

Article

Dynamic Graph Information Bottleneck for Traffic Prediction

Jing Pang ^{1,†}, Minzhe Wu ^{2,†}, Bingxue Xie ³, Yanqiu Bi ^{4,*}  and Zhongbin Luo ^{5,6,*} ¹ Course Hero, Redwood City, CA 94063, USA² Department of Civil and Coastal Engineering, University of Florida, Gainesville, FL 32611, USA³ School of Engineering and Applied Science, University of Virginia, Charlottesville, VA 22903, USA⁴ School of Civil Engineering, Chongqing Jiaotong University, Chongqing 400074, China⁵ College of Computer Science, Chongqing University, Chongqing 400044, China⁶ China Merchants Chongqing Communications Research & Design Institute Co., Ltd., Chongqing 400067, China

* Correspondence: biyanqiu@cqjtu.edu.cn (Y.B.); luozhongbin@stu.cqu.edu.cn (Z.L.)

† These authors contributed equally to this work.

Abstract

Traffic forecasting in large-scale urban networks must operate reliably under imperfect sensing conditions, where measurements may contain noise or missing values. Most existing spatio-temporal graph neural networks focus primarily on modeling spatial-temporal dependencies, while paying limited attention to the propagation of irrelevant or unstable information through dynamic graph structures. In this work, we propose a Dynamic Graph Information Bottleneck (DGIB) framework that enhances prediction stability by introducing task-aware representation compression into dynamic graph learning. Instead of relying solely on architectural complexity, DGIB explicitly regulates the information flow within spatio-temporal embeddings through a variational bottleneck objective. The model adaptively constructs time-evolving adjacency matrices, extracts spatial features via graph convolutions, captures temporal dependencies using recurrent modeling, and constrains the latent representation to retain only predictive content relevant to future traffic states. By jointly optimizing topology adaptation and information-theoretic regularization in an end-to-end manner, the proposed framework mitigates the amplification of noisy or redundant signals in dynamic graphs. Experiments on multiple benchmark traffic datasets demonstrate that DGIB achieves competitive forecasting accuracy while maintaining strong robustness under noisy and incomplete data scenarios.

Keywords: traffic prediction; dynamic graph information bottleneck; spatio-temporal graph neural networks; variational representation learning; intelligent transportation systems



Academic Editor: Manohar Das

Received: 18 November 2025

Revised: 13 December 2025

Accepted: 29 January 2026

Published: 1 February 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

Urban traffic systems exhibit intricate interaction patterns that vary across both space and time. Flows recorded at different sensing locations are neither independent nor stationary; instead, they are coupled through congestion spillover, route choice dynamics, and control strategies, leading to continuously evolving dependency structures [1–3]. Modeling these intertwined dynamics requires learning representations that can simultaneously account for spatial correlations and temporal transitions, which remains a central challenge in traffic forecasting research [4,5].

The increasing availability of large-scale sensing infrastructures and Internet-of-Things (IoT) deployments has significantly expanded the volume and resolution of traffic data.

While such data richness provides opportunities for fine-grained modeling, it also amplifies sources of uncertainty. Sensor readings are often affected by transient disturbances, partial observations, and cross-location heterogeneity [6,7]. When these irregularities propagate through deep architectures without explicit regulation, they may distort latent representations and compromise predictive stability.

Graph Neural Networks (GNNs) have emerged as an effective framework for modeling structured traffic data [8,9]. By representing sensors as nodes and encoding physical or functional relationships as edges, graph-based architectures explicitly model spatial correlations that are difficult to capture using grid-based or fully connected formulations. When combined with temporal modules such as Long Short-Term Memory (LSTM), Gated Recurrent Units (GRUs), Temporal Convolutional Networks (TCNs), or attention mechanisms, these models jointly learn spatial interactions and temporal evolution patterns, achieving competitive performance on multiple benchmark datasets [10,11].

To enhance modeling flexibility, recent research further introduces adaptive and dynamic graph structures that allow adjacency relationships to evolve with traffic conditions [10]. Methods such as DGCRN and its variants learn time-dependent topologies based on historical or current states, enabling the receptive field of the model to adjust according to transient congestion patterns or demand shifts. Dynamic graph formulations represent an important advancement beyond static topology assumptions, offering improved expressiveness for capturing the inherently non-stationary nature of urban mobility systems.

Despite these successes, several challenges remain unresolved. Most existing GNN-based models assume that both node features and graph structures are complete and noise-free, an assumption rarely satisfied in practice. In real transportation systems, traffic data are affected by numerous uncertain factors such as sensor failures, temporary road closures, weather conditions, and abrupt fluctuations in demand. These disturbances lead to highly nonlinear and nonstationary traffic dynamics. In addition, the spatial topology of road networks can evolve over time due to traffic control measures or construction activities, resulting in dynamic adjacency relationships. When models trained under idealized conditions encounter such data corruption or distribution shifts, their prediction accuracy often degrades substantially [12]. Therefore, enhancing the robustness and generalization ability of traffic forecasting models under imperfect data conditions has become a critical and timely research objective.

One promising approach to achieve robust learning is to explicitly filter out redundant or irrelevant information in both the input data and the learned graph representations. The Information Bottleneck (IB) principle [13,14] provides a rigorous theoretical foundation for this goal. The IB principle offers a general framework for learning representations that emphasize task-relevant structure while suppressing variability that does not contribute to prediction. The central idea is to construct a latent variable that preserves as much information as possible about the target output, yet remains minimally informative about the raw input. This trade-off is formalized by simultaneously reducing the mutual information between the input and the latent representation and increasing the mutual information between the latent representation and the prediction target. The resulting embeddings tend to be compact, robust to noise, and resistant to spurious correlations.

Importantly, in our setting, the bottleneck is not used as an aggressive dimensionality-reduction mechanism, but rather as a task-aware regularizer that is jointly optimized with the forecasting loss. This joint optimization explicitly constrains the compressed representation to remain maximally predictive of future traffic states, ensuring that task-critical information is preserved even under highly noisy, incomplete, or perturbed sensing conditions. As a result, any information removed by the bottleneck is, by construction, irrelevant to the forecasting objective, which enables the learned representations to retain

essential spatio-temporal traffic dependencies while becoming more resilient to sensor noise, missing values, and spurious environmental disturbances.

Motivated by these theoretical properties, recent studies have incorporated IB-style regularization into graph learning algorithms [15–17]. These works demonstrate that constraining the information flow during graph representation learning can enhance generalization, promote stable training behavior, and counteract overfitting—particularly in settings where graphs exhibit noisy attributes, missing edges, or heterogeneous structures. Such findings suggest that information-theoretic regularization offers a principled means to improve the reliability of graph neural networks in real-world, imperfect data environments.

Motivated by these advances, we introduce a novel and robust framework termed the Dynamic Graph Information Bottleneck (DGIB) for traffic prediction. DGIB integrates the information bottleneck principle into a dynamic spatio-temporal graph neural network architecture to achieve accurate and stable predictions under challenging data conditions. The proposed framework consists of three main components: (i) a dynamic graph construction module that adaptively models the evolving spatial dependencies among traffic sensors; (ii) a lightweight Graph Convolutional Network (GCN) combined with a Long Short Term Memory (LSTM) to capture spatial and temporal correlations, respectively; and (iii) a Variational Information Bottleneck (VIB) regularization term that enforces information compression on latent node embeddings.

In the dynamic graph construction stage, DGIB employs a time-aware attention mechanism to generate a series of temporally adaptive adjacency matrices. This design enables the model to learn how spatial relationships change with time, reflecting the real evolution of traffic flows. The spatial module then applies graph convolution operations on each dynamic graph snapshot, aggregating information from neighboring nodes to encode spatial dependencies. The temporal module, implemented as a LSTM, captures sequential patterns and temporal dynamics in the spatially aggregated representations. Finally, a VIB layer is incorporated to regulate the latent space of the learned spatio-temporal features. By minimizing the mutual information between the latent representation and the input while preserving information relevant to the prediction target, the VIB term effectively suppresses redundant or noisy signals. This process prevents noise propagation through the network and yields compact, task-specific representations that improve generalization under imperfect conditions.

Unlike existing methods that treat the graph topology as either fixed or purely data-driven, DGIB jointly optimizes the graph structure, temporal modeling, and information compression within a unified end-to-end learning framework. The model dynamically adjusts its receptive field according to the changing traffic states while maintaining robustness against data uncertainty. This synergistic integration of dynamic graph learning and information-theoretic regularization allows DGIB to achieve an optimal trade-off between accuracy and stability. Empirical analyses further demonstrate that the proposed model maintains high prediction accuracy across diverse datasets and remains resilient when the data contain missing or noisy measurements.

In summary, the main contributions of this paper are as follows:

- We propose a new framework, Dynamic Graph Information Bottleneck (DGIB), that integrates the information bottleneck principle into dynamic spatio-temporal graph neural networks for robust traffic prediction.
- DGIB introduces a Variational Information Bottleneck (VIB) regularization term on latent node representations within a GCN + LSTM backbone, effectively filtering out noise and redundant information during spatio-temporal learning.

- A time-aware dynamic graph construction module is designed to adaptively learn spatial dependencies that evolve with time, enabling the model to handle dynamic and irregular traffic patterns.
- Extensive experiments conducted on multiple benchmark traffic flow and speed datasets demonstrate that DGIB achieves competitive prediction accuracy and superior robustness under missing or noisy data conditions, outperforming several state-of-the-art baselines.

