

Article

An Operationally Interpretable Lag-Aware Directed Spatio-Temporal Graph Neural Network for Real-Time Freeway Traffic Forecasting

Yuanwei Guo ¹ , Junhao Lin ² , Zixuan Wang ² , Yutang Bi ²  and Hailiang Ye ^{2,*} 

¹ Guangzhou Institute of Internet of Things, Guangzhou 511462, China; csu20202020@163.com

² College of Sciences, China Jiliang University, Hangzhou 310018, China; linjh0412@163.com (J.L.); cur0327@163.com (Z.W.); chun13708421946@gmail.com (Y.B.)

* Correspondence: yehl@cjlu.edu.cn

Abstract

Reliable short-term freeway traffic forecasting is essential for proactive traffic management, including congestion warning, ramp metering support, and traveler information services. However, many existing forecasting models treat spatial dependencies as synchronous or weakly directional, limiting their ability to represent delayed upstream–downstream traffic propagation. This study proposes Traffic-DiMAGNet, a lightweight and interpretable lag-aware directed spatio-temporal graph neural network for real-time freeway flow forecasting. The model constructs a sparse directed sensor dependency graph by integrating physical road adjacency, training-set lead–lag traffic priors, and learnable source–target node embeddings. A lag-aware bidirectional propagation module then shifts inter-sensor messages according to estimated propagation delays, while directed random-walk normalization, directional gating, and multi-scale causal convolutions preserve asymmetric traffic semantics with low computational cost. Experiments on four Caltrans PeMS datasets show that Traffic-DiMAGNet consistently outperforms recurrent, diffusion-based, attention-based, and adaptive-graph baselines regarding MAE, RMSE, and MAPE values. The learned directed lag structures provide interpretable propagation information, and the lightweight architecture supports rolling 5 min forecasting, indicating practical potential for real-time freeway monitoring and proactive traffic management.

Keywords: freeway traffic forecasting; proactive traffic management; propagation delay; directed graph neural network; spatio-temporal graph learning; ramp metering support



Academic Editors: Tianwen Zhang and Jaeyoung Lee

Received: 7 July 2026

Revised: 6 August 2026

Accepted: 26 August 2026

Published: 30 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Short-term traffic flow forecasting is a fundamental task in intelligent transportation systems. Accurate forecasts over the next 15–60 min can support proactive freeway management, including congestion warning, ramp metering, variable speed control, dynamic route guidance, and traveler information services [1,2]. In these applications, traffic management agencies need not only point predictions of future flow, but also reliable indications of where demand growth, bottleneck activation, or congestion propagation may occur within a short operational time window. With the deployment of large-scale traffic sensing infrastructure, transportation agencies can continuously monitor traffic conditions over large freeway networks [3]. These sensor measurements provide rich information for data-driven forecasting models, but they also exhibit complex spatial interactions and rapidly changing temporal patterns [4].

Freeway traffic data differ from generic multivariate time series because they are strongly affected by corridor directionality and time-delayed propagation [5]. A traffic disturbance observed at one detector may influence other stations after several time intervals, and the direction and magnitude of this influence may change across traffic states. In addition, physically distant sensors can exhibit functional correlations due to recurrent commuting patterns, bottleneck spillbacks, or coordinated traffic control [6,7]. Therefore, relying only on a static road-network adjacency matrix may miss important non-local dependencies, whereas purely data-driven adaptive graphs may learn correlations that are difficult to interpret from a transportation perspective [8,9]. A practical traffic forecasting model should therefore jointly consider physical road connectivity, historical lead-lag relationships, adaptive sensor dependencies, and computational efficiency.

Spatio-temporal graph neural networks have become a major framework for traffic forecasting because they can jointly model temporal evolution and spatial dependencies over road networks. However, several limitations remain concerning how existing models construct spatial graphs and propagate traffic information. On the spatial side, most STGNNs operate on a predefined physical graph and model spatial dependencies using graph, Chebyshev, or diffusion convolutions [10–12]. Static graphs, however, cannot capture functionally correlated but physically distant nodes, and many models further simplify them as undirected or as single-direction propagation [13]. This simplification can hinder the modeling of asymmetric upstream–downstream effects that are central to freeway traffic propagation. Although adaptive graph learning methods such as MTGNN [14] and AGCRN [15] augment predefined topologies with learnable embeddings, they usually do not explicitly associate graph edges with traffic lead-lag delays.

Temporal modeling also needs to balance forecasting accuracy and computational efficiency. Recurrent models capture temporal dependencies but are difficult to parallelize, while attention-based models can be computationally expensive for long input sequences [16]. Temporal convolutional models improve efficiency through parallel computation and dilated kernels. However, fixed or single-scale temporal filters may be insufficient to simultaneously capture abrupt short-term fluctuations and smoother longer-term trends in freeway flow. This motivates a lightweight multi-scale temporal encoder that can enlarge the receptive field without introducing heavy attention or recurrent structures.

This paper proposes Traffic-DiMAGNet, a traffic-prior-constrained and lag-aware directed spatio-temporal graph neural network that explicitly addresses these spatial and temporal bottlenecks. It constructs a sparse directed fused graph by combining learnable source–target embeddings, road-network adjacency, and a training-set lead-lag traffic prior, and then performs lag-aware bidirectional propagation on the fused graph.

The main contributions of this study are summarized as follows:

- A traffic-prior-constrained sparse directed graph construction method is proposed for freeway sensor networks. It fuses learnable source–target node embeddings, road-network adjacency, and a training-set lead-lag traffic prior, while using graph sparsity and off-prior regularization to reduce implausible dependencies. This design allows the learned graph to capture both physical connectivity and non-local functional relationships that are relevant to freeway traffic propagation.
- A lag-aware bidirectional propagation module is developed to model delayed upstream and downstream traffic dependencies. Unlike synchronous graph diffusion, the proposed module shifts inter-sensor messages according to discrete edge delays estimated from lead-lag statistics, making the propagation process more consistent with the temporal evolution of freeway traffic disturbances and bottleneck effects.
- A direction-preserving propagation scheme is designed by combining directed random-walk normalization with feature-dependent directional gating. This scheme

preserves asymmetric source–target graph semantics, stabilizes multi-hop propagation, and adaptively balances forward and reverse traffic information under different traffic conditions.

- A lightweight multi-scale temporal convolutional encoder is integrated with the proposed spatial module for efficient short-term traffic forecasting. Experiments on four PeMS datasets evaluate forecasting accuracy, horizon-wise performance, ablation variants, computational cost, and learned graph interpretability, with emphasis on the model’s potential usefulness for real-time freeway monitoring and proactive traffic management.

2. Related Work

2.1. Spatio-Temporal Graph Neural Networks for Traffic Forecasting

Traffic flow forecasting has been widely studied with spatio-temporal graph neural networks because traffic sensors are naturally distributed over road networks and their observations exhibit both temporal dynamics and spatial correlations. Early STGNNs usually represent sensors as graph nodes and use predefined road-network structures to model spatial dependencies. STGCN combines graph convolution with temporal convolution to capture spatial and temporal patterns efficiently [10]. DCRNN models traffic diffusion on directed graphs through diffusion convolution and recurrent neural networks [11]. ASTGCN introduces attention mechanisms to capture dynamic spatial and temporal dependencies [12], while STSGCN constructs localized spatio-temporal graphs to model synchronous spatial–temporal correlations [17]. Graph WaveNet further integrates dilated temporal convolution with adaptive adjacency learning, improving both efficiency and spatial flexibility [18].

These methods have demonstrated the effectiveness of graph-based spatio-temporal modeling for traffic forecasting. However, many of them either rely heavily on predefined road-network topology or perform graph propagation in a synchronous manner. As a result, they may be insufficient for capturing non-local functional dependencies and delayed traffic propagation patterns that commonly occur in freeway networks [4,19,20]. This motivates more flexible graph learning and propagation mechanisms that can incorporate both transportation-domain priors and data-driven dependencies [21].

2.2. Adaptive Graph Learning for Traffic Forecasting

To overcome the limitations of fixed road-network graphs, adaptive graph learning has been introduced into traffic forecasting models. Graph WaveNet learns an adaptive adjacency matrix from trainable node embeddings to capture latent sensor dependencies beyond physical connectivity [18]. MTGNN constructs a graph structure from node embeddings and uses graph convolution to model multivariate temporal dependencies [14]. AGCRN further learns node-adaptive parameters and adaptive graph structures to improve sensor-specific representation learning [15]. Beyond these representative traffic forecasting models, fully connected and correlation-aware graph learning strategies have also been explored for general multivariate spatio-temporal time-series modeling [22,23]. In traffic forecasting, multi-spatio-temporal fusion networks and dynamic-adjacency recurrent models further demonstrate the importance of learning flexible and evolving spatial dependencies from data [24,25].

