

MGTTP: A multi-graph transformer model for traffic flow forecasting via bidirectional spatio-temporal interaction

Xiaolong Fan(范小龙) and Jialin He(何嘉林)

Citation: Chin. Phys. B, 2026, 35 (**7**): 078905. DOI: 10.1088/1674-1056/ae6635
http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

MGTTP: A multi-graph transformer model for traffic flow forecasting via bidirectional spatio-temporal interaction

Xiaolong Fan(范小龙) and Jialin He(何嘉林)
Chin. Phys. B, 2026, 35 (**6**): 068902. DOI: 10.1088/1674-1056/ae6635

A node importance prediction algorithm based on graph attention and contrastive learning

Jun Ai(艾均), Yuming Zhang(张玉明), Zhan Su(苏湛), Chenye Guo(郭晨晔), and Mingsong Li(李铭松)
Chin. Phys. B, 2026, 35 (**5**): 058901. DOI: 10.1088/1674-1056/ae0680

Critical station identification of metro networks based on the integrated topological-functional algorithm: A case study of Chengdu

Zi-Qiang Zeng(曾自强), Sheng-Jie He(何圣洁), and Wang Tian(田旺)
Chin. Phys. B, 2025, 34 (**2**): 028902. DOI: 10.1088/1674-1056/ad9734

Dynamic partition of urban network considering congestion evolution based on random walk

Zhen-Tong Feng(冯振通), Lele Zhang(张乐乐), Yong-Hong Wu(吴永洪), and Mao-Bin Hu(胡茂彬)
Chin. Phys. B, 2025, 34 (**1**): 018902. DOI: 10.1088/1674-1056/ad94e1

WT-FCTGN: A wavelet-enhanced fully connected time-gated neural network for complex noisy traffic flow modeling

Zhifang Liao(廖志芳), Ke Sun(孙轲), Wenlong Liu(刘文龙), Zhiwu Yu(余志武), Chengguang Liu(刘承光), and Yucheng Song(宋禹成)
Chin. Phys. B, 2024, 33 (**7**): 078901. DOI: 10.1088/1674-1056/ad3349

MGTTP: A multi-graph transformer model for traffic flow forecasting via bidirectional spatio-temporal interaction

Xiaolong Fan(范小龙)[1] and Jialin He(何嘉林)[1,2,3,†]

[1] *School of Computer Science, China West Normal University, Nanchong* 637001, *China*
[2] *The Internet of Things Perception and Big Data Analysis Key Laboratory of Nanchong City, Nanchong* 637000, *China*
[3] *Artificial Intelligence Key Laboratory of Nanchong, China West Normal University, Nanchong* 637000, *China*

(Received 21 December 2025; revised manuscript received 25 April 2026; accepted manuscript online 29 April 2026)

Accurate traffic flow forecasting hinges on modeling coupled spatio-temporal dependencies rather than treating space and time in isolation. Many prior methods process spatial and temporal features separately — either in series or in parallel — and then fuse them with simple operators, which weakens their ability to capture intrinsic space–time interactions. We propose multi-graph transformer for traffic flow forecasting (MGTTP), a framework with an innovatively designed bidirectional spatio-temporal interaction mechanism: temporal signals guide multi-graph spatial fusion, while spatial context guides attention-based temporal aggregation. It addresses the limitations of static spatial fusion in existing multi-graph models and the serial spatio-temporal modeling paradigm in vanilla transformer baselines, achieving deep coupled modeling of spatio-temporal features. First, MGTTP builds three complementary graphs — adjacency, reachability, and similarity — and applies temporal feature-guided attention to dynamically fuse their multi-dimensional spatial representations. Subsequently, a transformer encoder captures long-term temporal dependencies, with spatial feature-guided attention to aggregate the time series. Finally, a gated fusion module realizes the ultimate fusion of spatio-temporal features for prediction. Extensive experiments on four public real-world traffic datasets demonstrate that MGTTP outperforms all compared mainstream baseline models across all evaluation metrics, with statistically significant performance gaps, validating the effectiveness of the proposed bidirectional spatio-temporal interaction mechanism.

Keywords: traffic flow forecasting, graph neural networks, spatio-temporal modeling, attention mechanisms, transformer

PACS: 89.40.Bb, 05.60.Gg, 05.45.Tp, 07.05.Mh, 02.70.Hm

DOI: 10.1088/1674-1056/ae6635 **CSTR:** 32038.14.CPB.ae6635

1. Introduction

Traffic flow forecasting is a core capability of intelligent transportation systems (ITS), supporting congestion management, route optimization, and urban planning. With rapid urbanization and growing vehicle ownership, the need for timely and accurate forecasts is more pressing than ever. Historical data from loop detectors and global positioning system (GPS) devices exhibit rich spatio-temporal structure: spatial dynamics reflect road topology, multi-hop connectivity, and functional similarity across regions; temporal dynamics show daily/weekly cycles, seasonal trends, and regime shifts. Meanwhile, the raw traffic flow data inevitably contain noise interference caused by sensor failures, abnormal weather, and other factors during data acquisition, which also reduces the accuracy of traffic flow forecasting.[1] Effectively modeling both dimensions — and, crucially, their interactions — remains a central challenge.

Classical time-series models such as autoregressive integrated moving average (ARIMA) capture temporal correlations but struggle to encode the non-Euclidean spatial dependencies of road networks.[2] Deep learning has advanced the field, especially graph neural network (GNN)-based approaches that model traffic on graphs where nodes are sensors or intersections and edges capture connectivity or similarity.[3–6] However, existing methods have three core limitations that hinder deep spatio-temporal coupled modeling.

First, most methods rely on a single, predefined adjacency matrix: e.g., diffusion convolutional recurrent neural network (DCRNN)[7] and spatio-temporal graph convolutional network (STGCN)[8] only consider physical adjacency relations, limiting their ability to represent multi-facet spatial relations in traffic systems. Multi-graph fusion models, represented by spatio-temporal multi-graph convolutional network (ST-MGCN)[9] and temporal multi-graph convolutional network (TMGCN),[10] incorporate multi-dimensional spatial graph structures yet only integrate multi-graph features through static weighting or concatenation, without adaptively tuning the weights of distinct spatial relations based on the temporal dynamics of individual nodes. This results in a decoupling between spatial modeling and temporal features.

Second, recurrent neural network/long short-term memory (RNN/LSTM) models[11] face difficulties with very long-range dependencies and limited parallelism despite their excellent temporal modeling performance. Transformer-based

†Corresponding author. E-mail: hejialin32@126.com

© 2026 Chinese Physical Society and IOP Publishing Ltd. All rights, including for text and data mining, AI training, and similar technologies, are reserved.

spatio-temporal models such as the ST-transformer and ST large language model (ST-LLM)[12] excel at capturing long-range temporal dependencies, but they still follow a two-stage serial structure: "independently modeling spatial features first, then independently modeling temporal features, and finally performing shallow fusion", where spatio-temporal information flows only unidirectionally, failing to achieve deep feature-level interaction.

Most importantly, spatial and temporal modules are often designed separately and only loosely fused via simple serial or parallel feature fusion approaches, overlooking the intrinsic coupling between space and time, which leads to suboptimal prediction performance.

To address these issues, we present a multi-graph transformer for traffic flow forecasting (MGTTP), a multi-graph transformer with bidirectional spatio-temporal closed-loop interaction. Unlike existing approaches that simply combine temporal and spatial features via serial or parallel concatenation (i.e., a simple two-step sequential design), our model achieves deep coupling of spatio-temporal features through two mutually conditional steps forming a closed-loop bidirectional interaction: temporal-guided spatial fusion and spatial-guided temporal aggregation. This differs fundamentally from the unidirectional serial pipeline of existing methods: temporal features dynamically guide the fusion of three spatial views — adjacency, reachability, and similarity — while spatial features inform temporal attention to focus on the most relevant historical data. This design ensures that spatial feature extraction is conditioned on the temporal dynamics of individual nodes, while temporal feature aggregation is modulated by spatial context, overcoming the limitations of unidirectional information flow in conventional two-stage fusion paradigms and yielding deeply coupled representations beyond naive concatenation or static weighting.

The main contributions of this work are summarized as follows.