Overall, DGIB offers a principled and practical solution for reliable traffic forecasting in real-world environments characterized by incomplete and noisy observations. Its integration of dynamic graph learning with information-theoretic regularization provides a new perspective for developing robust spatio-temporal graph neural networks applicable not only to traffic prediction but also to other domains involving structured sequential data.

The remainder of this paper is organized as follows. Section 2 reviews related research on graph-based traffic prediction and information-theoretic representation learning. Section 3 presents the proposed Dynamic Graph Information Bottleneck (DGIB) model in detail, including its dynamic graph construction, spatio-temporal modeling, and variational regularization. Section 4 describes the experimental setup, datasets, evaluation metrics, and presents the performance comparison and ablation studies. Finally, Section 5 concludes this paper and discusses potential future research directions.

2. Related Work

2.1. Early Statistical and Machine Learning Methods

Early traffic flow prediction approaches were mainly based on statistical time-series models, such as the Historical Average (HA) [18] and the Auto-Regressive Integrated Moving Average (ARIMA) [19]. These models assume linear temporal correlations and require handcrafted features, limiting their ability to capture nonlinear and dynamic traffic behavior. With advances in sensing and computation, machine-learning methods such as Support Vector Regression (SVR) [20] and Kalman Filtering [21] were adopted. Hidden Markov Models (HMMs) have also been applied for traffic situation assessment and short-term prediction, demonstrating the importance of modeling latent traffic states [22]. While they improve short-term forecasting, they remain inadequate for large-scale, non-Euclidean, and highly correlated networks.

2.2. Deep Learning for Spatiotemporal Modeling

The emergence of deep learning enabled powerful sequence and spatial representation learning. Convolutional Neural Networks (CNNs) [23] and Recurrent Neural Networks (RNNs) [24–26] were first used on grid-structured maps, where traffic states are treated as images or sequences. Long Short-Term Memory (LSTM) [27,28] and Gated Recurrent Unit (GRU) [29] networks capture nonlinear temporal dependencies, while CNNs extract localized spatial patterns. However, grid segmentation distorts road geometry and cannot model the irregular topology of urban networks.

2.3. Graph Neural Networks and Dynamic Spatiotemporal Modeling

To overcome the limitations of grid-based models, Graph Neural Networks (GNNs) have become the dominant paradigm for traffic forecasting. A road network is represented as a graph $G = (V, E)$, where nodes correspond to sensors or intersections and edges denote physical or functional connectivity. The objective is to learn a mapping from historical node features to future states while jointly modeling spatial dependencies across nodes and temporal dynamics across time steps.

2.3.1. Spatial Modeling with Graph Convolutions

Early works extend convolution to the graph domain through spectral and spatial formulations. Spectral GCNs [30–33] define convolution via the graph Laplacian eigenbasis, enabling message aggregation from connected nodes weighted by normalized adjacency. Although mathematically elegant, spectral filters are computationally expensive for large graphs and require fixed topologies. Spatial GCNs perform direct neighborhood aggregation. The Diffusion Convolutional Recurrent Network (DCRNN) interprets information propagation as a bidirectional diffusion process along directed edges, integrating diffusion convolution with GRU-based sequence modeling. This formulation naturally captures asymmetric traffic flows and directional influences. Diffusion Graph Convolution (DGC) [8] and its variants improve efficiency on sparse graphs and alleviate over-smoothing by introducing multi-step random-walk diffusion kernels. Message Passing Neural Networks (MPNNs) [34,35] generalize GCNs by learning node-wise message and update functions, thus capturing higher-order and nonlinear spatial interactions. GraphSAGE [32] enables inductive learning on large or evolving networks by sampling and aggregating neighborhood embeddings, providing scalability for networks with thousands of sensors.

2.3.2. Temporal Modeling and Joint Architectures

Temporal dynamics are usually captured through recurrent, convolutional, or attention modules combined with GCNs. STGCN [36] stacks gated temporal convolutions and spatial graph convolutions in a fully convolutional manner, avoiding recurrent dependencies and enabling parallel training. Graph WaveNet introduces dilated causal convolutions for long temporal receptive fields and augments spatial learning with a data-adaptive adjacency matrix learned end-to-end, thereby uncovering hidden correlations beyond physical connectivity. AST-GCN [37] integrates spatial and temporal attention mechanisms that focus on short-term, daily, and weekly periodic patterns, improving long-horizon forecasts. GMAN [38] replaces convolutions with multi-head self-attention in both space and time, demonstrating strong long-range dependency modeling. These designs highlight a trend from local convolutional aggregation toward global attention-based architectures capable of modeling complex temporal rhythms.

2.3.3. Adaptive and Dynamic Graph Structures

While the above models assume static graphs, real traffic relationships evolve with congestion patterns, road closures, or external events. Adaptive Graph Convolutional Recurrent Networks (AGCRN) [39,40] learn node embeddings and construct adjacency matrices from data, capturing latent spatial dependencies without predefined topology. Dynamic Graph Convolutional Recurrent Networks (DGCRN) [41] and Double-Dynamic GCRN (DDGCRN) [42] further generate time-varying adjacency matrices, enabling the model to represent evolving spatial dependencies at each time step. D2STGNN [43] decomposes traffic signals into intrinsic and diffusion components, explicitly modeling both stationary and propagating dynamics. Other approaches adopt attention-based graph updates or neural ordinary differential equation (ODE) and controlled differential equation (CDE) frameworks to represent continuous-time evolution of traffic states. These adaptive and dynamic strategies significantly enhance the flexibility and expressiveness of GNN-based traffic forecasters, making them more suitable for real-world, non-stationary environments.

2.4. Information Bottleneck Representation Learning

The Information Bottleneck (IB) framework [13,44] provides a principled foundation for learning task-oriented representations by explicitly characterizing the trade-off between relevance and compression. Given an input variable X and a prediction target Y , the goal is to derive a latent representation Z that preserves the information in X that is most useful for predicting Y , while discarding nuisance variability and redundant patterns. In the ideal formulation, IB solves a constrained optimization problem in which the predictive relevance $I(Y; Z)$ is maximized under the requirement that the representation complexity $I(X; Z)$ does not exceed a predefined threshold α . This leads to an interpretation of Z as a minimal sufficient statistic of X for Y .

Directly solving this constrained problem is generally intractable when X , Y , and Z are high-dimensional or continuous. Consequently, most practical implementations rely on the IB Lagrangian:

$$\mathcal{L}_{\text{IB}} = I(Y; Z) - \beta I(X; Z), \quad (1)$$

where $\beta > 0$ balances sufficiency, quantified by $I(Y; Z)$, against simplicity, quantified by $I(X; Z)$. Larger values of β encourage stronger compression and typically yield more noise-resilient or sparse latent representations, while smaller values prioritize predictive fidelity at the cost of increased input dependence. A broad class of variational, kernel-based, and contrastive approaches can be viewed as practical relaxations of this Lagrangian.

In graph learning, the IB principle has been adapted to multiple contexts. For graph representation learning, IB-style regularization has shown promise in mitigating over-smoothing, suppressing redundant neighborhood information, and improving robustness to noisy attributes or missing edges [15]. IB has also been explored for graph interpretability, where the objective is to identify subgraphs that carry the highest predictive information with respect to a target label, as demonstrated by recent work such as BrainIB [17]. These studies illustrate the value of information-theoretic constraints for extracting stable and meaningful representations in static graph scenarios.

However, to the best of our knowledge, the IB framework has not been extended to dynamic graph convolutional neural networks for robust and generalizable traffic forecasting. Existing spatio-temporal GNNs predominantly focus on architectural innovations or dynamic adjacency construction, but they do not explicitly regulate the flow of information across evolving graph structures. Incorporating IB-based regularization into dynamic GNNs addresses this methodological gap by enabling suppression of irrelevant temporal-spatial fluctuations, enhancing model stability under distribution shifts, and promoting compact and transferable dynamic representations suitable for real-world traffic prediction applications.

3. Our Method

3.1. Overview

Our framework, illustrated in Figure 1, adopts a lightweight dynamic spatiotemporal design that integrates time-varying graph construction, graph-convolution-based spatial modeling, an LSTM module for temporal dependency learning, and a node-wise Variational Information Bottleneck (VIB) regularization. Given historical traffic observations $\{X_t \in \mathbb{R}^{N \times d}\}_{t=1}^T$, the model predicts future states $\{\hat{X}_t\}_{t=T+1}^{T+T_p}$ while simultaneously suppressing redundant or noisy dependencies through information-theoretic compression. The architecture consists of three main components: (i) dynamic graph generation; (ii) spatial modeling via GCN; (iii) temporal modeling using a Long Short-Term Memory (LSTM) network; and (iv) compact representation learning with VIB.

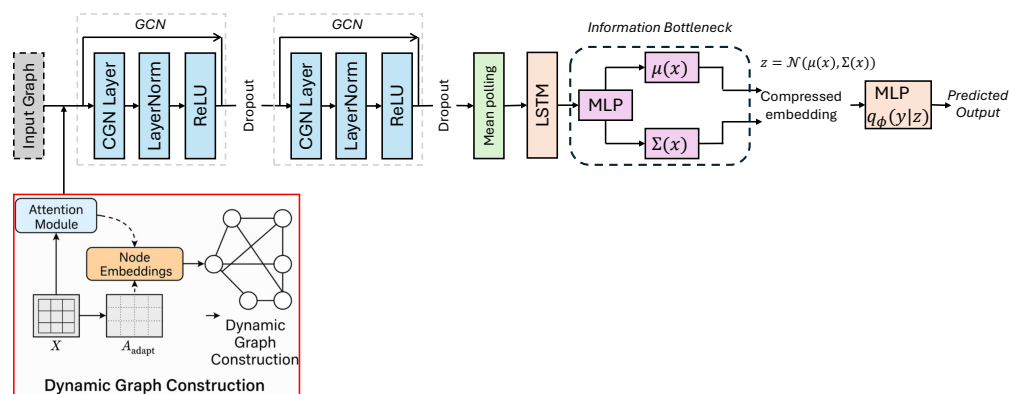


Figure 1. Conceptual illustration of the DGIB framework, showing dynamic graph construction, spatiotemporal feature extraction, and information-bottleneck-based compression for robust traffic prediction.

