Robustness of Spatio-Temporal Graph Neural Networks for Traffic Forecasting Under Realistic Sensor Degradation

Shreya N. Desai[1*]., Kasim Ishaque Ghanchi[2]., Ali Mehdi Mirza[3]., Atif Khan[4]

[1]San Francisco, USA

[2,3]Vidyalankar Institute of Technology, University of Mumbai, Mumbai, Maharashtra, India

[4]Fr. Conceicao Rodrigues Institute of Technology, University of Mumbai, Mumbai, Maharashtra, India

[*]Corresponding Author

DOI: https://doi.org/10.51244/IJRSI.2026.1307000373

Received: 08 August 2026; Accepted: 14 August 2026; Published: 20 August 2026

ABSTRACT

Short-term traffic forecasting plays an important role in intelligent transportation systems, as applications such as route guidance, adaptive traffic signal control, emergency response, congestion mitigation, and logistics planning depend on accurate estimates of future traffic states. Recent spatio-temporal graph neural network models, particularly Diffusion Convolutional Recurrent Neural Network (DCRNN) and Spatio-Temporal Graph Convolutional Network (STGCN), have improved traffic prediction by representing road sensors as graph nodes and jointly learning spatial and temporal relationships. However, strong performance on clean benchmark datasets does not always guarantee reliability in real-world conditions, where sensor reading may be missing, noisy, or unavailable due to hardware faults or communication issues. This paper provides a robust focused comparative evaluation of seven traffic forecasting approaches: Persistence, Historical Average, ARIMA, Random Forest, LSTM, STGCN, and DCRNN. Experiments are conducted on the METR-LA and PEMS-BAY speed datasets using 5-minute data intervals, a 12-step historical input sequence, and a 12-step forecasting horizon. Under clean-data conditions, DCRNN achieves the best overall MAE on both datasets, with 3.548 mph on METR-LA and 1.905 mph on PEMS-BAY. However, when 40% random missing input corruption is introduced, STGCN shows greater robustness than DCRNN; for METR-LA, STGCN's MAE increases by 27.0%, while DCRNN's MAE increases by 34.5%. Under complete sensor-failure conditions, the model ranking changes further, with the LSTM baseline showing greater stability than both graph-based models on METR-LA. Graph ablation analysis also indicates that temporal modeling accounts for most of the forecast improvement, while the evaluated graph topology provides only limited additional benefits. These findings suggest that traffic forecasting models should be assessed not only by clean-data accuracy but also by their robustness under degraded sensing conditions before deployment in real intelligent transportation systems.

Index Terms: Traffic forecasting, spatio-temporal graph neural networks, robustness, sensor failure, missing data, STGCN, DCRNN, intelligent transportation systems.

INTRODUCTION

Short-term traffic forecasting predicts future road speed, flow, or occupancy from recent sensor observations and supports route planning, adaptive signal control, emergency dispatch, congestion management, and autonomous mobility services [17], [18]. The task is difficult because traffic patterns are nonlinear, non-stationary, spatially coupled, and horizon dependent. A speed reduction at one detector can propagate to downstream detectors after several time steps, while rush-hour periodicity and incident-induced regime shifts create complex temporal dynamics.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Classical approaches such as Historical Average, ARIMA, and Random Forest provide useful forecasting baselines, but they do not explicitly exploit the graph structure of road networks [12], [13], [19]. Recurrent neural networks and LSTMs improve temporal modeling through gated memory mechanisms [8], yet they often treat the complete sensor vector as a flat multivariate sequence. Spatio-temporal graph neural networks address this limitation by representing sensors as graph nodes and propagating information over predefined or learned edges. DCRNN models traffic flow as diffusion over a directed graph [1], while STGCN combines graph convolution with gated temporal convolution [2].

Despite strong clean-data performance, ST-GNNs are not automatically reliable under degraded sensing conditions. Loop detectors may fail, communication links may drop packets, and maintenance gaps may produce random or structured missing values. Most forecasting benchmarks emphasize MAE, RMSE, and MAPE on cleaned datasets; however, deployment requires understanding how error changes when input quality deteriorates [9], [10].

This study investigates the robustness of traditional, deep-learning, and graph-based forecasting models under controlled sensor degradation using publicly available benchmark experiments and documented outputs [25]. The evaluation compares clean-data accuracy with robustness under random missing observations and complete sensor failure up to 40% corruption. The main contributions are: (1) an IEEE-style comparative formulation covering seven forecasting models and two public datasets; (2) a robustness protocol for random missing values and sensor-level failure; (3) an accuracy-versus-robustness analysis showing that DCRNN achieves the best clean accuracy while STGCN is often more stable under random missing inputs; (4) a graph ablation discussion indicating that temporal learning contributes most of the gain under the tested configuration; and (5) deployment guidance for selecting models under clean, noisy, and sensor-failure conditions.

Related Work

A. Classical and Machine Learning Traffic Forecasting

Early traffic forecasting studies relied heavily on statistical time-series models, including ARIMA and seasonal ARIMA formulations [13], [19]. These methods are computationally efficient and interpretable, but they assume relatively simple temporal structure and usually operate independently across sensors. Historical Average methods are useful as time-of-day baselines, but they cannot capture incident-driven nonlinear changes. Machine learning models such as Random Forest can model nonlinear relationships [12]; however, flattening the spatio-temporal input tends to remove the ordering and graph structure that are important in road networks.

B. Deep Temporal Models

Recurrent neural networks, particularly Long Short-Term Memory (LSTM) networks, are designed to capture sequential dependencies through gating mechanisms [8]. In traffic forecasting, LSTM can process recent sensor observations as a multivariate time series. However, a standard LSTM does not include an explicit spatial correlation operator, so it must learn spatial relationships implicitly from the full sensor vector. This can become inefficient as the number of sensors grows and as traffic propagation depends on road topology.

C. Spatio-Temporal Graph Neural Networks

DCRNN introduced diffusion convolution to represent traffic flow as a bidirectional diffusion process on a directed graph and combined it with an encoder-decoder recurrent architecture [1]. STGCN proposed a fully convolutional spatio-temporal block that combines Chebyshev graph convolution with gated temporal convolution, allowing efficient parallel training [2], [3]. Later models extended these ideas through adaptive graph learning and attention. Graph WaveNet learns adaptive dependency matrices with dilated temporal convolution [4], AGCRN uses node-adaptive graph structures and recurrent units [5], and ASTGCN applies attention mechanisms to capture spatial and temporal dependencies across multiple scales [6]. These developments show that traffic forecasting has moved from fixed-topology graph models toward adaptive and

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

attention-based spatio-temporal learning [7], [23]. Foundational graph convolution and graph attention studies also motivate the broader family of graph learning operators used in transportation networks [11], [20].