(i) We propose a bidirectional spatio-temporal interaction architecture incorporating two novel mechanisms: time-guided spatial feature fusion and space-guided temporal feature aggregation. This design realizes mutually conditional closed-loop modeling of spatio-temporal features, addressing the drawbacks of spatio-temporal separation and unidirectional information flow in existing methods.

(ii) We develop a multi-perspective spatial representation learning framework that simultaneously constructs three distinct graph types — adjacency, reachability, and similarity — each capturing complementary spatial dependencies arising from physical connectivity, path-based propagation, and functional similarity, respectively.

(iii) We design a peak-aware adaptive weighted loss function that assigns higher weights to samples corresponding to peak traffic periods, alleviating the critical limitation of conventional MSE loss — its insensitivity to prediction accuracy during peak traffic hours in real-world applications.

(iv) We conduct extensive experiments on four real-world public traffic datasets, demonstrating that the proposed MGTTP model achieves superior prediction performance over all compared mainstream baseline models across all key evaluation metrics, validating the effectiveness of each core component.

The remainder of this paper is organized as follows: Section 2 provides a review of related work; Section 3 details the methodology underlying the MGTTP model; Section 4 discusses experimental results; and Section 5 concludes with a summary and directions for future research.

2. Related work

2.1. Traffic flow forecasting based on graph neural networks

Early spatial modeling approaches predominantly employed convolutional neural networks (CNNs) to process gridded traffic data. However, these methods encountered inherent difficulties when applied to the non-Euclidean topology of real-world road networks. The advent of graph neural networks (GNNs) provides a robust framework to address this challenge. Notably, DCRNN, introduced by Li *et al.*,[7] integrates diffusion convolution with gated recurrent units to capture spatial dependencies on graph-structured data. Similarly, STGCN, proposed by Yu *et al.*,[3] models spatio-temporal correlations through the combination of graph convolution and temporal convolution operations. These seminal contributions underscore the efficacy of GNN-based methodologies in traffic flow forecasting.

Nonetheless, the majority of early GNN models rely on predefined static graphs constructed primarily on the basis of physical proximity, thereby constraining their capacity to represent non-local spatial dependencies. To mitigate this limitation, subsequent research has investigated adaptive graph construction strategies. For instance, Graph WaveNet, developed by Wu *et al.*,[13] employs learnable node embeddings to generate an adaptive adjacency matrix, while AGCRN, proposed by Bai *et al.*,[14] uncovers implicit node relationships in a fully data-driven manner. Although these adaptive approaches enhance model flexibility, the resultant graph structures often lack explicit physical interpretability.

2.2. Multi-graph fusion methods

In light of the limitations associated with single-graph structures, recent research has progressively shifted towards multi-graph fusion techniques. For example, ST-MGCN, introduced by Geng *et al.*,[9] integrates a proximity graph with a point-of-interest (POI) similarity graph. Similarly, TMGCN,

proposed by Lv *et al.*,[10] simultaneously models adjacency, functional similarity, and traffic connectivity among regions. These studies collectively highlight the promise of multi-graph fusion in capturing complex spatial dependencies. Nonetheless, existing approaches exhibit several inherent limitations: the dimensionality of fused graphs remains relatively constrained; fusion strategies commonly employ simplistic operations such as weighted summation or feature concatenation, which are essentially static fusion approaches; they lack adaptive mechanisms to dynamically adjust fusion weights according to the temporal dynamics of individual nodes; and critically, there is no interaction between spatial modeling and temporal features.

2.3. Spatio-temporal modeling based on attention mechanisms

In recent years, attention mechanisms have been extensively employed in spatio-temporal prediction tasks. The transformer architecture[15] has demonstrated superior capability in capturing long-range temporal dependencies via its self-attention mechanism. Xu *et al.*[16] applied the transformer framework to traffic forecasting, empirically validating its effectiveness in modeling long-term temporal dynamics. However, most existing attention-based methods predominantly follow the serial pipeline of "spatial modeling, temporal modeling, and feature fusion", without explicitly enabling bidirectional interaction and collaborative optimization between spatio-temporal features.

In contrast to these approaches, the bidirectional spatio-temporal interaction mechanism proposed herein not only integrates richer and more comprehensive graph information but also fosters deep, reciprocal interactions between spatial and temporal features. This design enables coupled spatio-temporal representation learning that transcends simplistic feature fusion strategies.

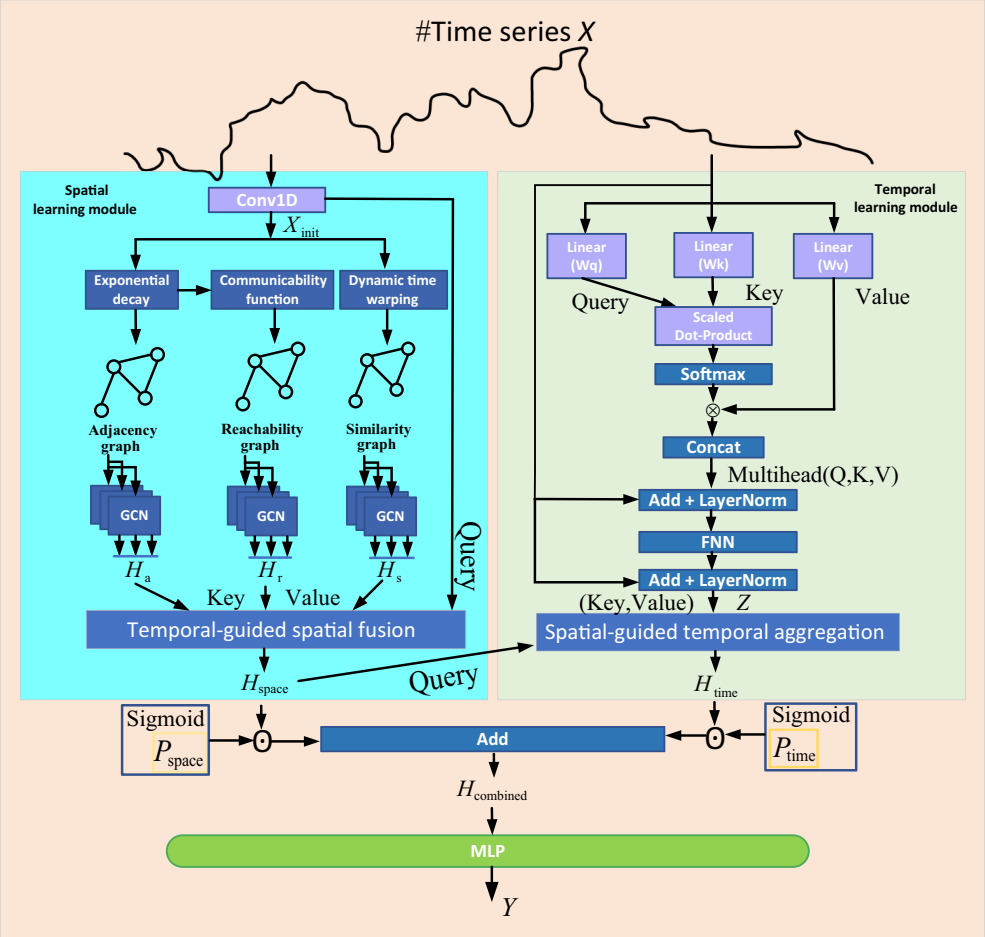


Fig. 1. Overall architecture of MGTTP with bidirectional spatio-temporal interaction.

2.4. Problem definition

Let $\boldsymbol{X} \in \mathbb{R}^{N\times T\times F}$ denote the historical traffic flow data, where N represents the number of nodes (i.e., traffic detectors) within the traffic network, T corresponds to the total number of historical time steps with a fixed input window size of $T = 12$ (each time step is 5 minutes, corresponding to 1 hour of historical data), and F indicates the feature dimension per time step. The raw input features include traffic flow, average speed, and

average occupancy; however, only the traffic flow value is used for prediction in this work, hence $F = 1$. The objective is to predict the future traffic flow $\boldsymbol{Y} \in \mathbb{R}^{N\times T'\times F}$ over the subsequent $T' = 12$ future time steps (corresponding to 1 hour of future data).