3.2. Dynamic Graph Construction

To capture the time-varying spatial dependencies among traffic sensors, we construct a dynamic adjacency matrix $A_t \in \mathbb{R}^{N \times N}$ at every time step t . The construction follows a structured attention mechanism that incorporates (i) historical traffic signals, (ii) temporal context, and (iii) learnable node embeddings. This section defines all symbols and describes the dynamic graph generation process.

3.2.1. Input Representations

Let $X_t \in \mathbb{R}^{N \times C}$ denote the traffic features at time t , where C is the number of channels (e.g., speed and flow). Two types of temporal context are used:

$$T_t^D \in \mathbb{R}^{d_e}, \quad T_t^W \in \mathbb{R}^{d_e}, \quad (2)$$

representing the day-of-week and time-of-day embeddings, respectively. In addition, two learnable node-embedding matrices,

$$E_o, E_t \in \mathbb{R}^{N \times d_e}, \quad (3)$$

encode upstream and downstream structural priors of the road network.

3.2.2. Contextual Feature Construction

For each node, we concatenate the multi-view features to obtain upstream and downstream representations:

$$D_t^{(u)} = \text{Concat}[X_t, T_t^D, T_t^W, E_o], \quad D_t^{(d)} = \text{Concat}[X_t, T_t^D, T_t^W, E_t]. \quad (4)$$

Both $D_t^{(u)}$ and $D_t^{(d)}$ are in $\mathbb{R}^{N \times d}$ after projection.

3.2.3. Directional Dynamic Adjacency

Scaled dot-product attention is used to compute directional correlation matrices:

$$P_{u,t}^{\text{dyn}} = \text{Softmax} \left(\frac{(D_t^{(u)} W_Q)(D_t^{(u)} W_K)^\top}{\sqrt{d}} \right), \quad (5)$$

$$P_{d,t}^{\text{dyn}} = \text{Softmax} \left(\frac{(D_t^{(d)} W_Q)(D_t^{(d)} W_K)^\top}{\sqrt{d}} \right), \quad (6)$$

where $W_Q, W_K \in \mathbb{R}^{d \times d}$ are learnable projection matrices.

The dynamic adjacency is obtained by bidirectional averaging:

$$A_t^{\text{dyn}} = \frac{1}{2} (P_{u,t}^{\text{dyn}} + P_{d,t}^{\text{dyn}}). \quad (7)$$

3.2.4. Fusion with Static and Adaptive Graphs

The final adjacency combines three components:

- A^{static} : physical road-network graph based on sensor distance or connectivity;
- A^{adapt} : data-driven graph learned from node embeddings

$$A^{\text{adapt}} = \text{ReLU}(E_o E_t^\top);$$

- A_t^{dyn} : time-dependent correlation graph.

Figure 2 provides examples of the three adjacency matrices considered in this study. In Figure 2a, the static graph encodes the inherent and nearly unchanging structure of the road network. Figure 2b depicts the adaptive graph, which is learned from data and updated through backpropagation to capture latent spatial correlations. Finally, Figure 2c illustrates the dynamic graph, where a distinct adjacency matrix is generated at each time step to represent evolving spatial patterns over time.

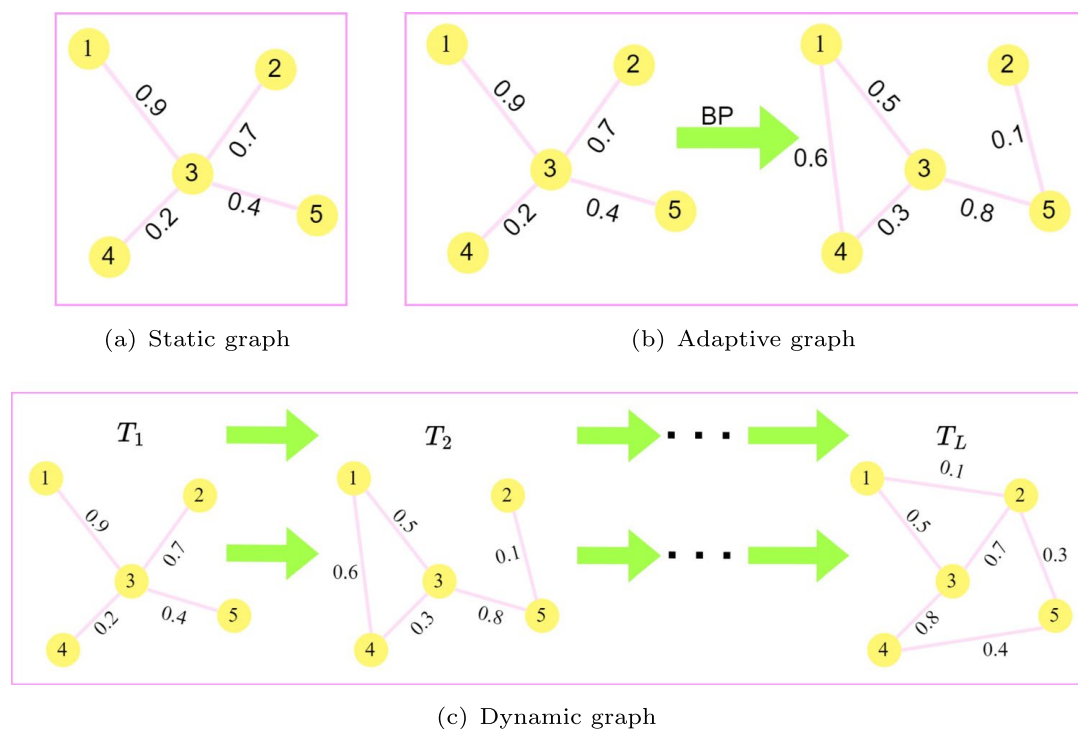


Figure 2. Illustration of the Three Graph Structures with Enhanced Visual Clarity. The circled numbers denote node indices, and the edge labels represent normalized connection weights.

These three graphs are fused by convex combination:

$$A_t = \alpha A^{\text{static}} + \beta A^{\text{adapt}} + \gamma A_t^{\text{dyn}}, \quad \alpha + \beta + \gamma = 1. \quad (8)$$

Graph Sparsification and Normalization

For numerical stability, we keep the top- k strongest neighbors per node:

$$A_t(i, j) = \begin{cases} A_t(i, j), & j \in \text{Top-}k(i), \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

Self-loops are added, and the matrix is symmetrically normalized:

$$\hat{A}_t = D_t^{-1/2}(A_t + I)D_t^{-1/2}, \quad D_t = \text{diag}(A_t \mathbf{1}). \quad (10)$$

The normalized matrices $\{\hat{A}_t\}$ are then used by the spatial GCN layers.

At each time step, the dynamic adjacency matrix is constructed using the current traffic observations, temporal embeddings, and learnable node embeddings. The raw affinity scores are then subject to sparsification and row-wise normalization to ensure numerical stability and prevent excessive graph densification. This procedure guarantees that the learned topology is both adaptive to traffic dynamics and well-conditioned for stable message passing.

3.3. Spatial Modeling via Graph Convolution

Given the dynamic normalized adjacency $\hat{A}_t$, the spatial dependencies are captured using a GCN module applied to each snapshot of the traffic state. The hidden representation at layer $\ell+1$ is computed as

$$H_t^{(\ell+1)} = \sigma(\hat{A}_t H_t^{(\ell)} W^{(\ell)}), \quad (11)$$

where $\sigma(\cdot)$ denotes a non-linear activation (ReLU) and $W^{(\ell)}$ is a trainable weight matrix. We initialize $H_t^{(0)} = X_t$, and one or two GCN layers are typically sufficient to model the local spatial patterns. Through the dynamic $\hat{A}_t$, the GCN adaptively updates its receptive field over time, enabling the model to track evolving spatial relationships in the traffic network.

3.4. Temporal Modeling via LSTM

After extracting spatial representations $\{H_t^{(sp)}\}_{t=1}^T$ from the GCN, temporal dependencies are modeled using a Long Short-Term Memory (LSTM) network. For each time step t , the LSTM updates are given by

$$i_t = \sigma(W_i H_t^{(sp)} + U_i h_{t-1} + b_i), \quad (12)$$

$$f_t = \sigma(W_f H_t^{(sp)} + U_f h_{t-1} + b_f), \quad (13)$$

$$o_t = \sigma(W_o H_t^{(sp)} + U_o h_{t-1} + b_o), \quad (14)$$

$$\tilde{c}_t = \tanh(W_c H_t^{(sp)} + U_c h_{t-1} + b_c), \quad (15)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \quad (16)$$

$$h_t = o_t \odot \tanh(c_t), \quad (17)$$

where i_t , f_t , and o_t denote the input, forget, and output gates, respectively; c_t and h_t represent the memory cell and hidden state; W_* and U_* are learnable weight matrices; and $\sigma(\cdot)$ denotes the sigmoid activation.

The LSTM processes the spatial embeddings sequentially and produces the final spatiotemporal representation $H_t^{(st)} = h_t$, which serves as input to the variational bottleneck module and the prediction head. The use of LSTM enables the framework to capture both short-term and long-term temporal dependencies while maintaining robustness to dynamic variations in the learned adjacency structures.

