

Dynamic Knowledge Driven Multi-Scale Spatiotemporal Fusion Framework for Reliable Multi-Horizon Traffic Flow Forecasting

Deepika M S, P. Deepa Shenoy and Venugopal K R

Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bangalore, India

Article history

Received: 02-05-2026

Revised: 16-08-2026

Accepted: 20-08-2026

Corresponding Author:

Deepika M S

Department of Computer Science and Engineering,
University Visvesvaraya College of Engineering,
Bangalore University,
Bangalore, India

Email:

deepikamsphd2020@gmail.com

Abstract: Accurate traffic flow forecasting remains a core challenge in intelligent transportation systems. Urban traffic patterns are characterised by complex spatiotemporal dependencies and inherent uncertainties, which complicate traffic forecasting. This study proposes a Dynamic Knowledge-Driven Spatiotemporal Fusion (DKDMSF) architecture to extract features of Multi-Scale traffic dynamics across various temporal traffic horizons. DKDMSF integrates Dynamic Spatial Attention (DSA) and Adaptive Graph Learning (AGL), a context-aware parallel encoder, and cross-modal fusion of variational inference for uncertainty quantification. DSA constructs a context-aware spatial inference graph by adaptively modifying node weights in response to changing traffic patterns. The AGL uncovers the optimal traffic network topology by learning node connectivity using dynamic traffic flow data. Context-aware parallel encoders extract temporal patterns via periodic temporal encoding and are fused with probabilistic uncertainty estimation for traffic forecasting. The proposed framework was evaluated on the METR-LA, PEMS-BAY, NYC-Taxi, SZ-Taxi, and TomTom datasets to demonstrate the model's robustness. The results demonstrate better performance and narrower confidence intervals than existing models across different prediction horizons, making it suitable for various traffic applications.

Keywords: Graph Neural Networks, Multi-Horizon Prediction, Intelligent Transportation Systems, Spatiotemporal Modelling, Traffic Forecasting, Uncertainty Quantification

Introduction

Urban traffic congestion is a critical global challenge owing to urbanisation in metropolitan areas. Congestion leads to various socio-economic and environmental problems, including impacts on economic productivity, quality of life, CO₂ emissions, and sustainability. Research from the Centre for Economics and Business Research (CERB, 2026) shows that urban commuters experience an average delay of 87 hours annually due to congestion. In rapidly developing countries like India, this number escalates. Bangalore, India's Silicon Valley, is the world's most congested city. According to a traffic analysis by TomTom, Bangalore experiences an average delay of 117 hours annually, with a peak-hour travel time of 49 minutes and 40 seconds for a 10-kilometre journey. These issues can be resolved by deploying an efficient Urban Traffic Management System (UTMS). Accurate traffic flow forecasting is an important part of UTMS; it enables dynamic route planning, emissions reduction, adaptive traffic signal management, ramp metering, emergency response, and demand-response transit

(Medina-Salgado et al., 2022). Authors in Kashyap et al. (2022); Gomes et al. (2023) demonstrate that traffic flow forecasting involves various temporal ranges from 2 minutes to monthly predictions; the classification of these prediction horizons is based on the changing traffic dynamics, and contributes to a specific set of applications in UTMS.

Several factors, interconnected across different temporal scales in the data, increase the complexity of multi-horizon traffic prediction, such as spatial dependencies, temporal dependencies, the non-linear nature of traffic patterns, congestion, variation in travel demands, external factors (events, weather), multi-modal data quality, and availability of reliable sources. The spatiotemporal patterns in traffic flow depend on the traffic zone, time, heterogeneous vehicles, and propagation of traffic to or from neighbouring traffic zones, sensor data noise, and missing values in the traffic data, making traffic flow prediction more challenging. The complexity of predicting uncertainty in multi-horizon traffic flow increases with the range of temporal dependencies and dynamic spatial interactions. Hence, there is a need for a

unified traffic prediction model capable of modelling local and global spatial dependencies, capturing short-term, mid-term, and long-term temporal correlations in a changing network topology, and predicting uncertainty. Naheliya et al. (2025) highlight the progression of traffic forecasting frameworks from statistical approaches, machine learning, and deep learning to Graph Neural Network (GNN) based transformer models that address the challenges in multi-scale traffic prediction. Existing models still face challenges in extracting features from heterogeneous traffic patterns; spatial dependencies vary across local, regional and city-wide traffic, finding traffic patterns from congestion onset and recovery, and interpreting the confidence level of the model predictions as discussed in this papers (Zong et al., 2026; Fan et al., 2025; Qiu et al., 2025; Carianni and Gemma, 2025). Graph-based transformer models construct a spatial graph from global traffic nodes, with forecast points held fixed. This limits the reliability of decision-making and robustness to changing traffic environments. Architectural complexities in multi-scale fusion models require substantial computational resources, making real-time deployment difficult.

To address these challenges, this paper proposes a parallel Dynamic Knowledge-Driven Multi-scale Spatiotemporal Fusion (DKDMSF) framework that combines temporal representations across all scales, adaptively learns spatial interactions, and quantifies the uncertainty of model predictions, thereby enabling robust multi-horizon traffic predictions across various urban environments. Existing approaches use single-feature extraction in sequential order; the proposed framework extracts local, global, and dynamic spatiotemporal features in parallel. The three key methodological innovations of the proposed DKDMSF framework are:

- (i) Context-Aware Dynamic Spatial Attention (DSA)
- (ii) Data-Driven Adaptive Graph Learning (AGL)
- (iii) Parallel feature-level fusion with uncertainty estimation

The contributions of this work are as follows:

- A unified parallel DKDMSF framework is proposed which combines adaptive spatiotemporal correlations and uncertainty prediction for reliable multi-horizon traffic forecasting
- Context-Aware Dynamic Spatial Attention (DSA) use dual-branch spatial attention with adaptive-gated temporal attention to fit the varying traffic conditions for adaptive representation learning in mixed traffic environments
- Identifying hidden spatial connectivity patterns through multi-hop node embedding
- Adaptive Graph Learning (AGL) module refines the predefined network graph through knowledge-

driven network topology according to dynamic traffic nodes

- Cross-modal parallel encoder fusion to combine temporal feature extraction with probabilistic uncertainty estimation

Related Work

Deep Learning Approaches for Traffic Prediction

The emergence of deep learning has fundamentally transformed traffic prediction methodologies by providing architectures specifically designed for spatiotemporal modelling, as highlighted by Kong et al. (2025); Wu (2024); Sayed et al. (2025); Ma et al. (2024); Harrou et al. (2024). Convolution Neural Networks (CNNs) demonstrate exceptional capability in capturing spatial correlations between adjacent road segments. In contrast, Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks effectively model temporal dependencies across varying time scales. Hybrid architectures of CNNs and LSTMs have demonstrated increased performance by simultaneously capturing spatial and temporal correlations. The introduction of attention mechanisms and transformer architectures further enhanced the ability to capture long-range dependencies in sequential data. Cai et al. (2020) adapted the transformer architecture for traffic forecasting, demonstrating the effectiveness of self-attention mechanisms in capturing temporal dependencies. Jiang et al. (2023) introduced PDFormer, a pyramid dilated attention mechanism for capturing multi-scale temporal patterns, demonstrating competitive performance across multiple datasets. Xu et al. (2021) employed Spatiotemporal transformers (STTrans), a combination of spatial and temporal attention mechanisms within a unified framework. Graph Neural Networks (GNNs) have provided a natural framework for modelling traffic networks. Graph Convolution Networks (GCNs) explicitly model road network topology, enabling effective spatial dependency modelling. Subsequent work by Chen et al. (2025); Wang et al. (2023); Ali et al. (2024) has explored various graph convolution operations and attention mechanisms.

Multi-Scale and Fusion Methodologies

Recent research has increasingly focused on multi-scale approaches to capture the parallel nature of traffic patterns. Cao et al. (2024) proposed a spatiotemporal multi scale graph convolution network architecture that adaptively integrates temporal and spatial features across different resolutions, enabling robust performance under fluctuating traffic conditions. The paper by Du et al. (2024) developed a feature-enhanced framework that exploits multiple temporal scales and spatial attention to improve the forecasting accuracy in dense urban networks. The authors of Dong et al. (2024) introduced MTESFormer. This transformer-based model combines temporal self-attention across multiple scales with spatial encoding to capture both periodic trends

and abrupt changes in traffic patterns. A multi-scale spatiotemporal graph neural network that fuses graph attention mechanisms with temporal encoding, allowing the model to adapt to granular variations in traffic flow while preserving the global structure, was discussed in Chen et al. (2025).

However, existing fusion approaches have significant limitations:

- (1) Methods such as ensemble-based approaches combine predictions from separate models rather than integrating architectures at the feature level
- (2) While MTESFormer (Chen et al., 2025) performs feature-level integration through multi-scale attention, it uses fixed temporal attention patterns and lacks adaptive spatial reasoning
- (3) Recent automated approaches by Jin et al. (2023) focus on dilated convolutions but do not address dynamic spatial connectivity learning

Uncertainty Quantification in Traffic Forecasting

Recent studies have emphasised the importance of Uncertainty Quantification (UQ) in traffic flow forecasting, highlighting its role in enhancing model reliability, interpretability, and generalisation. The paper by Qian et al. (2024) provides a comprehensive framework for categorizing and analyzing uncertainty in traffic prediction by integrating epistemic and aleatoric sources within a unified understanding of the model behaviour. Sengupta et al. (2024) proposed a Bayesian inference approach to quantify predictive uncertainty, improve forecast reliability,

and enhance model generalizability across diverse urban contexts. Mallick et al. (2024) employed a deep ensemble strategy within a spatiotemporal graph neural network framework to effectively capture the uncertainty arising from both model parameters and input variability. Wang et al. (2023) introduced a probabilistic graph neural network tailored for travel demand forecasting, demonstrating that explicitly modelling uncertainty at both the spatial and temporal levels significantly improves prediction robustness in dynamic transportation environments. Recent comprehensive surveys by Zong et al. (2026); Fan et al. (2025); Jin et al. (2023); Kong et al. (2025) have highlighted the need for more complex approaches that can handle the multi-scale characteristics of traffic dynamics while providing reliable uncertainty estimates. Specifically, existing multi-scale methods, such as MTESFormer, perform feature-level integration but lack adaptive spatial reasoning, whereas methods such as STTrans and STGAT use fixed attention patterns that cannot adapt to changing traffic conditions. Furthermore, recent automated spatiotemporal modelling approaches have shown promise but still rely on predefined graph structures and lack comprehensive uncertainty quantification.

Despite recent advances in traffic flow prediction, several key limitations persist in existing approaches, such as limited generalisation across regions, reliance on static graph topology, use of globally shared attention mechanisms, high computational overhead, sensitivity to data quality, and lack of integrated uncertainty estimation at various levels. The fundamental differences between the existing models and the proposed approach are shown in Table 1.

Table 1: Comparison of the Proposed DKDMSF Framework with Existing State-of-the-Art Traffic Forecasting Models

Model	Spatial Modeling	Dynamic Graph Learning	Knowledge-Driven Graph	Dynamic Spatial Attention	Multi-Scale Temporal Learning	Parallel Feature Fusion	U Q	Multi-Horizon Forecasting	Main Limitation
LSTM	X	X	X	X	✓	X	X	✓	Captures only temporal dependencies
CNN-LSTM	Local CNN	X	X	X	✓	Sequential	X	✓	Limited long-range spatial dependency modelling
STGCN	Graph Convolution	X (Static Graph)	X	X	✓	X	X	✓	Relies on predefined graph topology
DCRNN	Diffusion Graph	Partial	X	X	✓	X	X	✓	Graph opology remains largely fixed during training
Graph WaveNet	Adaptive Graph	Partial	X	X	✓	X	X	✓	Learns statistical correlations between graphs without prior traffic knowledge
ASTGCN	Graph Attention	X	X	Partial	✓	X	X	✓	Uses fixed attention after training.

STGAT	Graph Attention	X	X	✓	✓	X	✓	X	No adaptive graph refinement or uncertainty modelling. Relies on predefined spatial representation.
PDFormer	Transformer + Spatial Encoding	X	X	✓	✓	X	✓	X	Adaptive embedding but no knowledge-guided graph refinement. Focuses on temporal multi-scale learning without adaptive graph learning or uncertainty estimation.
STAEFormer	Adaptive Spatial-Temporal Embedding	Partial	X	✓	✓	X	✓	X	Unified adaptive framework integrating knowledge-driven graph refinement, context-aware attention, heterogeneous feature fusion, and probabilistic uncertainty estimation.
MTESFormer	Multi-scale Transformer	X	X	Partial	✓	Partial	✓	X	
Proposed DKDMSF	Dynamic Spatial Attention + Adaptive Graph Learning	✓	✓	✓	✓	✓ (CNN + LSTM + Transformer + DSA + AGL)	✓	✓	

This study addresses these gaps by proposing a DKDMSF architecture that integrates context-aware dynamic spatial attention, data-driven adaptive graph learning, and cross-modal parallel fusion with probabilistic uncertainty quantification within a unified framework, thereby overcoming the limitations of existing feature- and prediction-level combination strategies.