2.5. Overall model architecture

The MGTTP model is grounded in the core principle of bidirectional spatio-temporal closed-loop interaction and comprises four essential components: (1) initial temporal feature extraction, which supplies temporal context to guide subsequent spatial feature learning; (2) multi-view spatial representation learning, involving the construction and fusion of feature representations derived from multiple spatial graphs; (3) long-term temporal dependency modeling, which leverages the transformer architecture to capture complex temporal patterns across extended time horizons; and (4) bidirectional spatio-temporal interaction fusion, facilitating deep reciprocal interaction between spatial and temporal features to generate the final prediction. Figure 1 provides an overview of MGTTP, integrating multi-graph spatial learning with bidirectional spatio-temporal interaction.

2.6. Initial temporal feature extraction

To effectively capture the temporal dynamics inherent to each node within the spatial feature extraction process, we begin by deriving an initial temporal feature vector for each individual node. Given the historical traffic flow matrix for node i, denoted as $\boldsymbol{X}_i \in \mathbb{R}^{T\times F}$, a one-dimensional convolutional neural network (1D CNN) is applied

$$x_{\text{init},i} = \text{Conv1D}(\boldsymbol{X}_i) \in \mathbb{R}^{d'}, \tag{1}$$

where $\text{Conv1D}(\cdot)$ denotes the 1D convolution operator and d' represents the dimensionality of the convolutional output feature. The complete hyperparameters of the 1D convolution are set as: kernel size $= 3$, stride $= 1$, padding $= 1$, which ensures the output sequence length is consistent with the input, and the output feature dimension $d' = 64$. This vector encapsulates the essential temporal dynamic characteristics of node i and constitutes the key input for the subsequent time-guided spatial feature fusion.

2.7. Multi-perspective spatial representation learning

2.7.1. Multi-graph construction

(i) Adjacency graph construction The adjacency graph $G_a(V, E, \boldsymbol{W}_a)$ is employed to characterize local spatial dependencies among nodes within the traffic network. An edge $(i, j) \in E$ is established if nodes i and j are directly connected by a road segment. The corresponding edge weight is defined by an exponential decay function dependent on the physical distance d_{ij} between nodes:

$$\boldsymbol{W}_a(i, j) = \mathrm{e}^{-\alpha d_{ij}}, \tag{2}$$

where α denotes the attenuation coefficient, which physically represents the strength of distance decay and governs the influence range of local adjacency relationships. This parameter is determined via grid search on the validation set and is fixed at 0.1 throughout this work. This setting achieves an optimal balance between capturing local correlations and filtering out noise. The optimality of this value has been verified through the hyperparameter ablation experiment detailed in Subsection 4.5.2, and the parameter remains fixed across all datasets.

(ii) Reachability graph construction The reachability graph $G_r(V, E, W_r)$ is designed to capture path dependencies among nodes from a global perspective, thereby accounting for the indirect influence propagated through multi-hop connections. In practical traffic networks, certain intersections may lack direct geographic adjacency yet form potential traffic propagation pathways via intermediate nodes. For instance, as illustrated in Fig. 2, although nodes A and F are not directly connected by a road segment, congestion occurring at node A can indirectly impact traffic conditions at node F through other intermediary nodes.

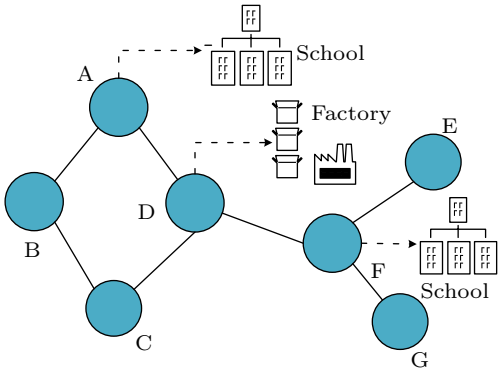


Fig. 2. An example of a toy urban road network.

To effectively capture the indirect propagation relationships between nodes, we employ the communicability function to model global reachability. Originally introduced by Estrada *et al.*,[17] the communicability function quantifies the capacity for information transmission between two nodes within a complex network by considering all possible paths, including multi-hop connections. Concretely, it represents the cumulative influence between nodes as the matrix exponential of the adjacency matrix, thereby assigning greater weight to shorter paths. Mathematically, the matrix exponential of the adjacency matrix can be expanded as a Taylor series

$$\mathrm{e}^{\boldsymbol{W}_a} = \sum_{k=0}^{\infty} \frac{\boldsymbol{W}_a^k}{k!} = \boldsymbol{I} + \boldsymbol{W}_a + \frac{\boldsymbol{W}_a^2}{2!} + \frac{\boldsymbol{W}_a^3}{3!} + \cdots, \tag{3}$$

where the term with $k = 0$ corresponds to the identity matrix $\boldsymbol{I}$ (node self-loops, 0-hop paths), the term with $k = 1$ corresponds to $\boldsymbol{W}_a$ (direct node connections, 1-hop paths), and the terms with $k \geq 2$ correspond to multi-hop indirect paths with two or more hops.

Building upon this idea, we define the reachability weight matrix $\boldsymbol{W}_r$ from the adjacency weight matrix $\boldsymbol{W}_a$ associated with the underlying graph

$$\boldsymbol{W}_r = \mathrm{e}^{\boldsymbol{W}_a} - \boldsymbol{W}_a - \boldsymbol{I}. \tag{4}$$

Here, e^{W_a} represents the matrix exponential of W_a, serving as the communicability function within the graph. This function can be interpreted as a weighted summation over all possible paths between nodes, thereby encapsulating both direct and indirect connectivity. The identity matrix $\boldsymbol{I}$ is subtracted to remove trivial self-loops; the subsequent subtraction of $\boldsymbol{W}_a$ eliminates the information of direct 1-hop connections, which are exclusively modeled by the adjacency graph. This design ensures that $\boldsymbol{W}_r$ only retains the cumulative influence of all indirect paths with a length ≥ 2 between nodes, realizing complete functional decoupling between the adjacency and reachability graphs. The adjacency graph exclusively models 1-hop direct local connections, while the reachability graph focuses on multi-hop global path propagation, reducing feature redundancy in multi-graph fusion. In real-world traffic scenarios, 1-hop congestion effects are captured by the adjacency graph, while cross-node indirect propagation effects modeled by the reachability graph serve as effective supplements.

To improve computational efficiency and highlight valid paths, a threshold ε_r is applied to $\boldsymbol{W}_r$ for sparsification. For all datasets in this study, the sparsity threshold ε_r is uniformly set to 10^{-5}, representing the lower effective weight bound of multi-hop paths. Its value is determined via grid search on the validation set and remains fixed across all datasets, ensuring computational control while preserving core multi-hop paths.

For computational complexity and numerical stability, we implement:

(i) Computational complexity optimization: The matrix exponential $\mathrm{e}^{\boldsymbol{W}_a}$ is computed using the industry-standard Scaling and Squaring algorithm, with time complexity $O(N^3)$, where N is the number of nodes. Combined with sparsity thresholding, this approach reduces computational overhead while maintaining accuracy, rendering graph convolution complexity comparable to that of the sparse adjacency matrix. This allows scalability to large-scale networks, such as the PEMS07 dataset.

(ii) Numerical stability control: Prior to computing the matrix exponential, spectral normalization is applied to W_a to constrain its spectral radius below 1, avoiding numerical explosion. Double-precision floating-point arithmetic further ensures stability for large-scale graphs.

(iii) Similarity graph construction: The similarity graph captures functional relationships between nodes that, while not necessarily spatially proximal, exhibit analogous traffic patterns. For instance, intersections in different regions near schools may share similar commuting patterns. To quantify these similarities, we employ the dynamic time warping (DTW) algorithm, denoted as $\mathrm{DTW}(\cdot,\cdot)$, which calculates the optimal nonlinear alignment distance between two time series.[18] Unlike point-wise metrics such as Euclidean distance, DTW mitigates sensitivity to temporal shifts and scaling, better capturing the dynamic shapes of traffic flow.

Formally, given two time series $A = (a_1, a_2, \ldots, a_n)$ and $B = (b_1, b_2, \ldots, b_m)$, DTW constructs a cost matrix and applies dynamic programming to find the minimal cumulative cost path from $(1,1)$ to (n,m). For efficiency, we adopt FastDTW,[19] which approximates DTW with near-linear complexity by constraining the search space of the warping path, e.g., via the Sakoe–Chiba band.