Although adaptive graph learning improves the flexibility of spatial dependency modeling, learned graphs can be difficult to interpret when they are generated mainly from latent embeddings. They may also introduce implausible edges that are weakly supported by road connectivity or historical traffic propagation patterns. Recent studies have therefore emphasized the need for more interpretable and transportation-aware

graph learning mechanisms [9]. Different from purely embedding-driven adaptive graphs, Traffic-DiMAGNet constructs a sparse directed fused graph by combining learnable source–target node embeddings, road-network adjacency, and a training-set lead–lag traffic prior. Sparsity and off-prior regularization are further used to reduce implausible dependencies and improve graph interpretability.

2.3. Direction-Aware and Lag-Aware Traffic Dependency Modeling

Directionality and time-delayed propagation are fundamental characteristics of freeway traffic. A disturbance observed at one detector may affect downstream or upstream detectors after several time intervals, and the propagation direction can vary with traffic states and corridor structures. To capture these complex dynamics, several existing models have incorporated directed spatial dependencies. For instance, DCRNN employs bidirectional diffusion processes on directed graphs to model traffic propagation [11]. Subsequent dynamic graph models and attention-based spatio-temporal graph neural networks (STGNNs) have further attempted to capture time-varying or asymmetric dependencies among sensors [7,26]. More recently, time-lag-aware models have explicitly investigated delayed correlations in traffic forecasting, highlighting the significance of temporal offsets in spatial interactions [1,5].

Despite these advancements, two critical gaps remain in the existing literature. First, lead–lag relationships are often utilized implicitly through attention weights or dynamic correlations, rather than being incorporated as explicit traffic priors for graph construction. Second, most graph propagation modules aggregate neighboring information synchronously at the same time step, even when the physical influence between two sensors is inherently delayed. Such synchronous propagation fails to fully reflect the physical and functional propagation processes of freeway traffic disturbances. To address the delay issue, recent state-of-the-art models such as TLAST [5], PDFormer [21], and DSM-STWave [27] have explicitly modeled delayed correlations using Transformer-based architectures. While these advanced models achieve remarkable predictive accuracy by capturing global dynamic dependencies, they inherently incur high computational complexity and large parameter sizes, which significantly limits their feasibility for real-time, resource-constrained edge deployment.

To bridge these gaps while maintaining computational efficiency, Traffic-DiMAGNet adopts a fundamentally different design philosophy focused on lightweight efficiency and explicit physical interpretability. Instead of relying on heavy self-attention mechanisms, our model utilizes explicit discrete time shifts combined with graph convolutions. Specifically, it estimates discrete edge delays from training-set lead–lag statistics and shifts inter-sensor messages accordingly during bidirectional graph propagation. This design directly encodes the physical propagation delay into the message-passing mechanism with linear complexity, effectively addressing the aforementioned synchronous propagation gap without the heavy computational overhead of Transformers. Therefore, while Transformer-based models excel in absolute accuracy, Traffic-DiMAGNet provides a highly efficient, low-latency alternative that strictly balances predictive performance with the stringent computational requirements of real-time freeway monitoring systems.

2.4. Efficient Temporal Modeling

Temporal modeling is another key component of short-term traffic forecasting. Recurrent models, such as those used in DCRNN and AGCRN, can capture sequential dependencies but are relatively difficult to parallelize and may become inefficient for long input sequences. Attention-based models improve the ability to model long-range temporal dependencies, but their computational cost can increase rapidly with sequence length [16].

Continuous-time graph recurrent formulations, including latent ODE-based graph RNNs, provide another way to represent temporal evolution in multichannel time-series forecasting [28]. More broadly, recent reviews of deep learning on spatio-temporal graphs have summarized the methodological landscape of recurrent, convolutional, attention-based, and continuous-time spatio-temporal models [29].

However, fixed-scale temporal filters may be insufficient for traffic flow data, where abrupt short-term fluctuations and smoother longer-term trends often coexist. Multi-scale convolutional designs can provide a more flexible temporal receptive field while maintaining computational efficiency. Therefore, Traffic-DiMAGNet adopts a lightweight multi-scale dilated temporal convolutional encoder with gating. This design allows the model to capture temporal patterns at different scales without relying on heavy recurrent or attention-based modules.

3. Proposed Method

3.1. Problem Definition

Let $\mathcal{G}_r = (\mathcal{V}, \mathcal{E}_r)$ denote the road-network graph, where $\mathcal{V} = \{1, \dots, N\}$ is the set of traffic sensors and $N = |\mathcal{V}|$ is the number of nodes. The physical road adjacency matrix is denoted by $\mathbf{R} \in \mathbb{R}_+^{N \times N}$. In this study, $\mathbf{R}$ is derived directly from the benchmark-provided road topology: $\mathbf{R}_{ij} = 1$ if a directed physical connection from sensor j to sensor i exists in the dataset, and $\mathbf{R}_{ij} = 0$ otherwise. In this paper, we adopt the convention that an entry $\mathbf{A}_{ij}$ represents a directed dependency from source node j to target node i . For a generic node-feature matrix $\mathbf{Z} \in \mathbb{R}^{N \times C}$, where $\mathbf{Z}_j \in \mathbb{R}^C$ denotes the feature vector of node j , graph aggregation is expressed as $(\mathbf{AZ})_i = \sum_{j=1}^N \mathbf{A}_{ij} \mathbf{Z}_j$. Given historical observations over T time steps, the input tensor is denoted as $\mathbf{X} \in \mathbb{R}^{B \times C_x \times N \times T}$, where B is the batch size and C_x is the number of input features. The goal is to predict future traffic flow over the next H time steps, denoted as $\hat{\mathbf{Y}} \in \mathbb{R}^{B \times N \times H}$. The forecasting task is thus formulated as learning a mapping $\hat{\mathbf{Y}} = F_{\Theta}(\mathbf{X}; \mathbf{R}, \mathbf{P}, \mathbf{T}_{\text{lag}})$, where $\mathbf{P}$ is the lead-lag traffic prior estimated from the training set, $\mathbf{T}_{\text{lag}}$ is the corresponding discrete delay matrix, and Θ denotes all learnable parameters. The overall architecture of the proposed Traffic-DiMAGNet is illustrated in Figure 1. The model comprises four main components: (1) training-set lead-lag prior estimation, (2) traffic-prior-constrained sparse directed graph construction, (3) lightweight multi-scale temporal encoding, and (4) lag-aware bidirectional graph propagation with feature-dependent directional gating.

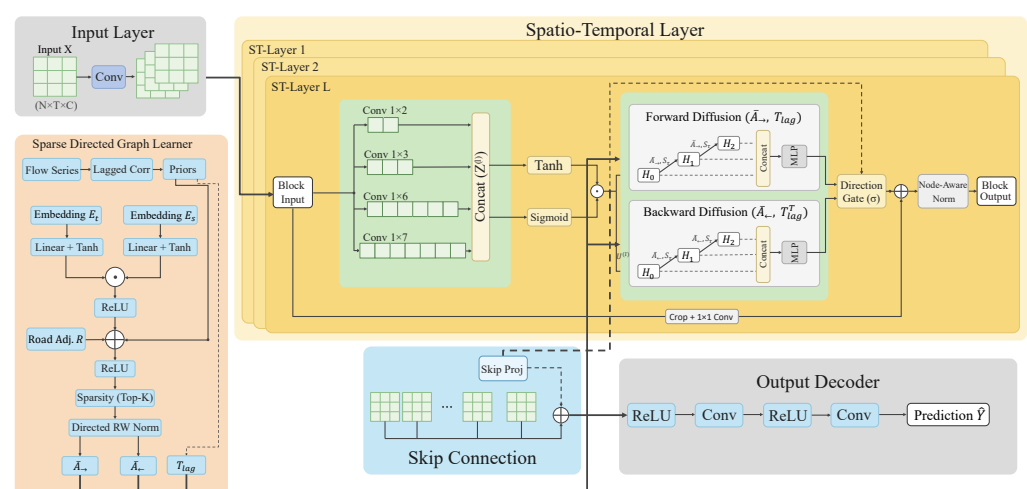


Figure 1. Structure of the Traffic-DiMAGNet model. The dashed lines represent the skip connections.