D. Robustness and Missing Data

Real traffic sensor networks commonly experience missing values due to sensor malfunctions, maintenance outages, and communication errors. Related studies have explored graph-based forecasting with missing data, including graph Markov networks and graph convolutional approaches designed for incomplete traffic observations [9], [10]. However, many studies focus mainly on improving prediction accuracy under missing values rather than directly comparing clean-data accuracy and corruption robustness across statistical, deep temporal, and graph-based model families. This gap motivates the robustness-oriented benchmark analyzed in this paper.

Problem Formulation

The traffic network is modeled as a graph G = (V, E, A), where V contains N traffic sensor nodes, E represents spatial relationships between sensors, and $A \in \mathbb{R}^{N\times N}$ is the weighted adjacency matrix describing graph connectivity. At time step t, the observed traffic state is denoted by $X_t \in R^{N\times F}$, where F is the number of measured features. In this study, F = 1 because the forecasting task uses traffic speed measured in miles per hour.

Given a historical observation window of length T, the objective is to learn a parameterized forecasting function f_θ(·) that predicts the next H traffic states:

$$Y^t + 1{:}\, t + H = f\theta(Xt - T + 1{:}\, t, A)$$

where A provides the road-network structure and θ denotes the trainable model parameters. In this benchmark, T = 12 and H = 12. Since each observation corresponds to a 5-minute interval, the models use the previous 60 minutes of traffic speed data to forecast the next 60 minutes.

$$X_{t-T+1:t} = M \odot X_{t-T+1:t}$$

where ⊙ denotes element-wise multiplication. Two degradation settings are evaluated. In the random-missing setting, individual sensor-time observations are removed at random. In the sensor-failure setting, all historical observations from selected sensors are removed. The model receives the corrupted input, but the target future sequence remains clean. This design makes it possible to measure how strongly each forecasting model depends on complete sensor information and how reliably it performs when part of the sensing system fails. The overall data and experimental protocol, including historical input corruption, forecasting, and comparison with ground-truth observations, is illustrated in **Fig. 1**.

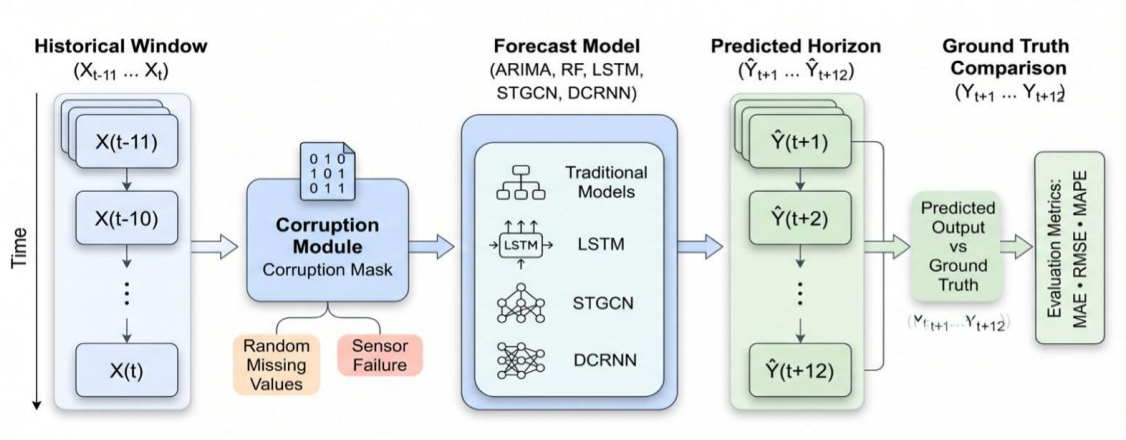

Fig.1. Robustness Evaluation Framework for Traffic Forecasting Under Sensor Failures and Missing Data A. Datasets

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

The benchmark uses two public traffic speed datasets. METR-LA contains 207 loop detectors from Los Angeles highways, while PEMS-BAY contains 325 detectors from the San Francisco Bay Area. Both datasets use 5-minute sampling intervals and are widely used benchmarks for spatio-temporal traffic forecasting [1], [18]. The dataset characteristics are summarized in **Table I**.

Table I: Dataset Statistics

Property	**METR-LA**	**PEMS-BAY**
Location	Los Angeles	Bay Area
Sensors	207	325
Timesteps	34,272	52,116
Duration	Mar.–Jun. 2012	Jan.–Jun. 2017
Interval	5 min	5 min
Feature	Speed (mph)	Speed (mph)
Train/Val/Test	70/10/20	70/10/20
Adjacency size	207×207	325×325
Nonzero entries	447	2,457
Sparsity	98.96%	97.67%
Avg. conn./node	2.2	7.6

B. Preprocessing and Leakage Prevention

Zero-valued traffic readings in freeway datasets often correspond to sensor malfunction or missing signals rather than true roadway speed. Therefore, missing or invalid values are handled using chronological imputation: forward fill, backward fill, and column-mean replacement. This preserves temporal continuity while avoiding unrealistic zero-speed artifacts.

To avoid temporal information leakage, the data are divided chronologically into 70% training, 10% validation, and 20% testing. Unlike random splits, chronological partitioning preserves the natural evolution of traffic conditions and ensures that future observations are not used during model development. Normalization is performed using training statistics only according to the Z-score transformation

$$z = \frac{x - \mu_{train}}{\sigma_{train}}$$

where μtrain and σtrain denote the mean and standard deviation computed exclusively from the training subset. The same statistics are subsequently applied to the validation and test sets, preventing information leakage from future observations.

The normalized data are converted into supervised learning samples using a sliding-window strategy. Each input sequence consists of **12 historical time steps (60 minutes)**, while the corresponding target consists of the subsequent **12 future time steps (60 minutes)**. Sequence generation is performed independently within each data partition so that no sample crosses train-validation or validation-test boundaries.

