just Accepted

Aiming at the task of traffic flow prediction, this paper proposes DSTT-HF model, and through a large number of experiments on pems04 and pems08 public data sets, compares it with traditional time series model (GRU, LSTM), graph neural network model (GAT, GCN) and advanced spatio-temporal fusion model (astgcn, stgcn), and comprehensively verifies the performance advantages of the model. The main research conclusions are as follows:

Short time scale prediction accuracy is leading: DSTT-HF model can accurately capture the time-space coupling characteristics of traffic flow, and has high short-term prediction accuracy, providing reliable data support for real-time traffic control.

The stability of long-time scale prediction is excellent: on the two data sets, the three key indicators of MAE, RMSE and SMAPE maintain the lowest level in all time steps, and the error growth curve is more gentle than the baseline model. DSTT-HF model can effectively restrain the error accumulation in long-term forecast, and can meet the requirements of long-term traffic flow forecast.

The effectiveness and practical value of the model architecture are significant: it can not only accurately mine the time sequence characteristics of traffic flow (such as the change law of peak traffic flow), but also better reflect the spatial correlation between nodes in the road network.

JUST Accepted

Acknowledgments

This work was supported by the Natural Science Foundation of Fujian Province and the Scientific Research Fund of Fujian University of Technology. The authors thank Prof. Fumin Zou for supervision and Dr. Qinqin Cai for technical support.

Funding

This work was supported by Fuzhou Municipal Science and Technology Bureau – Fuzhou-Xiamen-Quanzhou National Independent Innovation Demonstration Zone Collaborative Innovation Special Project: Collaborative Innovation Platform for the Development and Industrialization of Three-Electric Systems in New Generation Motor Loaders (Project No.2024-P-007, Funding No.20252400231); the Foundation for Cultivated Young Talents of Fujian Province, China (Grant No. 2025350443); and the 2025 Doctoral Scientific Research Startup Fund (Grant No. ZZYB2507). The funding bodies had no role in the design of the study, the collection, analysis and interpretation of research data, nor the writing and revision of this manuscript.

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

Author contribution statement

Jin Hu: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data Curation, Writing—Original Draft, Visualization.

Fumin Zou: Resources, Funding acquisition, Project administration, Supervision, Writing—Review & Editing.

Qinqin Cai: Methodology, Software, Validation, Formal analysis, Data Curation, Writing—Review & Editing.

Data availability statement

The data that support the findings of this study are available from the corresponding author upon reasonable request. However, restrictions apply to the availability of these data, which were used under license for the current study, and so are not publicly available. Data are however available from the authors upon reasonable request and with permission of the relevant authorities.

Use of AI statement

During the preparation of this research, we used ChatGPT to improve grammar and readability. After using this tool, we reviewed and edited the content as needed and take full responsibility for content of the publication.

References

[1] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. Adv. Neural Inf. Process. Syst. 2017, 30: 5998-6008. https://doi.org/10.48550/arXiv.1706.03762.

[2] Feng M, Meunier J. Skeleton graph-neural-network-based human action recognition: A survey[J]. Sensors, 2022, 22(6): 2091.

[3] Jiang J, Han C, Zhao W X, et al. PDFormer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7-14 February 2023; 2023, 37(4): 4365-4373. https://doi.org/10.1609/aaai.v37i4.25593.

[4] Xu B, Shen H, Cao Q, et al. Graph wavelet neural network. arXiv preprint arXiv:1904.07785, 2019. https://doi.org/10.48550/arXiv.1904.07785.

[5] Chen H, Shi Z. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. Remote Sens. 2020, 12(10): 1662. https://doi.org/10.3390/rs12101662.

[6] Song C, Lin Y, Guo S, et al. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7-12 February 2020; 2020, 34(1): 914-921. https://doi.org/10.1609/aaai.v34i01.5438.

[7] Zhu J, Wang Q, Tao C, et al. AST-GCN: Attribute-augmented spatiotemporal graph convolutional network for traffic forecasting. IEEE Access 2021, 9: 35973-35983.