3.2. Training-Set Lead-Lag Traffic Prior

To incorporate historical traffic propagation patterns, we estimate a lead-lag prior exclusively from the training set. Let $z_i(t)$ denote the normalized traffic flow series of node i . For each ordered pair (j, i) , where j is the source and i is the target, we compute the lagged correlation

$$\rho_{ij}(\tau) = \text{Corr}\left(\{z_j(t - \tau)\}_{t=\tau+1}^{T_{\text{tr}}}, \{z_i(t)\}_{t=\tau+1}^{T_{\text{tr}}}\right), \quad \tau \in \{0, 1, \dots, \tau_{\text{max}}\}, \quad (1)$$

where T_{tr} is the training sequence length. The optimal discrete delay τ_{ij}^* and its corresponding prior strength p_{ij}^* are identified as

$$\tau_{ij}^* = \arg \max_{\tau \in \{0, \dots, \tau_{\text{max}}\}} \rho_{ij}(\tau), \quad p_{ij}^* = \max(0, \rho_{ij}(\tau_{ij}^*)). \quad (2)$$

To suppress noisy edges, we introduce a minimum prior-strength threshold $\delta_p \geq 0$. The final prior matrices are constructed as

$$(\mathbf{P}_{ij}, \mathbf{T}_{\text{lag}, ij}) = \begin{cases} (p_{ij}^*, \tau_{ij}^*), & p_{ij}^* \geq \delta_p, \\ (0, 0), & p_{ij}^* < \delta_p. \end{cases} \quad (3)$$

Self-connections are explicitly set to zero ($\mathbf{P}_{ii} = 0, \mathbf{T}_{\text{lag}, ii} = 0$) as self-loops are handled during graph normalization. Consequently, $\mathbf{P}$ quantifies the historical leading strength from j to i , while $\mathbf{T}_{\text{lag}}$ records the estimated propagation delay.

3.3. Traffic-Prior-Constrained Sparse Directed Graph Learning

While the physical adjacency matrix $\mathbf{R}$ encodes road connectivity, it fails to capture non-local functional dependencies. Traffic-DiMAGNet therefore learns an adaptive directed graph constrained by both physical and lead-lag priors. We assign each node i a target-side embedding $\mathbf{e}_i^t \in \mathbb{R}^d$ and a source-side embedding $\mathbf{e}_i^s \in \mathbb{R}^d$, where d denotes the embedding dimension, to model asymmetric dependencies. The transformed embeddings are

$$\tilde{\mathbf{e}}_i^t = \tanh(\beta \mathbf{e}_i^t \mathbf{W}_t), \quad \tilde{\mathbf{e}}_j^s = \tanh(\beta \mathbf{e}_j^s \mathbf{W}_s), \quad (4)$$

where $\mathbf{W}_t, \mathbf{W}_s \in \mathbb{R}^{d \times d}$ are trainable projection matrices and $\beta > 0$ controls the nonlinearity strength. The embedding-based directed affinity from source node j to target node i is computed as

$$\mathbf{A}_{ij}^{\text{emb}} = \text{ReLU}\left[\tanh(\beta \tilde{\mathbf{e}}_i^t (\tilde{\mathbf{e}}_j^s)^\top)\right]. \quad (5)$$

To integrate the learned affinity with the domain priors, we fuse the embedding graph, traffic prior, and road adjacency using fixed scaling factors:

$$\mathbf{A}_{ij}^{\text{score}} = \text{ReLU}(\mathbf{A}_{ij}^{\text{emb}} + \lambda_p \mathbf{P}_{ij} + \gamma \mathbf{R}_{ij}), \quad (6)$$

where λ_p and γ are non-negative hyperparameters controlling the influence of the lead-lag prior and physical connectivity, respectively. To ensure sparsity and computational efficiency, we retain only the top- K_s neighbors for each node:

$$\mathbf{A}_{ij} = \begin{cases} \mathbf{A}_{ij}^{\text{score}}, & j \in \mathcal{N}_i^{K_s}, \\ 0, & \text{otherwise,} \end{cases} \quad (7)$$

where $\mathcal{N}_i^{K_s} = \text{TopK}(\{\mathbf{A}_{ij}^{\text{score}}\}_{j=1}^N, K_s)$ denotes the index set of the K_s largest affinities for node i . The resulting $\mathbf{A}$ is a sparse directed fused graph that serves as the primary spatial structure for both subsequent message passing and graph regularization.

To discourage implausible dependencies that contradict historical traffic propagation, we impose a graph regularization term during training:

$$\mathcal{L}_{\text{graph}} = \lambda_s \|\mathbf{A}\|_1 + \lambda_o \left\| (\mathbf{1} - \mathbf{M}^{\text{pri}}) \odot \mathbf{A} \right\|_1, \quad (8)$$

where $\mathbf{1} \in \mathbb{R}^{N \times N}$ is an all-ones matrix, $\mathbf{M}_{ij}^{\text{pri}} = \mathbb{I}(\mathbf{P}_{ij} > 0)$ acts as a traffic-prior support mask, and $\lambda_s, \lambda_o \geq 0$ control the sparsity and off-prior penalties, respectively. The overall training objective combines this graph regularization term $\mathcal{L}_{\text{graph}}$ with the primary forecasting loss.

3.4. Directed Random-Walk Normalization

Because the learned graph is directed, symmetric normalization may weaken the asymmetric source–target semantics. Therefore, we adopt directed random-walk normalization. For any non-negative matrix $\mathbf{M} \in \mathbb{R}^{N \times N}$, we define the degree-clipped row-stochastic operator as

$$\tilde{\mathbf{D}}_{\mathbf{M}} = \text{diag}(\max(\mathbf{M}\mathbf{1}, \epsilon)), \quad \text{RW}(\mathbf{M}) = \tilde{\mathbf{D}}_{\mathbf{M}}^{-1} \mathbf{M}, \quad (9)$$

where $\mathbf{1} \in \mathbb{R}^{N \times 1}$ is an all-ones column vector, $\epsilon > 0$ is a small lower bound (e.g., 10^{-6}) to prevent division by zero, and $\max(\cdot, \epsilon)$ applies element-wise clipping to the out-degree vector. This formulation preserves exact row-stochasticity for nodes with non-zero out-degree and aligns with the degree-clamping strategy used in our implementation.

After adding self-loops, the forward and reverse normalized adjacency matrices are obtained as

$$\bar{\mathbf{A}}^{\rightarrow} = \text{RW}(\mathbf{A} + \mathbf{I}), \quad \bar{\mathbf{A}}^{\leftarrow} = \text{RW}(\mathbf{A}^{\top} + \mathbf{I}). \quad (10)$$

Here, $\bar{\mathbf{A}}^{\rightarrow}$ facilitates information propagation along the learned source-to-target direction, while $\bar{\mathbf{A}}^{\leftarrow}$ handles reverse-direction diffusion. This normalization strictly preserves the asymmetric graph semantics established in the problem definition and stabilizes multi-hop message passing. The resulting normalized matrices, together with the discrete delay matrix $\mathbf{T}_{\text{lag}}$, are subsequently fed into the lag-aware bidirectional propagation module to model asynchronous traffic diffusion.

3.5. Lightweight Multi-Scale Temporal Encoding

The raw input $\mathbf{X} \in \mathbb{R}^{B \times C_x \times N \times T}$ is first projected into a hidden feature space:

$$\mathbf{H}^{(0)} = \phi_{\text{in}}(\mathbf{X}) \in \mathbb{R}^{B \times C \times N \times T}, \quad (11)$$

where ϕ_{in} denotes a 1×1 convolution that maps the input channels C_x to the hidden dimension C . Let $\mathbf{H}^{(l)}$ denote the input to the l -th spatio-temporal block. To capture temporal patterns at multiple scales, we employ parallel dilated causal convolutions with a kernel size set $\mathcal{K} = \{k_1, k_2, \dots, k_m\}$. The multi-scale temporal response at layer l is computed as

$$\mathbf{Z}^{(l)} = \text{Concat}_{k \in \mathcal{K}} \left[\text{CausalConv}_{k, d_l}^{(l)} \left(\mathbf{H}^{(l)} \right) \right], \quad (12)$$

where $\text{CausalConv}_{k, d_l}^{(l)}(\cdot)$ applies a causal convolution along the temporal dimension with kernel size k and dilation factor d_l , while operating independently across the node and channel dimensions. Causal padding is applied to ensure the temporal length remains as T without future information leakage. The dilation factor grows exponentially with depth to expand the receptive field:

$$d_l = d_0 \eta^{l-1}, \quad l = 1, \dots, L, \quad (13)$$

where $d_0 \geq 1$ is the base dilation, $\eta > 1$ is the growth factor, and L is the total number of temporal layers.

To adaptively fuse multi-scale features and suppress high-frequency noise, a gated linear unit is applied:

$$\mathbf{U}^{(l)} = \tanh(\mathbf{W}_f^{(l)} * \mathbf{Z}^{(l)}) \odot \sigma(\mathbf{W}_g^{(l)} * \mathbf{Z}^{(l)}), \quad (14)$$

where $*$ denotes a 1×1 convolution, $\mathbf{W}_f^{(l)}$ and $\mathbf{W}_g^{(l)}$ are learnable weight tensors, $\sigma(\cdot)$ is the sigmoid activation, and $\odot$ denotes element-wise multiplication. The resulting gated representation $\mathbf{U}^{(l)}$ is then forwarded to the lag-aware bidirectional graph propagation module to model spatial dependencies, while simultaneously being accumulated through a skip connection for the final prediction decoder.