3.5. Variational Information Bottleneck Formulation

The information bottleneck principle aims to learn a compressed representation Z of the spatio-temporal features H that preserves task-relevant information about the target variable Y while discarding nuisance variations. Formally, the optimization objective follows the classical trade-off:

$$\min_{q_\phi(z|h)} -I(Z;Y) + \beta I(Z;H), \quad (18)$$

where $\beta > 0$ controls the balance between compression and prediction fidelity. However, direct computation of the mutual information (MI) terms in Equation (18) is intractable. Following the variational information bottleneck framework [14], we introduce tractable variational bounds for both $I(Z;Y)$ and $I(Z;H)$.

3.5.1. Variational Upper Bound of $-I(Z;Y)$

$$I(Z;Y) = \iint p(y,z) \log \frac{p(y|z)}{p(y)} dydz = \iint p(y,z) \log p(y|z) dydz + H(Y), \quad (19)$$

where $H(Y)$ is the entropy of labels and can be ignored during training. Since $p(y|z)$ is intractable, we approximate it with a variational distribution $q_\phi(y|z)$. By the non-negativity of the Kullback–Leibler (KL) divergence between $p(y|z)$ and $q_\phi(y|z)$,

$$\int p(y|z) \log p(y|z) dy - \int p(y|z) \log q_\phi(y|z) dy \geq 0, \quad (20)$$

we obtain the following upper bound:

$$-I(Z;Y) \leq - \iint p(y,z) \log q_\phi(y|z) dydz. \quad (21)$$

3.5.2. Variational Upper Bound of $I(Z;H)$

Similarly, the mutual information $I(Z;H)$ can be expressed as

$$I(Z;H) = \iint p(z,h) \log \frac{p(z|h)}{p(z)} dzdh. \quad (22)$$

Introducing a variational prior $r(z)$ to approximate $p(z)$ yields

$$I(Z;H) \leq \iint p(z,h) \log \frac{p(z|h)}{r(z)} dzdh. \quad (23)$$

Combining Equations (21) and (23), the variational upper bound of the information bottleneck objective in Equation (18) can be written as

$$\mathcal{L}_{\text{VIB}} = \mathbb{E}_{p(y,h)q_\phi(z|h)} [-\log q_\phi(y|z)] + \beta \text{KL}(q_\phi(z|h) \parallel r(z)), \quad (24)$$

where $q_\phi(z|h) = \mathcal{N}(\mu(h), \text{diag}(\sigma^2(h)))$ and $r(z) = \mathcal{N}(0, I)$ are Gaussian encoder and prior, respectively. The first term corresponds to the prediction loss (e.g., mean absolute error in regression), while the second KL term regularizes the latent representation, enforcing compression and robustness.

The above objective naturally integrates with our dynamic GCN+LSTM backbone: H denotes the spatio-temporal embedding obtained after LSTM module, $q_\phi(z|h)$ is implemented as a lightweight two-layer MLP, and $q_\phi(y|z)$ is realized by the prediction

head $f_\psi(\cdot)$. This variational formulation enables end-to-end optimization using stochastic reparameterization.

3.6. Training Procedure

The entire training pipeline is summarized in Algorithm 1. At each iteration, the attention-based module produces a dynamic adjacency matrix, and the resulting graph snapshots are processed by the GCN and LSTM modules to capture spatial and temporal dependencies, and the latent representations are regularized by the VIB objective in Equation (24).

The resulting model retains the efficiency of standard GCN+LSTM architectures while leveraging dynamic graph adaptivity and information-theoretic regularization. The VIB term effectively filters out irrelevant temporal noise and enhances robustness against fluctuations in the dynamic adjacency matrices.

3.7. End-to-End Training

The proposed DGIB framework is optimized in a fully end-to-end manner. All modules, including dynamic graph construction, spatial graph convolutions, temporal modeling, variational encoding, and the prediction head, form a single computational graph whose parameters are updated jointly via backpropagation.

At each training iteration, the model receives an input sequence $\{X_t\}_{t=1}^T$. The dynamic graph construction module first computes a time-varying adjacency matrix A_t using Equations (5)–(8), incorporating traffic signals, temporal embeddings, and learnable node embeddings. After sparsification and normalization, the resulting matrices $\{\hat{A}_t\}$ parameterize the spatial message passing of the GCN layers.

The GCN module produces spatial embeddings $\{H_t^{(\text{sp})}\}$, which are sequentially aggregated by the LSTM module to obtain spatio-temporal representations $\{H_t^{(\text{st})}\}$. These representations are then passed through the variational encoder $q_\phi(z|h)$, producing Gaussian latent variables whose reparameterized samples $\{z_t\}$ feed the prediction head $f_\psi(\cdot)$. For a given historical input sequence, the prediction head $f_\psi(\cdot)$ directly outputs the corresponding future sequence $\{\hat{Y}_{t+1}, \dots, \hat{Y}_{t+\tau}\}$ in a single forward pass.

The entire network is trained by minimizing the variational information bottleneck objective in Equation (24), which combines the prediction loss with the KL regularization on the latent distribution. Crucially, the gradients of this objective propagate through all components, including the dynamic graph formation steps. Thus, the attention-based graph generator, node embeddings, GCN filters, LSTM parameters, VIB encoder, and prediction head are jointly optimized to produce spatially and temporally coherent representations.

This end-to-end formulation ensures that the learned dynamic graphs are directly aligned with the forecasting objective under the information-theoretic constraint, and that the entire system adapts holistically to data-driven temporal variations, structural dependencies, and robustness requirements.

It is important to emphasize that the dynamic topology refinement process is not iterative at inference time. Once the model parameters are fixed, the refined adjacency matrices are generated through a single forward pass without feedback loops across prediction horizons. Therefore, the topology remains stable during real-time deployment and does not suffer from error accumulation caused by recursive refinement.

Algorithm 1 End-to-End Training of the Dynamic Graph Information Bottleneck (DGIB) Model

Require: Input sequence $\{X_t\}_{t=1}^T$; static graph A^{static} ; node embeddings E_o, E_t ; temporal embeddings T_t^D, T_t^W ; information-bottleneck coefficient β_{IB} ; learning rate η .

1: **while** not converged **do**

2: **for** $t = 1$ to T **do**

3: Construct upstream and downstream contextual features:

$$D_t^{(u)} = \text{Concat}(X_t, T_t^D, T_t^W, E_o), \quad D_t^{(d)} = \text{Concat}(X_t, T_t^D, T_t^W, E_t).$$

4: Compute directional attentional graphs:

$$P_{u,t}^{\text{dyn}} = \text{Softmax}\left(\frac{(D_t^{(u)} W_Q)(D_t^{(u)} W_K)^\top}{\sqrt{d}}\right),$$

$$P_{d,t}^{\text{dyn}} = \text{Softmax}\left(\frac{(D_t^{(d)} W_Q)(D_t^{(d)} W_K)^\top}{\sqrt{d}}\right).$$

5: Compute dynamic adjacency:

$$A_t^{\text{dyn}} = \frac{1}{2}(P_{u,t}^{\text{dyn}} + P_{d,t}^{\text{dyn}}).$$

6: Fuse with static and adaptive graphs:

$$A_t = \alpha A^{\text{static}} + \beta A^{\text{adapt}} + \gamma A_t^{\text{dyn}}.$$

7: Sparsify and normalize:

$$\hat{A}_t = D_t^{-1/2}(A_t + I)D_t^{-1/2}.$$

8: **end for**

9: Apply GCN to obtain spatial embeddings:

$$H_t^{(\text{sp})} = \sigma(\hat{A}_t X_t W_{\text{sp}}).$$

10: Apply LSTM over $\{H_t^{(\text{sp})}\}$ to obtain spatio-temporal embeddings:

$$H_1^{(\text{st})}, \dots, H_T^{(\text{st})} = \text{LSTM}(H_1^{(\text{sp})}, \dots, H_T^{(\text{sp})}).$$

11: Encode latent variables (mean and variance):

$$(\mu_t, \log \sigma_t^2) = \text{MLP}_\phi(H_t^{(\text{st})}).$$

12: Reparameterize:

$$z_t = \mu_t + \sigma_t \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

13: Predict multi-step output:

$$\hat{Y}_{t+1:t+\tau} = f_\psi(z_t).$$

14: Compute total loss:

$$\mathcal{L} = \text{MAE}(\hat{Y}_{t+1:t+\tau}, Y_{t+1:t+\tau}) + \beta_{\text{IB}} \cdot \text{KL}(q_\phi(z|h) \| r(z)).$$

15: Update all parameters

$$\{\theta_{\text{graph}}, \theta_{\text{GCN}}, \theta_{\text{LSTM}}, \phi, \psi\}$$

using Adam.

16: **end while**

4. Experiments

To comprehensively assess the effectiveness of the proposed model, we performed extensive experiments on four publicly available spatiotemporal traffic datasets that are widely adopted in recent studies. This section first outlines the overall experimental configuration, including dataset descriptions, baseline models, and parameter settings. We then report comparative results against a series of representative approaches to evaluate predictive accuracy and efficiency. Furthermore, a set of ablation studies was designed to examine the contribution of each major component within the proposed framework. Finally, visual analyses of both ground-truth and predicted traffic states are presented to intuitively demonstrate the model's capability in capturing dynamic spatiotemporal patterns.