Fig. 2. illustrates the overall preprocessing and evaluation framework used in this study and includes the following sequential steps: data cleaning, chronological splitting, train-only normalization, sequence creation, graphing, model building, and robustness assessment with no leakage.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

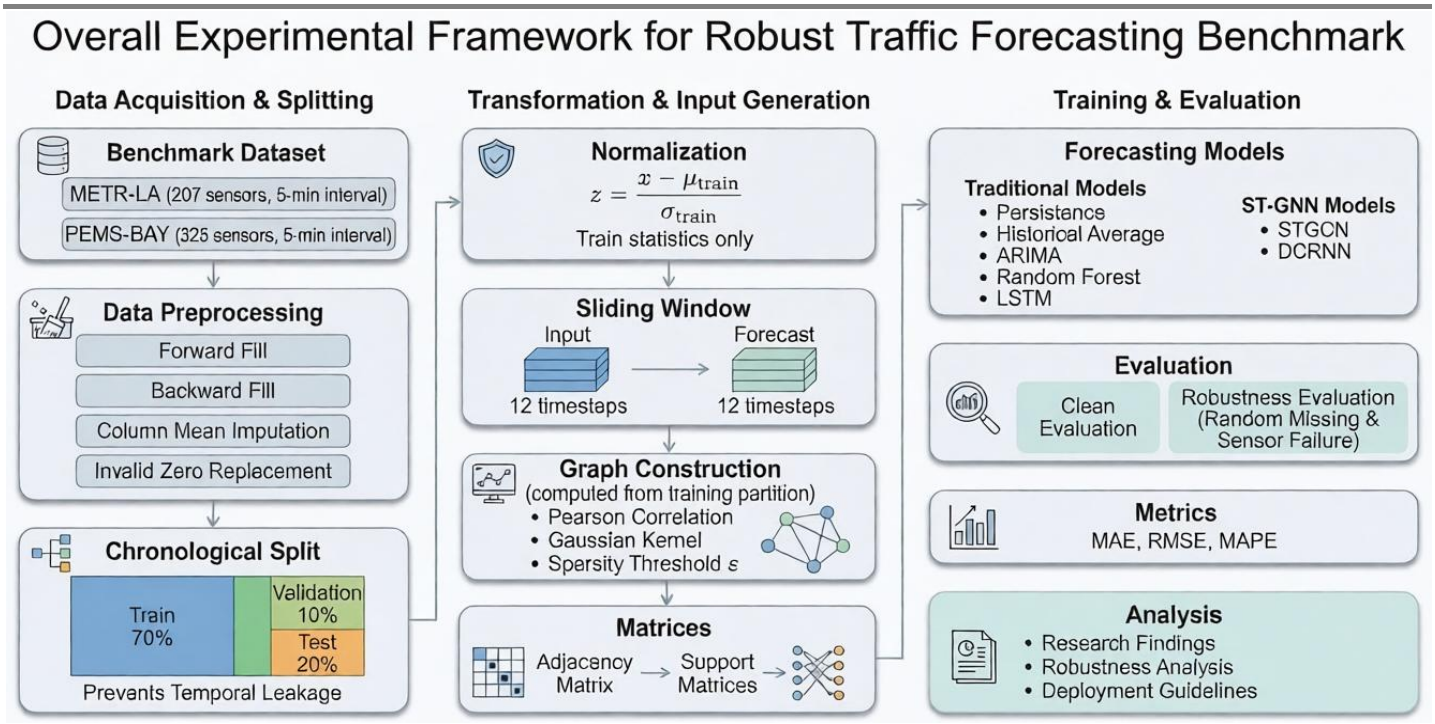

Fig. 2. Overall experimental framework of the proposed robustness benchmark, illustrating data preprocessing, leakage prevention, graph construction, model training, controlled sensor degradation experiments, and performance evaluation.

C. Graph Construction

The traffic network is represented as a weighted graph G = (V, E, A), where each node corresponds to a traffic sensor and A encodes pairwise spatial dependence. To reduce spatial leakage, the adjacency matrix is constructed from the training data rather than from the complete dataset [25].

Pairwise Pearson correlation coefficients $C_{i,j}$ are first computed between all sensor time series. A Gaussian kernel transforms correlations into edge weights:

$$A_{i,j} = exp\left(-\left(1 - C_{i,j}\right)^2 / 2\,\sigma^2\right)$$

where σ = 0.1 controls the kernel bandwidth. Entries below a sparsity threshold ε = 0.3 are set to zero, and self-loops are added ($A_{i,i} = 1$). The resulting adjacency matrix captures only the strongest spatial relationships.

For STGCN, the adjacency matrix is symmetrically normalized as $\hat{A} = D^{-1/2}AD^{-1/2}$ and used to compute Chebyshev polynomial supports $T_0(\tilde{L}), \ldots, T_K(\tilde{L})$ of the scaled graph Laplacian up to order K = 3. [2], [3]. For DCRNN, forward and backward random-walk transition matrices $P_f = D^{-1}A$ and $P_b = D^{-1}A^{\mathrm{T}}$ are computed at diffusion steps K = 2, producing four support matrices that model bidirectional traffic propagation [1]. To examine the sensitivity of model performance to graph topology, ablation experiments using identity and random adjacency matrices are presented in Section VII-D.

Models

Seven forecasting models are evaluated under the same experimental protocol, including statistical, machine learning, deep temporal, and graph neural network approaches. The same chronological split, input length, forecasting horizon, and evaluation metrics are used for all models, and all reported errors are converted back to the original speed unit before comparison. The characteristics of the evaluated model families are summarized in **Table II.**

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Table II: Compared Models

Model	Spatial Learning	Temporal Learning
Persistence	None	Copy last value
Historical Avg.	None	Time-of-day lookup
ARIMA(2,1,2)	None	Per-sensor statistical model
Random Forest	Implicit via flattened input	Independent horizon regressors
LSTM	Implicit	2-layer recurrent memory
STGCN	Chebyshev spectral graph convolution	Gated temporal CNN
DCRNN	Bidirectional diffusion convolution	DCGRU encoder-decoder

A. Baselines

Persistence predicts each future time step using the most recently observed value. Historical Average estimates future traffic speed using the average training-set value corresponding to the same time of day. ARIMA(2,1,2) is independently fitted to each sensor time series. Random Forest employs 100 decision trees with a maximum depth of 15 using flattened sliding-window inputs. The LSTM baseline consists of a two-layer recurrent architecture with a hidden dimension of 64.