For nodes i and j, with daily traffic sequences $\boldsymbol{X}_{i,k}$ and $\boldsymbol{X}_{j,k}$ for day k, the alignment distance is computed via FastDTW. The average distance over T days is

$$\mathrm{DTW}_{\mathrm{avg}}(i,j) = \frac{1}{T}\sum_{k=1}^{T} \mathrm{FastDTW}(\boldsymbol{X}_{i,k}, \boldsymbol{X}_{j,k}). \tag{5}$$

This average distance is transformed into a similarity weight via exponential decay:

$$W_s(i,j) = \mathrm{e}^{-\mathrm{DTW}_{\mathrm{avg}}(i,j)}. \tag{6}$$

Graph sparsification retains edges exceeding a threshold ε_s, set to 10^{-3} for all datasets. This parameter represents the effective lower bound of functional similarity and is fixed via grid search, ensuring a balance between capturing cross-regional similarity and controlling graph sparsity.

The core hyperparameters of the three graphs are fixed via grid search to ensure clear physical meaning, avoid semantic ambiguity, and enhance generalization stability. Fixed thresholds also support the reproducibility and practical engineering deployment of traffic flow forecasting models.

2.7.2. Graph convolution feature extraction

Following the construction of the adjacency graph G_{a}, reachability graph G_{r}, and similarity graph G_{s}, graph convolutional networks (GCNs) are employed to extract spatial dependency information embedded within these graphs from the input features. Each GCN utilizes the initial temporal feature matrix $x_{\mathrm{init},i}$ (Subsection 3.3) as input. For a given graph $G_t = (V, E, W_t)$, where $t \in \{\mathrm{a},\mathrm{r},\mathrm{s}\}$ denotes the adjacency, reachability, and similarity graphs respectively, normalized graph convolution operations are applied

$$\hat{\boldsymbol{W}}_t = \boldsymbol{D}_t^{-\frac{1}{2}}(\boldsymbol{W}_t + \boldsymbol{I})\boldsymbol{D}_t^{-\frac{1}{2}}, \tag{7}$$

where $\boldsymbol{D}_t$ is the degree matrix corresponding to $\boldsymbol{W}_t$, and $\boldsymbol{I}$ is the identity matrix to include self-loops. A multi-layer GCN then extracts spatial features. The graph convolution at layer l is formulated as

$$\boldsymbol{H}_t^{(l+1)} = \sigma\left(\hat{\boldsymbol{W}}_t \boldsymbol{H}_t^{(l)} \Theta_t^{(l)}\right), \tag{8}$$

where $\Theta_t^{(l)}$ denotes the trainable weight matrix at layer l, and $\sigma(\cdot)$ represents a nonlinear activation function. After L layers, the spatial representations $\boldsymbol{H}_t \in \mathbb{R}^{N\times d_s}$ are obtained, capturing local spatial dependencies ($\boldsymbol{H}_a$), global path dependencies ($\boldsymbol{H}_r$), and functional similarity dependencies ($\boldsymbol{H}_s$) among the nodes.

2.7.3. Time-guided spatial feature fusion

Given the distinct temporal dynamics of nodes, each node may rely differently on the three spatial relationships under varying traffic conditions. Unlike transformer-based baselines such as ST-transformer and ST-LLM, which use mean pooling, fixed-step extraction, or independent temporal modeling, and only perform static multi-graph feature fusion, we design a temporal-guided attention mechanism to dynamically fuse the three types of spatial features. This mechanism is the first core component of our bidirectional spatio-temporal interaction design, allowing temporal features to guide spatial modeling and breaking the decoupling between spatial and temporal modules in existing methods.

For each node i, its initial temporal feature $x_{\text{init},i}$ serves as the query, while the corresponding spatial features $h_{i,t}$ act as keys and values. The attention weight $\alpha_{i,t}$ for graph $t \in \{a, r, s\}$ is computed as

$$\alpha_{i,t} = \frac{\exp\left((\boldsymbol{W}_q x_{\text{init},i})^{\mathrm{T}}(\boldsymbol{W}_k h_{i,t})\right)}{\sum_{t'\in\{a,r,s\}} \exp\left((\boldsymbol{W}_q x_{\text{init},i})^{\mathrm{T}}(\boldsymbol{W}_k h_{i,t'})\right)}, \tag{9}$$

where $\boldsymbol{W}_q$ and $\boldsymbol{W}_k$ denote learnable projection matrices. The fused spatial feature for node i is obtained via a weighted sum

$$h_{\text{space},i} = \sum_{t\in\{a,r,s\}} \alpha_{i,t} h_{i,t}. \tag{10}$$

Collectively, the fused spatial feature matrix $\boldsymbol{H}_{\text{space}} \in \mathbb{R}^{N\times d}$ is constructed by aggregating these node-level representations. This mechanism enables the model to adaptively adjust the importance of different spatial relations based on real-time temporal dynamics of each node, rather than using a fixed fusion strategy. The fused spatial features will subsequently serve as the query vector for the spatial-guided temporal aggregation module (Subsection 3.5.2), forming the second half of our bidirectional spatio-temporal interaction closed loop.

2.8. Long-term temporal dependency modeling

2.8.1. Time encoding based on transformer

To capture temporal dependencies across multiple scales — such as daily peaks, weekly cycles, and irregular holiday patterns — we employ a multi-layer transformer encoder to process the historical sequence $\boldsymbol{X}_i \in \mathbb{R}^{T\times F}$ for each node i. At layer l, temporal dependencies are modeled via a multi-head attention mechanism

$$\text{MultiHead}(Q,K,V) = \text{Concat}(\text{head}_1,\ldots,\text{head}_h)\boldsymbol{W}_o, \tag{11}$$

where each attention head is computed as

$$\text{head}_i = \text{Attention}(Q\boldsymbol{W}_i^Q, K\boldsymbol{W}_i^K, V\boldsymbol{W}_i^V), \tag{12}$$

and the scaled dot-product attention is defined by

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_h}}\right)V. \tag{13}$$

The output of each attention layer is followed by a residual connection and a position-wise feed-forward network (FFN)

$$\begin{aligned}\boldsymbol{Z}_i^{(l)} = \text{LayerNorm}\Big\{ \boldsymbol{Z}_i^{(l-1)} + \text{FFN}\Big[\text{LayerNorm}\big(\boldsymbol{Z}_i^{(l-1)} \\ +\text{MultiHead}(Q,K,V)\big)\Big]\Big\}.\end{aligned} \tag{14}$$

After passing through L transformer layers, the final temporal feature sequence $\boldsymbol{Z}_i^{(L)} \in \mathbb{R}^{T\times d_{\text{model}}}$ is obtained, effectively encoding multi-scale temporal patterns.

2.8.2. Spatially guided temporal feature aggregation

Traffic temporal patterns are often strongly correlated with the spatial attributes of nodes. For example, nodes in commercial districts exhibit weekday peaks, whereas residential nodes may emphasize weekends and nighttime traffic. Motivated by this observation, we propose a spatially guided temporal feature aggregation mechanism that dynamically selects and weights historical time intervals based on the spatial features learned in Subsection 3.4.3.

Formally, for node i, let $h_{\text{space},i}$ denote the spatial feature obtained via time-guided fusion, and let the temporal feature sequence from the transformer encoder be $\boldsymbol{Z}_i^{(L)} = [z_{i,1}, z_{i,2}, \ldots, z_{i,T}] \in \mathbb{R}^{T\times d}$. We employ a cross-modal attention mechanism where the spatial feature acts as the query and the temporal features as the keys and values

$$\beta_{i,t} = \frac{\exp\left((\boldsymbol{W}_q' h_{\text{space},i})^{\mathrm{T}}(\boldsymbol{W}_k' z_{i,t})\right)}{\sum_{t'=1}^{T} \exp\left((\boldsymbol{W}_q' h_{\text{space},i})^{\mathrm{T}}(\boldsymbol{W}_k' z_{i,t'})\right)}. \tag{15}$$

The aggregated temporal feature for node i is then

$$h_{\text{time},i} = \sum_{t=1}^{T} \beta_{i,t} \cdot z_{i,t}, \tag{16}$$

where $\boldsymbol{W}_q'$ and $\boldsymbol{W}_k'$ are learnable projection matrices.

This mechanism enables the model to adaptively extract the most relevant temporal patterns conditioned on the spatial context of each node. By assembling the temporal features of all nodes, the final temporal representation matrix $\boldsymbol{H}_{\text{time}} \in \mathbb{R}^{N\times d}$ is produced.