We evaluated the proposed framework on four large-scale, real-world traffic datasets that are commonly used in spatiotemporal forecasting research. The statistical characteristics of these datasets are summarized in Table 1. Each dataset contains the average vehicular speed (measured in miles per hour) collected from loop detectors deployed across different road segments. To ensure consistency and comparability with prior studies [8,42], we adopted standard preprocessing procedures, including data normalization, temporal alignment, and the removal of incomplete records. A brief description of each dataset is provided below.

- PEMS7(M): This dataset corresponds to the medium-scale subset of District 7 in California, which covers the Los Angeles metropolitan area. It contains measurements from 228 sensors collected over a period of four months (May–August 2012). Each record represents the average traffic speed observed during a 5 min interval.
- PEMS7(L): The large-scale subset of District 7 includes a wider coverage of the Los Angeles road network with 1029 sensors deployed across major highways. It provides approximately four months of data with the same temporal resolution as PEMS7(M), reflecting a more complex spatial dependency structure due to its higher sensor density.
- PEMS4: This dataset covers District 4, which encompasses the San Francisco Bay Area. It includes 307 sensors and records traffic speed data collected over two months (January–February 2018). Compared with the PEMS7 series, PEMS4 exhibits more pronounced regional heterogeneity and temporal fluctuations, making it suitable for evaluating model generalization across geographic domains.
- PEMS8: The PEMS8 dataset is collected from District 8 in California, covering the San Bernardino region. It consists of 170 sensors and contains two months of data (July–August 2016). This dataset features diverse traffic dynamics caused by varying congestion levels and environmental factors, providing a complementary benchmark for cross-district performance assessment.

The main statistical properties of the four benchmark datasets are presented in Table 1, including the number of sensors, temporal length, and data collection period.

Table 1. Summary of the four benchmark datasets used in this study.

Dataset	Number of Sensors	Timesteps	Time Range
PEMS7(M)	228	12,672	1 May 2012–31 August 2012
PEMS7(L)	1029	12,672	1 May 2012–31 August 2012
PEMS4	307	16,992	1 January 2018–28 February 2018
PEMS8	170	17,856	1 July 2016–31 August 2016

For each dataset, the spatial adjacency matrix was constructed according to the physical topology of the corresponding road network, where edge weights reflect the pairwise distance between sensors. The traffic speed measurements were standardized using Z-score

normalization to ensure comparable scales across all nodes. To maintain consistency with prior studies, we adopted the same data partitioning strategy as in earlier benchmark works. Time-series samples were generated using a sliding-window mechanism with a window size of $L \times \tau$, where the first L intervals represent historical observations and the following τ intervals correspond to prediction targets. For all four datasets (PEMSD7(M), PEMS7(L), PEMS4, and PEMS8), approximately 60% of the sequences were used for model training, 20% for validation, and the remaining 20% for testing. The models were evaluated on 1 h prediction horizons (12 time steps) using three standard metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE).

To quantitatively evaluate the forecasting performance, four commonly used metrics are adopted: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2). Their formulations are given as follows:

$$\text{MAE} = \frac{1}{N\tau S} \sum_{s=1}^S \sum_{h=1}^{\tau} \sum_{i=1}^N \left| \hat{Y}_{i,t_s+h}^{(s)} - Y_{i,t_s+h}^{(s)} \right| \quad (25)$$

$$\text{RMSE} = \sqrt{\frac{1}{N\tau S} \sum_{s=1}^S \sum_{h=1}^{\tau} \sum_{i=1}^N \left(\hat{Y}_{i,t_s+h}^{(s)} - Y_{i,t_s+h}^{(s)} \right)^2} \quad (26)$$

$$\text{MAPE} = \frac{100\%}{N\tau S} \sum_{s=1}^S \sum_{h=1}^{\tau} \sum_{i=1}^N \left| \frac{\hat{Y}_{i,t_s+h}^{(s)} - Y_{i,t_s+h}^{(s)}}{Y_{i,t_s+h}^{(s)}} \right| \quad (27)$$

$$R^2 = 1 - \frac{\sum_{s=1}^S \sum_{h=1}^{\tau} \sum_{i=1}^N \left(\hat{Y}_{i,t_s+h}^{(s)} - Y_{i,t_s+h}^{(s)} \right)^2}{\sum_{s=1}^S \sum_{h=1}^{\tau} \sum_{i=1}^N \left(Y_{i,t_s+h}^{(s)} - \bar{Y} \right)^2} \quad (28)$$

Here, N denotes the number of sensors, τ the forecasting horizon, and S the number of test samples. $Y_{i,t_s+h}^{(s)}$ and $\hat{Y}_{i,t_s+h}^{(s)}$ represent the ground-truth and predicted traffic speeds of sensor i at horizon h for the s -th test instance, respectively. All metrics are first computed for each sensor and each forecasting horizon, and then averaged across sensors, horizons, and test samples, following the standard per-node per-horizon evaluation protocol in traffic forecasting.

Following common practice, Z-score normalization is applied during model training. For evaluation, all predictions and ground-truth values are transformed back to their original scale using the training-set statistics. Therefore, the MAE and RMSE values reported in this paper are expressed in physical units (miles per hour), enabling direct interpretation of real-world forecasting accuracy.

To comprehensively evaluate the performance of the proposed model, we compared it against ten representative baseline approaches, ranging from traditional statistical models to recent graph-based deep learning methods. A brief description of each baseline is provided below.

HA [45]: The Historical Average method predicts future traffic states using the mean of historical observations at the same time as previous days. It serves as a simple yet effective statistical benchmark.

ARIMA [46]: The Auto-Regressive Integrated Moving Average model captures linear temporal correlations by combining autoregressive and moving-average components. It is one of the most classical and widely used time-series forecasting techniques.

VAR [47]: The Vector Auto-Regression model extends ARIMA to multivariate time series, jointly modeling dependencies among multiple traffic sensors through linear interactions.

DCRNN [8]: The Diffusion Convolutional Recurrent Neural Network models spatial dependencies using bidirectional diffusion processes on directed graphs and temporal dynamics through gated recurrent units (GRUs).

STGCN [48]: The Spatio-Temporal Graph Convolutional Network applies graph convolutional layers for spatial feature extraction and temporal gated convolution layers for sequence modeling, achieving efficient end-to-end learning.

STSGCN [49]: The Spatial-Temporal Synchronous Graph Convolutional Network introduces a localized synchronous modeling mechanism to jointly learn spatial and short-term temporal correlations within a unified graph block.

ASTGCN [50]: The Attention-based Spatial-Temporal Graph Convolutional Network employs spatial and temporal attention mechanisms to adaptively weight node influence and capture multi-period dependencies, such as recent, daily, and weekly patterns.

DGCRN [41]: The Dynamic Graph Convolutional Recurrent Network constructs time-varying adjacency matrices to capture evolving spatial dependencies, integrating dynamic graph generation with recurrent temporal modeling.

Graph WaveNet [51]: A convolution-based spatio-temporal graph neural network that integrates stacked dilated causal temporal convolutions with a self-adaptive adjacency matrix learned from node embeddings, enabling the model to capture long-range temporal dependencies and uncover hidden spatial correlations beyond the given road network structure.

D2STGNN [52]: A decoupled dynamic spatio-temporal graph neural network that explicitly separates traffic signals into diffusion and inherent components via a residual decomposition mechanism and an estimation gate. It further incorporates a dynamic graph learning module based on self-attention to capture time-varying spatial dependencies for improved traffic forecasting.

DDGCRN [42]: The Double Dynamic Graph Convolutional Recurrent Network extends DGCRN by jointly updating spatial and temporal graphs, enabling adaptive modeling of complex spatiotemporal variations.

STGODE [53]: The Spatial-Temporal Graph ODE Network captures continuous-time spatiotemporal dependencies through tensor-based ordinary differential equations (ODEs), allowing the construction of deeper and more flexible architectures.

RDSTGNN [54]: The Robust Dynamic Spatio-Temporal Graph Neural Network introduces a dual-module architecture that filters abnormal traffic patterns and models periodic behaviors, enabling more stable forecasting under noisy or highly irregular conditions.

4.1. Results and Discussion

Tables 2 and 3 summarize the predictive performance of the proposed DGIB model and all baseline methods across the four benchmark datasets, where MAE and RMSE are reported as per-node per-horizon averages in physical units (mph). Several important observations can be drawn from the comparative results.

First, DGIB consistently achieves the best overall performance across all datasets and evaluation metrics. Compared with traditional statistical methods such as HA, ARIMA, and VAR, DGIB yields substantially lower per-node prediction errors, demonstrating the advantage of deep spatio-temporal modeling in capturing nonlinear traffic dynamics. When compared with early deep learning baselines, including DCRNN, STGCN, and STSGCN, DGIB further reduces both MAE and RMSE by a clear margin, highlighting the benefit of adaptive graph modeling and effective temporal dependency learning.

Table 2. Performance comparison on PEMS D7(M) and D7(L). MAE and RMSE are reported as per-node per-horizon averages in miles per hour (mph) after inverse Z-score normalization. Best results are in **bold**.