Methods

In this section, the problem formulation and the modules of the proposed DKDMSF framework are explained in detail.

Problem Formulation

Let $G = (V, E, A_0)$, be represented as a directed traffic network topology, where V denotes the set of N sensors in the traffic network, and $A_0 \in R^{N \times N}$ represents the adjacency matrix for constructing spatial relationships between the nodes. The historical traffic patterns observed over a window of length T are represented as $X \in \mathbb{R}^{T \times N \times F}$, where F denotes the feature dimension. Given historical traffic patterns $X^{(t-T+1:t)}$, the objective is to learn a mapping function that predicts the future traffic states $Y^{(t+1:t+H)} \in \mathbb{R}^{H \times N \times 1}$ for the next H time steps. To capture hidden and dynamic spatial dependencies, the model further learns the adaptive adjacency matrix A_{adapt} and performs traffic forecasting as $Y = F(X, A; \theta)$, where A combines predefined and dynamically learned

spatial structures.

Dynamic Knowledge Driven Spatiotemporal Fusion Framework Multi-Scale

The proposed DKDMSF framework depicts an end-to-end pipeline for multi-horizon traffic forecasting with uncertainty quantification. This framework integrates knowledge-driven adaptive traffic network topology learning with cross-modal, context-driven, gated parallel encoders and uncertainty-modelling fusion to capture varying spatiotemporal characteristics for reliable multi-horizon traffic forecasting. The structured pipeline of the framework, shown in Fig. 1, is as follows: Data acquisition, feature engineering, context-aware Dynamic Spatial Attention (DSA), knowledge-driven Adaptive Graph Learning (AGL), hypothesis generation via a context-gated parallel encoder, fusion of uncertainty estimates, and multi-horizon traffic forecasting.

Data Acquisition and Pre Processing

The summary of the dataset collected for the study is given in Table 2. Traffic data are collected using sensors and GPS sources, and this heterogeneous raw traffic pattern is then provided as input to the DKDMSF framework after preprocessing steps in Fig. 2. During preprocessing, the historical traffic patterns in each dataset are converted to an input tensor representing both cyclic temporal patterns and localised spatial dependencies using distance-aware weighting.

$$A_{ij} = \begin{cases} \frac{1}{d_{ij}+1} & \text{if } d_{ij} \leq \tau \\ 0 & \text{otherwise} \end{cases}; \text{ where } d_{ij} \text{ represents the}$$

distance between the nodes in the traffic network i and j , and τ represents the threshold of node connectivity. The hidden spatial features are computed for each time step using the equations $Neighbor\ Agg_{t,i} = \sum_{j=1}^N A_{ij} \cdot X_{t,j}$; $SpatialVar_{t,i} = Var(\{A_{ij} : j \in N(i)\})$, where $N(i)$ represents the neighbouring node in the traffic network i . The local spatial interactions are

captured in this representation before the graph refinement.

Cyclic temporal encoding is employed to extract the temporal features using the sine and cosine functions, as shown below:

$$f_{hour}(h) = \left[\sin\left(\frac{2\pi h}{24}\right), \cos\left(\frac{2\pi h}{24}\right) \right]; f_{day}(d) = \left[\sin\left(\frac{2\pi d}{7}\right), \cos\left(\frac{2\pi d}{7}\right) \right]; f_{month}(m) = \left[\sin\left(\frac{2\pi m}{12}\right), \cos\left(\frac{2\pi m}{12}\right) \right]$$

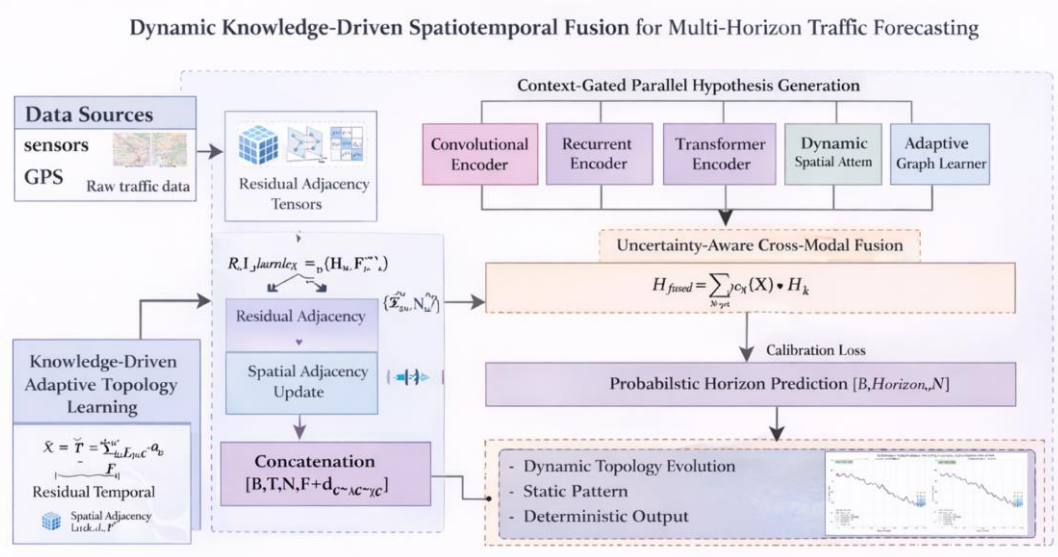


Fig. 1: Dynamic Knowledge Driven Spatiotemporal Fusion Architecture

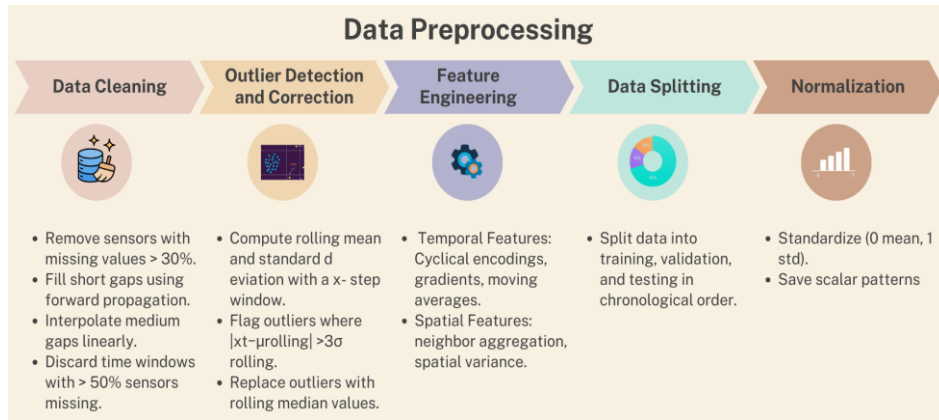


Fig. 2: Data preprocessing pipeline for spatiotemporal feature extraction

Table 2: Summary of Datasets Used for Traffic Flow Prediction

Dataset	METR-LA	TomTom	PEMS-BAY	NYC-Taxi	SZ-Taxi
Source	LA Metro	TomTom	Caltrans	NYC TLC	Shenzhen Gov
Format	HDF5	CSV	HDF5	CSV	CSV
Coverage	207 sensors, 4 months	20 segments, 36.5 days	325 sensors, 6 months	266 zones, 6 months	156 zones, 3 months
Resolution	5-minute	1-minute	5-minute	30-minute	30-minute
Data Size	34,272×207	52,560×20	52,116×325	8,760×266	4,380×156

This encoding ensures that captured traffic patterns span temporally adjacent periods, such as 23:59 and 00:01.

Feature Engineering

The input tensors are converted to meaningful representations for the parallel encoder using a knowledge-driven adaptive network topology and residual adjacency tensor construction. The predefined adjacency matrix is the foundation for knowledge-driven adaptive network topology construction; to capture changing traffic dynamics, residuals from temporal patterns are added to prior knowledge of traffic network connectivity. This allows the model to learn the hidden, time-varying interactions between nodes in the network that are not explicitly connected, while retaining the basic graph structure. This is further refined using the residual adjacency tensor by encoding multi-hop and global spatial dependencies. Through this, the model can handle long-range dependencies and be robust to incomplete or noisy network connectivity information. The spatial and temporal features are concatenated to provide a unified input for parallel encoders. This concatenation bridges the gap between the knowledge-driven network topology and feature-level modelling, facilitating joint spatiotemporal learning.

Hypothesis Generation Through Context Gated Parallel Encoders

This module is the core of the proposed DKSMF framework and enables the modelling of the multi-perspective nature of changing traffic patterns. The enriched input tensors from the previous module were projected into a shared representation space and processed simultaneously through multiple parallel encoder blocks. Parallel encoders include CNN, LSTM, Transformer, DSA and AGL modules. Each encoder is designed to capture distinct aspects of traffic behaviour at different spatial and temporal scales, which a single encoder cannot capture.

CNN Encoder: To enable spatial convolution operations, the input tensor X was reconstructed using the equation:

$$X_{CNN} = \text{Reshape}(X) \in \mathbb{R}^{(B \times T) \times F \times N \times 1}$$

Where B denotes the batch size. 2D convolutions are applied to multiple layers to capture the local spatial dependencies and short-term temporal patterns:

$$H_{CNN}^{(1)} = \text{ReLU}\left(\text{Conv2D}\left(X_{CNN}; W_{CNN}^{(1)}\right)\right)$$

$$H_{CNN}^{(1)} = \text{ReLU}\left(\text{Conv2D}\left(H_{CNN}^{(1)}; W_{CNN}^{(1)}\right)\right)$$

$$H_{CNN}^{(3)} = \text{ReLU}\left(\text{Conv2D}\left(H_{CNN}^{(2)}; W_{CNN}^{(1)}\right)\right)$$

To obtain fixed-size representations, global adaptive

pooling is applied using the equation:

$$H_{CNN} = \text{AdaptivePool}(H_{CNN}^{(3)}) \in \mathbb{R}^{B \times T \times d}$$

Where d denotes the hidden dimension, this is essential for modelling immediate neighbour-node interactions in traffic.

Network

LSTM encoder: Sequential temporal dependencies for each sensor are extracted independently by the LSTM encoder. For sensor i , the temporal sequence is processed as follows:

$$h_t^{(i)}, c_t^{(i)} = \text{LSTM}(x_t^{(i)}, h_{t-1}^{(i)}, c_{t-1}^{(i)}; \theta_{LSTM})$$

Where $h_t^{(i)}$ and $c_t^{(i)}$ are the hidden and cell states, respectively. The output of the LSTM is aggregated from all the sensors: $H_{LSTM} = \frac{1}{N} \sum_{i=1}^N [h_1^{(i)}, h_2^{(i)}, \dots, h_T^{(i)}]$ It is important to capture the evolution of traffic flow and periodic traffic patterns.

Transformer Encoder: Long-range temporal dependencies are captured by the transformer model through a self-attention mechanism. For each sensor i , the input sequence is expressed as:

$$Q^{(i)} = X^{(i)} W_Q; K^{(i)} = X^{(i)} W_K; V^{(i)} = V^{(i)} W_V$$

The multi-head self-attention is as follows:

$$\text{Attention}(Q^{(i)}, K^{(i)}, V^{(i)}) = \text{softmax}\left(\frac{Q^{(i)} K^{(i)T}}{\sqrt{d_k}}\right) V^{(i)}$$

The multi-head self-attention integrates h attention heads:

$$\begin{aligned} \text{Multi-head}(Q^{(i)}, K^{(i)}, V^{(i)}) \\ = \text{Concat}(\text{head}_1, \dots, \text{head}_n) W_O \end{aligned}$$

Multiple layers of residual connections and layer normalisation are applied in transformer encoders:

$$Z_l^{(i)} = \text{LayerNorm}\left(X^{(i)} + \text{MultiHead}(X^{(i)})\right)$$

$$H_{Trans}^{(i)} = \text{LayerNorm}\left(Z_l^{(i)} + \text{FFN}\left(Z_l^{(i)}\right)\right)$$

The outputs are aggregated from all sensors using the equation $\text{rans} = 1 \ Ni = 1NH \ Transi$, which is essential for identifying distant temporal relationships.

Dynamic Spatial Attention (DSA) Module

The DSA module is the first key innovation in the proposed framework based on context-aware spatiotemporal reasoning, as shown in Fig. 3. At each timestep, DSA dynamically learns which traffic nodes are important for varying temporal scales and constructs network topology adaptively in real-time through spatial attention.