3.6. Lag-Aware Bidirectional Graph Propagation

Standard graph diffusion aggregates neighboring features at the same time step. However, freeway traffic propagation is often delayed. To model this property, Traffic-DiMAGNet shifts inter-sensor messages according to the estimated edge delay matrix $\mathbf{T}_{\text{lag}}$.

Given the gated temporal representation $\mathbf{U}^{(l)} \in \mathbb{R}^{B \times C \times N \times T}$ from the temporal encoder, we first define a causal shift operator $\mathcal{S}_\tau(\cdot)$ along the time dimension:

$$[\mathcal{S}_\tau(\mathbf{H})]_{b,c,i,t} = \begin{cases} \mathbf{H}_{b,c,i,t-\tau}, & t > \tau \\ 0, & t \leq \tau \end{cases}, \quad (15)$$

where $\mathbf{H}$ denotes a generic feature tensor, and $\mathcal{S}_0(\mathbf{H}) = \mathbf{H}$. We directly utilize the discrete delay matrix $\mathbf{T}_{\text{lag}}$, where self-loop delays are already set to zero as described in the prior estimation. For each discrete delay $\tau \in \{0, \dots, \tau_{\max}\}$, we construct delay-specific adjacency masks using an element-wise indicator function $\mathbb{I}(\cdot)$:

$$\bar{\mathbf{A}}_\tau^\rightarrow = \bar{\mathbf{A}}^\rightarrow \odot \mathbb{I}(\mathbf{T}_{\text{lag}} = \tau), \quad \bar{\mathbf{A}}_\tau^\leftarrow = \bar{\mathbf{A}}^\leftarrow \odot \mathbb{I}(\mathbf{T}_{\text{lag}}^\top = \tau), \quad (16)$$

where $\bar{\mathbf{A}}^\rightarrow$ and $\bar{\mathbf{A}}^\leftarrow$ are the forward and reverse normalized adjacency matrices obtained via directed random-walk normalization. The lag-aware propagation operators then perform spatial aggregation at temporally shifted steps:

$$\mathcal{P}^\rightarrow(\mathbf{H}) = \sum_{\tau=0}^{\tau_{\max}} \bar{\mathbf{A}}_\tau^\rightarrow \mathcal{S}_\tau(\mathbf{H}), \quad \mathcal{P}^\leftarrow(\mathbf{H}) = \sum_{\tau=0}^{\tau_{\max}} \bar{\mathbf{A}}_\tau^\leftarrow \mathcal{S}_\tau(\mathbf{H}), \quad (17)$$

where matrix multiplication is applied along the node dimension, and the time dimension remains aligned via causal padding.

To capture multi-hop dependencies, mixed-order propagation is performed recursively in both directions. For the forward branch:

$$\mathbf{H}_0^{(l,\rightarrow)} = \mathbf{U}^{(l)}, \quad \mathbf{H}_m^{(l,\rightarrow)} = \alpha \mathbf{U}^{(l)} + (1 - \alpha) \mathcal{P}^\rightarrow(\mathbf{H}_{m-1}^{(l,\rightarrow)}), \quad m = 1, \dots, M_p, \quad (18)$$

where M_p is the propagation depth and $\alpha \in (0, 1)$ is a residual mixing coefficient that balances local self-features and aggregated neighborhood information. The reverse branch follows an identical recurrence with $\mathcal{P}^\leftarrow$.

The multi-order states are concatenated along the channel dimension and projected back to the original feature space:

$$\mathbf{Y}^{(l,\rightarrow)} = \phi_{\rightarrow}^{(l)} \left(\text{Concat}_{m=0}^{M_p} \mathbf{H}_m^{(l,\rightarrow)} \right), \quad \mathbf{Y}^{(l,\leftarrow)} = \phi_{\leftarrow}^{(l)} \left(\text{Concat}_{m=0}^{M_p} \mathbf{H}_m^{(l,\leftarrow)} \right), \quad (19)$$

where $\phi_{\rightarrow}^{(l)}$ and $\phi_{\leftarrow}^{(l)}$ are 1×1 convolutions that map $C(M_p + 1)$ channels to C .

3.7. Feature-Dependent Directional Gating

The relative importance of forward and reverse traffic propagation varies across nodes, time steps, and traffic regimes. Instead of symmetrically aggregating both directions, we employ a feature-dependent directional gate:

$$\mathbf{G}^{(l)} = \sigma \left(\phi_{\text{gate}}^{(l)} \left([\mathbf{U}^{(l)}; \mathbf{Y}^{(l,\rightarrow)}; \mathbf{Y}^{(l,\leftarrow)}] \right) \right), \quad (20)$$

where $[\cdot; \cdot]$ denotes channel-wise concatenation, yielding $3C$ input channels. The gate projection $\phi_{\text{gate}}^{(l)}$ is a 1×1 convolution that maps the concatenated features to C channels, and $\sigma(\cdot)$ is the sigmoid function. The spatial output of layer l is then computed as

$$\mathbf{Q}^{(l)} = \mathbf{G}^{(l)} \odot \mathbf{Y}^{(l,\rightarrow)} + (\mathbf{1} - \mathbf{G}^{(l)}) \odot \mathbf{Y}^{(l,\leftarrow)}, \quad (21)$$

where $\odot$ denotes element-wise multiplication. When $\mathbf{G}^{(l)}$ approaches $\mathbf{1}$, forward propagation dominates; conversely, values near $\mathbf{0}$ emphasize reverse diffusion. This gating mechanism enables Traffic-DiMAGNet to dynamically balance upstream and downstream traffic influences under varying flow conditions.

3.8. Spatio-Temporal Fusion, Normalization, and Decoder

To stabilize layer-wise optimization and preserve multi-scale representations, we integrate residual refinement with skip accumulation. The skip pathway is updated as

$$\mathbf{S}^{(l)} = \mathbf{S}^{(l-1)} + \phi_{\text{skip}}^{(l)}(\mathbf{Q}^{(l)}), \quad \mathbf{S}^{(0)} = \mathbf{0}, \quad (22)$$

where $\phi_{\text{skip}}^{(l)}$ is a 1×1 convolution that projects $\mathbf{Q}^{(l)}$ to a fixed skip channel dimension. The hidden state for the subsequent layer is obtained via residual fusion:

$$\tilde{\mathbf{H}}^{(l+1)} = \mathbf{Q}^{(l)} + \phi_{\text{res}}^{(l)}(\mathcal{C}(\mathbf{H}^{(l)})), \quad (23)$$

where $\phi_{\text{res}}^{(l)}$ is a 1×1 projection, and $\mathcal{C}(\cdot)$ represents a temporal cropping operation that discards the initial steps to align the sequence length with the reduced receptive field caused by causal convolutions.

To mitigate distribution shifts across heterogeneous sensors, we apply node-aware layer normalization:

$$\mathbf{H}_{b,i,:}^{(l+1)} = \gamma_i \odot \text{LN}(\tilde{\mathbf{H}}_{b,i,:}^{(l+1)}) + \beta_i, \quad (24)$$

where $\text{LN}(\cdot)$ computes statistics over the channel and temporal dimensions for each node i , and $\gamma_i, \beta_i \in \mathbb{R}^C$ are node-specific learnable affine parameters.

After L spatio-temporal blocks, the aggregated skip representation is decoded into the prediction horizon:

$$\hat{\mathbf{Y}} = \Theta_2 * \text{ReLU}(\Theta_1 * \mathbf{S}^{(L)}), \quad (25)$$

where Θ_1 and Θ_2 are 1×1 convolutional kernels that sequentially map the skip features to the target horizon H . The final output is reshaped to $\hat{\mathbf{Y}} \in \mathbb{R}^{B \times N \times H}$.

3.9. Training Objective

The primary prediction loss is formulated as the masked mean absolute error:

$$\mathcal{L}_{\text{pred}} = \frac{1}{|\Omega|} \sum_{(b,i,h) \in \Omega} |\hat{\mathbf{Y}}_{b,i,h} - \mathbf{Y}_{b,i,h}|, \quad (26)$$

where Ω denotes the set of valid observations after excluding padded or missing entries. The overall training objective combines the prediction loss with the graph regularization term $\mathcal{L}_{\text{graph}}$:

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \mathcal{L}_{\text{graph}}. \quad (27)$$

Optimization is performed via the Adam optimizer with a multi-step learning-rate decay schedule.