Model	PEMSD7(M)				PEMSD7(L)			
	MAE	RMSE	MAPE (%)	R^2	MAE	RMSE	MAPE (%)	R^2
HA	5.80	8.20	24.60	0.590	6.10	8.60	25.10	0.565
ARIMA	4.50	6.90	19.60	0.655	4.90	7.20	20.10	0.635
VAR	3.90	5.70	18.90	0.770	4.20	6.10	19.80	0.752
DCRNN	3.05	4.90	12.10	0.875	3.30	5.20	12.80	0.862
STGCN	3.00	4.95	11.70	0.880	3.25	5.25	12.40	0.866
STSGCN	2.95	4.85	11.40	0.885	3.20	5.15	12.10	0.870
ASTGCN	2.85	4.70	10.80	0.892	3.05	4.95	11.50	0.878
DGCRN	2.75	4.55	10.30	0.900	2.95	4.80	11.00	0.887
Graph WaveNet	2.90	4.75	11.14	0.888	3.10	5.05	11.84	0.874
D2STGNN	2.60	4.35	9.79	0.910	2.80	4.65	10.45	0.897
DDGCRN	2.70	4.45	9.98	0.904	2.90	4.75	10.70	0.892
STGODE	2.62	4.38	9.72	0.907	2.82	4.68	10.35	0.897
RDSTGNN	2.68	4.42	9.85	0.905	2.88	4.72	10.55	0.890
DGIB (Ours)	2.45	4.10	8.75	0.918	2.65	4.35	9.02	0.909

Table 3. Performance comparison on PEMS D4 and D8. MAE and RMSE are reported as per-node per-horizon averages in miles per hour (mph) after inverse Z-score normalization. Best results are in **bold**.

Model	PEMSD4				PEMSD8			
	MAE	RMSE	MAPE (%)	R^2	MAE	RMSE	MAPE (%)	R^2
HA	4.94	7.67	28.00	0.672	4.52	6.76	24.10	0.689
ARIMA	4.77	7.15	25.00	0.703	4.45	6.34	20.50	0.715
VAR	3.19	4.85	17.20	0.857	2.74	4.12	14.30	0.878
DCRNN	2.77	4.34	14.20	0.911	2.20	3.45	10.95	0.924
STGCN	2.74	4.51	13.80	0.913	2.22	3.50	11.30	0.921
STSGCN	2.73	4.37	13.70	0.914	2.21	3.47	11.35	0.923
ASTGCN	2.70	4.25	13.40	0.918	2.16	3.36	10.80	0.929
DGCRN	2.55	4.10	12.95	0.924	2.07	3.22	10.22	0.933
Graph WaveNet	2.67	4.34	13.46	0.917	2.16	3.39	10.87	0.926
D2STGNN	2.52	4.06	12.78	0.926	2.05	3.20	10.06	0.935
DDGCRN	2.52	4.05	12.70	0.927	2.04	3.19	10.03	0.936
STGODE	2.46	3.99	12.45	0.931	2.01	3.15	9.81	0.939
RDSTGNN	2.49	3.97	12.55	0.929	1.99	3.13	9.72	0.938
DGIB (Ours)	2.39	3.91	12.10	0.935	1.93	3.08	9.40	0.943

Second, DGIB remains competitive when compared with more recent graph-based forecasting approaches such as ASTGCN, DGCRN, DDGCRN, STGODE, and RDSTGNN. Across all four datasets, DGIB consistently achieves the lowest MAE and RMSE values together with the highest R^2 scores. On large-scale networks such as PEMS D7(L), DGIB attains a per-node MAE of approximately 2.65 mph, outperforming strong dynamic graph baselines by a consistent margin. Similar trends are observed on PEMS D7(M), PEMS D4, and PEMS D8, indicating that the proposed method generalizes well across different network sizes and traffic conditions.

Third, the improved performance of DGIB can be attributed to its information-bottleneck-based design, which encourages the model to retain compact yet task-relevant spatio-temporal representations. By explicitly constraining the learned graph embeddings to suppress redundant or noisy correlations while preserving predictive dependencies, DGIB enhances generalization and robustness without relying on additional adversarial training or heavy regularization. Moreover, the dynamic graph update mechanism enables the model to adapt to time-varying topological relationships, providing an advantage over static or weakly adaptive graph structures.

Finally, beyond predictive accuracy, DGIB also offers improved interpretability. The learned bottleneck representations emphasize the most influential spatial connections and temporal patterns for traffic forecasting, which can provide useful insights for traffic management and decision-making. Overall, these results demonstrate that integrating information-theoretic compression with dynamic graph modeling yields a robust and effective framework for spatio-temporal traffic prediction.

4.2. Ablation Study

To understand the impact of each major component in DGIB, we conducted extensive ablation experiments on the PEMS D7(M) and PEMS D4 datasets, representing medium- and small-scale traffic networks, respectively. Four model variants were designed:

- DGIB w/o VIB: removes the Variational Information Bottleneck term from the objective function, disabling information compression during training.
- DGIB w/o Dynamic Graph: fixes the adjacency matrix to a static road topology derived from sensor distance, without temporal updates.
- DGIB w/o Node Embeddings: constructs dynamic graphs only from observed traffic features, excluding learnable node embeddings that encode structural priors.
- GCN+GRU Baseline: employs a conventional GCN for spatial feature extraction and a GRU for temporal modeling, serving as a simplified backbone without dynamic graphs or information bottleneck.

Table 4 reports the ablation results of different DGIB variants under the same per-node per-horizon evaluation protocol. As shown in the table, all ablated models exhibit consistently degraded performance compared with the full DGIB, confirming the complementary contributions of the dynamic graph and information bottleneck components. Specifically, removing the variational information bottleneck (VIB) results in a clear increase in MAE on both PEMS D7(M) and PEMS D4, indicating that the bottleneck constraint plays an important role in filtering redundant spatio-temporal dependencies and improving generalization. Disabling the dynamic graph update also leads to noticeable performance degradation, highlighting the necessity of modeling time-varying spatial correlations in traffic networks. In addition, removing node embeddings causes moderate but consistent performance drops, especially on larger networks, suggesting their effectiveness in capturing latent structural similarities among sensors. The simple GCN+GRU baseline performs the worst among all variants, further demonstrating the benefit of integrating information-theoretic compression with dynamic graph modeling.

Overall, these results demonstrate that both the dynamic graph module and the information bottleneck contribute significantly to DGIB's strong predictive capability. The dynamic graph ensures spatial adaptability to changing traffic conditions, while the VIB constraint enforces compact yet informative latent representations, yielding consistent gains across network scales.

Table 4. Ablation results of DGIB components on PEMS D7(M) and PEMS D4. MAE and RMSE are reported as per-node per-horizon averages in miles per hour (mph) after inverse normalization. Best results are in **bold**.

PEMS D7(M)				
Model Variant	MAE	RMSE	MAPE (%)	R^2
GCN+GRU Baseline	3.04	4.95	12.58	0.880
DGIB w/o Node Emb.	2.61	4.31	9.89	0.903
DGIB w/o Dynamic Graph	2.59	4.28	9.81	0.906
DGIB w/o VIB	2.63	4.34	10.01	0.902
Full DGIB	2.45	4.06	8.75	0.918
PEMS D4				
Model Variant	MAE	RMSE	MAPE (%)	R^2
GCN+GRU Baseline	2.73	4.27	14.11	0.910
DGIB w/o Node Emb.	2.41	3.89	12.78	0.924
DGIB w/o Dynamic Graph	2.38	3.86	12.56	0.928
DGIB w/o VIB	2.44	3.94	12.84	0.923
Full DGIB	2.29	3.76	12.10	0.935

4.3. Hyperparameter Analysis

To further examine the stability and generalization of DGIB, we analyzed the effects of several key hyperparameters on the PEMS D7(M) dataset, namely the information-bottleneck coefficient β , the number of neighbors k in dynamic graph construction, and the hidden feature dimension d_h of the encoder. Each parameter was varied individually while keeping the remaining settings fixed.

4.3.1. Effect of the Information-Bottleneck Coefficient β

The parameter β in Equation (24) controls the trade-off between information compression and predictive sufficiency. As illustrated in Figure 3, the forecasting performance initially improves as β increases from a small value, indicating that a moderate level of information bottleneck helps suppress redundant or noisy dependencies. When β becomes overly large, the per-node MAE starts to increase, suggesting that excessive compression leads to the loss of task-relevant information. This U-shaped trend demonstrates that an appropriately balanced bottleneck strength enables DGIB to retain informative spatio-temporal representations while improving generalization performance.

4.3.2. Effect of the Number of Neighbors k

The hyperparameter k controls the number of nearest neighbors retained for each sensor in the adaptive graph. As shown in Figure 4, the forecasting performance improves as k increases from small values, indicating that incorporating a moderate amount of local spatial context helps capture meaningful inter-sensor correlations. When k becomes too large, the per-node MAE gradually increases, suggesting that excessive neighborhood sizes introduce noisy or weakly relevant connections and reduce modeling efficiency. Overall, a moderate neighborhood size (e.g., $k = 10$) provides a favorable trade-off between locality, robustness, and computational cost, and is adopted in all experiments.

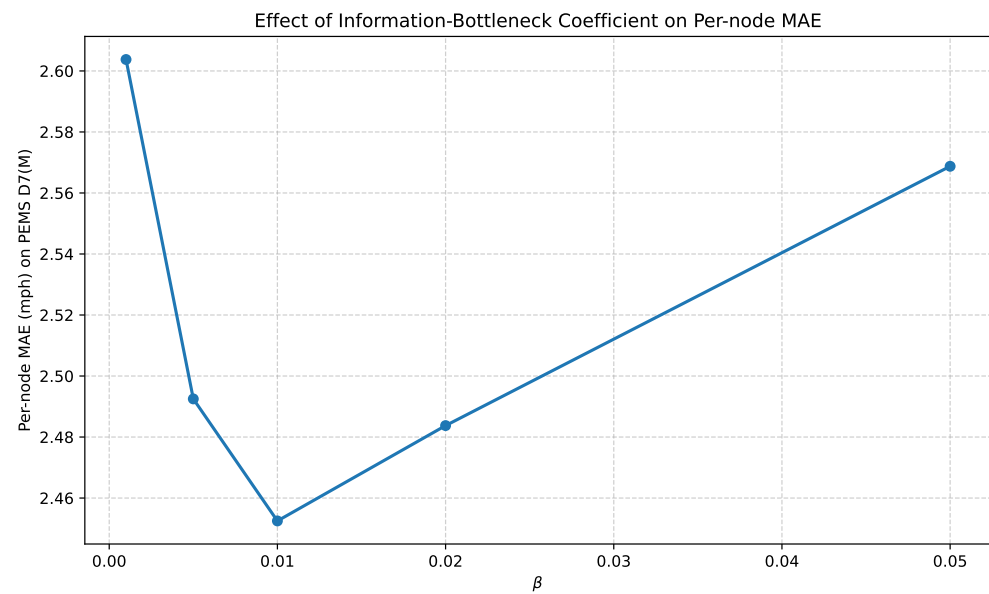


Figure 3. Effect of the information-bottleneck coefficient β on MAE for the PEMS D7(M) dataset.