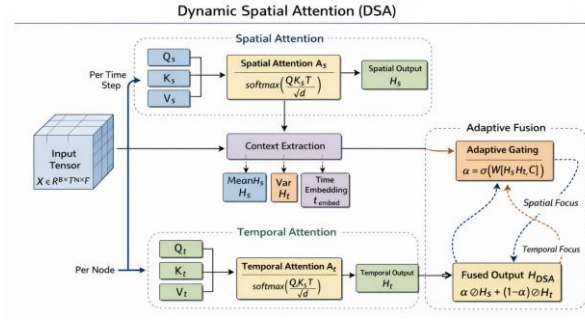


Fig. 3: Dynamic Spatial Attention Architecture

For input $X \in \mathbb{R}^{B \times T \times N \times F}$, context-aware spatial reasoning is performed through each timestep t , spatial attention is computed across nodes as:

$$Q_s^{(t)} = X_{:,t,:,:} W_{Q_s}; K_s^{(t)} = X_{:,t,:,:} W_{K_s}; V_s^{(t)} = X_{:,t,:,:} W_{V_s};$$

$$A_s^{(t)} = \text{softmax} \left(\frac{Q_s^{(t)} K_s^{(t)T}}{\sqrt{d_k}} \right); H_t^{(i)} = A_s^{(t)} V_s^{(t)}$$

For each node n , temporal attention is computed for each timestep as:

$$Q_t^{(n)} = X_{:,n,:} W_{Q_t}; K_t^{(n)} = X_{:,n,:} W_{K_t}; V_t^{(n)} = X_{:,n,:} W_{V_t};$$

$$A_t^{(n)} = \text{softmax} \left(\frac{Q_t^{(n)} K_t^{(n)T}}{\sqrt{d_k}} \right); H_t^{(n)} = A_t^{(n)} V_t^{(n)}$$

Based on traffic conditions, DSA dynamically balances the spatial and temporal contributions through adaptive temporal gating:

$$C_{\text{context}} = \text{MLP}([\text{mean}(H_s); \text{var}(H_t); \text{tembed}])$$

$$\alpha = \sigma(W_{\text{gate}}[H_s; H_t; C_{\text{context}}] + b_{\text{gate}})$$

$$H_{\text{DSA}} = \alpha \odot H_s + (1 - \alpha) \odot H_t$$

Where C_{context} captures the traffic environment, t_{embed} represents the temporal embeddings, and α provides context-aware gating. Through this innovation, the model adapts attention patterns based on changing traffic conditions, such as prioritising spatial attention during incidents and temporal attention during regular patterns.

Adaptive Graph Learning (AGL) Module

The second key innovation of the proposed framework is AGL. The architecture is illustrated in Fig. 4. Based on observed traffic flow patterns in the network, the AGL automatically determines the optimal spatial network topology by dynamically learning which nodes are connected and strongly connected in the traffic network. This data-driven topology-construction capability allows the model to adapt to diverse, complex traffic environments by discovering hidden spatial relationships within traffic networks.

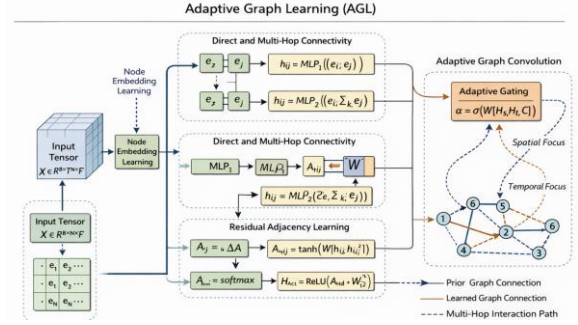


Fig. 4: Adaptive Graph Learning Architecture

All feature representations are projected into a common latent space; hence, heterogeneous features can be comparable.

Given the base adjacency matrix $A_{\text{base}} \in \mathbb{R}^{N \times N}$, which includes the prior information of the traffic road network and represents the physical connection of road topology. AGL performs data-driven topology discovery via node embedding learning without altering the road topology. Each node i has a learnable embedding $e_i \in \mathbb{R}^{d_e} : E = [e_1, e_2, \dots, e_N] \in \mathbb{R}^{N \times d_e}$.

AGL introduces multi-hop connectivity discovery by:

- $h_{ij}^1 = \text{MLP}_1([e_i; e_j])$ (direct connection)
- $h_{ij}^2 = \text{MLP}_2([e_i; \sum_k e_k; e_j])$ (2-hop connection);
- $\Delta A_{ij} = \tanh(W_{\text{combine}}[h_{ij}^1; h_{ij}^2])$ (multi-hop fusion).

The adaptive adjacency topology is constructed by combining base and learned components:

$$A_{\text{learned}} = A_{\text{base}} + \lambda \Delta A; A_{\text{final}} = \text{softmax}(A_{\text{learned}})$$

Where λ is the scaling factor for residual corrections. The learned adjacency topology is used for feature propagation as follows: $H^{(1)} = \text{ReLU}(A_{\text{final}} X W^{(1)})$; $H_{\text{AGL}} = \text{ReLU}(A_{\text{final}} H^{(1)} W^{(2)})$.

The term knowledge-driven refers to the AGL module, because the adaptive graph maintains the underlying road network architecture while enabling learning of dynamic relationships within the traffic network.

Uncertainty Aware Cross Modal Fusion

The DKDMSF framework combines features from all parallel encoders via cross-model attention-based weighting, learning the complementary relationships among different encoder outputs for the same input. Algorithm 1 presents the complete training procedure for a parallel fusion model. Each feature representation from the parallel encoder is projected from a hidden space of dimension d using learnable weight matrices to match the feature dimensionality required for aggregation. Let $\mathcal{H} = \{H_{\text{CNN}}, H_{\text{LSTM}}, H_{\text{Trans}}, H_{\text{DSA}}, H_{\text{AGL}}\}$ be the encoder output set, and let $\mathcal{W} = \{W_{\text{CNN}}, W_{\text{LSTM}}, W_{\text{Trans}}, W_{\text{DSA}}, W_{\text{AGL}}\}$ be

the projection weights. The projection of each encoder output is defined as: $H_k^1 = H_k W_k, \forall k \in \{1,2,3,4,5\}$, where k indexes the five feature sources in H .

The projected parallel encoder features are stacked to prepare features for fusion:

$$H_{stack} = \text{Stack}([H_1', H_2', H_3', H_4', H_5']) \in \mathbb{R}^{B \times T \times 5 \times d}$$

$$Q_{fusion} = H_{stack} W_Q, K_{fusion} = H_{stack} W_K, \\ V_{fusion} = H_{stack} W_V,$$

$$A_{cross} = \text{softmax}\left(\frac{Q_{fusion} K_{fusion}^T}{\sqrt{d}}\right); \alpha = \\ \text{softmax}(A_{cross} V_{fusion} W_{att}) \in \mathbb{R}^{B \times T \times 5}$$

The encoded features are stacked and fused via the cross-modal attention block, which learns the associations between the encoder outputs and assigns corresponding attention weights based on their impact on the forecasting task. The attention weights are used to fuse the projected representations of the encoders along with their original counterparts in the residual fusion operation as described by the equation $H_{fused} = \sum_{k=1}^5 \alpha_{:,k} \odot H_{stack}[:,k] + R_{residual}$.

Where $R_{residual}$ preserves the original feature information, enabling the model to learn the most relevant features and discard noisy or redundant ones. Where $\odot$ denotes the element-wise multiplication. The robustness of the predictions was evaluated using variational inference to provide confidence intervals for the uncertainty estimates. Algorithm 2 describes the inference procedure for uncertainty quantification. Variational dropout was applied with probability p during training. $H_{dropout} = \text{Variational Dropout}(H_{fused}, p)$.

The M forward passes are performed with variational dropout for uncertainty quantification during inference:

$$\{\hat{Y}_1, \hat{Y}_2, \dots, \hat{Y}_M\} = \{\mathcal{F}(X; \theta, \text{dropout} = \text{True})\}_{m=1}^M$$

The predictive mean and uncertainty are estimated as:

$$\mu \leftarrow \left(\frac{1}{M}\right) \sum_{m=1}^M \hat{Y}_m; \sigma^2 \leftarrow \left(\frac{1}{M}\right) \sum_{m=1}^M (\hat{Y}_m - \mu)^2$$

Algorithm 1: Parallel Multi-Scale Fusion Training Algorithm

Input: Training data $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^{|\mathcal{D}|}$, adjacency matrix A
Hyperparameters: learning rate η , batch size B , epochs E
Output: Trained model parameters θ^*
1. Initialise model parameters θ
2. Initialise optimiser (Adam) with learning rate η
3. Compute enhanced adjacency matrix
4. Generate periodic temporal features
5. for epoch $e = 1$ to E do
6. Shuffle training data $\mathcal{D}$

7. for each batch $B \subset \mathcal{D}$ of size B do
8. Extract batch data $\{X_b, Y_b\}_{b \in B}$
9. // Forward pass through Parallel encoders
10. $H_{CNN} \leftarrow \text{CNN} - \text{Encoder}(X_b)$
11. $H_{LSTM} \leftarrow \text{LSTM} - \text{Encoder}(X_b)$
12. $H_{Trans} \leftarrow \text{Transformer} - \text{Encoder}(X_b)$
13. $H_{DSA} \leftarrow \text{DSA} - \text{Module}(X_b)$
14. $H_{AGL}, A_{learned} \leftarrow \text{AGL} - \text{Module}(X_b)$
15. // Multi-scale fusion
16. $H_{fused} \leftarrow \text{Fusion} -$
Framework($H_{CNN}, H_{LSTM}, H_{Trans}, H_{DSA}, H_{AGL}$)
17. // Prediction with uncertainty
18. $\hat{Y}_b, \sigma_b^2 \leftarrow \text{Prediction} - \text{Head}(H_{fused})$
19. // Compute loss
20. $L \leftarrow \text{Composite} - \text{Loss}(Y_b, \hat{Y}_b, \sigma_b^2, A_{learned})$
21. // Backward pass and optimisation
22. Compute gradients $\nabla \theta \mathcal{L}$
23. Update parameters $\theta \leftarrow \theta - \eta \nabla \theta \mathcal{L}$
24. end for
25. // Validation and early stopping
26. Evaluate model on validation set
27. if validation loss increases for patience epochs then
28. Break training loop
29. end if
30. end for
31. return Optimised parameters θ^*

Algorithm 2: Inference with Probabilistic Uncertainty Quantification

Require: Test input X_{test} , trained model $\mathcal{F}(\cdot; \theta^*)$
Require: Number of variational samples M .
Ensure: Predictions μ and uncertainties σ^2
1. Initialise prediction storage $\{\hat{Y}_m\}_{m=1}^M$
2. for sample $m = 1$ to M do
3. Perform variational dropout during inference
4. $\hat{Y}_m \leftarrow \mathcal{F}(X_{test}; \theta^*, \text{dropout} = \text{True})$
5. end for
6. // Compute predictive statistics
7. $\mu \leftarrow \left(\frac{1}{M}\right) \sum_{m=1}^M \hat{Y}_m$
8. $\sigma^2 \leftarrow \left(\frac{1}{M}\right) \sum_{m=1}^M (\hat{Y}_m - \mu)^2$
9. return μ, σ^2

Composite Loss Function

The composite loss function combines prediction accuracy, uncertainty estimation, and graph regularisation to train the model.

$$L = L_{pred} + \lambda L_{reg} + \beta L_{unc} + \gamma L_{graph}$$

$$\text{Where: } L_{pred} = \left(\frac{1}{BHN}\right) \sum_{b=1}^B \sum_{h=1}^H \sum_{n=1}^N (Y_{b,h,n} - \hat{Y}_{b,h,n})^2 \\ L_{reg} = \|\theta\|_2^2$$

$$L_{unc} = \frac{1}{BHN} \sum_{b=1}^B \sum_{h=1}^H \sum_{n=1}^N \sigma_{b,h,n}^2$$

$$L_{\{graph\}} = ||A_{\{learned\}}||_1$$

This loss function determines the predicted uncertainty and actual prediction errors. The graph regularisation term L_{graph} was used to stabilise the training, improve generalisation, and prevent overconfident predictions.

Experiments

This section outlines the datasets and experimental procedures used to evaluate the proposed DKDSF framework.

Dataset

The DKDSF framework was evaluated using the following five datasets: METR-LA and PEMS-BAY (He, 2025), TomTom (traffic analytics, 2025), NYC-Taxi (NYC, 2023), and SZ-Taxi (lehaifeng, 2023). The characteristics of the datasets are presented in Table 2.

Table 3 presents the primary characteristics of the TomTom dataset. The values in the table, with an average travel time of 58.42 seconds and a standard deviation of 42.32 seconds, show notable changes in traffic due to heterogeneous conditions in Bangalore, India.

Experimental Configuration

The experimental configurations used in this study are presented in Table 4. The baseline models are selected for comparison with the proposed model across

the various categories of available traffic forecasting methods.

The LSTM and CNN-LSTM models fall under the category of recurrent and hybrid sequence learning methods, respectively. The STGCN, DCRNN, and Graph WaveNet models have been chosen from the category of graph-based convolution.