4. Experiment

4.1. Datasets and Baselines

In our experiments, we utilize four real-world traffic flow datasets collected from the California Performance Measurement System (PeMS), including PeMS03, PeMS04, PeMS07, and PeMS08 [17]. These datasets are collected by loop detectors installed on freeways and are aggregated into 5 min intervals, yielding 12 observations per hour for each sensor station. The detailed statistics of these datasets are summarized in Table 1.

We handle missing values using linear interpolation to preserve temporal continuity, as the missing rates in these datasets are relatively low. Additionally, all data are normalized using z-score normalization to eliminate the influence of different scales and transform the data into a standard normal distribution. To capture the spatial correlations among sensing regions, we construct a road-network-based spatial graph, where nodes represent sensor locations and edges encode the physical connectivity of the transportation network.

Table 1. Detailed statistics of the PeMS datasets.

Dataset	Nodes	Edges	Time Steps	Time Range (mm/yyyy)	Missing Rate (%)	Data Type
PeMS03	358	547	26,208	09/2018–11/2018	0.67	Flow
PeMS04	307	340	16,992	01/2018–02/2018	3.18	Flow, speed, occupancy
PeMS07	883	866	28,224	05/2017–08/2017	0.45	Flow
PeMS08	170	259	17,856	07/2016–08/2016	0.69	Flow, speed, occupancy

We compare the proposed model with 16 representative baselines, including:

- **DSANet** [30]: Integrates a dual-stage attention mechanism with graph convolutions to capture long-term temporal dependencies and spatial correlations in multivariate time series.
- **STGCN** [10]: Integrates spectral graph convolutions with temporal convolutional layers to jointly model spatial and temporal features.
- **DCRNN** [11]: Employs a diffusion process over graph structures within a sequence-to-sequence recurrent framework to capture spatial dependencies.
- **Graph WaveNet** [18]: Employs adaptive adjacency matrices and dilated causal convolutions to model dynamic spatial dependencies without relying on predefined graphs.
- **ASTGCN** [12]: Integrates temporal, spatial, and channel-wise attention mechanisms with graph convolutions to capture dynamic spatio-temporal patterns.
- **AGCRN** [15]: Learns node-specific adaptive adjacency matrices, combined with gated recurrent units for dynamic temporal modeling.
- **STFGNN** [31]: Proposes a joint spatial-temporal fusion mechanism to simultaneously learn spatial topology and temporal dynamics directly from data.
- **STG-ODE** [32]: Models continuous-time spatial-temporal dynamics using neural ordinary differential equations to capture fine-grained traffic evolution.

- **Z-GCNETs** [33]: Introduces time-aware zero-shot learning mechanisms to generalize spatial dependencies to unseen or sparsely observed nodes.
- **TAMP** [34]: Captures multi-scale traffic patterns through specialized temporal aggregation and spatial message passing mechanisms.
- **DSTAGNN** [26]: Constructs dynamic spatial-temporal graphs using multi-head attention mechanisms to adaptively model evolving traffic dependencies.
- **FOGS** [35]: Learns flow-oriented graph structures to filter high-frequency noise and enhance robust spatial feature extraction.
- **STSGCN** [17]: Synchronizes spatial and temporal feature learning using localized spatio-temporal subgraphs to avoid information loss during propagation.
- **LSTGCN** [36]: Decouples long-term and short-term traffic patterns, applying distinct graph convolutional modules to model periodic and recent trends.
- **KD-Pruning** [37]: Applies knowledge distillation and network pruning techniques to compress spatio-temporal models while preserving predictive accuracy.
- **LightST** [38]: Designs a highly efficient, lightweight architecture that optimally balances computational cost and predictive performance for real-time applications.

4.2. Experimental Settings

To evaluate the performance of our proposed model and other methods, we utilize three commonly adopted metrics: the mean absolute error (MAE), root mean squared error (RMSE), and the mean absolute percentage error (MAPE). Following the common traffic forecasting setting, we use the past $T = 12$ time steps as the input and predict the next $H = 12$ time steps for all datasets and baselines. All experiments are implemented using PyTorch (version 2.1.2) on a single NVIDIA RTX 5090 32GB GPU (NVIDIA Corporation, Santa Clara, CA, USA).

Data Preparation. Following standard protocols in traffic forecasting, all datasets are chronologically split into training, validation, and test sets with a ratio of 6:2:2. The input traffic flow data are normalized using z-score normalization to mitigate scale differences across different sensors.

Training Strategy. The proposed Traffic-DiMAGNet is optimized using the Adam optimizer with an initial learning rate of 0.003. To ensure stable optimization, we apply gradient clipping with a maximum norm of 5 and adopt a multi-step learning rate decay strategy with a decay factor of 0.3. The model is trained to minimize the masked mean absolute error (MAE) and employs an early stopping mechanism based on the validation loss to prevent overfitting.

Model Hyperparameters. For a fair comparison, the architectural hyperparameters of the Traffic-DiMAGNet are kept consistent across all four datasets. The model consists of $L = 3$ spatio-temporal layers with a graph propagation depth of $M_p = 2$. The hidden channel dimensions for the temporal convolution, residual connection, skip connection, and output decoding are set to 32, 32, 64, and 128, respectively. The node embedding dimension is $d = 40$, and the dropout rate is set to 0.3. In the proposed traffic-prior-constrained graph learning module, the maximum lead-lag prior is configured as $\tau_{\max} = 3$, the sparsity level is $K_s = 20$, and the multi-scale temporal kernel set is $\mathcal{K} = \{2, 3, 6, 7\}$. The fusion weights for the traffic prior and physical road network are set to $\lambda_p = 0.5$ and $\gamma = 0.2$, respectively. Furthermore, the graph sparsity and off-prior regularization weights are set to $\lambda_s = 10^{-3}$ and $\lambda_o = 10^{-4}$. The detailed hyperparameter settings are summarized in Table 2.

Table 2. Core hyperparameter settings of Traffic-DiMAGNet.

Category	Hyperparameter	Value
Task Setup	Input length T /Horizon H	12/12
	Data split (Train/Val/Test)	0.6/0.2/0.2
Training	Initial learning rate/Decay factor	0.003/0.3
	Gradient clipping/Loss	5/Masked MAE
	Early-stop patience	20
Architecture	ST layers L /Propagation depth M_p	3/2
	Node embedding dimension d /Dropout	40/0.3
	Embedding nonlinearity strength β	3
	Sparsity level K_s	20
	Conv/Res/Skip/End channels	32/32/64/128
	Multi-scale kernel set $\mathcal{K}$	$\{2, 3, 6, 7\}$
	Base dilation d_0 /Growth factor η	1/2
	Propagation mixing coefficient α	0.05
Prior and Reg.	Max. lead-lag prior lag $\tau_{\max}$	3
	Min. prior-strength threshold δ_p	0.0
	Traffic-prior weight λ_p	0.5
	Road-prior weight γ	0.2
	Graph sparsity weight λ_s	10^{-3}
	Off-prior weight λ_o	10^{-4}

4.3. Performance Comparison

Table 3 summarizes the prediction performance across the four PeMS datasets. Overall, Traffic-DiMAGNet consistently achieves the best results across all evaluation metrics, significantly outperforming both classical spatio-temporal GNNs (e.g., STGCN, GWN) and recent topology-learning or ODE-based models (e.g., STG-ODE, DSTAGNN). These improvements are particularly evident on the larger-scale PeMS07 and PeMS08 datasets, indicating that the proposed adaptive graph construction and direction-aware propagation mechanisms effectively capture complex and asymmetric spatial dependencies. Moreover, Traffic-DiMAGNet maintains a lower MAPE and RMSE across datasets with varying traffic conditions, demonstrating strong robustness and generalization capability.

In addition, baseline models equipped with single-scale temporal kernels or fixed spatial graphs exhibit noticeable performance degradation, especially under long-sequence forecasting settings. In contrast, the multi-scale dilated temporal module and the bidirectional propagation mechanism enable Traffic-DiMAGNet to model short-term fluctuations and long-term trends simultaneously. As a result, Traffic-DiMAGNet delivers more stable prediction accuracy across all metrics, thoroughly validating the effectiveness of its spatio-temporal design.

To further validate that the superiority of Traffic-DiMAGNet is not due to random chance, a statistical significance analysis is performed. As recommended, we employ the Wilcoxon signed-rank test on the test samples to examine whether the performance improvements over baseline models are statistically meaningful, rather than treating different metrics as independent trials. For each dataset, we compare the per-sample prediction errors (e.g., absolute errors) between Traffic-DiMAGNet and each baseline. Under the null hypothesis of the Wilcoxon signed-rank test, the median difference between the paired errors is zero.