Importantly, the spatial features from the time-guided spatial fusion module serve as the query for this spatial-guided temporal aggregation, forming a closed-loop bidirectional spatio-temporal interaction. This design fundamentally differs from conventional two-step serial attention paradigms, achieving deep coupling between spatial and temporal representations.

2.9. Final prediction

2.9.1. Adaptive gating fusion mechanism

Following the bidirectional spatio-temporal interaction, we obtain the coupled spatial and temporal feature matrices $\boldsymbol{H}_{\text{space}}$ and $\boldsymbol{H}_{\text{time}}$, respectively. The relative importance of these features may vary across traffic conditions, forecasting horizons, and node types. For instance, spatial propagation effects dominate during peak hours, while historical temporal patterns may be more influential during off-peak intervals.

To account for this variability, we design a parameterized gating fusion mechanism that adaptively modulates the contribution of spatial and temporal features at the feature-dimension level. Formally, we introduce two learnable gating matrices $\boldsymbol{P}_{\text{space}}$ and $\boldsymbol{P}_{\text{time}} \in \mathbb{R}^{N\times d}$ and compute the fused features as

$$\boldsymbol{H}_{\text{combined}} = \big(\boldsymbol{H}_{\text{space}} \odot \sigma(\boldsymbol{P}_{\text{space}})\big) + \big(\boldsymbol{H}_{\text{time}} \odot \sigma(\boldsymbol{P}_{\text{time}})\big), \tag{17}$$

where $\sigma(\cdot)$ denotes the sigmoid function, constraining gating weights to $[0,1]$, and $\odot$ represents element-wise multiplication. Through end-to-end training, the gating parameters learn to optimally balance the contributions of spatial and temporal features for each node and feature dimension, improving adaptability and predictive accuracy.

2.9.2. Prediction output and loss function design

The fused spatio-temporal feature $\boldsymbol{H}_{\text{combined}}$ is subsequently mapped to the prediction space via a multi-layer perceptron (MLP) to forecast traffic flow over the upcoming T' time steps

$$\hat{\boldsymbol{Y}}_i = \text{MLP}(h_{\text{combined},i}) \in \mathbb{R}^{T'\times F}. \tag{18}$$

Given the characteristics of traffic flow forecasting, the conventional mean squared error (MSE) loss treats all time intervals equally, neglecting the critical importance of accuracy during peak traffic periods. To mitigate this, we introduce a peak-aware adaptive weighted loss

$$\mathcal{L} = \frac{1}{N\cdot T'\cdot F}\sum_{i=1}^{N}\sum_{t=1}^{T'}\sum_{f=1}^{F} w_{i,t,f}\cdot(\hat{\boldsymbol{Y}}_{i,t,f} - \boldsymbol{Y}_{i,t,f})^2, \tag{19}$$

where the adaptive weight $w_{i,t,f}$ is defined as

$$w_{i,t,f} = 1 + \rho\cdot\text{ReLU}\Big(\frac{Y_{i,t,f}-\mu_i}{\sigma_i} - \tau\Big). \tag{20}$$

Here, μ_i and σ_i are the historical mean and standard deviation of traffic flow at node i, ensuring cross-node comparability; τ is a peak threshold controlling the onset of additional weighting; ρ is the peak weight coefficient controlling the amplification for high-demand periods. In this study, $\tau = 1.0$ and $\rho = 0.5$ are determined via grid search on the validation set and fixed across all datasets. This design follows conventions in spatio-temporal forecasting and is inspired by the focusing parameter γ in focal loss,[20] ensuring training stability and interpretability. The ReLU function guarantees non-negativity of additional weights.

3. Experiment

3.1. Dataset

We evaluated the proposed MGTTP model on four public traffic network datasets: PEMS03, PEMS04, PEMS07, and PEMS08. These datasets were originally compiled and released by Song *et al.* (2020),[8] and collected from traffic monitoring systems in four distinct regions of California, USA.

All data are derived from the Performance Measurement System (PeMS) operated by the California Department of Transportation (Caltrans) and are aggregated at 5-minute intervals, yielding 288 time points per working day. Prior to modeling, all input features were standardized using Z-score normalization.

Detailed statistics of these datasets are summarized in Table 1. For experimental evaluation, each dataset is chronologically partitioned into training, validation, and test sets in a 6:2:2 ratio, without any random shuffling. This chronological split strictly adheres to the standard protocol for time-series forecasting, ensuring that future data are never used to inform past predictions.

Table 1. Description and statistical summary of the traffic datasets.

Datasets	Nodes	Edges	TimeSteps	MissingRatio
PEMS03	358	547	26208	0.672%
PEMS04	307	340	16992	3.182%
PEMS07	883	866	28224	0.452%
PEMS08	170	295	17856	0.696%

3.2. Baseline methods

We evaluated the proposed MGTTP model against a suite of mainstream representative baseline methods commonly employed in traffic flow forecasting. All baseline models were configured with core architecture parameters strictly following the optimal settings reported in their original publications, and uniformly adapted to the 12-step input and 12-step output prediction window used throughout this work.

FC-LSTM: Fully connected long short-term memory network, a classical recurrent architecture capable of modeling long-term temporal dependencies in time series data (Sutskever *et al.*, 2014).[21]

DCRNN: Diffusion convolutional recurrent neural network, integrating diffusion convolution within gated recurrent units under an encoder-decoder framework to capture spatial dependencies in traffic networks (Li *et al.*, 2017).[7]

STGCN: Spatio-temporal graph convolutional network, combining graph convolution with one-dimensional temporal convolution to jointly extract spatial and temporal features from traffic data (Yu *et al.*, 2017).[3]

ASTGCN(R): Attention-based spatio-temporal graph convolutional network, enhancing modeling capability via spatio-temporal attention mechanisms. We adopt the variant that models only recent periodic components for fairness (Guo *et al.*, 2019).[22]

Graph WaveNet: A graph neural network framework incorporating adaptive adjacency matrices alongside dilated one-dimensional convolutions to automatically learn spatial dependencies (Wu *et al.*, 2019).[13]

STSGCN: Spatial-temporal synchronous graph convolutional network, synchronously modeling spatial and temporal relationships via local spatio-temporal subgraph modules (Song *et al.*, 2020).[9]

STGODE: Spatio-temporal graph neural ordinary differential equation, introducing continuous graph neural networks to capture dynamic spatio-temporal evolution in traffic forecasting (Fang *et al.*, 2021).[23]

ST-LLM: Spatio-temporal large language model, treating each location's time step as a token, learning spatio-temporal embeddings for spatial positions and global temporal patterns, and fusing them through convolutional operations with a partially frozen attention mechanism to adapt large language models for global spatio-temporal dependency modeling (Liu *et al.*, 2024).[12]

3.3. Experimental setup

To ensure consistency, reproducibility, and fairness, all experiments maintained identical data preprocessing pipelines, dataset partitioning strategies, input/output window settings, training environments, and hyperparameter tuning budgets across MGTTP and all baseline models.

All models were implemented using the PyTorch framework and trained on hardware consisting of an Intel(R) Xeon(R) Gold 6133 CPU @ 2.50 GHz and an NVIDIA GeForce RTX 3090 GPU. During training, hyperparameters were consistent across all models: the Adam optimizer was used with an initial learning rate of 0.001, a cosine annealing learning rate scheduler was applied, the batch size was set to 32, and the maximum number of training epochs was 100. Early stopping was employed, terminating training if the validation RMSE did not decrease for 10 consecutive epochs.

The core architectural hyperparameters of MGTTP are summarized in Table 2. For transparency and fairness, the main architectural settings of four strong baseline models are presented in Table 3, strictly following their original optimal configurations and adapted to the 12-step input/output prediction window. The remaining baseline models (FC-LSTM, DCRNN, STGCN, ASTGCN) adhere to the above hyperparameter tuning rules.

Table 2. Summary of core hyperparameters of the MGTTP model.

Module category	Hyperparameter name	Symbol	Value
Initial temporal feature extraction module	Output feature dimension of 1D convolution	d'	64
	1D convolution kernel size/stride/padding	–	3/1/1
Multi-view spatial feature extraction module (GCN)	Number of graph convolutional layers	L	2
	Hidden dimension of graph convolution	d_s	64
Transformer temporal encoding module	Number of transformer encoder layers	L_{trans}	3
	Number of self-attention heads	h	8
	Model hidden dimension	d_{model}	64
	Intermediate dimension of feed-forward network	d_{ffn}	256
	Dropout Rate	p_{drop}	0.1

Table 3. Key architecture settings of core strong baseline models.