The ASTGCN and STGAT models have been chosen as representatives of graph attention-based methods. The latest state-of-the-art spatiotemporal traffic forecasting methods based on Transformers, such as STAEFormer, STID, STDMAE, PDFormer, and MTESFormer, have also been selected. All datasets used in this study were split into 70:15:15 ratios for training, validation, and testing, respectively. The model was trained using the AdamW optimiser with learning-rate scheduling. Early stopping was applied with a patience of 25 epochs to prevent overfitting.

Loss weights λ , β , γ are Regularization weight, Uncertainty weight and Graph sparsity weight, respectively. The regularisation weight $\lambda = 0.01$ is chosen based on a grid search over $\{0.001, 0.005, 0.01, 0.05, 0.1\}$, which yields the best validation accuracy across the evaluation metrics. Uncertainty weight β controls epistemic uncertainty penalty $L_{\{unc\}}$, the value is 0.1 to avoid overconfident intervals.

Graph sparsity weight $\gamma = 0.001$ to consider the sparsity in the adjacency matrix because the real-time traffic network structure is sparse; only a minimum number of sensors meaningfully interact with neighbouring nodes.

Table 3: Statistical Summary of the TomTom Traffic Dataset

Feature	Mean	Std. Dev.	Min	Max
travelTimeSec	58.42	42.31	19.00	259.00
delaySec	28.76	36.89	0.00	232.00
stops	0.52	0.37	0.00	1.50
queueLengthMeters	112.45	156.78	0.00	1440.46
length	283.97	101.25	219.91	436.52

Table 4: Experimental Setup for Parallel Spatio-temporal Network

Category	Details
Framework	PyTorch, VS Code, Numpy
Hardware	NVIDIA RTX 3090, Intel Core i7 10 th generation.
Dataset	METR-LA, TomTom, PEMS-BAY, NYC-Taxi, SZ-Taxi
Optimizer	AdamW with learning rate scheduler (reduces LR by 0.5 on validation loss plateau)
Hyperparameters	Hidden Dimension: 128; Attention Heads: 8; LSTM Layers: 2; Transformer Layers: 2; Dropout: 0.15; Batch Size: 8; Learning Rate: 0.001; Weight Decay: 1×10^{-5} ; Patience: 25 epochs; Loss Weights: $\lambda = 0.01$, $\beta = 0.1$, $\gamma = 0.001$; Variational Samples: $M = 50$; random seed = 42
Evaluation Metrics	Prediction Accuracy: MAE, RMSE, MAPE, R ² ; Uncertainty Estimates: PICP, MPIW, Calibration Error, Reliability Score
Baseline Models	LSTM (Li et al., 2018), GCN (Li et al., 2018), ASTGCN (Zhu et al., 2021), STGAT (Huang et al., 2020), PDFormer (Jin et al., 2023), MTESFormer (Dong et al., 2024), Graph WaveNet (Wu, 2024), DCRNN (Li et al., 2019;), STGCN (Yu et al., 2018), MegaCRN (Jiang et al., 2023), STAEformer (Liu et al., 2023), STID (Shao et al., 2022), STDMAE (Gao et al., 2024)
Prediction Horizons	METR-LA: 15, 30, 45, 60 min; TomTom: 1, 2, 5, 10, 15, 30, 45, 60 min; PEMS-BAY: 15, 30, 45, 60 min; NYC-Taxi: 15, 30, 45, 60 min; SZ-Taxi: 15, 30, 45, 60 min

The variational sample M is the number of Monte Carlo forward passes that control the latency and quality of the estimated uncertainty during the traffic forecasting; $M = 50$ provides statistically stable and reliable uncertainty estimation.

Evaluation Metrics

Four standard metrics presented below were employed for a comprehensive evaluation, and Prediction Interval Coverage Probability (PICP) and Mean Prediction Interval Width (MPIW) were employed for uncertainty quantification:

$$MAE = \frac{1}{N} \sum_{i=1}^N |Y_i - \hat{Y}_i| ; MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|Y_i - \hat{Y}_i|}{Y_i} \times 100 ;$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (Y_i - \hat{Y}_i)^2} ; R^2 = 1 - \frac{\sum_{i=1}^N (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^N (Y_i - \bar{Y})^2}$$

Table 5: Performance Comparison on the METR-LA Dataset

Model	15-min Prediction				30-min Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	3.99	9.6%	8.21	0.623	5.15	12.7%	10.45	0.578
SVR	3.28	8.3%	7.12	0.651	4.42	10.9%	8.89	0.612
LSTM	3.44	8.7%	6.89	0.742	3.98	9.8%	7.92	0.698
GCN-LSTM	2.77	6.9%	5.69	0.768	3.15	7.9%	6.45	0.721
DCRNN	2.69	6.7%	5.51	0.775	3.09	7.7%	6.28	0.728
Graph WaveNet	2.65	6.5%	5.43	0.779	3.05	7.5%	6.21	0.732
STGCN	2.71	6.8%	5.58	0.773	3.12	7.8%	6.35	0.725
ASTGCN	2.75	7.51%	5.62	0.781	3.31	9.3%	6.98	0.745
STGAT	2.66	6.09%	5.12	0.789	3.01	8.14%	6.12	0.752
MegaCRN	2.63	6.4%	5.38	0.783	3.02	7.6%	6.18	0.735
STAEformer	2.59	6.2%	5.29	0.791	2.97	7.3%	6.05	0.758
STID	2.57	6.1%	5.24	0.794	2.95	7.2%	6.01	0.761
STDMAE	2.55	6.0%	5.19	0.797	2.92	7.0%	5.97	0.764
PDFormer	2.81	7.67%	5.45	0.795	3.20	9.19%	6.46	0.761
MTESFormer	2.73	7.10%	5.32	0.801	2.89	7.8%	5.94	0.768
Proposed DKDMSF	2.52	5.9%	5.16	0.809	2.91	6.9%	5.98	0.772
Model	45-min Prediction				60-min Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	6.45	16.1%	13.21	0.512	7.23	18.4%	15.67	0.445
SVR	5.42	13.8%	11.23	0.534	6.15	15.7%	12.89	0.478
LSTM	4.67	11.2%	9.34	0.634	5.23	12.8%	10.89	0.587
GCN-LSTM	3.53	8.8%	7.37	0.667	3.99	10.0%	8.21	0.621
DCRNN	3.47	8.6%	7.29	0.671	3.93	9.8%	8.14	0.625
Graph WaveNet	3.43	8.4%	7.21	0.675	3.89	9.6%	8.07	0.629
STGCN	3.49	8.7%	7.33	0.669	3.95	9.9%	8.18	0.623
ASTGCN	3.98	9.6%	8.21	0.689	4.45	10.7%	9.34	0.651
STGAT	3.29	8.3%	6.98	0.698	3.46	9.79%	7.19	0.663
MegaCRN	3.35	8.2%	7.05	0.693	3.78	9.4%	7.89	0.641
STAEformer	3.21	7.9%	6.87	0.705	3.58	9.1%	7.45	0.671
STID	3.18	7.8%	6.82	0.708	3.54	9.0%	7.39	0.674
STDMAE	3.14	7.6%	6.76	0.712	3.49	8.8%	7.31	0.678
PDFormer	3.46	9.9%	7.67	0.712	3.62	10.91%	7.47	0.681
MTESFormer	3.18	9.1%	6.78	0.706	3.37	9.62%	7.14	0.675
Proposed DKDMSF	2.89	7.4%	6.05	0.721	2.95	8.6%	6.58	0.691

Results and Analysis

Overall Performance Comparison

Tables 5, 6, 7, and 8 present comprehensive results on the METR-LA, PEMS-BAY, NYC-Taxi, and SZ-Taxi datasets, respectively, across multiple prediction horizons.

Tables 9 and 10 show the results of the proposed DKDMSF model on the TomTom dataset for both the short and medium prediction horizons.

The proposed DKDMSF framework demonstrated consistent improvements over baseline approaches. Evaluation of the five datasets validated the generalizability of the proposed DKDMSF framework across diverse urban environments. The consistent performance improvements across all five datasets (METR-LA, TomTom, PEMS-BAY, NYC-Taxi, and SZ-Taxi) demonstrate the robustness and applicability of the proposed approach to UTMS.

Table 6: Performance Comparison on PEMS-BAY Dataset

Method	15-min Prediction				30-min Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	2.89	7.2%	5.67	0.634	3.45	8.9%	6.78	0.589
SVR	2.67	6.8%	5.34	0.658	3.21	8.4%	6.45	0.612
LSTM	2.45	6.3%	4.89	0.689	2.98	7.8%	5.89	0.645
GCN-LSTM	2.23	5.9%	4.56	0.712	2.78	7.3%	5.67	0.671
DCRNN	2.18	5.7%	4.48	0.718	2.71	7.1%	5.58	0.678
Graph WaveNet	2.15	5.6%	4.41	0.723	2.67	6.9%	5.51	0.683
STGCN	2.21	5.8%	4.52	0.715	2.74	7.2%	5.62	0.675
ASTGCN	2.19	5.7%	4.45	0.721	2.69	7.0%	5.54	0.681
STGAT	2.12	5.5%	4.32	0.729	2.63	6.8%	5.43	0.689
MegaCRN	2.16	5.6%	4.38	0.725	2.71	7.0%	5.56	0.679
STAEformer	2.08	5.3%	4.25	0.735	2.58	6.6%	5.32	0.695
STID	2.05	5.2%	4.21	0.739	2.54	6.5%	5.28	0.699
STDMAE	2.02	5.1%	4.17	0.743	2.51	6.4%	5.24	0.703
PDFormer	2.09	5.4%	4.28	0.732	2.61	6.7%	5.38	0.692
MTESFormer	2.01	5.0%	4.15	0.745	2.49	6.3%	5.21	0.706
Proposed DKDMSF	1.94	4.8%	4.02	0.758	2.38	6.0%	5.05	0.719
Method	45-min Prediction				60-min Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	4.12	10.8%	7.89	0.534	4.67	12.3%	8.78	0.489
SVR	3.89	10.2%	7.56	0.567	4.34	11.6%	8.23	0.523
LSTM	3.45	9.1%	6.78	0.612	3.89	10.4%	7.45	0.578
GCN-LSTM	3.21	8.5%	6.34	0.645	3.67	9.8%	7.12	0.612
DCRNN	3.15	8.3%	6.27	0.651	3.58	9.6%	7.05	0.618
Graph WaveNet	3.09	8.1%	6.19	0.657	3.51	9.4%	6.97	0.624
STGCN	3.18	8.4%	6.31	0.648	3.62	9.7%	7.08	0.615
ASTGCN	3.12	8.2%	6.23	0.654	3.54	9.5%	7.01	0.621
STGAT	3.05	8.0%	6.15	0.661	3.47	9.3%	6.89	0.628
MegaCRN	3.11	8.1%	6.21	0.656	3.53	9.4%	6.99	0.622
STAEformer	2.98	7.8%	6.05	0.668	3.38	9.0%	6.78	0.635
STID	2.94	7.7%	6.01	0.672	3.34	8.9%	6.74	0.639
STDMAE	2.91	7.6%	5.97	0.675	3.31	8.8%	6.71	0.642
PDFormer	3.02	7.9%	6.09	0.665	3.41	9.1%	6.82	0.632
MTESFormer	2.89	7.5%	5.94	0.678	3.28	8.7%	6.67	0.645
Proposed DKDSF	2.76	7.2%	5.78	0.691	3.12	8.3%	6.45	0.658

Table 7: Performance Comparison on NYC-Taxi Dataset

Method	15-min Prediction				30-min Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	3.89	11.2%	7.23	0.534	4.23	12.8%	7.89	0.512
SVR	3.56	10.4%	6.78	0.567	3.89	11.6%	7.34	0.545
LSTM	3.12	9.1%	6.23	0.612	3.45	10.2%	6.78	0.589
GCN-LSTM	2.89	8.5%	5.78	0.645	3.12	9.4%	6.23	0.623
DCRNN	2.83	8.3%	5.69	0.651	3.05	9.1%	6.15	0.631
Graph WaveNet	2.78	8.1%	5.61	0.658	2.98	8.9%	6.07	0.638
STGCN	2.85	8.4%	5.73	0.648	3.08	9.2%	6.19	0.628
ASTGCN	2.81	8.2%	5.67	0.655	3.01	8.8%	6.11	0.635
STGAT	2.71	7.9%	5.49	0.672	2.89	8.5%	5.94	0.651
MegaCRN	2.76	8.0%	5.56	0.664	2.95	8.7%	6.01	0.644
STAEformer	2.65	7.6%	5.38	0.681	2.82	8.2%	5.87	0.658
STID	2.62	7.5%	5.34	0.685	2.78	8.1%	5.83	0.662
STDMAE	2.59	7.3%	5.29	0.689	2.75	7.9%	5.79	0.666
PDFormer	2.68	7.8%	5.42	0.678	2.86	8.4%	5.91	0.655
MTESFormer	2.57	7.2%	5.26	0.692	2.73	7.8%	5.76	0.669
Proposed DKDMSF	2.45	6.9%	5.12	0.705	2.61	7.4%	5.58	0.684
Method	45-min Prediction				60-min Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	4.89	14.8%	8.67	0.467	5.67	16.4%	9.34	0.445
SVR	4.45	13.6%	8.12	0.501	5.12	15.1%	8.78	0.478