In our experimental setup, we evaluate the models across $m = 4$ datasets. At a significance level of $\alpha = 0.05$, we assert that Traffic-DiMAGNet is significantly better than a baseline if the Wilcoxon test rejects the null hypothesis (p -value < 0.05) on a dataset. Specifically, against each of the 16 evaluated baseline models, Traffic-DiMAGNet consistently

achieves statistically significant improvements across all four datasets (i.e., 4 wins out of 4, with p -values < 0.05). Therefore, by the statistical comparison between two algorithms on each single dataset based on the Wilcoxon signed-rank test, Traffic-DiMAGNet outperforms all 16 baseline models with significant superiority, confirming that its performance gains are robust and not due to random chance.

Table 3. Results on the four datasets.

Models	PeMS03			PeMS04			PeMS07			PeMS08		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
DSANet	21.29	34.55	23.21%	22.79	35.77	17.12%	31.36	49.11	14.43%	17.14	26.96	11.32%
DCRNN	17.99	30.31	18.34%	24.70	38.12	14.17%	25.22	38.61	11.82%	17.86	27.83	11.45%
STGCN	17.55	30.42	17.34%	22.70	35.55	14.59%	25.33	39.34	11.21%	18.02	27.83	11.40%
Graph WaveNet	19.12	32.77	18.89%	25.45	39.70	17.29%	26.39	41.50	11.97%	19.13	31.05	12.68%
ASTGCN	17.34	29.56	17.21%	22.93	35.22	16.56%	24.01	37.87	10.73%	18.25	28.06	11.64%
LSTGCN	17.94	29.85	16.98%	21.53	33.86	13.18%	27.31	41.16	11.98%	17.73	26.76	11.30%
STSGCN	17.48	29.21	16.78%	21.19	33.65	13.90%	24.26	39.03	10.21%	17.13	26.86	10.96%
AGCRN	15.98	28.25	15.23%	19.83	32.26	12.97%	22.37	36.55	9.12%	15.95	25.22	10.09%
STFGNN	16.77	28.34	16.30%	19.83	31.88	13.02%	22.07	35.80	9.21%	16.64	26.22	10.60%
STGODE	16.50	27.84	16.69%	20.84	32.82	13.77%	22.99	37.54	10.14%	16.81	25.97	10.62%
Z-GCNETs	16.64	28.15	16.39%	19.50	31.61	12.78%	21.77	35.17	9.25%	15.76	25.11	10.01%
TAMP	16.46	28.44	15.37%	19.74	31.74	13.22%	21.84	35.42	9.24%	16.36	25.98	10.15%
FOGS	15.89	25.74	15.13%	19.74	31.66	13.05%	21.28	34.88	8.95%	15.73	24.92	9.88%
DSTAGNN	15.57	27.21	14.68%	19.30	31.46	12.72%	21.42	34.51	9.01%	15.67	24.77	9.94%
KD-Pruning	17.12	29.87	17.06%	21.22	34.63	14.15%	24.55	38.17	11.90%	17.46	27.09	11.77%
LightST	15.11	24.74	14.41%	19.21	31.31	12.70%	20.78	33.95	8.98%	15.43	24.52	9.84%
Ours	14.89	24.12	14.06%	18.87	31.18	12.35%	20.38	33.11	8.57%	14.84	24.27	9.58%

4.4. Ablation Study

To evaluate the individual contributions of the core components in Traffic-DiMAGNet, we conduct a comprehensive ablation study on the PeMS07 and PeMS08 datasets. We design five variant models by systematically removing or replacing key modules: (1) w/o Bi-Prop: removing the lag-aware bidirectional graph propagation and directional gating; (2) w/o Adaptive Graph: disabling the traffic-prior-constrained graph learning and reverting to the predefined static physical graph R ; (3) w/o Spatial: discarding spatial message passing entirely to form a pure temporal model; (4) w/o Multi-Scale: replacing the multi-scale gated temporal module with a standard single-kernel causal convolution; and (5) w/o Node Norm: substituting the node-aware layer normalization with a shared standard layer normalization.

The quantitative results are summarized in Table 4, and in Figures 2 and 3. The full model achieves the best overall performance across all metrics, confirming the necessity of each design.

Table 4. Ablation study results on PeMS07 and PeMS08 datasets.

Variant	PeMS07			PeMS08		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE
w/o Bi-Prop	20.62	34.03	8.95%	15.24	24.46	9.83%
w/o Adaptive Graph	21.04	34.47	9.62%	15.06	24.31	10.30%
w/o Spatial	20.93	34.26	9.56%	15.01	24.35	9.64%
w/o Multi-Scale	20.98	35.78	8.81%	15.02	24.45	9.82%
w/o Node Norm	23.04	37.53	9.97%	16.00	25.98	10.29%
Ours	20.38	33.11	8.57%	14.84	24.27	9.58%

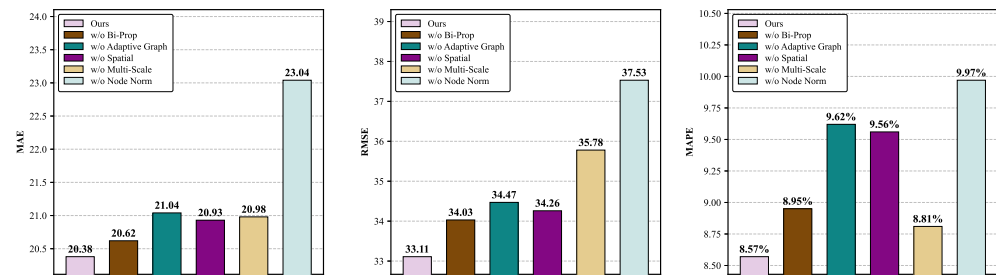


Figure 2. Ablation study on PeMS07.

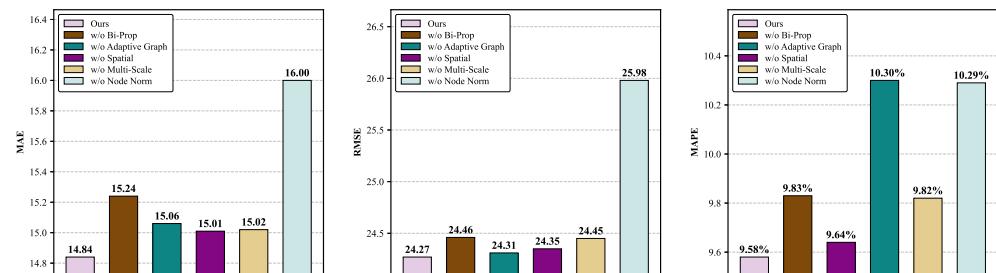


Figure 3. Ablation study on PeMS08.

Several key observations can be drawn from the ablation results. First, the full Traffic-DiMAGNet model achieves the best overall performance across most metrics, validating the necessity of its core designs. The most pronounced performance degradation is observed when the node-aware layer normalization is replaced with a shared standard normalization (w/o Node Norm). Specifically, this modification increases the MAE by 1.16 and the RMSE by 1.71 on PeMS08, with an even more severe deterioration on PeMS07 (MAE +2.66, RMSE +4.42). This substantial drop underscores the high heterogeneity in traffic sensor data distributions, indicating that node-specific affine transformations are indispensable for stabilizing training dynamics and effectively aligning feature scales across the spatial graph.

Furthermore, the results confirm the consistent effectiveness of the spatial and multi-scale temporal modeling components. Discarding spatial message passing entirely (w/o Spatial) leads to uniform performance degradation across both datasets (e.g., the MAE increases by 0.17 on PeMS08 and 0.55 on PeMS07), demonstrating that explicit spatial topology modeling is fundamental for accurate traffic forecasting. Complementing this, disabling the traffic-prior-constrained adaptive graph learning (w/o Adaptive Graph) yields notable increases in the MAPE (+0.72% on PeMS08 and +1.05% on PeMS07). This confirms that the static physical adjacency matrix R alone is inadequate for modeling complex, non-local spatial dependencies like upstream–downstream congestion spillbacks, and the learned sparse directed graph successfully captures these dynamic interactions. Similarly, substituting the multi-scale gated temporal module with a single-kernel convolution (w/o Multi-Scale) consistently impairs prediction accuracy, particularly evident in the RMSE increase of 2.67 on PeMS07. This validates that traffic time series exhibit multi-frequency temporal patterns, which require varied receptive fields to capture both short-term fluctuations and long-term periodic trends effectively.

Finally, regarding the lag-aware bidirectional graph propagation (w/o Bi-Prop), although the improvements in absolute error metrics are marginal, the full model consistently achieves a lower MAPE. In traffic forecasting, metrics like MAE and RMSE are often dominated by high-flow nodes like major highways during peak hours, which can mask prediction inaccuracies in low-flow or transitional regions. The consistent reduction in MAPE demonstrates that the lag-aware mechanism effectively captures directional temporal dependencies and traffic lag effects. This enables the model to maintain robust relative

prediction accuracy across varying flow conditions, preventing overfitting to high-volume sensors and ensuring a more balanced and reliable forecasting performance across the entire network.