B. STGCN

STGCN [2] captures spatio-temporal dependencies through a fully convolutional architecture that avoids recurrent computation. The network contains two stacked Spatio-Temporal Convolution (ST-Conv) blocks followed by an output projection layer. Each ST-Conv block applies a gated temporal convolution, a Chebyshev spectral graph convolution, and a second gated temporal convolution, followed by normalization and dropout.

The gated temporal convolution operates along the time axis using a one-dimensional convolution with kernel size 3. The output is split into two parts, P and Q, along the channel dimension, and a Gated Linear Unit (GLU) activation is applied:

$$h = P \odot \sigma(Q)$$

where σ(·) is the sigmoid function and ⊙ denotes element-wise multiplication. The learned gate σ(Q) controls which temporal features propagate forward, providing a data-dependent filtering mechanism [15].

The graph convolution approximates spectral filtering on the graph Laplacian using Chebyshev polynomials of order K:

$$g_{(}\theta_{)} \star x = \Sigma_{k=0}{}^{K}\theta_{k} \cdot T_{k}(\tilde{L}) \cdot x$$

where $\mathbf{T_k(\tilde{L})}$ is the k-th Chebyshev polynomial of the scaled Laplacian

$$\mathbf{\tilde{L} = (2/\lambda_{max})L - I}$$

where $\boldsymbol{\theta_k}$ are learnable filter coefficients. This formulation approximates localized spectral filtering without explicit eigen decomposition and restricts information propagation to a K-hop neighborhood [3]. For sparse graph implementations the computation scales as O(K|E|); dense implementations scale with the full graph size.

In this benchmark, the channel progression is **[1, 16, 32, 64]** with Chebyshev order **K = 3**. The final output convolution collapses the temporal dimension and produces all 12 forecast horizons simultaneously, enabling

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

fully parallel inference across both the spatial and temporal axes. The STGCN architecture is illustrated in **Fig. 3**.

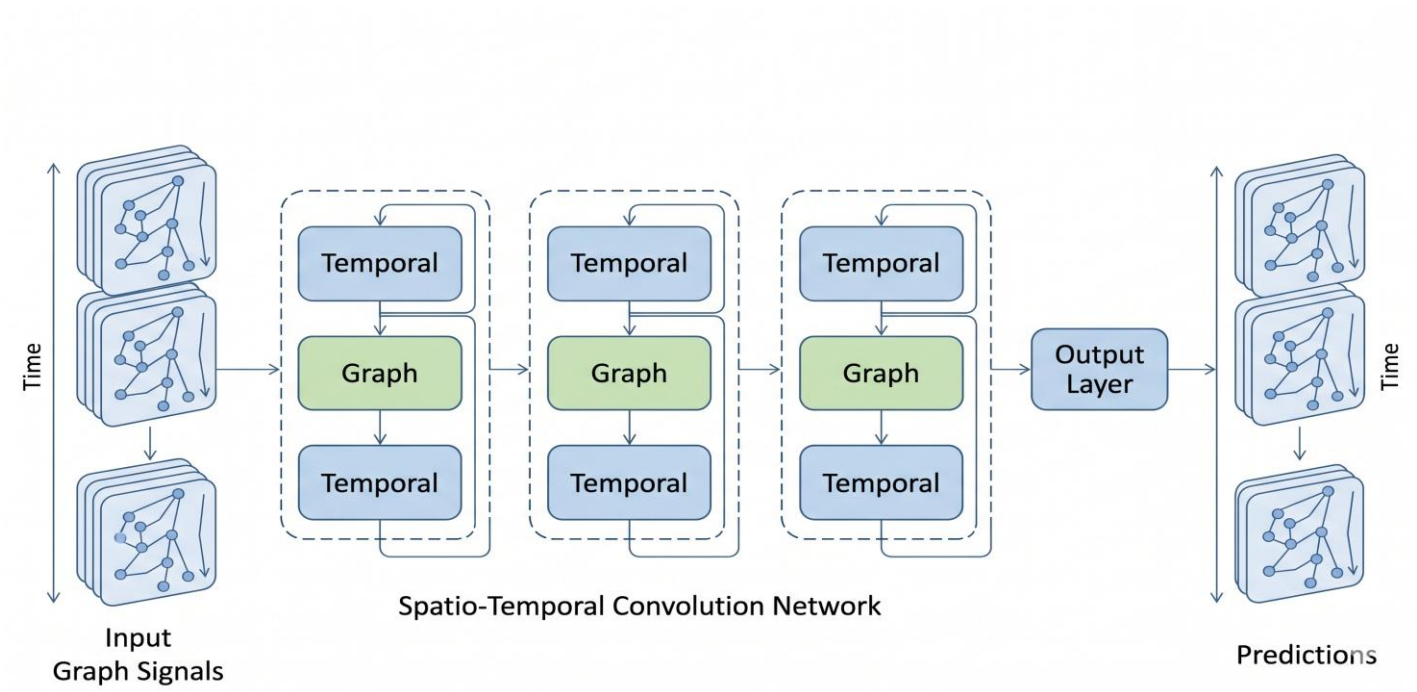

Fig. 3. STGCN architecture.

C. DCRNN

DCRNN [1] models traffic flow as a diffusion process on a directed graph and captures temporal dynamics through sequence-to-sequence architecture. The key architectural component is the Diffusion Convolutional GRU (DCGRU) cell, which replaces the fully connected weight matrices in a standard GRU with diffusion convolutions over the graph.[14].

The diffusion convolution aggregates information from **K-hop** neighborhoods via bidirectional random walks. The forward transition matrix $\mathbf{P_{(f)}} = \mathbf{D_{(out)}^{-1}}\ \mathbf{W}$ models downstream traffic propagation, while the backward transition matrix $\mathbf{P_{(b)}} = \mathbf{D_{(in)}^{-1}}\ \mathbf{W^T}$ captures upstream influences. The diffusion convolution over input **X** is defined as:

$$X \star_{(G)} f_{(\theta)} = \Sigma_{k=0}^{K}(\theta_{k,1} \cdot P_{(f)}^{k} + \theta_{k,2} \cdot P_{(b)}^{k})X$$

where $\mathbf{P_{(f)}^k}$ and $\mathbf{P_{(b)}^k}$ are the k-th powers of the respective transition matrices, $\boldsymbol{\theta_{k,1}}$ and $\boldsymbol{\theta_{k,2}}$ are learnable parameters, and $\mathbf{P^0 = I}$ represent the identity (self-connection).