Baseline model	Core architecture settings	Fixed key hyperparameters
Graph WaveNet	2 stacked spatio-temporal convolution blocks; 8 dilated causal convolution layers per block	Dilation factor sequence: [1,2,1,2,1,2,1,2]; adaptive graph embedding dimension = 10; hiddendimension = 64; diffusion step $K = 2$
STSGCN	3 spatio-temporal synchronous convolution modules; 3 graph convolutional layers per module	Hidden dimension = 64; input window = 12; prediction window = 12
STGODE	TCN–ODE-TCN sandwich structure; tensor-based ODE solver	TCN hidden dimensions = $[64, 32, 64]$; 3 STGODE blocks; ODE step size = 1; adjacency threshold $\sigma = 10$, $\varepsilon = 0.5$
ST-LLM	6-layer GPT2-based transformer; partially frozen attention strategy	Spatio-temporal embedding dimension = 64; input window = 12; prediction window = 12

For performance evaluation, three standard metrics are adopted: mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean squared error (RMSE). All baseline models are based on their official open-source implementations, and the reported results represent the mean of three independent runs with different random seeds, ensuring stability and reproducibility by mitigating randomness.

3.4. Comparative analysis of model performance

To evaluate the effectiveness of the proposed MGTTP model, we conducted a comprehensive quantitative comparison against all selected baseline methods on the four PEMS datasets, using the evaluation metrics defined in Subsection 4.3. The results, summarized in Table 4, consistently demonstrate that MGTTP achieves state-of-the-art prediction performance across all metrics and datasets.

Specifically, MGTTP attains MAE values of 15.49, 19.37, 21.62, and 15.50, and RMSE values of 26.83, 31.42, 34.85, and 24.47 on PEMS03, PEMS04, PEMS07, and PEMS08, respectively, outperforming all comparative baselines, including the recent ST-LLM, on most evaluation metrics.

Table 4. Performance comparison of the proposed MGTTP model and baseline methods on the PEMS datasets.

Datasets	Metric	FC-LSTM	DCRNN	STGCN	ASTGCN	Graph WaveNet	STSGCN	STGODE	ST-LLM	MGTTP
PEMS03	MAE	21.33 ± 0.24	18.18 ± 0.15	17.49 ± 0.46	17.69 ± 1.43	19.85 ± 0.03	17.48 ± 0.15	16.05 ± 0.15	15.78 ± 0.13	15.49 ± 0.11
	MAPE(%)	23.33 ± 4.23	18.91 ± 0.82	17.15 ± 0.45	19.40 ± 2.24	19.31 ± 0.49	16.78 ± 0.20	16.69 ± 0.20	15.42 ± 0.17	14.73 ± 0.15
	RMSE	35.11 ± 0.50	30.31 ± 0.25	30.12 ± 0.70	29.66 ± 1.68	32.94 ± 0.18	29.21 ± 0.56	27.84 ± 0.26	27.15 ± 0.23	26.83 ± 0.20
PEMS04	MAE	26.24 ± 0.20	24.70 ± 0.22	22.70 ± 0.64	22.93 ± 1.29	25.45 ± 0.03	21.19 ± 0.10	20.84 ± 0.18	19.89 ± 0.15	19.37 ± 0.12
	MAPE(%)	19.30 ± 0.40	17.12 ± 0.37	14.59 ± 0.21	16.56 ± 1.36	17.29 ± 0.24	13.90 ± 0.05	13.77 ± 0.22	13.24 ± 0.19	12.92 ± 0.18
	RMSE	40.49 ± 0.21	38.12 ± 0.26	35.55 ± 0.75	35.22 ± 1.90	39.70 ± 0.04	33.65 ± 0.20	32.82 ± 0.29	31.87 ± 0.25	31.42 ± 0.21
PEMS07	MAE	29.96 ± 0.42	25.30 ± 0.52	25.38 ± 0.49	28.05 ± 2.34	26.85 ± 0.05	24.26 ± 0.14	22.59 ± 0.20	21.89 ± 0.17	21.62 ± 0.14
	MAPE(%)	14.34 ± 0.53	11.66 ± 0.33	11.08 ± 0.18	13.92 ± 1.65	12.12 ± 0.41	10.21 ± 1.65	10.14 ± 0.24	9.47 ± 0.20	9.19 ± 0.17
	RMSE	43.94 ± 0.57	38.58 ± 0.70	38.78 ± 0.58	42.57 ± 3.31	42.78 ± 0.07	39.03 ± 0.27	37.54 ± 0.32	36.24 ± 0.28	34.85 ± 0.24
PEMS08	MAE	22.20 ± 0.18	17.86 ± 0.03	18.02 ± 0.14	18.61 ± 0.40	19.13 ± 0.08	17.13 ± 0.09	16.81 ± 0.16	15.94 ± 0.12	15.50 ± 0.09
	MAPE(%)	15.02 ± 0.59	11.45 ± 0.03	11.40 ± 0.10	13.08 ± 1.00	12.68 ± 0.57	10.96 ± 0.07	10.62 ± 0.20	10.18 ± 0.15	9.84 ± 0.11
	RMSE	33.06 ± 0.32	27.83 ± 0.05	27.83 ± 2.20	28.16 ± 0.48	31.05 ± 0.07	26.80 ± 0.18	25.97 ± 0.27	24.92 ± 0.22	24.47 ± 0.15

Compared with classical graph-based models such as DCRNN and STGCN, which integrate graph convolution with recurrent architectures, MGTTP achieves a relative improvement of 10% to 15% in RMSE across all datasets, highlighting the effectiveness of the multi-graph fusion strategy in capturing spatial heterogeneity. Notably, even against stronger baselines such as Graph WaveNet and STGODE, which leverage adaptive adjacency matrices and neural ordinary differential equations (ODEs) respectively, MGTTP maintains superior predictive performance. This advantage is primarily attributed to its richer spatial relation modeling and fine-grained bidirectional spatio-temporal attention mechanism, which enables more accurate traffic forecasting across diverse traffic conditions and datasets.

It is particularly noteworthy that MGTTP consistently outperforms the recently proposed ST-LLM method across all datasets. Although ST-LLM innovatively leverages large language models by treating time steps as tokens and designing spatio-temporal embeddings to capture global patterns, its general-purpose architecture lacks task-specific specialization for traffic prediction. In contrast, MGTTP is explicitly tailored for traffic flow forecasting, enabling more precise modeling of the unique spatio-temporal dynamics inherent in traffic data. For example, on PEMS04, MGTTP reduces MAE by 2.6% relative to ST-LLM (from 19.89 to 19.37), while on PEMS08, the improvement reaches 2.8% (from 15.94 to 15.50).

Compared with STGODE, the second-best performing model, MGTTP achieves notable MAE improvements of 7.1% on PEMS04 and 7.8% on PEMS08. In particular, MGTTP reduces MAPE to 9.19% and 9.84% on PEMS07 and PEMS08, respectively, surpassing all compared baselines and demonstrating superior accuracy in low-error scenarios. These consistent and significant performance gains underscore the effectiveness of the MGTTP architecture in capturing complex spatio-temporal dependencies, dynamics, and heterogeneity in traffic flow data.

In terms of stability, MGTTP exhibits smaller performance standard deviations than all baseline models across the datasets, indicating enhanced robustness under different random seeds.

To verify the statistical significance of the improvements, we conducted a one-tailed paired samples t-test between MGTTP and ST-LLM at a 95% confidence level. The test settings are as follows:

Paired objects: Prediction errors (MAE, RMSE, MAPE) of MGTTP and ST-LLM for each test sample across three independent repeated experiments (with different random seeds) were paired one-to-one.

Prediction horizons: The test was performed over all 12 prediction steps (5–60 min).

Hypotheses: The null hypothesis H_0 states that "there is no significant difference between the prediction errors of MGTTP and ST-LLM," while the one-tailed alternative hypothesis H_1 states that "the prediction error of MGTTP is significantly lower than that of ST-LLM." A p-value < 0.05 indicates statistical significance.

The p-values of the paired t-test are summarized in Table 5. All metrics yielded p-values below 0.05, leading to rejection of H_0. This confirms that MGTTP's performance improvements are statistically significant and not attributable to

random fluctuations.