4.5. Model Efficiency

At the same time as achieving state-of-the-art predictive accuracy, computational efficiency is crucial for the real-time deployment and iterative training of traffic forecasting systems. In this section, we evaluate the computational complexity of Traffic-DiMAGNet in terms of parameter counts (Params), theoretical floating point operations (FLOPs), and actual inference latency. To ensure a rigorous and fair comparison, all models are profiled under the exact same hardware environment and input configurations on the PeMS04 dataset.

As summarized in Table 5, Traffic-DiMAGNet achieves an optimal trade-off between model scale and theoretical complexity. It requires merely 8.38 G FLOPs and 0.55 M parameters, representing the lowest FLOPs among all compared baselines, while maintaining a highly competitive parameter size and achieving a low inference latency of 13.38 ms/batch. This reduces the computational overhead of dense-graph models like STSGCN and ASTGCN, as well as other baselines like LightST, while remaining in a similar lightweight parameter magnitude as Graph WaveNet. This efficiency validates that our proposed sparse directed graph construction and multi-scale temporal modules effectively mitigate redundant computations, ensuring high viability for real-time intelligent transportation systems.

Table 5. Computational efficiency comparison on PeMS04. ↓ indicates that lower values are better.

Models	Params (M) ↓	FLOPs (G) ↓	Latency (ms/batch) ↓
ASTGCN	2.25	28.25	26.05
Graph WaveNet	0.34	18.40	7.63
STG-ODE	0.71	20.66	33.42
STSGCN	2.02	48.33	18.36
KD-Pruning	0.39	16.90	2.06
LightST	0.75	24.52	3.03
Ours	0.55	8.38	13.38

4.6. Sensitivity Analysis of the Maximum Lead-Lag Prior Lag

To rigorously justify the selection of the maximum lead-lag prior lag $\tau_{max} = 3$, we conduct a comprehensive sensitivity analysis on the PeMS08 dataset using three random seeds. First, we investigate the statistical distribution of the selected edges by expanding the candidate lag search space to $\tau_{max} = 12$. The results reveal that the vast majority of statistically significant edges naturally fall within the short-lag regime. Specifically, the proportions of edges with $\tau \leq 3$ are 94.96%, 79.33%, 91.43%, and 80.06% for PeMS03, PeMS04, PeMS07, and PeMS08, respectively. The overlap rates with the edge set derived from $\tau_{max} = 3$ exceed 88% across all datasets. For lags L_4 to L_{11} , the distribution forms a flat, long-tail noise floor without any secondary peaks, and the marginal correlation gain per edge is negligible (<0.004), which is far below the noise threshold (~ 0.01). This confirms that $\tau_{max} = 3$ is sufficient to capture the dominant lagged dependencies without introducing spurious edges.

Second, to quantitatively evaluate the impact of the maximum lag threshold on predictive accuracy, we perform a sensitivity scan on PeMS08 with $\tau_{max} \in \{1, 2, 3, 4, 6\}$. The detailed MAE results are summarized in Table 6. A block rank-permutation test across seeds yields $Q = 4.533$ and $p = 0.406$, indicating no statistically significant difference in performance across different τ_{max} values. As shown in the table, the performance curve exhibits

a flat plateau rather than a sharp optimum. Although $\tau_{max} = 6$ shows a nominally lower MAE, its 95% confidence interval heavily overlaps with that of $\tau_{max} = 3$, and the marginal effect size is smaller than the random seed drift.

Table 6. Sensitivity analysis of τ_{max} on the PeMS08 dataset.

τ_{max}	MAE
1	14.83 ± 0.06
2	14.85 ± 0.04
3	14.84 ± 0.03
4	14.85 ± 0.04
6	14.83 ± 0.03

Beyond the MAE metric, the RMSE values follow a similar flat trend (ranging from 24.13 to 24.26), further confirming the stability of the model across different lag thresholds. However, increasing the τ_{max} incurs a notable computational overhead. Specifically, increasing the τ_{max} to 6 increases the training time by approximately 15% (from ~ 16.5 s/epoch for $\tau_{max} = 3$ to ~ 19 s/epoch), and significantly raises the peak memory usage to 19.69 GiB. Therefore, $\tau_{max} = 3$ is justified as a conservative and efficient choice located on the flat accuracy–cost plateau, corresponding to a 15 min propagation window that aligns well with typical freeway shockwave dynamics.

5. Conclusions

This paper proposed Traffic-DiMAGNet, an interpretable and lightweight spatio-temporal graph neural network for short-term freeway traffic flow forecasting, designed to address asymmetric spatial dependencies and time-delayed traffic propagation. To this end, a traffic-prior-constrained sparse directed graph learning mechanism was developed to integrate physical road adjacency, historical lead–lag traffic priors, and learnable source–target node embeddings, with sparsity constraints and off-prior regularization further enhancing the transportation interpretability of the learned graph. Furthermore, a lag-aware bidirectional propagation module was introduced to shift inter-sensor messages according to estimated discrete edge delays, providing a more traffic-consistent representation of delayed upstream and downstream influences. Combined with directed random-walk normalization, feature-dependent directional gating, and a lightweight multi-scale temporal convolutional encoder, the model effectively captures both short-term fluctuations and long-range temporal patterns while preserving asymmetric source–target semantics and maintaining computational efficiency.

Extensive experiments on four real-world Caltrans PeMS datasets demonstrated that Traffic-DiMAGNet consistently outperforms representative recurrent, diffusion-based, attention-based, and adaptive-graph baselines in terms of MAE, RMSE, and MAPE values, with stable predictive performance across different forecasting horizons. Ablation studies further confirmed the critical contributions of the directed graph construction, lag-aware propagation, directional gating, and multi-scale temporal encoding components. Beyond numerical accuracy, the learned sparse directed graphs provide valuable transportation interpretability by revealing potential non-local functional correlations and delayed propagation relationships. In practical applications, Traffic-DiMAGNet serves as a reliable forecasting and decision-support component for proactive freeway management. Beyond improving point prediction accuracy, the interpretable directed lag relationships explicitly reveal the spatio-temporal propagation paths and time delays of traffic disturbances. These insights offer crucial spatio-temporal priors for downstream operational tasks, such as anticipating bottleneck activations for congestion warning, and informing the timing and location of ramp metering or variable speed control. While this study focuses on the

forecasting framework and graph interpretability rather than directly evaluating closed-loop traffic-control strategies, the learned directional dependencies provide actionable intelligence that can significantly enhance the situational awareness and decision-making process of traffic management systems.

Despite these advantages, several limitations should be noted. First, the lead–lag traffic prior matrix $\mathbf{P}$ and the delay matrix $\mathbf{T}_{\text{lag}}$ are estimated once from the training data and remain fixed during inference. Consequently, they primarily capture long-term average propagation patterns and may not fully reflect time-varying shockwave speeds and directional influences under changing traffic regimes or incidents. Although the proposed feature-dependent directional gate $\mathbf{G}^{(l)}$ can partially compensate for such mismatches by adaptively reweighting forward and reverse propagation branches based on current latent features, it does not explicitly update the underlying prior graph structure or lag assignments online. Therefore, it improves robustness to prior mismatch but cannot fully replace a dynamic time-varying prior estimation mechanism. Second, the experiments were conducted on Caltrans PeMS freeway datasets, and the transferability of the model to dense urban arterial networks or mixed freeway–arterial systems requires further investigation. Third, although the proposed model shows favorable computational efficiency, actual deployment performance also depends on data transmission, preprocessing, and integration with existing traffic management platforms. Future research will focus on exploring online adaptive updating of lead–lag priors, incorporating exogenous variables (e.g., weather, incidents) to improve robustness under non-recurrent conditions, and investigating hardware-aware optimization to further reduce inference latency for real-time intelligent transportation systems.

Author Contributions: Conceptualization, Y.G., J.L. and H.Y.; methodology, J.L.; software, J.L.; validation, Z.W., Y.B. and H.Y.; formal analysis, Y.G. and H.Y.; investigation, Z.W. and Y.B.; data curation, Z.W.; writing—original draft preparation, J.L.; writing—review and editing, H.Y.; visualization, J.L. and Z.W.; resources, Y.G.; project administration, Y.G.; funding acquisition, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Xinmiao Talent Program of Zhejiang Province, grant number 250709.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available from the corresponding author upon reasonable request. Restrictions apply due to privacy.

Acknowledgments: The authors sincerely thank the editor and the reviewers for their valuable comments and helpful suggestions, which greatly improved this manuscript.