The DCGRU cell applies this diffusion convolution within the standard GRU gating mechanism:

$$r = \sigma\big(DC([X_t; H_{t-1}])\big)$$

$$u = \sigma\big(DC([X_t; H_{t-1}])\big)$$

$$C = tanh\big(DC([X_t; r \odot H_{t-1}])\big)$$

$$H_t = u \odot H_{t-1} + (1-u) \odot C$$

where **DC(·)** denotes the diffusion convolution, **r** is the reset gate, **u** is the update gate, and **C** is the candidate hidden state.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Through a two-layer DCGRU stack, the encoder processes the complete 12-step input sequence and summarizes the past traffic state. The decoder generates future predictions sequentially, using teacher forcing during training to stabilize learning while gradually encouraging autoregressive forecasting [25].

In this benchmark, the hidden dimension is **64**, the number of DCGRU layers is **2**, and **K = 2** diffusion steps are used, producing four support matrices ($\mathbf{P_{(f)}^1}$, $\mathbf{P_{(b)}^1}$, $\mathbf{P_{(f)}^2}$, $\mathbf{P_{(b)}^2}$). The DCRNN encoder-decoder architecture is shown on **Fig. 4** [1].

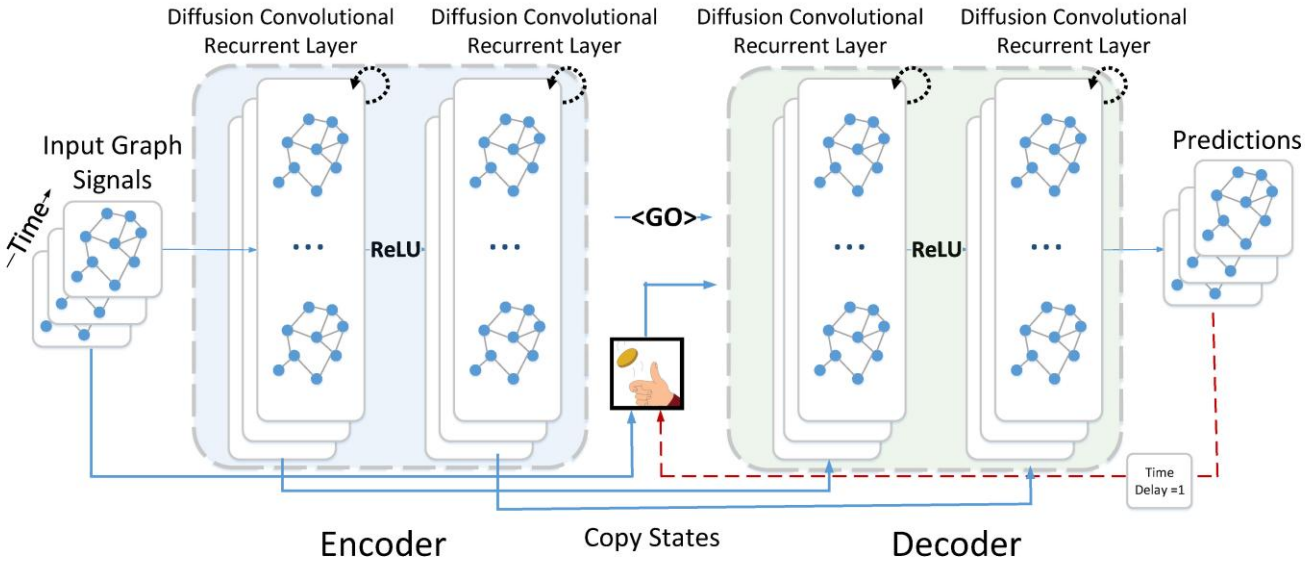

Fig. 4. DCRNN encoder-decoder architecture [1].

Evaluation Metrics And Robustness Protocol

A. Metrics

All models are evaluated on de-normalized predictions so that reported errors remain in real-world traffic speed units (miles per hour). Three metrics are used:

$$MAE = \frac{1}{Q}\sum_{q=1}^{Q}|\hat{y}_q - y_q|$$

$$RMSE = \sqrt{\frac{1}{Q}\sum_{q=1}^{Q}(\hat{y}_q - y_q)^2}$$

$$MAPE = \frac{100}{Q}\sum_{q=1}^{Q}\left|\frac{\hat{y}_q - y_q}{y_q}\right|$$

where Q is the total number of predictions, $\hat{\mathbf{y}}_{(\mathbf{q})}$ is the predicted value, and $\mathbf{y}_{(\mathbf{q})}$ is the ground-truth value in the original scale. MAE is the primary metric because it gives an interpretable average error magnitude. RMSE emphasizes large deviations and identifies occasional extreme errors, while MAPE provides a percentage error measure but can be sensitive to near-zero ground-truth values.

Performance is reported at three forecast horizons corresponding to prediction steps **3**, **6**, and **12**, which represent **15-minute**, **30-minute**, and **60-minute** forecasts, respectively. This multi-horizon evaluation captures the expected degradation in prediction quality as the forecast window increases.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

B. Corruption Scenarios

To evaluate model robustness under realistic sensor degradation, two corruption scenarios are applied to the test-set input sequences using the binary mask formulation defined in Section III.

Random missing. Each input entry $\mathbf{M}(t,n)$ is independently sampled from a Bernoulli distribution with probability $\mathbf{1 - p}$, producing spatially and temporally scattered data gaps. This scenario simulates intermittent communication failures or transient sensor malfunctions that affect individual readings independently.

Sensor failure. $\lfloor p \times N \rfloor$ sensors are selected uniformly at random, and all their historical observations are set to zero:

$$\mathbf{M}(t,n) = 0, \text{ for all } t, \text{ for each selected sensor } n.$$

This produces spatially correlated degradation that simulates complete hardware failures, power outages, or prolonged maintenance events affecting entire monitoring stations.