Table 5. P-value results of the paired t-test between MGTTP and ST-LLM.

Dataset	MAE	MAPE	RMSE
PEMS03	0.023	0.031	0.018
PEMS04	0.017	0.024	0.021
PEMS07	0.028	0.019	0.025
PEMS08	0.015	0.020	0.016

3.5. Ablation study

To rigorously assess the effectiveness of each core component of the proposed MGTTP model, we conducted two categories of ablation experiments following the single-variable control principle: (i) Necessity ablation of core modules, which evaluates the performance contribution of each essential design. (ii) Optimality ablation of key hyperparameters, which verifies the validity of the selected core hyperparameter settings.

All ablation experiments were performed on two representative datasets, PEMS04 and PEMS08, while keeping the training environment and all non-pertinent hyperparameters fixed to ensure fair comparison.

3.5.1. Necessity ablation of core modules

The results of the necessary ablation experiments are summarized in Table 6, illustrating the contribution of each individual module to the overall predictive performance.

Table 6. Component analysis of the MGTTP model.

	PEMS04			PEMS08		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE
-H_{time}	20.64	15.54	33.18	17.15	12.87	26.42
-Adjacency graph	19.84	13.31	32.33	16.29	10.74	25.70
-Reachability graph	19.80	13.25	32.20	16.57	11.36	25.72
-I–W_{a}	19.76	13.22	32.18	16.31	10.76	25.64
-Similarity graph	19.79	13.02	32.21	16.28	10.73	25.65
-Adaptive gating fusion + Feature concatenation	19.94	13.58	32.56	16.74	10.81	25.78
-Peak-aware adaptive weighted loss + (MSE) loss	19.87	13.47	32.49	16.55	10.76	25.67
MGTTP	19.37	12.92	31.42	15.50	9.84	24.47

3.5.1.1. Effectiveness of the temporal feature extraction module

We first evaluate the contribution of the temporal feature extraction module. As shown in Table 6, the variant "-H_{time}", which omits the spatio-temporal attention mechanism and simply averages the transformer outputs at the final time step, exhibits notable performance degradation. On PEMS04, MAE increases from 19.37 to 20.64. These results confirm that the spatio-temporal attention mechanism effectively integrates spatial context when aggregating temporal features, enabling richer and more informative temporal representations.

3.5.1.2. Effectiveness of the multi-graph fusion strategy

To assess the contribution of the multi-graph fusion strategy, we conducted ablations by removing each graph structure — adjacency, reachability, or similarity — individually. Excluding any single graph consistently increases prediction error, indicating that the multi-graph fusion framework outperforms any two-graph combination. A comparison with the full MGTTP model (Table 4) shows that all ablated variants (-adjacency, -reachability, -similarity) degrade performance across all metrics. This highlights that adjacency, reachability, and similarity graphs capture complementary spatial dependencies essential for accurate traffic forecasting. Notably, the relative importance of each graph varies between datasets: on PEMS04, removing adjacency has the largest impact (MAE 19.84), whereas on PEMS08, omitting either adjacency or similarity yields significant degradation, reflecting differences in network topology and functional zoning.

3.5.1.3. Validity of the reachability graph design

We further validate the design choice in Eq. (3) by constructing an ablation variant $W_{\text{r}} = \text{e}^{W_{\text{a}}}$, retaining self-loops and the 1-hop adjacency term. This variant shows clear performance drops: on PEMS04, MAE rises from 19.37 to 19.76 and RMSE from 31.42 to 32.18; on PEMS08, MAE rises from 15.50 to 16.31 and RMSE from 24.47 to 25.64. Retaining I and W_{a} introduces redundant modeling between reachability and adjacency graphs, validating the necessity of the original functional decoupling in the multi-graph design.

3.5.1.4. Effectiveness of the adaptive gating fusion mechanism

The adaptive gating fusion mechanism is evaluated by replacing it with simple feature concatenation (– adaptive gating, + feature concatenation). As reported in Table 6, this ablation leads to substantial performance degradation: on PEMS04, MAE increases from 19.37 to 19.94; on PEMS08, from 15.50 to 16.74. These results demonstrate that the adaptive gating mechanism dynamically modulates the contributions of spatial and temporal features at both node and feature levels, adjusting to heterogeneous traffic scenarios. In contrast, fixed concatenation fails to capture the varying importance of spatio-temporal information, highlighting the gating module as critical for high-performance prediction.

3.5.1.5. Effectiveness of the peak-aware loss function

Finally, the peak-aware adaptive weighted loss function is evaluated by replacing it with the standard MSE (i.e., peak-aware loss vs. MSE). Table 6 shows clear performance deterioration: on PEMS04, RMSE increases from 31.42 to 32.49; on PEMS08, from 24.47 to 25.67, corresponding to a 4.90% relative increase. The peak-aware loss enhances accuracy during critical periods such as peak traffic hours by dynamically

weighting samples according to practical importance. Compared with the uniform MSE loss, it guides the model to focus on high-impact periods, supporting effective traffic management and practical deployment.

3.5.2. Optimality ablation of key hyperparameters

3.5.2.1. Optimality ablation of hyperparameters for multi-graph structures

In this set of experiments, we strictly adhere to the single-variable control principle: all other model parameters are fixed at their optimal values while only the core hyperparameters of the target graph are varied. This allows for a direct assessment of the optimality of the hyperparameter values adopted in this work. All hyperparameters were initially determined via grid search on the PEMS04 validation set and then fixed across all other datasets. The ablation results demonstrate the generalization stability of these fixed settings across similar traffic datasets.

(i) Decay coefficient of the adjacency graph

To verify the optimality of the adjacency graph decay coefficient, we conducted an ablation study varying its value while keeping all other hyperparameters unchanged. The results, summarized in Table 7, show that the selected decay coefficient yields the lowest prediction error. Deviating from this value leads to a gradual degradation in both MAE and RMSE, indicating that the chosen coefficient effectively balances the influence of neighboring nodes, capturing local spatial dependencies without over-smoothing or under-weighting adjacent information.

Table 7. Ablation results of the decay coefficient for the adjacency graph.

α value	PEMS04			PEMS08		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE
0.01	19.75	13.18	32.16	15.82	10.21	24.81
0.1	19.37	12.92	31.42	15.50	9.84	24.47
1.0	19.69	13.11	32.09	15.79	10.15	24.76

The experimental results indicate that the decay coefficient $\alpha = 0.1$ selected in this work is optimal. When $\alpha = 0.01$, the decay intensity is too weak; this allows correlations from overly distant nodes to persist and introduces noise interference. Conversely, when $\alpha = 1.0$, the decay intensity is excessively strong, retaining only the connections of very close neighboring nodes and discarding valid spatial information from adjacent road segments. The choice of $\alpha = 0.1$ effectively balances local correlation capture with noise filtering, achieving the best prediction performance.

(ii) Sparsity threshold of the reachability graph

To validate the optimal sparsity threshold ε_r for the reachability graph, we conducted an ablation study with varying threshold values, and the results are summarized in Table 8. The experiments show that $\varepsilon_r = 10^{-5}$ is the optimal value. A higher threshold, $\varepsilon_r = 10^{-4}$, is overly strict, filtering out many valid multi-hop paths and failing to fully capture global reachability correlations. In contrast, a lower threshold, $\varepsilon_r = 10^{-6}$, is too permissive, introducing numerous edges with extremely low weights, which increases computational overhead and the risk of overfitting. Setting $\varepsilon_r = 10^{-5}$ retains the core multi-hop paths while maintaining graph sparsity and computational efficiency.

Table 8. Ablation results of the sparsity threshold for the reachability graph.

ε_r value	PEMS04	PEMS04	PEMS04	PEMS08	PEMS08	PEMS08
	MAE	MAPE	RMSE	MAE	MAPE	RMSE
10^{-4}	19.71	13.14	32.12	15.77	10.13	24.74
10^{-5}	19.37	12.92	31.42	15.50	9.84	24.47
10^{-6}	19.63	13.07	32.03	15.71	10.09	24.70

(iii) Sparsity threshold of the similarity graph

To determine the optimal sparsity threshold ε_s for the similarity graph, we conducted an ablation study with different threshold values, and the results are summarized in Table 9. The experiments indicate that the selected value of $\varepsilon_s = 10^{-5}$ achieves the best balance. A higher threshold, $\varepsilon_s = 10^{-4}$, is excessively restrictive, removing many valid similarity-based edges and weakening the model's ability to capture functional correlations between nodes. Conversely, a lower threshold, $\varepsilon_s = 10^{-6}$, is overly permissive, introducing a large number of weak and potentially noisy connections, which increases computational cost and risks overfitting. Setting $\varepsilon_s = 10^{-5}$ effectively preserves meaningful functional similarity edges while maintaining sparsity and efficiency in the graph structure.