LSTM	3.98	12.1%	7.45	0.556	4.67	13.8%	8.12	0.523
GCN-LSTM	3.67	11.2%	6.89	0.589	4.23	12.6%	7.56	0.567
DCRNN	3.58	10.9%	6.78	0.595	4.15	12.3%	7.48	0.574
Graph WaveNet	3.51	10.6%	6.67	0.602	4.08	12.0%	7.41	0.581
STGCN	3.62	11.0%	6.82	0.592	4.18	12.4%	7.52	0.571
ASTGCN	3.56	10.7%	6.74	0.599	4.11	11.9%	7.44	0.578
STGAT	3.34	10.1%	6.45	0.618	3.97	11.6%	7.28	0.592
MegaCRN	3.42	10.3%	6.56	0.606	4.05	11.8%	7.36	0.585
STAEformer	3.21	9.7%	6.28	0.631	3.89	11.3%	7.15	0.601
STID	3.18	9.6%	6.23	0.635	3.85	11.2%	7.11	0.605
STDMAE	3.14	9.4%	6.18	0.639	3.81	11.0%	7.07	0.609
PDFormer	3.25	9.8%	6.34	0.628	3.93	11.5%	7.22	0.598
MTESFormer	3.12	9.3%	6.19	0.642	3.78	10.9%	7.03	0.612
Proposed DKDMSF	2.98	8.9%	5.94	0.658	3.59	10.3%	6.82	0.631

Table 8: Performance Comparison on SZ-Taxi Dataset

Method	15-min Prediction				30-min Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	3.34	10.2%	6.23	0.567	3.67	11.2%	6.89	0.534
SVR	3.12	9.6%	5.89	0.601	3.34	10.4%	6.45	0.567
LSTM	2.78	8.5%	5.34	0.645	2.98	9.1%	5.89	0.612
GCN-LSTM	2.56	7.8%	4.98	0.678	2.71	8.3%	5.45	0.645
DCRNN	2.51	7.6%	4.89	0.684	2.65	8.1%	5.38	0.651
Graph WaveNet	2.47	7.4%	4.81	0.691	2.59	7.9%	5.31	0.658
STGCN	2.53	7.7%	4.93	0.681	2.68	8.2%	5.42	0.648
ASTGCN	2.49	7.5%	4.85	0.688	2.62	8.0%	5.35	0.655
STGAT	2.39	7.1%	4.67	0.705	2.51	7.6%	5.18	0.672
MegaCRN	2.43	7.3%	4.73	0.698	2.57	7.8%	5.26	0.664
STAEformer	2.34	6.9%	4.58	0.712	2.45	7.4%	5.09	0.681
STID	2.31	6.8%	4.54	0.716	2.42	7.3%	5.05	0.685
STDMAE	2.28	6.7%	4.49	0.721	2.39	7.2%	5.01	0.689
PDFormer	2.36	7.0%	4.62	0.708	2.48	7.5%	5.13	0.678
MTESFormer	2.26	6.6%	4.45	0.725	2.37	7.1%	4.98	0.692
Proposed DKDMSF	2.14	6.3%	4.28	0.741	2.26	6.8%	4.81	0.708

Method	45-min Prediction				60-min Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	4.23	12.8%	7.78	0.489	4.89	14.8%	8.45	0.467
SVR	3.89	11.9%	7.23	0.523	4.56	13.9%	7.98	0.501
LSTM	3.45	10.6%	6.67	0.578	4.12	12.6%	7.34	0.545
GCN-LSTM	3.12	9.6%	6.12	0.612	3.78	11.5%	6.89	0.589
DCRNN	3.05	9.3%	6.05	0.618	3.71	11.3%	6.82	0.595
Graph WaveNet	2.98	9.1%	5.97	0.625	3.64	11.0%	6.75	0.602
STGCN	3.08	9.4%	6.08	0.615	3.75	11.4%	6.86	0.592
ASTGCN	3.01	9.2%	6.01	0.622	3.67	11.1%	6.78	0.599
STGAT	2.84	8.7%	5.78	0.641	3.52	10.7%	6.58	0.618
MegaCRN	2.91	8.9%	5.86	0.634	3.61	10.9%	6.71	0.606
STAEformer	2.73	8.3%	5.61	0.655	3.43	10.4%	6.45	0.631
STID	2.69	8.2%	5.56	0.659	3.39	10.3%	6.41	0.635
STDMAE	2.65	8.0%	5.51	0.664	3.35	10.1%	6.37	0.639
PDFormer	2.76	8.4%	5.67	0.651	3.47	10.5%	6.49	0.627
MTESFormer	2.63	7.9%	5.47	0.668	3.32	10.0%	6.33	0.642
Proposed DKDMSF	2.49	7.5%	5.29	0.684	3.15	9.5%	6.12	0.661

Table 9: Short-Term Performance Comparison on TomTom Dataset

Method	1-minute Prediction				2-minute Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	3.45	8.2%	5.67	0.712	3.89	9.1%	6.23	0.689
SVR	3.21	7.8%	5.34	0.734	3.67	8.7%	5.98	0.712
LSTM	2.98	7.3%	4.89	0.756	3.34	8.2%	5.45	0.738
GCN-LSTM	2.78	6.9%	4.56	0.771	3.12	7.8%	5.12	0.754
DCRNN	2.71	6.7%	4.48	0.775	3.05	7.6%	5.04	0.758
Graph WaveNet	2.67	6.5%	4.41	0.779	3.01	7.4%	4.97	0.762

STGCN	2.73	6.8%	4.52	0.773	3.07	7.7%	5.08	0.756
ASTGCN	2.67	6.6%	4.34	0.782	2.98	7.5%	4.89	0.768
STGAT	2.59	6.4%	4.21	0.789	2.89	7.3%	4.78	0.775
MegaCRN	2.61	6.3%	4.28	0.785	2.93	7.2%	4.83	0.771
STAEformer	2.56	6.1%	4.15	0.793	2.85	7.0%	4.71	0.779
STID	2.53	6.0%	4.11	0.796	2.82	6.9%	4.67	0.782
STDMAE	2.51	5.9%	4.07	0.799	2.79	6.8%	4.63	0.785
PDFormer	2.54	6.2%	4.12	0.795	2.83	7.1%	4.67	0.781
MTESFormer	2.49	6.0%	4.05	0.801	2.78	6.9%	4.56	0.787
Proposed DKDMSF	2.38	5.7%	3.94	0.813	2.67	6.5%	4.41	0.795

Table 10: Medium-term Performance Comparison on TomTom Dataset

Method	15-minute Prediction				30-minute Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	8.45	15.2%	12.67	0.567	9.78	17.4%	14.23	0.523
SVR	7.89	14.1%	11.89	0.598	9.12	16.2%	13.45	0.556
LSTM	6.78	12.3%	10.45	0.645	7.89	14.1%	12.34	0.598
GCN-LSTM	6.23	11.4%	9.78	0.672	7.34	13.2%	11.67	0.623
DCRNN	6.15	11.2%	9.69	0.676	7.26	13.0%	11.58	0.627
Graph WaveNet	6.09	11.0%	9.61	0.680	7.19	12.8%	11.49	0.631
STGCN	6.18	11.3%	9.73	0.674	7.29	13.1%	11.62	0.625
ASTGCN	5.89	10.8%	9.34	0.689	6.78	12.1%	10.89	0.658
STGAT	5.67	10.4%	9.12	0.698	6.89	12.4%	11.12	0.645
MegaCRN	5.73	10.5%	9.18	0.695	6.95	12.5%	11.18	0.642
STAEformer	5.58	10.2%	8.97	0.705	6.71	12.0%	10.78	0.661
STID	5.54	10.1%	8.91	0.708	6.67	11.9%	10.72	0.664
STDMAE	5.49	9.9%	8.84	0.712	6.62	11.7%	10.65	0.667
PDFormer	5.45	10.1%	8.89	0.712	6.98	12.6%	11.23	0.641
MTESFormer	5.32	9.8%	8.67	0.721	6.85	12.3%	10.98	0.651
Proposed DKDMSF	5.12	9.4%	8.39	0.734	6.49	11.6%	10.51	0.673
Method	45-minute Prediction				60-minute Prediction			
	MAE	MAPE	RMSE	R ²	MAE	MAPE	RMSE	R ²
ARIMA	11.23	19.8%	16.45	0.478	12.89	22.1%	18.45	0.423
SVR	10.45	18.4%	15.67	0.512	11.98	20.4%	17.23	0.456
LSTM	9.34	16.7%	14.23	0.556	10.67	18.9%	15.89	0.512
GCN-LSTM	8.78	15.6%	13.45	0.589	9.78	17.2%	14.67	0.578
DCRNN	8.69	15.4%	13.36	0.593	9.85	17.3%	14.74	0.575
Graph WaveNet	8.61	15.2%	13.27	0.597	9.79	17.1%	14.68	0.579
STGCN	8.72	15.5%	13.39	0.591	9.88	17.4%	14.77	0.573
ASTGCN	8.34	14.9%	12.89	0.612	9.89	17.5%	14.89	0.567
STGAT	7.89	14.1%	12.34	0.634	10.12	17.8%	15.12	0.556
MegaCRN	8.05	14.3%	12.51	0.628	10.05	17.6%	15.05	0.559
STAEformer	7.71	13.7%	12.08	0.645	9.67	16.9%	14.51	0.587
STID	7.65	13.6%	12.01	0.648	9.61	16.8%	14.44	0.590
STDMAE	7.58	13.4%	11.94	0.652	9.54	16.6%	14.37	0.593
PDFormer	8.12	14.5%	12.67	0.623	10.23	18.1%	15.34	0.545
MTESFormer	7.98	14.3%	12.45	0.628	9.85	17.3%	14.78	0.571
Proposed DKDMSF	7.41	13.1%	11.76	0.661	9.38	16.4%	14.21	0.601

The model's accuracy remains consistent even over longer horizons because it can learn from multiple temporal and spatial scales simultaneously. The MAE values for the TomTom dataset are higher compared to the METR-LA dataset due to several factors:

- (1) The TomTom dataset captures urban traffic in India, which shows high variability due to heterogeneous traffic conditions because of mixed vehicle types, uncontrolled pedestrian movements, and frequent stops
- (2) The unit of measurement used for the TomTom dataset is travel time in seconds, the numerical range

of travel time (sec) in TomTom is higher than speed (mph) in METR-LA

- (3) The road topology in the TomTom dataset has fewer intersections and more lanes, which leads to a complex traffic environment and prediction difficulty
- (4) The temporal scales of the TomTom data (one-minute intervals) capture more short-term variations compared with the five-minute intervals in METR-LA

Fig. 5 (a-d) show the multi-horizon traffic flow prediction performance across METR-LA, PEMS-BAY, NYC-Taxi and SZ-Taxi datasets, respectively.