Both scenarios are evaluated at corruption ratios

$$p \in \{0.0, 0.1, 0.2, 0.3, 0.4\}.$$

For each non-zero corruption ratio, five independent random seeds are used to generate different corruption masks, and results are reported as **mean ± standard deviation**. The target output **Y** remains uncorrupted in all cases, ensuring that the evaluation measures the model's ability to produce accurate forecasts from degraded inputs. Model degradation is quantified as the relative increase in MAE compared to the clean baseline:

$$\text{Degradation}(p) = [(\text{MAE}(p) - \text{MAE}(0)) / \text{MAE}(0)] \times 100\%$$

where $\text{MAE}(0)$ is the error under clean conditions and $\text{MAE}(p)$ is the error at corruption ratio p. This metric enables direct comparison of robustness across models with different baseline accuracy levels.

Together, these two corruption scenarios evaluate robustness against both diffuse random information loss and structured spatial sensor failures, enabling a comprehensive assessment of forecasting reliability under realistic deployment conditions.

RESULTS AND DISCUSSION

A. Clean-Data Accuracy

The clean-data forecasting performance of all seven models across both datasets is summarized in **Table III**. DCRNN achieves the lowest overall MAE on both datasets: 3.548 mph on METR-LA and 1.905 mph on PEMS-BAY. This advantage may be attributable to DCRNN's bidirectional diffusion convolution, which models asymmetric upstream-downstream traffic propagation, combined with the autoregressive decoder that conditions each prediction step on prior outputs.

STGCN achieves the second-best overall MAE on METR-LA (3.634 mph), but it ranks below LSTM and Random Forest on PEMS-BAY (2.299 vs. 2.071 and 2.113 mph, respectively). This dataset-dependent behavior may be related to graph density: PEMS-BAY has 7.6 average connections per node compared with 2.2 for METR-LA. In the denser PEMS-BAY graph, the K = 3 Chebyshev convolution aggregates information from a larger effective neighborhood, which may lead to excessive smoothing of local traffic patterns. This interpretation should be treated as a hypothesis and would require additional experiments varying in K across graph densities.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

A notable crossover emerges at longer forecast horizons. At 60 minutes, STGCN achieved the best MAE on METR-LA (4.242 vs. DCRNN's 4.541), a 6.6% advantage. One possible explanation is that STGCN generates all 12 horizon predictions simultaneously, whereas DCRNN's autoregressive decoder generates predictions sequentially. On longer horizons, small prediction errors in early decoder steps may accumulate through this sequential chain, producing compounding degradation that parallel architectures avoid. The horizon-wise MAE trends for METR-LA and PEMS-BAY are illustrated in **Figs. 7 and 8**, respectively.

The RMSE comparison provides complementary insight. STGCN achieves the lowest RMSE on METR-LA (6.291 vs. DCRNN's 6.672), indicating that while DCRNN produces lower average errors, STGCN produces fewer extreme outlier predictions. This property has practical significance for traffic management systems where catastrophic mispredictions—even if rare—can trigger inappropriate control actions. The multi-metric forecasting performance of the evaluated models on METR-LA is illustrated in **Fig. 5**, while the corresponding results for PEMS-BAY are presented in **Fig. 6**.

LSTM outperforms all classical baselines on both datasets (MAE 3.707 on METR-LA, 2.071 on PEMS-BAY), confirming that learned temporal representations capture nonlinear traffic dynamics that statistical models cannot. The performance gap between LSTM and Random Forest is modest (1.3% on METR-LA), suggesting that temporal ordering provides limited additional benefit over feature-engineered input windows for short-term prediction.

Table III: Clean-Data Forecasting Performance (Lower Is Better)

Dataset	**Model**	**Type**	**15 min MAE**	**30 min MAE**	**60 min MAE**	**Overall MAE**	**RMSE**	**MAPE (%)**
METR-LA	DCRNN	GNN	2.868	3.540	4.541	3.548	6.672	9.95
METR-LA	STGCN	GNN	3.214	3.605	4.242	3.634	6.291	9.75
METR-LA	LSTM	DL	2.988	3.685	4.759	3.707	7.027	10.66
METR-LA	Random Forest	ML	3.049	3.753	4.798	3.758	7.142	10.80
METR-LA	Persistence	Base	3.161	3.831	4.905	3.856	7.806	9.94
METR-LA	ARIMA(2,1,2)	Stat.	3.324	3.947	5.009	3.990	8.126	10.38
METR-LA	Historical Avg.	Base	5.120	5.120	5.120	5.120	8.777	15.93
PEMS-BAY	DCRNN	GNN	1.472	1.942	2.519	1.905	4.026	4.37
PEMS-BAY	LSTM	DL	1.523	2.083	2.864	2.071	4.500	4.90
PEMS-BAY	Random Forest	ML	1.553	2.135	2.920	2.113	4.665	5.01
PEMS-BAY	Persistence	Base	1.591	2.171	3.039	2.170	5.123	4.67
PEMS-BAY	ARIMA(2,1,2)	Stat.	1.747	2.292	3.119	2.294	5.489	5.00
PEMS-BAY	STGCN	GNN	2.077	2.302	2.604	2.299	4.232	5.20
PEMS-BAY	Historical Avg.	Base	3.559	3.559	3.559	3.559	6.827	8.76