Table 9. Ablation results of the sparsity threshold for the similarity graph.

ε_s value	PEMS04	PEMS04	PEMS04	PEMS08	PEMS08	PEMS08
	MAE	MAPE	RMSE	MAE	MAPE	RMSE
10^{-2}	19.82	13.15	32.25	15.91	10.27	24.89
10^{-3}	19.37	12.92	31.42	15.50	9.84	24.47
10^{-4}	19.68	13.10	32.08	15.76	10.12	24.75

(iii) Sparsity threshold of the similarity graph

To determine the optimal sparsity threshold ε_s for the similarity graph, we conducted an ablation study with different threshold values. The result, summarized in Table 9, shows that the selected value $\varepsilon_s = 10^{-3}$ achieves the best trade-off. When $\varepsilon_s = 10^{-2}$, only node pairs with extremely high similarity are retained, which fails to fully capture cross-regional functional correlations. Conversely, $\varepsilon_s = 10^{-4}$ introduces many weak and redundant edges, causing noise interference. The choice of $\varepsilon_s = 10^{-3}$ effectively balances the extraction of meaningful functional similarity and the sparsity of the

graph, validating the scientific rationale of this hyperparameter setting.

3.5.2.2 Optimality ablation of hyperparameters for the peak-aware loss function

We further investigate the optimal values of τ and ρ in the peak-aware adaptive weighted loss function. A grid search experiment was performed on the PEMS04 validation set, and the results are summarized in Table 10. The findings confirm that the selected values of $\tau = 1.0$ and $\rho = 0.5$ achieve the best prediction performance, effectively emphasizing peak traffic periods while maintaining overall model stability and accuracy.

Table 10. Ablation results of hyperparameters for the peak-aware loss function (PEMS04).

$\tau\backslash\rho$	0.2	0.5	0.8
0.5	19.62/31.78	19.51/31.62	19.58/31.71
1.0	19.45/31.55	19.37/31.42	19.42/31.51
1.5	19.55/31.68	19.48/31.59	19.53/31.65

Note: Values in the table are presented in the format of MAE/RMSE.

The experimental results indicate that the selected hyperparameter combination $\tau = 1.0$ and $\rho = 0.5$ is optimal. When τ is too small or ρ is too large, the model excessively emphasizes extreme peak values, neglecting regular traffic patterns during off-peak periods. Conversely, if τ is too large or ρ too small, the peak-aware mechanism becomes ineffective, and prediction accuracy for peak hours cannot be substantially improved. These findings confirm the rationality of the adopted hyperparameter values.

3.5.3. Conclusion of ablation experiments

The ablation experiments systematically validate the rationality of the MGTTP model design. The necessity ablation demonstrates that all four core components — spatially-guided temporal aggregation, multi-graph fusion strategy, adaptive gating fusion mechanism, and peak-aware loss function — are indispensable for achieving high prediction performance. The optimality ablation further confirms the scientific validity of the key hyperparameter settings, providing rigorous quantitative support for the core conclusions of this study.

4. Conclusion

In this paper, we propose MGTTP, a multi-graph transformer model for traffic flow forecasting based on bidirectional spatio-temporal interaction. By adopting a bidirectional mechanism of time-guided spatial fusion and space-guided temporal aggregation, the model overcomes the limitations of conventional spatio-temporal separate modeling strategies. Simultaneously, through parallel fusion of adjacency, reachability, and similarity graphs, MGTTP effectively captures multi-dimensional spatial dependencies in traffic networks.

Extensive experiments on four PEMS datasets demonstrate that MGTTP outperforms all selected mainstream baseline models across key evaluation metrics, achieving statistically significant improvements over state-of-the-art methods. Ablation studies further verify that the bidirectional spatio-temporal interaction and multi-graph fusion modules play critical roles in capturing latent and complementary semantic information in traffic data. Both adaptive gating fusion and the peak-aware loss function are effective components for boosting predictive performance, and the individual modules exhibit strong complementary synergy.

Looking forward, although MGTTP achieves favorable predictive performance, real-world traffic systems are influenced by diverse exogenous factors such as weather, holidays, and special events, and cross-domain generalization remains a critical challenge. Future work will focus on two key extensions: (i) incorporating heterogeneous exogenous information as additional node attributes or via a global contextual vector through a dedicated information fusion module; and (ii) evaluating and optimizing cross-city generalization performance on datasets with varying road networks, traffic patterns, and data sources. We anticipate that these enhancements will further improve prediction accuracy, robustness to anomalies, interpretability, and practical deployment potential, advancing MGTTP toward more intelligent and adaptable traffic forecasting systems.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62176217), the Sichuan Provincial Science and Technology Program, China (Grant No. 2018RZ0081), the Major Science and Technology Special Project of Sichuan Province (Grant No. 2026YFTH0029), and the Fundamental Research Funds of China West Normal University (Grant No. 17E063).

References

[1] Liao Z F, Sun K, Liu W L, Yu Z W, Liu C G and Song Y C 2024 *Chin. Phys. B* **33** 078901
[2] Hamed M M, Al-Masaeid H R and Said Z M B 1995 *J. Transp. Eng.* **121** 249
[3] Yu B, Yin H and Zhu Z 2017 *IJCAI* 3634
[4] Yin X Y, Wu Z H, Wang S, Wang Y and Yu P S 2021 *IEEE Trans. Intell. Transp. Syst.* **23** 4927
[5] Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H and Zhang W 2021 *Proc. AAAI Conf. Artif. Intell.* **35** 11106
[6] Jiang W and Luo J 2022 *Expert Syst. Appl.* **207** 117921
[7] Li Y, Yu R, Shahabi C and Liu Y 2017 arXiv:1707.01926 [cs.LG]
[8] Song C, Lin Y, Guo S and Wan H 2020 *Proc. AAAI Conf. Artif. Intell.* **34** 914
[9] Geng X, Li Y, Wang L, Zhang L, Yang Q, Ye J and Liu Y 2019 *Proc. AAAI Conf. Artif. Intell.* **33** 3656
[10] Lv M, Hong Z, Chen L, Chen T, Zhu T and Ji Y 2020 *IEEE Trans. Intell. Transp. Syst.* **22** 3337
[11] Sherstinsky A 2020 *Phys. D* **404** 132306
[12] Liu C, Yang S, Xu Q, Wang Y, Zhao J and Wang J 2024 *Proc. IEEE Int. Conf. Mobile Data Manag.* 31

[13] Wu Z, Pan S, Long G, Jiang J and Zhang C 2019 arXiv:1906.00121 [cs.LG]
[14] Bai L, Yao L, Li C, Wang X and Wang C 2020 *Adv. Neural Inf. Process. Syst.* **33** 17804
[15] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser Ł and Polosukhin I 2017 *Adv. Neural Inf. Process. Syst.* **30** 5998
[16] Xu M, Dai W, Liu C, Gao X, Lin W, Qi G J and Xiong H 2020 arXiv prep 2020 arXiv:2001.02908 [eess.SP]
[17] Estrada E and Hatano N 2008 *Phys. Rev. E* **77** 036111
[18] Berndt D J and Clifford J 1994 *Proc. 3rd Int. Conf. Knowl. Discov. Data Min.* 359
[19] Salvador S and Chan P 2007 *Intell. Data Anal.* **11** 561
[20] Lin T Y, Goyal P, Girshick R, He K and Dollár P 2017 *Proc. IEEE Int. Conf. Comput. Vis.* 2980
[21] Sutskever I, Vinyals O and Le Q V 2014 *Adv. Neural Inf. Process. Syst.* **27** 3104
[22] Guo S, Lin Y, Feng N, Song C and Wan H 2019 *Proc. AAAI Conf. Artif. Intell.* **33** 922
[23] Fang Z, Long Q, Song G and Xie K 2021 *Proc. ACM SIGKDD Conf. Knowl. Discov. Data Min.* 364