[8] Xu Y, Han L, Zhu T, et al. Generic dynamic graph convolutional network for traffic flow forecasting. Inf. Fusion 2023, 100: 101946.

Just Accepted

[9] Mu H ,Aljeri N ,Boukerche A .Hierarchical multi-scale spatio-temporal semantic graph convolutional network for traffic flow forecasting[J].Journal of Network and Computer Applications,2025,238: 104166.https://doi.org/10.1016/j.jnca.2025.104166.

[10] Feng N, Guo S N, Song C, et al. Multi-component spatial-temporal graph convolutional network for traffic flow forecasting. J. Softw. 2019, 30(3): 759-769. https://doi.org/10.13328/j.cnki.jos.005697.

[11] Xie J, Meng Y, Zhao Y, Nguyen A, Yang X, Zheng Y. Dynamic semantic based spatial graph convolution network for skeleton based human action recognition[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Vancouver: AAAI Press, 2024, 38(6): 6225 6233. https://doi.org/10.1609/aaai.v38i6.28440..

[12] Bai L, Yao L, Li C, et al. Adaptive graph convolutional recurrent network for traffic forecasting[J]. Advances in neural information processing systems, 2020, 33: 17804-17815.

[13] Wang H, Chen J, Pan T, et al. Stgformer: Efficient spatiotemporal graph transformer for traffic forecasting[J]. arXiv preprint arXiv:2410.00385, 2024.

[14] Liu Z, Ding F, Dai Y, et al. Spatial-temporal graph convolution network model with traffic fundamental diagram information informed for network traffic flow prediction[J]. Expert Systems with Applications, 2024, 249: 123543.

[15] Wang L, Ning N, Liu Y, et al. Dynamic Spatial-Temporal Dual Graph Neural Networks for Urban Traffic Prediction[C]//Pacific Rim International Conference on Artificial Intelligence. Singapore: Springer Nature Singapore, 2023: 365-376.

[16] Gu R, Tan Q, Liu Y, et al. ADMFormer: An Adaptive-Decomposition Transformer with Time-Varying Masked Spatial Attention for Traffic Forecasting[J]. arXiv preprint arXiv:2605.25543, 2026.

[17] Shi P, Lu Q, Chen J, et al. AMGST: Adaptive multi-graph convolution and spatiotemporal attention network for traffic forecasting[J]. PloS one, 2026, 21(6): e0342235.

[18] Li P, Tian Z, Shang C, et al. Traffic Flow Prediction Model Based on Transformer and Dynamic Graph Convolutional Networks[J]. IET Intelligent Transport Systems, 2025, 19(1): e70111.

[19] Zheng C, Fan X, Pan S, et al. Spatio-temporal joint graph convolutional networks for traffic forecasting[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 36(1): 372-385.

[20] Ali A, Ali R, Asad M, Yang L, Alsarhan T, Bai X. Exploiting attention-driven weather-aware multimodal spatio-temporal fusion for urban traffic flow prediction[J]. Future Generation Computer Systems, 2026, 183: 108559.https://doi.org/10.1016/j.future.2026.108559.

Just Accepted

Author biography



Hu, Jin (2002–), male, born in Changde, Hunan Province, China. M.Sc. candidate, School of Electronic, Electrical and Physics, Fujian University of Technology. Research interests: ETC data mining and analysis. E-mail: 2336282846@qq.com; Tel.: +86-18651578412. Address: School of Electronic, Electrical and Physics, Fujian University of Technology, No. 33 South Xuefu Road, Shangjie Town, Minhou County, Fuzhou 350118, Fujian, China. (Corresponding author).



Zou, Fumin (1976–), male, Professor and Ph.D. Supervisor, Fujian University of Technology; Fujian Key Laboratory of Automotive Electronics and Electric Drive Technology. E-mail: fmzou@fjut.edu.cn; Tel.: +86-13178035505..



Cai, Qiqin (1994–), male, Ph.D., School of Equipment Manufacturing and Vehicle Engineering, Zhangzhou Institute of Technology. E-mail: 742349408@qq.com; Tel.: +86-18144056202..

Just Accepted