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

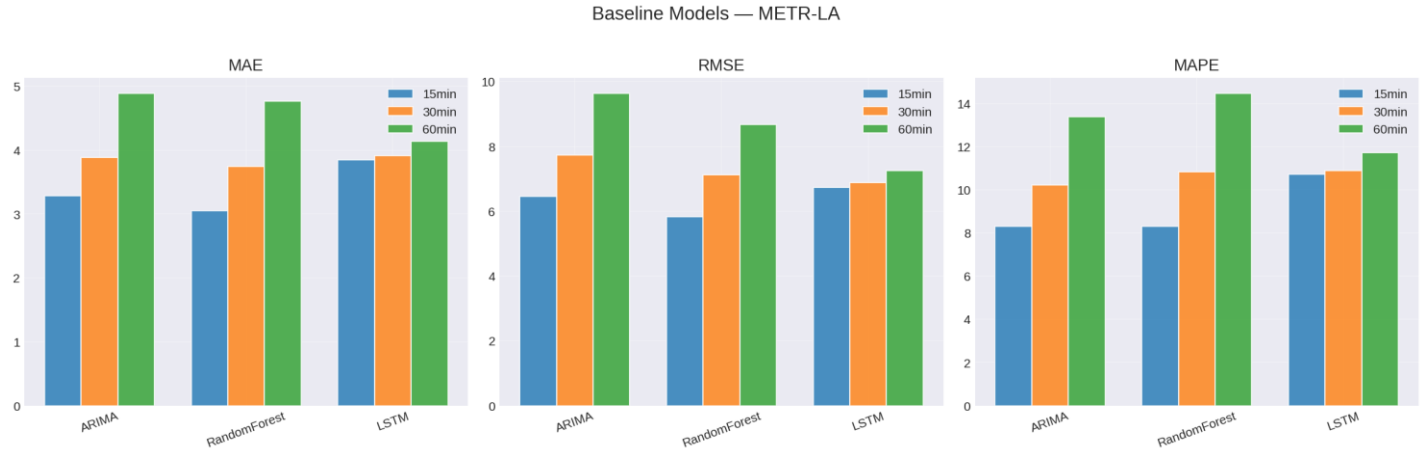

Fig. 5. Multi-metric performance comparison on METR-LA (MAE, RMSE, MAPE). DCRNN achieves the lowest MAE, but STGCN achieves the lowest RMSE, indicating fewer extreme prediction errors.

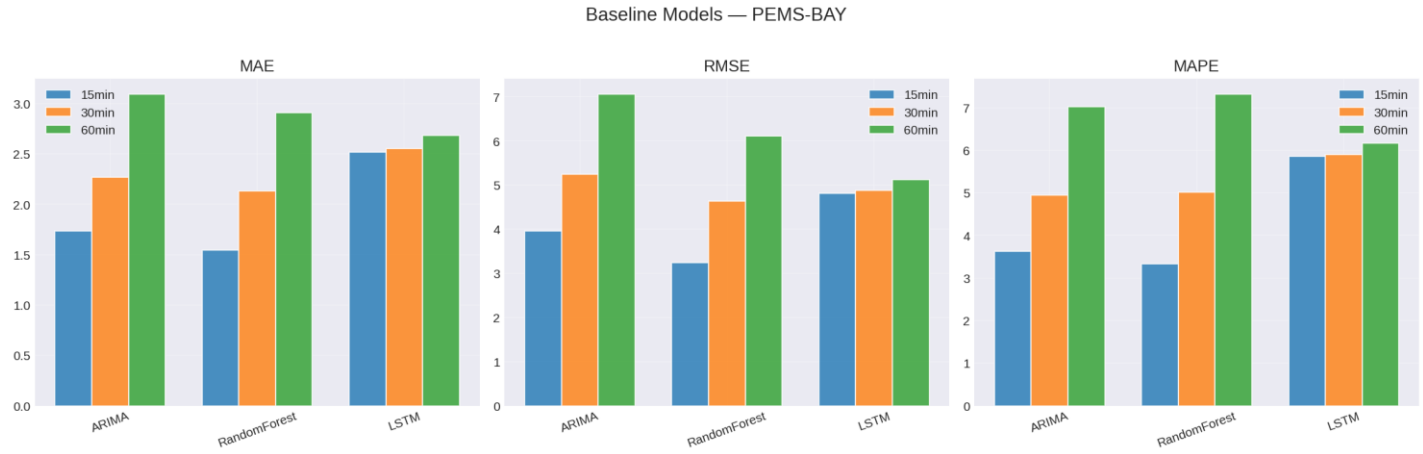

Fig. 6. Multi-metric performance comparison on PEMS-BAY (MAE, RMSE, MAPE). DCRNN leads across all three metrics. STGCN ranks below LSTM and Random Forest on MAE, suggesting graph density may interact with spectral convolution

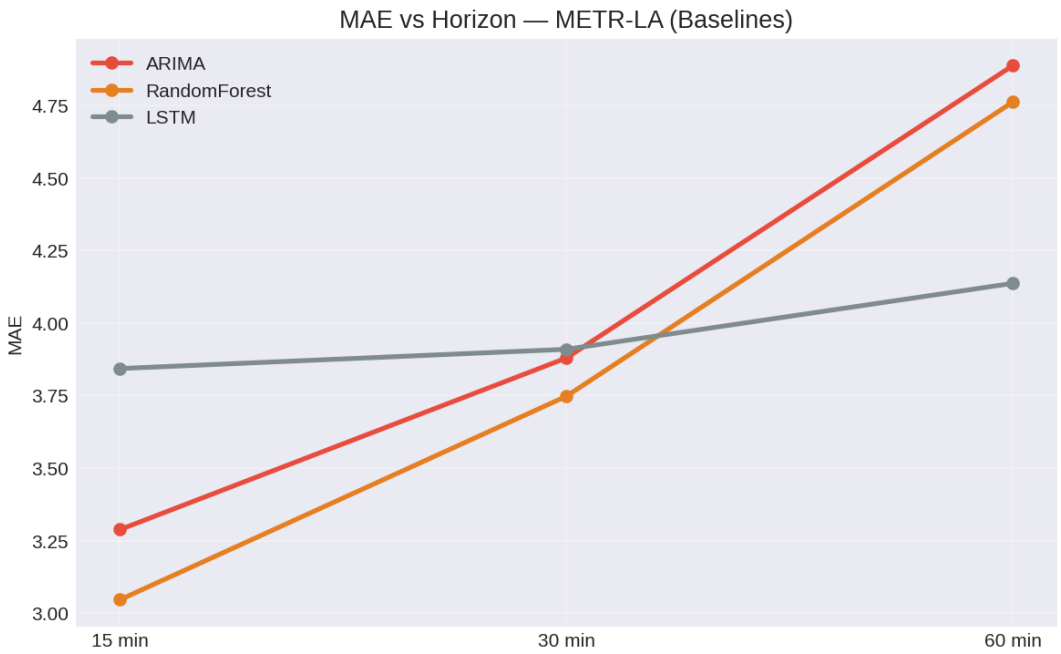

Fig. 7. Per-horizon MAE on METR-LA. STGCN surpasses DCRNN at the 60-minute horizon due to the absence of autoregressive error compounding.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