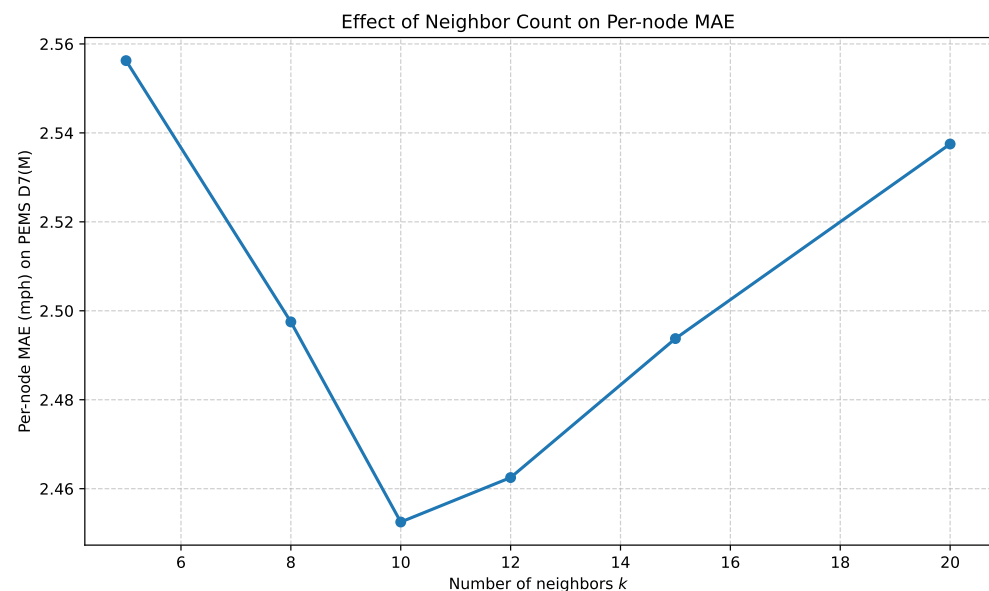


Figure 4. Effect of the number of neighbors k in dynamic graph construction on MAE.

4.3.3. Effect of the Hidden Dimension d_h

As illustrated in Figure 5, increasing the hidden feature dimension d_h initially leads to improved forecasting accuracy, indicating that a moderate increase in representational capacity helps the model capture informative spatio-temporal patterns. When d_h becomes excessively large, the per-node MAE no longer decreases and may slightly increase, suggesting diminishing returns and potential overfitting. This trend confirms that the information bottleneck mechanism effectively regularizes the model, allowing DGIB to achieve strong performance without relying on overly large latent dimensions.

Overall, DGIB demonstrates consistent and stable performance across a wide range of hyperparameter values, validating the robustness and ease of tuning of the proposed framework for real-world traffic forecasting applications.

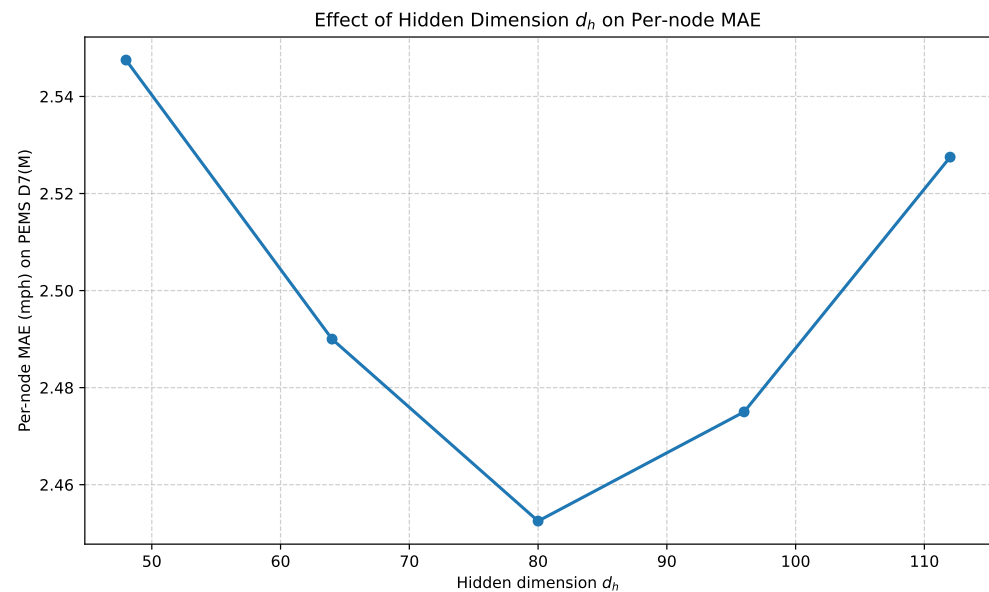


Figure 5. Effect of the hidden dimension d_h on MAE.

5. Conclusions

Accurate and reliable traffic prediction remains a fundamental challenge due to the highly dynamic, nonlinear, and non-Euclidean nature of urban road networks. In this paper, we proposed a Dynamic Graph Neural Network (DGIB) that integrates dynamic graph learning with an information-bottleneck-based structure compression mechanism. Unlike conventional models that rely on fixed or single-type adjacency matrices, the proposed framework adaptively captures multi-scale spatiotemporal dependencies while suppressing redundant and noisy correlations. Through extensive experiments on benchmark datasets, DGIB consistently outperformed representative baselines, demonstrating its superior capacity to model evolving traffic dynamics and maintain stable performance under diverse traffic conditions.

The presented framework provides new insights into how information-theoretic regularization can enhance the generalization of graph-based forecasting models. By enforcing a compact representation that retains only task-relevant dependencies, DGIB bridges the gap between dynamic graph modeling and principled information compression. Beyond traffic forecasting, this approach has potential applications in other spatiotemporal domains such as environmental monitoring, energy demand prediction, and human mobility analysis.

Future work will focus on several directions. First, integrating probabilistic or uncertainty-aware modules may further improve the model's robustness and interpretability. Second, following the standardized single-city benchmarking protocol adopted in this study, extending the proposed framework to cross-city, large-region, or federated learning settings could further enhance its scalability, generalization ability, and privacy preservation. Finally, incorporating causal or counterfactual reasoning into the dynamic graph learning process may enable more trustworthy and policy-relevant traffic forecasting in intelligent transportation systems.

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

Funding: This research was supported by the National R&D Program of China (Grant No. 2023YFB2504704); the National Natural Science Foundation of China (Grant Nos. 52208424, 52208416, 52078091, and 52108399); the Natural Science Foundation of Chongqing Municipality (Grant No. CSTB2022NSCQ-MSX0503); the China Postdoctoral Science Foundation (Grant No. 2022M710545); and the Chongqing Municipal Education Commission Foundation (Grant Nos. KJQN202200745 and KJQN202300728). The authors gratefully acknowledge their financial support.

Data Availability Statement: The data presented in this study are openly available in Performance Measurement System, at https://dot.ca.gov/programs/traffic-operations/mpr/pems-source?utm_source=chatgpt.com, accessed on 1 August 2025.

Conflicts of Interest: Author Zhongbin Luo was employed by the company China Merchants Chongqing Communications Research & Design Institute Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IB	Information Bottleneck
GIB	Graph Information Bottleneck
GNN	Graph Neural Network