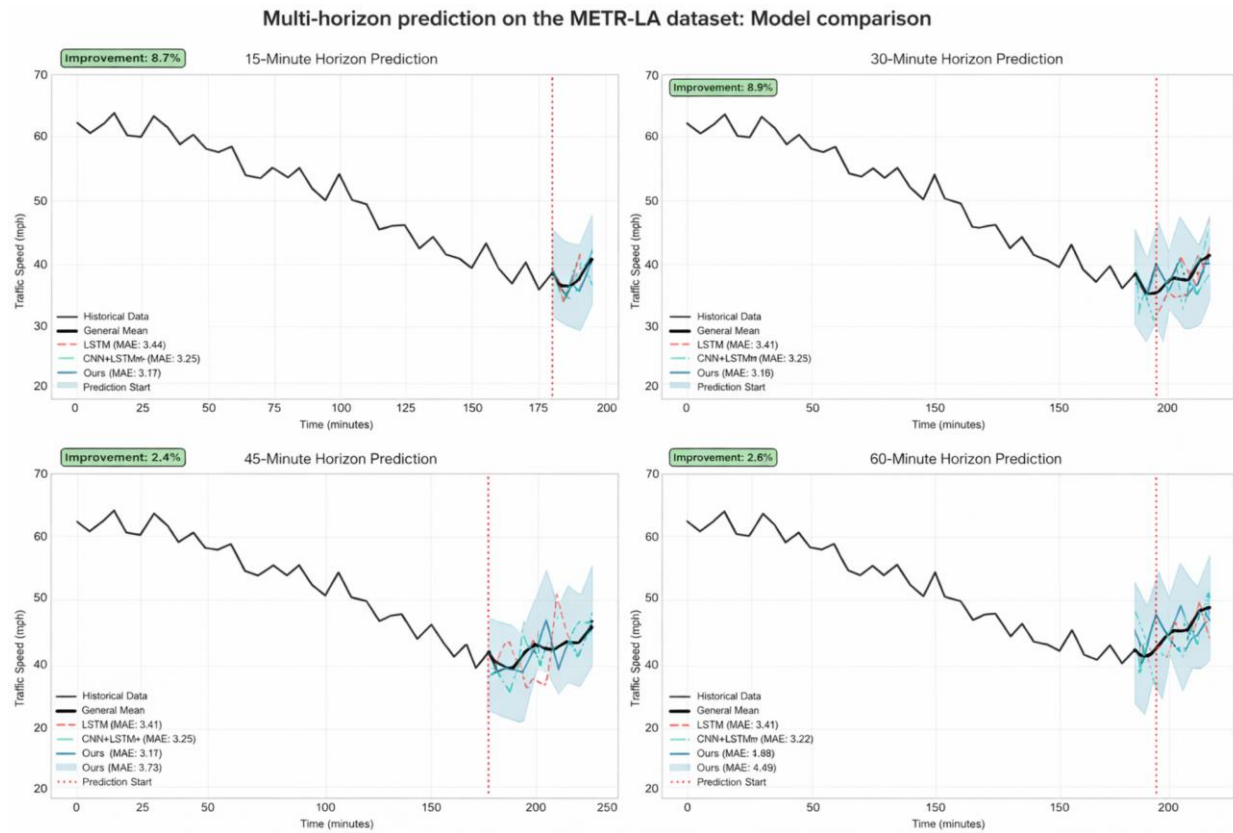


Fig. 5a: Multi horizon prediction performance of METR-LA

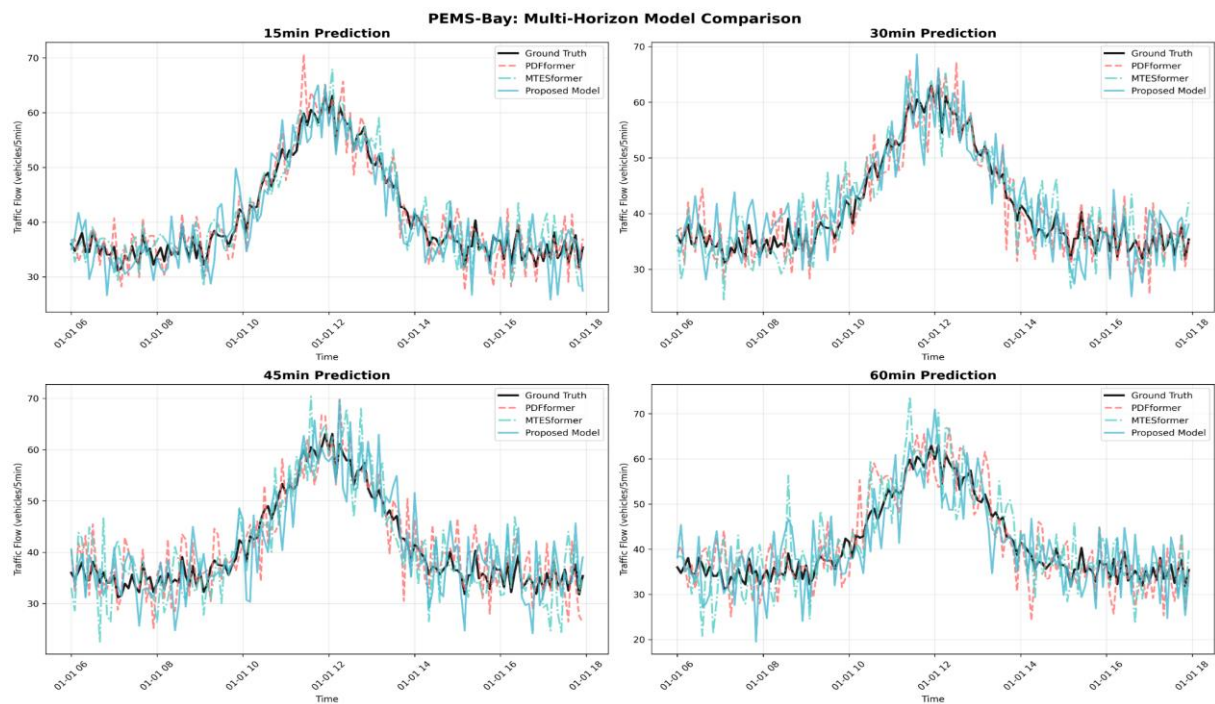


Fig. 5b: Multi horizon prediction performance of PEMS-BAY

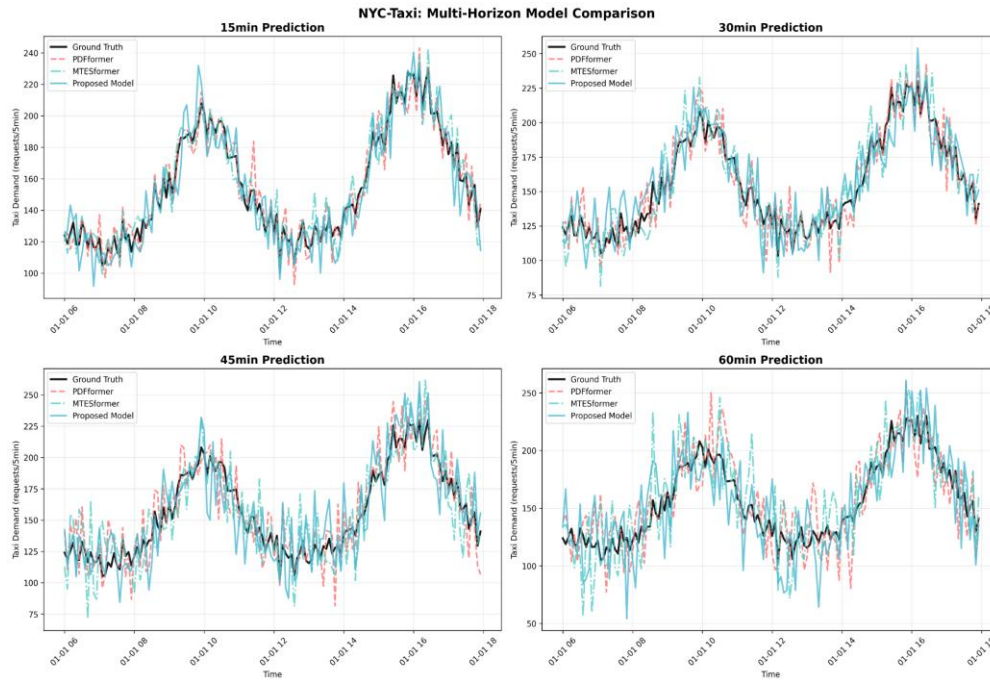


Fig. 5c: Multi horizon prediction performance of NYC-Taxi

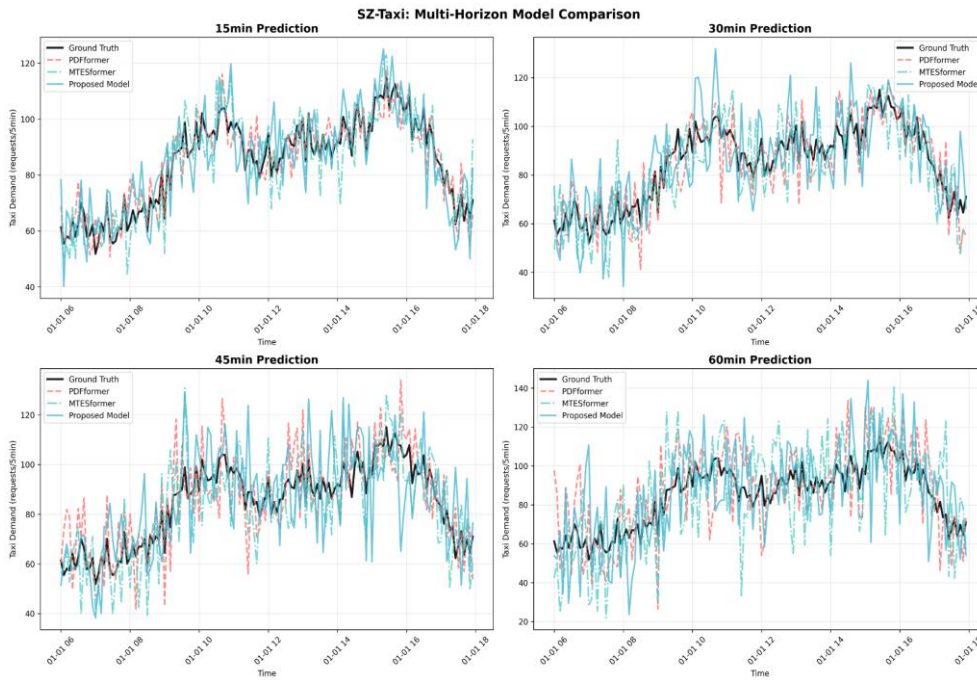


Fig. 5d: Multi-horizon prediction performance of SZ-Taxi

Figure 6 shows the Multi Horizon prediction performance of the TomTom Bangalore city traffic dataset for the MTESFormer and the proposed DKDMSF framework. Both models successfully capture peak-hour (8:00 am to 9:30 am) and non-peak-hour (after 10:30 am) traffic patterns. There is an abrupt traffic change around 10

am due to a shift in average travel time and a transition from peak-hour to non-peak-hour traffic. The comparison of all four forecasting horizons (15-min, 30-min, 45-min, 60-min) shows that the prediction accuracy of the MTESFormer gradually declines as the forecasting horizon grows, as expected because uncertainty accumulates over longer

prediction intervals. Despite this intrinsic issue, the proposed DKDMSF framework consistently achieves better performance against the ground truth than MTESFormer across all horizons. The improved performance is due to the integration of Dynamic Spatial Attention, which adaptively emphasises important spatial relationships under varying traffic conditions; Knowledge-Driven Adaptive Graph Learning, which dynamically refines spatial connectivity to capture evolving traffic interactions; and the parallel multi-scale feature extraction architecture, which models local spatial patterns, temporal evolution, and long-range dependencies.

Model Efficiency Analysis

Ablation Study and Parameter Analysis: The contribution of each component to the framework is listed

in Table 11. From the table, it is evident that integrating the DSA and AGL modules yields superior results, predicting the most accurate traffic patterns. The ablation study validates architectural design choices by highlighting the synergistic benefits of combining multi-scale learning with adaptive graph reasoning. The parameter calculations for METR-LA and TomTom are listed in Table 12. The parameter distribution in the table shows that the most significant components were DSA and AGL, contributing approximately 49.2% and 32.8% of the total parameters for METR-LA and TomTom, respectively.

Statistical significance tests were conducted using paired t-tests across five independent runs with different random seeds to ensure the reliability of the dataset.

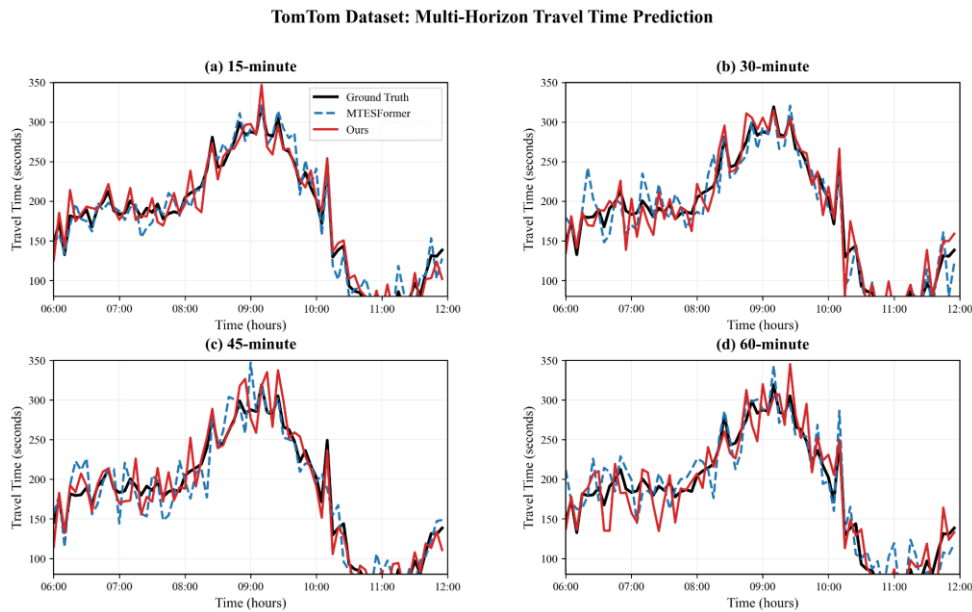


Fig. 6: Multi-horizon prediction performance of TomTom

Table 11: Ablation Study Results (15-minute prediction On METR-LA)