Conflicts of Interest: The authors declare that the first author is affiliated with the Guangzhou Institute of Internet of Things, which is a public academic and research institution. This affiliation does not constitute a conflict of interest in relation to the research presented in this manuscript. The authors declare no other conflicts of interest.

References

1. Cao, R.; Wang, D.Z.W. ImFCP: An iterative framework for completing and predicting spatially and temporally incomplete direction-based traffic data. *Transp. Res. Part C Emerg. Technol.* **2025**, *177*, 105172. [[CrossRef](#)]
2. Li, J.; Yu, C.; Shen, Z.; Su, Z.; Ma, W. A survey on urban traffic control under mixed traffic environment with connected automated vehicles. *Transp. Res. Part C Emerg. Technol.* **2023**, *154*, 104258. [[CrossRef](#)]
3. Kong, W.; Guo, Z.; Liu, Y. Spatio-temporal pivotal graph neural networks for traffic flow forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; pp. 8627–8635.

4. Jin, G.; Liang, Y.; Fang, Y.; Shao, Z.; Huang, J.; Zhang, J.; Zheng, Y. Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. *IEEE Trans. Knowl. Data Eng.* **2023**, *36*, 5388–5408. [[CrossRef](#)]
5. Zheng, Q.; Shao, M.; Zhang, Y. TLAST: A time-lag aware spatial-temporal transformer for traffic flow forecasting. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 13144–13158. [[CrossRef](#)]
6. Zhang, Q.; Chang, J.; Meng, G.; Xiang, S.; Pan, C. Spatio-temporal graph structure learning for traffic forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 7–12 February 2020; pp. 1177–1185.
7. Ali, A.; Naeem, H.M.Y.; Sharafian, A.; Qiu, L.; Wu, Z.; Bai, X. Dynamic multi-graph spatio-temporal learning for citywide traffic flow prediction in transportation systems. *Chaos Solitons Fract.* **2025**, *199*, 116898. [[CrossRef](#)]
8. Zhang, W.; Zhu, F.; Lv, Y.; Tan, C.; Liu, W.; Zhang, X.; Wang, F.Y. AdapGL: An adaptive graph learning algorithm for traffic prediction based on spatio-temporal neural networks. *Transp. Res. Part C Emerg. Technol.* **2022**, *139*, 103659. [[CrossRef](#)]
9. Tygesen, M.N.; Pereira, F.C.; Rodrigues, F. Unboxing the graph: Towards interpretable graph neural networks for transport prediction through neural relational inference. *Transp. Res. Part C Emerg. Technol.* **2023**, *146*, 103946. [[CrossRef](#)]
10. Yu, B.; Yin, H.; Zhu, Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), Stockholm, Sweden, 13–19 July 2018; pp. 3634–3640.
11. Li, Y.; Yu, R.; Shahabi, C.; Liu, Y. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In Proceedings of the International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018.
12. Guo, S.; Lin, Y.; Feng, N.; Song, C.; Wan, H. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 922–929.
13. Su, Y.; Du, J.; Li, Y.; Li, X.; Liang, R.; Hua, Z.; Zhou, J. Trajectory forecasting based on prior-aware directed graph convolutional neural network. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 16773–16785. [[CrossRef](#)]
14. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; Zhang, C. Connecting the dots: Multivariate time series forecasting with graph neural networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, San Diego, CA, USA, 22–26 August 2020; pp. 753–763.
15. Bai, L.; Yao, L.; Li, C.; Wang, X.; Wang, C. Adaptive graph convolutional recurrent network for traffic forecasting. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 17804–17815.
16. Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; Sun, L. Transformers in time series: A survey. In Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI), Macao, China, 19–25 August 2023; pp. 6778–6786.
17. Song, C.; Lin, Y.; Guo, S.; Wan, H. 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; pp. 914–921.
18. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Zhang, C. Graph WaveNet for deep spatial-temporal graph modeling. In Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI), Macao, China, 10–16 August 2019; pp. 1907–1913.
19. Wu, S.; Hu, Y. Traffic flow prediction based on spatio-temporal aggregated graph neural networks. *Transp. Res. Rec.* **2025**, *2679*, 573–588. [[CrossRef](#)]
20. Su, J.; Wang, H.; Przystupa, K.; Kochan, O.; Liu, D. Dynamic spatio-temporal residual hypergraph convolutional networks for traffic flow forecasting. *Transp. Res. Rec.* **2025**, *2679*, 596–614. [[CrossRef](#)]
21. Jiang, J.; Han, C.; Zhao, W.X.; Wang, J. PDFormer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In Proceedings of the 37th AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; pp. 4365–4373.
22. Wang, Y.; Xu, Y.; Yang, J.; Wu, M.; Li, X.; Xie, L.; Chen, Z. Fully-connected spatial-temporal graph for multivariate time-series data. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; pp. 15715–15724.
23. Zheng, Y.; Koh, H.Y.; Jin, M.; Chi, L.; Phan, K.T.; Pan, S.; Chen, Y.P.P.; Xiang, W. Correlation-aware spatial-temporal graph learning for multivariate time-series anomaly detection. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *35*, 11802–11816. [[CrossRef](#)] [[PubMed](#)]
24. Zhao, W.; Zhang, S.; Zhou, B.; Wang, B. Multi-spatio-temporal fusion graph recurrent network for traffic forecasting. *Eng. Appl. Artif. Intell.* **2023**, *124*, 106615. [[CrossRef](#)]
25. Shi, Z.; Zhang, Y.; Wang, J.; Qin, J.; Liu, X.; Yin, H.; Huang, H. DAGCRN: Graph convolutional recurrent network for traffic forecasting with dynamic adjacency matrix. *Expert Syst. Appl.* **2023**, *227*, 120259. [[CrossRef](#)]
26. Lan, S.; Ma, Y.; Huang, W.; Wang, W.; Yang, H.; Li, P. DstAGNN: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting. In Proceedings of the 39th International Conference on Machine Learning (ICML), Baltimore, MD, USA, 17–23 July 2022; pp. 11906–11917.
27. Liang, Y.; Liu, S.; Bai, Y.; Gong, Y.; Zhu, T. DSM-STWave: Enhancing traffic flow prediction for both offline and online scenarios. *Neurocomputing* **2026**, *661*, 131836. [[CrossRef](#)]

28. Zhan, F.; Zhou, X.; Li, S.; Jia, D.; Song, H. Learning latent ODEs with graph RNN for multi-channel time series forecasting. *IEEE Signal Process. Lett.* **2023**, *30*, 1432–1436. [[CrossRef](#)]
29. Zeghina, A.; Leborgne, A.; Le Ber, F.; Vacavant, A. Deep learning on spatio-temporal graphs: A systematic review, methodological landscape, and research opportunities. *Neurocomputing* **2024**, *594*, 127861. [[CrossRef](#)]
30. Huang, S.; Wang, D.; Wu, X.; Tang, A. DsaNet: Dual self-attention network for multivariate time series forecasting. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM), Beijing, China, 3–7 November 2019; pp. 2129–2132.
31. Li, M.; Zhu, Z. Spatial-temporal fusion graph neural networks for traffic flow forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; pp. 4189–4196.
32. Fang, Z.; Long, Q.; Song, G.; Xie, K. Spatial-temporal graph ODE networks for traffic flow forecasting. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Virtual, 14–18 August 2021; pp. 364–373.
33. Chen, Y.; Segovia, I.; Gel, Y.R. Z-GCNETs: Time zigzags at graph convolutional networks for time series forecasting. In Proceedings of the 38th International Conference on Machine Learning (ICML), Virtual, 18–24 July 2021; pp. 1684–1694.
34. Chen, Y.; Segovia-Dominguez, I.; Coskunuzer, B.; Gel, Y. TAMP-S2GCNets: Coupling time-aware multipersistence knowledge representation with spatio-supra graph convolutional networks for time-series forecasting. In Proceedings of the International Conference on Learning Representations (ICLR), Virtual, 25–29 April 2022.
35. Rao, X.; Wang, H.; Zhang, L.; Li, J.; Shang, S.; Han, P. FOGS: First-order gradient supervision with learning-based graph for traffic flow forecasting. In Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI), Vienna, Austria, 23–29 July 2022; pp. 3926–3932.
36. Han, X.; Gong, S. LST-GCN: Long short-term memory embedded graph convolution network for traffic flow forecasting. *Electronics* **2022**, *11*, 2230. [[CrossRef](#)]
37. Izadi, M.; Safayani, M.; Mirzaei, A. Knowledge distillation on spatial-temporal graph convolutional network for traffic prediction. *Int. J. Comput. Appl.* **2025**, *47*, 45–56. [[CrossRef](#)]
38. Zhang, Q.; Gao, X.; Wang, H.; Yiu, S.M.; Yin, H. Efficient traffic prediction through spatio-temporal distillation. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 25 February–4 March 2025; pp. 1093–1101.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.