References

- Li, F.; Feng, J.; Yan, H.; Jin, G.; Yang, F.; Sun, F.; Jin, D.; Li, Y. Dynamic Graph Convolutional Recurrent Network for Traffic Prediction: Benchmark and Solution. *ACM Trans. Knowl. Discov. Data* **2023**, *17*, 1–21. [\[CrossRef\]](#)
- Liu, S.; Wang, X. An Improved Transformer Based Traffic Flow Prediction Model. *Sci. Rep.* **2025**, *15*, 8284. [\[CrossRef\]](#)
- Rahmani, S.; Baghbani, A.; Bouguila, N.; Patterson, Z. Graph Neural Networks for Intelligent Transportation Systems: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 8846–8885. [\[CrossRef\]](#)
- Jianbei, L.; Weiwei, L.; Yuanyuan, J.; Zhongbin, L. Research on Expressway Operation Risk Based on Multi Factor Safety Entropy Model. In *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*; IEEE: Piscataway, NJ, USA, 2018; pp. 430–435.
- Shao, Y.; Luo, Z.; Wu, H.; Han, X.; Pan, B.; Liu, S.; Claudel, C.G. Evaluation of two improved schemes at non-aligned intersections affected by a work zone with an entropy method. *Sustainability* **2020**, *12*, 5494. [\[CrossRef\]](#)
- Zhu, D.; Zhu, Z.; Gong, X.; Ye, D.; Li, C.; Chen, J. A Nonlinear Spatiotemporal Optimization Method of Hypergraph Convolution Networks for Traffic Prediction. *Intell. Autom. Soft Comput.* **2023**, *37*, 3. [\[CrossRef\]](#)
- Liu, K.; Chen, Z.; Yamamoto, T.; Tuo, L. Exploring the Impact of Spatiotemporal Granularity on the Demand Prediction of Dynamic Ride-Hailing. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 104–114. [\[CrossRef\]](#)
- 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.
- Zhang, D.; Wang, P.; Ding, L.; Wang, X.; He, J. Spatio-Temporal Contrastive Learning-Based Adaptive Graph Augmentation for Traffic Flow Prediction. *IEEE Trans. Intell. Transp. Syst.* **2024**, *26*, 1304–1318. [\[CrossRef\]](#)
- Li, H.; Yang, S.; Song, Y.; Luo, Y.; Li, J.; Zhou, T. Spatial Dynamic Graph Convolutional Network for Traffic Flow Forecasting. *Appl. Intell.* **2023**, *53*, 14986–14998. [\[CrossRef\]](#)
- Zhang, H.; Wang, H.; Zhang, X.; Gong, L. Research on Traffic Flow Forecasting Based on Dynamic Spatial-Temporal Transformer. *Transp. Res. Rec.* **2024**, *2678*, 301–313. [\[CrossRef\]](#)
- Liu, Y.; Shen, G.; Liu, N.; Han, X.; Xu, Z.; Zhou, J.; Kong, X. Traffic Data Imputation via Knowledge Graph-Enhanced Generative Adversarial Network. *PeerJ Comput. Sci.* **2024**, *10*, e2408. [\[CrossRef\]](#)
- Tishby, N.; Pereira, F.C.; Bialek, W. The Information Bottleneck Method. In *Proceedings of the 37th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, USA, 22–24 September 1999; pp. 368–377.
- Alemi, A.A.; Fischer, I.; Dillon, J.V.; Murphy, K. Deep Variational Information Bottleneck. In *Proceedings of the International Conference on Learning Representations (ICLR)*, Toulon, France, 24–26 April 2017; pp. 1–12.
- Wu, T.; Ren, H.; Li, P.; Leskovec, J. Graph information bottleneck. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 20437–20448.
- Yu, J.; Xu, T.; Rong, Y.; Bian, Y.; Huang, J.; He, R. Graph Information Bottleneck for Subgraph Recognition. In *Proceedings of the International Conference on Learning Representations, Virtual Event, Austria*, 3–7 May 2021.

17. Zheng, K.; Yu, S.; Li, B.; Jenssen, R.; Chen, B. Brainib: Interpretable brain network-based psychiatric diagnosis with graph information bottleneck. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *36*, 13066–13079. [[CrossRef](#)]
18. Ahmed, M.S.; Cook, A.R. Short-term traffic forecasting: A statistical approach. *Transp. Res. Rec.* **1980**, *772*, 1–9.
19. Williams, B.M.; Hoel, L.A. Forecasting time series with ARIMA model: An application to traffic volume forecasting. *Transp. Res. Rec.* **2003**, *1836*, 1–9.
20. Wu, C.H.; Ho, J.M.; Lee, D.T. Short-term traffic flow forecasting with support vector regression. *IEEE Trans. Intell. Transp. Syst.* **2004**, *5*, 276–281. [[CrossRef](#)]
21. Wang, Y.; Papageorgiou, M. Real-time freeway traffic state estimation: An extended Kalman filter approach. *Transp. Res. Part B Methodol.* **2005**, *39*, 141–167. [[CrossRef](#)]
22. Luo, Z.; Shi, H.; Liu, W.; Jin, Y. HMM-Based Traffic Situation Assessment and Prediction Method. In *CICTP 2020*; American Society of Civil Engineers: Reston, VA, USA, 2020; pp. 4205–4217.
23. Ma, X.; Tao, Z.; Wang, Y.; Yu, H.; Wang, Y. Deep convolutional neural networks for short-term traffic flow prediction. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, Anchorage, AK, USA, 14–19 May 2017; pp. 1–6.
24. Lv, Y.; Duan, Y.; Kang, W.; Li, Z.; Wang, F.Y. Deep recurrent neural networks for traffic flow prediction. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3470–3479.
25. Zhang, J.; Zheng, Y.; Qi, D. Traffic flow forecasting with spatial-temporal graphs and long short-term memory networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, San Francisco, CA, USA, 4–9 February 2017; pp. 3889–3895.
26. Yao, H.; Tang, X.; Wei, H.; Zheng, G.; Li, Z. A deep learning approach for traffic flow prediction based on spatio-temporal features. *Transp. Res. Part C Emerg. Technol.* **2020**, *119*, 102695.
27. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
28. Ma, X.; Ma, Y.; Li, Z.; Wang, Y.; Wang, F.Y. Traffic flow prediction with long short-term memory networks (LSTM). In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, Seoul, Republic of Korea, 28 June–1 July 2015; pp. 153–158.
29. Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Gated recurrent unit networks for traffic flow prediction. *arXiv* **2014**, arXiv:1406.1078.
30. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, Toulon, France, 24–26 April 2017.
31. Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph attention networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, Vancouver, BC, Canada, 30 April–3 May 2018.
32. Hamilton, W.; Ying, R.; Leskovec, J. Inductive representation learning on large graphs. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, Long Beach, CA, USA, 4–9 December 2017; pp. 1024–1034.
33. Zhou, J.; Cui, G.; Zhang, S.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; Sun, M. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4–24.
34. Gilmer, J.; Schoenholz, S.S.; Riley, P.F.; Vinyals, O.; Dahl, G.E. Neural message passing for quantum chemistry. In *Proceedings of the International Conference on Machine Learning (ICML)*, Sydney, Australia, 6–11 August 2017; pp. 1263–1272.
35. Yu, B.; Yin, H.; Zhu, Z. Message passing neural networks for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, New Orleans, LA, USA, 2–7 February 2018; pp. 5691–5698.
36. Han, H.; Zhang, M.; Hou, M.; Zhang, F.; Wang, Z.; Chen, E.; Wang, H.; Ma, J.; Liu, Q. STGCN: A spatial-temporal aware graph learning method for POI recommendation. In *Proceedings of the 2020 IEEE International Conference on Data Mining (ICDM)*, Sorrento, Italy, 17–20 November 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1052–1057.
37. Zhu, J.; Wang, Q.; Tao, C.; Deng, H.; Zhao, L.; Li, H. AST-GCN: Attribute-augmented spatiotemporal graph convolutional network for traffic forecasting. *IEEE Access* **2021**, *9*, 35973–35983. [[CrossRef](#)]
38. Zheng, C.; Fan, X.; Wang, C.; Qi, J. Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 1234–1241.
39. Bai, L.; Yao, H.; Li, Z. Adaptive graph convolutional recurrent network for traffic forecasting. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, Virtual, 6–12 December 2020.
40. Jiang, J.; Guo, S.; Li, C.; Cui, P.; Song, C. Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 204–216.
41. Han, M.; Xu, H.; Xue, Y.; Zhang, J. Dynamic graph convolutional recurrent network for traffic forecasting. *Inf. Sci.* **2021**, *579*, 304–321.
42. Yang, T.; He, W.; Yang, K.; Guo, S. DDGCRN: Double dynamic graph convolutional recurrent network for traffic flow forecasting. *Knowl.-Based Syst.* **2024**, *283*, 111039.
43. Chen, Z.; Shen, Y.; Shi, X. D2STGNN: Disentangled dynamic spatial-temporal graph neural network for traffic flow prediction. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 6940–6952.
44. Yu, S.; Giraldo, L.G.S.; Príncipe, J.C. Information-Theoretic Methods in Deep Neural Networks: Recent Advances and Emerging Opportunities. In *Proceedings of the IJCAI*, Montreal, QC, Canada, 19–27 August 2021; pp. 4669–4678.

45. Hamilton, J.D. *Time Series Analysis*; Princeton University Press: Princeton, NJ, USA, 2020.
46. Box, G.E.P.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*, 5th ed.; John Wiley & Sons: Hoboken, NJ, USA, 2015.
47. Brandt, P.T.; Williams, J.T. *The Analysis of Time Series: An Introduction*; Cambridge University Press: London, UK, 2007.
48. 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.
49. Song, C.; Lin, Y.; Guo, S.; Wan, H.; Ouyang, K.; Xu, Z. Spatial-Temporal Synchronous Graph Convolutional Networks: A New Framework for Spatial-Temporal Network Data Modeling. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 914–921.
50. 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.
51. 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, Macao, China, 10–16 August 2019; pp. 1907–1913.
52. Shao, Z.; Zhang, Z.; Wei, W.; Wang, F.; Xu, Y.; Cao, X.; Jensen, C.S. Decoupled dynamic spatial-temporal graph neural network for traffic forecasting. *Proc. Vldb Endow.* **2022**, *15*, 2733–2746. [[CrossRef](#)]
53. Fang, Z.; Zhang, C.; Li, Z.; Wang, W. STGODE: Spatial-Temporal Graph Ordinary Differential Equation Networks for Continuous-Time Traffic Forecasting. *IEEE Trans. Intell. Transp. Syst.* **2023**, *Early Access*.
54. Huang, Y.; Han, W.; Xie, Y. Robust dynamic spatio-temporal graph neural network for traffic forecasting. *Appl. Intell.* **2025**, *55*, 917. [[CrossRef](#)]

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.