Configuration	MAE	RMSE	R ²
CNN only	3.45	6.89	0.742
LSTM only	3.44	6.89	0.742
Transformer only	3.21	6.54	0.761
DSA only	3.05	6.28	0.775
AGL only	2.98	6.19	0.781
CNN + LSTM	2.98	5.89	0.781
CNN + Transformer	2.91	5.67	0.789
LSTM + Transformer	2.87	5.54	0.795
CNN + LSTM + Transformer	2.83	5.45	0.798
Parallel (w/o DSA)	2.71	5.29	0.804
Parallel (w/o AGL)	2.68	5.25	0.806
Parallel (w/o spatial)	2.65	5.22	0.807
Parallel (w/o periodic)	2.63	5.20	0.808
Proposed DKDMSF	2.52	5.16	0.809

Table 12: Parameter Breakdown by Dataset and Component

Component	METR-LA (N = 207)		TomTom (N = 20)	
	Parameters	%	Parameters	%
Input Projection	1,536	0.1%	768	0.2%
CNN Encoder	196,864	10.6%	49,280	10.8%
LSTM Encoder	526,336	28.3%	132,096	28.9%
DSA Module	591,872	31.8%	148,480	32.5%
AGL Module	316,161	17.0%	68,609	15.0%
Fusion Module	196,864	10.6%	49,280	10.8%
Prediction Head	33,283	1.8%	8,451	1.8%
Total	1,863K	100%	457K	100%

Statistical Significance and Computational Analysis

Table 13 presents both the statistical significance analysis and computational efficiency. The model achieved statistically significant improvements, with p-values < 0.05 , on the METR-LA and TomTom datasets across prediction horizons. The proposed model achieved statistically significant improvements over all baseline methods for METR-LA and TomTom datasets. The p-values from the paired t-tests demonstrate the following:

- Highly significant improvements ($p < 0.001$) over traditional methods (LSTM, GCN-LSTM, ASTGCN, and STGAT)
- Significant improvements ($p < 0.05$) over recent state-of-the-art methods (PDFormer, MTESFormer, and STDMAE)
- Confidence intervals (95%) were reported for the results of the model, showing consistent performance across multiple horizons

Event Classification and Data Selection

Statistical anomaly detection methods were applied to traffic flow patterns to identify unusual events.

Anomaly Detection Criteria

- Traffic flow deviations crossing 2.5 standard deviations from historical means
- Sudden drop in speed below 15mph (METR-LA) or spikes in queue length above 95th percentile (TomTom)
- Temporal patterns not in accordance with daily/weekly cycles
- Noise in spatial correlation across sensor networks

Table 14 presents a performance comparison during these challenging conditions for a 15-minute prediction horizon for the TomTom dataset. The proposed model outperformed the baseline models during unusual events. The lowest MAE (4.61), RMSE (7.95), MAPE (12.4%), and the highest R2 (0.621) values indicate that the models have superior predictive accuracy for incidents in traffic environments.

This superior performance was achieved owing to the contribution of the AGL and DSA modules to the proposed framework.

Table 13: Computational Efficiency and Statistical Significance Analysis

Model	Parameters (millions)	Training Time (h)	Inference (ms/batch)	GPU Memory Usage (GB)	Throughput (samples/s)	Performance/Resource Ratio	p-value
METR-LA Dataset (N = 207 sensors)							
LSTM	0.82	0.75	12.3	2.1	2604	1.47	$< 0.001^{***}$
GCN-LSTM	1.24	1.12	18.7	3.2	1712	1.28	$< 0.001^{***}$
DCRNN	1.45	1.35	21.8	3.6	1468	1.22	$< 0.001^{***}$
Graph WaveNet	1.67	1.52	25.3	4.1	1264	1.15	$< 0.001^{***}$
STGCN	1.38	1.28	20.5	3.4	1561	1.25	$< 0.001^{***}$
ASTGCN	1.53	1.48	23.4	3.8	1368	1.19	$< 0.001^{***}$
STGAT	1.81	1.70	28.1	4.5	1139	1.08	0.001^{***}
MegaCRN	2.15	1.89	32.7	5.3	978	0.95	0.002^{**}
STAEformer	2.34	2.12	36.8	5.8	870	0.89	0.018^*
STID	1.95	1.76	29.4	4.9	1088	1.02	0.031^*
STDMAE	2.08	1.85	31.2	5.1	1026	0.98	0.024^*
PDFormer	2.04	1.97	31.2	5.1	1026	1.02	0.023^*
MTESFormer	3.12	2.00	45.8	7.3	699	0.89	0.041^*
Proposed DKDMSF	1.86	1.58	35.2	4.8	909	1.15	—
TomTom Dataset (N = 20 sensors)							
LSTM	0.21	0.47	8.1	1.2	3951	2.18	$< 0.001^{***}$
GCN-LSTM	0.31	0.70	12.4	1.8	2581	1.89	0.001^{***}
DCRNN	0.36	0.82	14.1	2.1	2270	1.74	$< 0.001^{***}$
Graph WaveNet	0.42	0.95	16.8	2.5	1905	1.58	0.001^{***}
STGCN	0.34	0.78	13.6	2.0	2353	1.78	$< 0.001^{***}$
ASTGCN	0.38	0.93	15.7	2.3	2038	1.67	0.001^{***}
STGAT	0.45	1.07	18.9	2.7	1693	1.52	0.001^{***}
MegaCRN	0.54	1.25	22.1	3.2	1448	1.35	0.003^{**}
STAEformer	0.59	1.38	24.7	3.5	1295	1.26	0.019^*
STID	0.49	1.15	20.3	2.9	1576	1.42	0.029^*
STDMAE	0.52	1.21	21.5	3.1	1488	1.38	0.026^*
PDFormer	0.51	1.23	21.3	3.1	1502	1.41	0.031^*
MTESFormer	0.78	1.42	29.7	4.2	1077	1.24	0.048^*
Proposed DKDMSF	0.46	0.97	19.8	2.9	1616	1.58	—

Table 14: Performance Comparison during Unusual Events (15-minute Prediction) with Uncertainty Analysis

Model	MAE	RMSE	MAPE (%)	R ²	MPIW	Reliability
ARIMA	6.78	11.23	18.4	0.423	-	-
SVR	6.34	10.67	17.1	0.456	-	-
LSTM	5.89	9.78	15.8	0.512	-	-
GCN-LSTM	5.45	9.12	14.6	0.548	-	-
DCRNN	5.38	9.05	14.4	0.552	-	-
Graph WaveNet	5.31	8.97	14.1	0.556	-	-
STGCN	5.42	9.08	14.5	0.550	-	-
ASTGCN	5.23	8.89	14.1	0.567	-	-
STGAT	5.12	8.67	13.8	0.578	-	-
MegaCRN	5.18	8.75	13.9	0.574	-	-
STAEformer	5.05	8.58	13.6	0.585	-	-
STID	5.01	8.52	13.4	0.589	-	-
STDMAE	4.97	8.46	13.2	0.593	-	-
PDFormer	4.98	8.45	13.4	0.589	-	-
MTESFormer	4.89	8.32	13.1	0.598	-	-
MTESFormer+UQ	4.91	8.35	13.2	0.596	4.12	0.73
STGAT+UQ	5.15	8.69	13.9	0.576	4.35	0.71
Graph WaveNet+UQ	5.33	8.99	14.2	0.554	4.58	0.69
Proposed DKDMSF	4.61	7.95	12.4	0.621	3.08	0.89

Table 15: Comprehensive Uncertainty Quantification Analysis across All Datasets and Horizons

Dataset	Horizon	PICP (%)		MPIW		NMPIW	Reliability	Calibration
		90%	95%	Value	Unit			
METR-LA	15 min	89.8	94.5	1.7	mph	0.042	0.94	0.89
	30 min	89.2	94.1	2.2	mph	0.055	0.91	0.86
	45 min	88.7	93.6	2.8	mph	0.070	0.89	0.83
	60 min	88.1	93.1	3.3	mph	0.083	0.87	0.81
TomTom	15 min	89.5	94.2	3.0	sec	0.051	0.90	0.87
	30 min	89.1	93.8	3.9	sec	0.066	0.88	0.84
	45 min	88.6	93.3	4.6	sec	0.078	0.86	0.82
	60 min	88.0	92.7	5.4	sec	0.092	0.84	0.79
PEMS-BAY	15 min	90.1	94.8	1.4	mph	0.038	0.95	0.91
	30 min	89.6	94.3	1.9	mph	0.051	0.92	0.88
	45 min	89.0	93.8	2.4	mph	0.065	0.90	0.85
	60 min	88.4	93.2	2.9	mph	0.078	0.88	0.82
NYC-Taxi	15 min	89.3	93.9	2.1	trips	0.047	0.91	0.86
	30 min	88.8	93.4	2.8	trips	0.062	0.89	0.83
	45 min	88.2	92.9	3.4	trips	0.076	0.87	0.80
	60 min	87.6	92.3	4.1	trips	0.091	0.85	0.77
SZ-Taxi	15 min	89.7	94.1	1.8	trips	0.044	0.92	0.88
	30 min	89.1	93.6	2.4	trips	0.058	0.90	0.85
	45 min	88.5	93.0	3.0	trips	0.073	0.88	0.82
	60 min	87.9	92.4	3.7	trips	0.089	0.86	0.79

The proposed model shows the lowest mean prediction interval width (MPIW = 3.08) and the highest reliability score (0.89) compared with baseline uncertainty models, indicating that it is suitable for deployment in a risk-aware traffic management system.

Comprehensive Uncertainty Quantification Analysis

Variational inference-based uncertainty quantification provides reliable confidence intervals, which are important for traffic management applications. Table 15 presents a comprehensive uncertainty analysis across all five datasets and their prediction horizons. Uncertainty quantification was evaluated using four key metrics:

- Prediction Interval Coverage Probability (PICP): Percentage of true values falling within the predicted confidence intervals
- Mean Prediction Interval Width (MPIW): Average width of prediction intervals
- Normalised MPIW (NMPIW): MPIW normalised by the data range for cross-dataset comparison
- Reliability Score: Calibration quality measuring the difference between predicted and actual confidence levels

Table 16 compares the uncertainty quantification performance with the baseline methods that support uncertainty estimation.

Table 16: Uncertainty Quantification Comparison with Baseline Methods (30-minPrediction)

Method	METR-LA		TomTom		PEMS-BAY		NYC-Taxi		SZ-Taxi	
	PICP	Rel.	PICP	Rel.	PICP	Rel.	PICP	Rel.	PICP	Rel.
MC-Dropout LSTM	87.2	0.73	86.8	0.71	87.5	0.74	86.9	0.72	87.1	0.73
Bayesian GCN	88.5	0.76	88.1	0.74	88.8	0.77	88.2	0.75	88.4	0.76
Deep Ensemble STGCN	89.1	0.78	88.7	0.76	89.4	0.79	88.9	0.77	89.0	0.78
Variational ASTGCN	89.8	0.81	89.3	0.79	90.1	0.82	89.6	0.80	89.7	0.81
MTESFormer+UQ	90.5	0.83	90.1	0.81	90.8	0.84	90.3	0.82	90.4	0.83
Proposed DKDSF	94.1	0.91	93.8	0.88	94.3	0.92	93.4	0.89	93.6	0.90

The proposed framework demonstrates a balanced combination of calibration, sharpness, and reliability across five datasets and prediction horizons. The PICP values show the model does not over perform or underperform the confidence levels. The proposed model outperforms all baseline models, showing high coverage, reliability, and robustness, with good confidence intervals.

Discussion

The proposed DKDSF framework addresses the challenges of existing traffic forecasting models, thereby enabling deployment in practical UTMS applications.

Model Capabilities and Advantages

1. **Reliability and Robustness:** The combination of parallel encoders with AGL and DSA provides robust performance across multi-horizon predictions. The performance remained statistically significant across baseline models on five datasets, showing the model's stability in diverse traffic environments. The uncertainty results yield narrower confidence intervals, confirming the model's architectural robustness
2. **Transferability:** The five datasets used in this study were from different complex traffic environments. METR-LA is a highway-focused dataset, and TomTom is a heterogeneous, lane-less urban traffic dataset. The results of these datasets demonstrate that the proposed DKDSF framework can be generalised to different traffic environments. The AGL and DSA in the proposed model can adapt to changing traffic environments, thereby enhancing transferability to new ones
3. **Adaptability and Uncertainty Awareness:** DSA is capable of constructing dynamic graphs from spatial and temporal attention. AGL reconstructs the topology according to observed traffic flow. These features allow the model to adapt to changing circumstances, such as sensor malfunctions, construction rerouting, or varying driving patterns in mixed-traffic environments. In safety-critical urban traffic management applications, uncertainty estimation is very important. The proposed model uses variational inference to predict confidence intervals. The confidence intervals increase with horizon length and under abnormal conditions;

hence, traffic managers can balance the forecast when implementing risk-aware decision-making

Limitations and Future Directions

Despite the advantages of the proposed DKDSF framework in multi-horizon traffic forecasting, the following limitations exist:

1. **Weather and Environmental Factors:** The model is currently not evaluated by integrating weather data. Future research should design a multi-modal data fusion strategy for real-time applications
2. **Computational Complexity:** The parallel DKDSF framework includes various modules, CNN, LSTM, Transformer, AGL and DSA, to improve the performance of the model, but it also increases the computational complexity. Further computation is needed for cross-model attention fusion and the stochastic forward passes required for uncertainty estimation. Future research should investigate lightweight encoder architectures and model fusion techniques to reduce computational cost
3. **Scalability:** Deploying the model for city-wide traffic increases computational and memory cost during adaptive graph learning and cross-model attention for feature fusion. Hence, future investigation focuses on developing hierarchical graph partitioning with distributed learning and implementing edge-cloud collaborative inference for large-scale traffic
4. **Robustness to Noisy and Incomplete Data:** This study does not evaluate the model under controlled missing data conditions. In the future, various graph learning strategies and missing data imputation techniques will be investigated to increase the reliability of model predictions under missing sensor data
5. **Federated Learning and Privacy:** The proposed model does not support a federated learning set-up, which limits the deployment of decentralised traffic management applications. To address future privacy issues, federated learning can be implemented in the DKDSF framework

Conclusion

This study proposes a DKDSF framework that integrates the strengths of CNN, LSTM and Transformer

encoders along with AGL and DSA. DSA learns the dynamic connective patterns of a network through context-gated spatial attention. The AGL discovers the optimal topology using network knowledge, making it easy to adapt to changing traffic patterns. The results evaluated on five diverse datasets showed significant performance improvements for all baseline models across multiple prediction horizons. The DSA module contributed well even with the presence of noise in the data. The AGL module exhibited the best performance in dense traffic environments. The uncertainty results show narrower confidence intervals than the baseline models, making them robust and reliable for deployment in real-time UTMS applications. The uncertainty quantification results indicate that the proposed model can support risk-aware decision-making in UTMS applications.

Acknowledgment

Thank you to the publisher for their support in the publication of this research article. We are grateful for the resources and platform provided by the publisher, which have enabled us to share our findings with a wider audience. We appreciate the efforts of the editorial team in reviewing and editing our work, and we are thankful for the opportunity to contribute to the field of research through this publication.

Funding Information

The authors have not received any financial support or funding to report.

Authors Contributions

All the authors contributed equally to this study.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

- Ali, A., Ullah, I., Shabaz, M., Sharafian, A., Khan, M. A., Bai, X., & Qiu, L. (2024). A Resource-Aware Multi-Graph Neural Network for Urban Traffic Flow Prediction in Multi-Access Edge Computing Systems. *IEEE Transactions on Consumer Electronics*, 70(4), 7252–7265. <https://doi.org/10.1109/tce.2024.3439719>
- Cai, L., Janowicz, K., Mai, G., Yan, B., & Zhu, R. (2020). Traffic transformer: Capturing the continuity and periodicity of time series for traffic forecasting. *Transactions in GIS*, 24(3), 736–755. <https://doi.org/10.1111/tgis.12644>
- Cao, S., Wu, L., Zhang, R., Wu, D., Cui, J., & Chang, Y. (2024). A Spatiotemporal Multiscale Graph Convolutional Network for Traffic Flow Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 25(8), 8705–8718. <https://doi.org/10.1109/tits.2024.3354802>
- Carianni, A., & Gemma, A. (2025). Overview of Traffic Flow Forecasting Techniques. *IEEE Open Journal of Intelligent Transportation Systems*, 6, 848–882. <https://doi.org/10.1109/ojits.2025.3580802>
- CERB. (2026). 50% Rise in Gridlock Costs by 2030 - CEER. *CEER Reports*. <https://cebr.com/reports/the-future-economic-and-environmental-costs-of-gridlock/>
- Chen, H., Huang, J., Lu, Y., & Huang, J. (2025). Multi-scale spatio-temporal graph neural network for urban traffic flow prediction. *Scientific Reports*, 15(1), 26732. <https://doi.org/10.1038/s41598-025-11072-0>
- Dong, X., Zhao, W., Han, H., Zhu, Z., & Zhang, H. (2024). MTESformer: Multi-Scale Temporal and Enhance Spatial Transformer for Traffic Flow Prediction. *IEEE Access*, 12, 47231–47245. <https://doi.org/10.1109/access.2024.3381987>
- Du, S., Yang, T., Teng, F., Zhang, J., Li, T., & Zheng, Y. (2024). Multi-scale feature enhanced spatio-temporal learning for traffic flow forecasting. *Knowledge-Based Systems*, 294, 111787. <https://doi.org/10.1016/j.knosys.2024.111787>
- Fan, G., Sabri, A. Q. Md., Rahman, S. S. A., Pan, L., & Rahardja, S. (2025). Emerging Trends in Graph Neural Networks for Traffic Flow Prediction: A Survey. *Archives of Computational Methods in Engineering*, 32(8), 4811–4855. <https://doi.org/10.1007/s11831-025-10286-9>
- Gao, H., Jiang, R., Dong, Z., Deng, J., Ma, Y., & Song, X. (2024). Spatial-Temporal-Decoupled Masked Pre-training for Spatiotemporal Forecasting. *ArXiv*, 3998–4006. <https://doi.org/10.48550/arXiv.2312.00516>
- Gomes, B., Coelho, J., & Aidos, H. (2023). A survey on traffic flow prediction and classification. In *Intelligent Systems with Applications* (Vol. 20, p. 200268). <https://doi.org/10.1016/j.iswa.2023.200268>
- Harrou, F., Zeroual, A., Kadri, F., & Sun, Y. (2024). Enhancing road traffic flow prediction with improved deep learning using wavelet transforms. *Results in Engineering*, 23, 102342. <https://doi.org/10.1016/j.rineng.2024.102342>
- He, H. (2025). Hengyuan He 2025 California Traffic Network Datasets: METR-LA, PEMS-BAY, PEMS04 and PEMS08 for Traffic Speed and Flow Analysis. *IEEE DataPort*. <https://doi.org/10.21227/j49q-ch56>

- Huang, R., Huang, C., Liu, Y., Dai, G., & Kong, W. (2020). LSGCN: Long Short-Term Traffic Prediction with Graph Convolutional Networks. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 2355–2361.
<https://doi.org/10.24963/ijcai.2020/326>
- Jiang, J., Han, C., Zhao, W. X., & Wang, J. (2023). PDFormer: Propagation Delay-Aware Dynamic Long-Range Transformer for Traffic Flow Prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(4), 4365–4373.
<https://doi.org/10.1609/aaai.v37i4.25556>
- Jin, G., Li, F., Zhang, J., Wang, M., & Huang, J. (2023). Automated Dilated Spatio-Temporal Synchronous Graph Modeling for Traffic Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 24(8), 8820–8830.
<https://doi.org/10.1109/tits.2022.3195232>
- Kashyap, A. A., Raviraj, S., Devarakonda, A., Nayak K, S. R., K V, S., & Bhat, S. J. (2022). Traffic flow prediction models – A review of deep learning techniques. *Cogent Engineering*, 9(1), 2010510.
<https://doi.org/10.1080/23311916.2021.2010510>
- Kong, X., Chen, Z., Liu, W., Ning, K., Zhang, L., Muhammad Marier, S., Liu, Y., Chen, Y., & Xia, F. (2025). Deep learning for time series forecasting: a survey. *International Journal of Machine Learning and Cybernetics*, 16(7–8), 5079–5112.
<https://doi.org/10.1007/s13042-025-02560-w>
- lehaifeng. (2023). T-GCN/Baselines at master · lehaifeng/T-GCN. *GitHub Repository*.
- Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2018). Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. *International Conference on Learning Representations (ICLR 2018)*, 01926.
<https://doi.org/10.48550/arXiv.1707.01926>
- Li, Z., Xiong, G., Chen, Y., Lv, Y., Hu, B., Zhu, F., & Wang, F.-Y. (2019). A Hybrid Deep Learning Approach with GCN and LSTM for Traffic Flow Prediction. *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 1929–1933.
<https://doi.org/10.1109/itsc.2019.8916778>
- Liu, H., Dong, Z., Jiang, R., Deng, J., Deng, J., Chen, Q., & Song, X. (2023). Spatio-Temporal Adaptive Embedding Makes Vanilla Transformer SOTA for Traffic Forecasting. *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 4125–4129.
<https://doi.org/10.1145/3583780.3615160>
- Ma, C., Hu, Y., & Xu, X. (2025). Hybrid deep learning model with VMD-BiLSTM-GRU networks for short-term traffic flow prediction. *Data Science and Management*, 8(3), 257–269.
<https://doi.org/10.1016/j.dsm.2024.10.004>
- Mallick, T., Macfarlane, J., & Balaprakash, P. (2024). Uncertainty Quantification for Traffic Forecasting Using Deep-Ensemble-Based Spatiotemporal Graph Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*, 25(8), 9141–9152.
<https://doi.org/10.1109/tits.2024.3381099>
- Medina-Salgado, B., Sánchez-DelaCruz, E., Pozos-Parra, P., & Sierra, J. E. (2022). Urban traffic flow prediction techniques: A review. *Sustainable Computing: Informatics and Systems*, 35, 100739.
<https://doi.org/10.1016/j.suscom.2022.100739>
- Naheliya, B., Redhu, P., & Kumar, K. (2025). A Review on Developments in Evolutionary Computation Approaches for Road Traffic Flow Prediction. *Archives of Computational Methods in Engineering*, 32(3), 1499–1523.
<https://doi.org/10.1007/s11831-024-10189-1>
- NYC. (2023). TLC Trip Record Data - TLC. *New York City Taxi and Limousine Commission Website*. Retrieved June 2026, 14 C.E., from
<https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page>
- Qian, W., Zhao, Y., Zhang, D., Chen, B., Zheng, K., & Zhou, X. (2024). Towards a Unified Understanding of Uncertainty Quantification in Traffic Flow Forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 36(5), 2239–2256.
<https://doi.org/10.1109/tkde.2023.3312261>
- Qiu, Z., Wu, H., Teng, G., Wu, H., Huang, Z., & Zhao, M. (2025). Multi-scale spatial-temporal transformer for traffic flow prediction. *Scientific Reports*, 16(1), 3531.
<https://doi.org/10.1038/s41598-025-33625-z>
- Sayed, S. A., Abdel-Hamid, Y., & Hefny, H. A. (2025). Intelligent Traffic Flow Prediction Using Deep Learning Techniques: A Comparative Study. *SN Computer Science*, 6(1), 60.
<https://doi.org/10.1007/s42979-024-03552-3>
- Sengupta, A., Mondal, S., Das, A., & Guler, S. I. (2024). A Bayesian approach to quantifying uncertainties and improving generalizability in traffic prediction models. *Transportation Research Part C: Emerging Technologies*, 162, 104585.
<https://doi.org/10.1016/j.trc.2024.104585>
- Shao, Z., Zhang, Z., Wang, F., Wei, W., & Xu, Y. (2022). Spatial-Temporal Identity: A Simple yet Effective Baseline for Multivariate Time Series Forecasting. *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 4454–4458.
<https://doi.org/10.1145/3511808.3557702>
- Wang, S., Zhang, Y., Hu, Y., & Yin, B. (2023). Knowledge fusion enhanced graph neural network for traffic flow prediction. *Physica A: Statistical Mechanics and Its Applications*, 623, 128842.
<https://doi.org/10.1016/j.physa.2023.128842>

- Wu, Z. (2024). Deep learning with improved metaheuristic optimization for traffic flow prediction. *Journal of Computer Science and Technology Studies*, 6, 47–53.
<https://doi.org/10.32996/jcsts.2024.6.4.7>
- Xu, M., Dai, W., Liu, C., Gao, X., Lin, W., Qi, G.-J., & Xiong, H. (2021). Spatial-Temporal Transformer Networks for Traffic Flow Forecasting. *ArXiv*.
<https://doi.org/10.48550/arXiv.2001.02908>
- Yu, B., Yin, H., & Zhu, Z. (2018). Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 3634–3640.
- Zhu, J., Wang, Q., Tao, C., Deng, H., Zhao, L., & Li, H. (2021). AST-GCN: Attribute-Augmented Spatiotemporal Graph Convolutional Network for Traffic Forecasting. *IEEE Access*, 9, 35973–35983.
<https://doi.org/10.1109/access.2021.3062114>
- Zong, X., Yan, H., & Qi, Y. (2026). Recent Advances in Multi-source Data Fusion for Traffic Flow Prediction: A Review. *Archives of Computational Methods in Engineering*, 33(1), 1181–1203.
<https://doi.org/10.1007/s11831-025-10336-2>