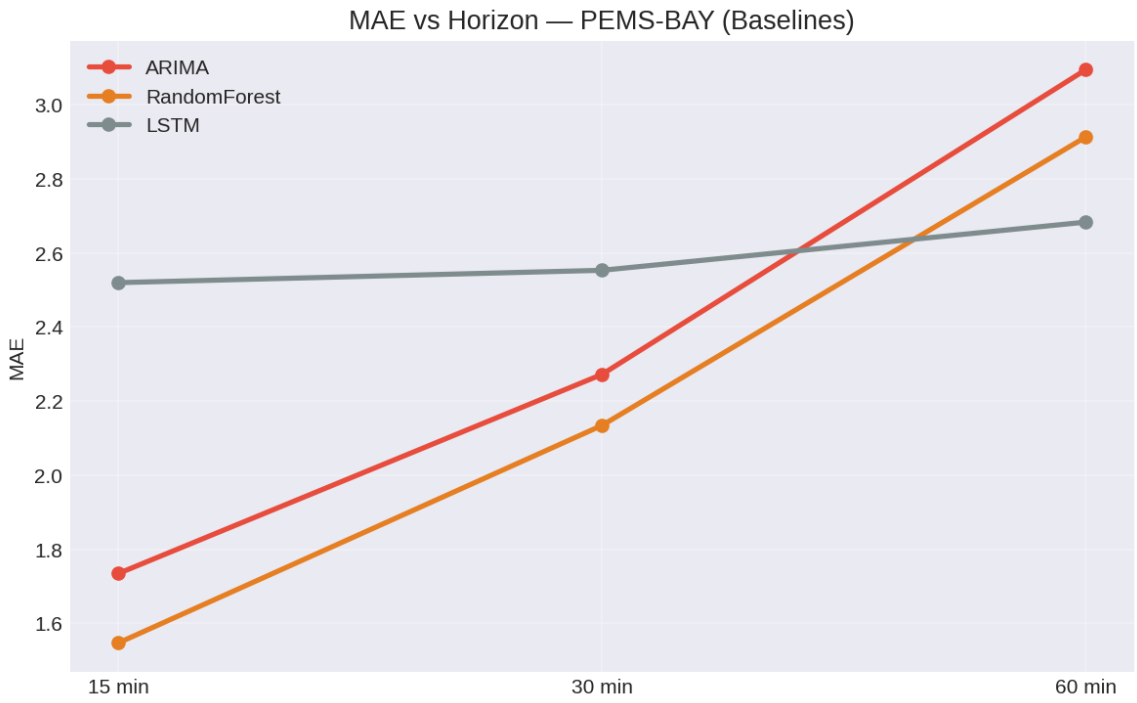

Fig. 8. Per-horizon MAE on PEMS-BAY. DCRNN maintains dominance across all horizons on this dataset.

B. Random Missing Data Robustness

Table IV presents the multi-metric robustness results under random missing corruption, where each input entry is independently zeroed with probability p. All corrupted results are reported as mean ± standard deviation across five independent corruption seeds.

The central finding is that the accuracy ranking and the robustness ranking are inversely related. DCRNN, which achieves the best clean accuracy, exhibits the highest MAE degradation on both datasets: +34.5% on METR-LA and +70.3% on PEMS-BAY at p=0.4. Conversely, STGCN degrades by only +27.0% and +39.1% respectively, making it the most robust GNN under this corruption type.

The RMSE degradation data provides independent corroboration of this pattern. On PEMS-BAY, DCRNN's RMSE increases by +56.4% (4.026→6.295) compared to STGCN's +36.8% (4.232→5.789). The disproportionate RMSE growth relative to MAE growth for DCRNN (+56.4% RMSE vs. +70.3% MAE on PEMS-BAY) indicates that corruption does not merely increase average errors but also generates occasional extreme mispredictions. This is consistent with the autoregressive decoder hypothesis: when the encoder produces corrupted hidden states, the decoder's sequential prediction chain can amplify small initial errors into catastrophic long-horizon predictions.

STGCN's relative robustness may be explained by two architectural properties. First, the Chebyshev spectral graph convolution acts as a spatial low-pass filter: when individual input entries are randomly zeroed, the convolution aggregates values from the K-hop neighborhood, effectively performing spatial smoothing over corrupted entries. Second, the gated temporal convolution produces all horizon predictions in parallel, avoiding sequential error propagation.

MAPE degradation reveals a scale-dependent dimension. On PEMS-BAY, DCRNN's MAPE increases by +92.5% at 40% corruption (4.37→8.42%), the highest among all models. Since MAPE normalizes by ground-truth magnitude, this indicates that DCRNN's errors are disproportionately concentrated on lower-speed observations, which is consistent with corrupted zero-valued inputs biasing the diffusion convolution toward underestimation.

The standard deviations under random missing are remarkably small (σ_MAE $\leq$ 0.008 on both datasets), indicating that this corruption type produces consistent degradation largely independent of which specific

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

entries are removed. The degradation trends under random missing corruption and structured sensor failure on METR-LA are compared in **Fig. 9**.

Table IV: Multi-Metric Robustness Under Random Missing (40% Corruption)

Dataset	Model	Clean MAE	40% MAE	MAE Δ	Clean RMSE	40% RMSE	RMSE Δ	40% MAPE	MAPE Δ
METR-LA	STGCN	3.634	4.615±0.004	+27.0%	6.291	7.838	+24.6%	14.41%	+47.9%
METR-LA	Persistence	3.856	4.970±0.003	+28.9%	7.806	9.314	+19.3%	14.73%	+48.1%
METR-LA	RF	3.758	4.898±0.002	+30.3%	7.142	8.521	+19.3%	15.14%	+40.1%
METR-LA	LSTM	3.707	4.885±0.001	+31.8%	7.027	8.313	+18.3%	15.13%	+41.9%
METR-LA	DCRNN	3.548	4.772±0.004	+34.5%	6.672	8.140	+22.0%	14.83%	+49.0%
PEMS-BAY	STGCN	2.299	3.198±0.002	+39.1%	4.232	5.789	+36.8%	8.03%	+54.6%
PEMS-BAY	LSTM	2.071	3.351±0.001	+61.8%	4.500	6.624	+47.2%	8.80%	+79.3%
PEMS-BAY	DCRNN	1.905	3.244±0.001	+70.3%	4.026	6.295	+56.4%	8.42%	+92.5%

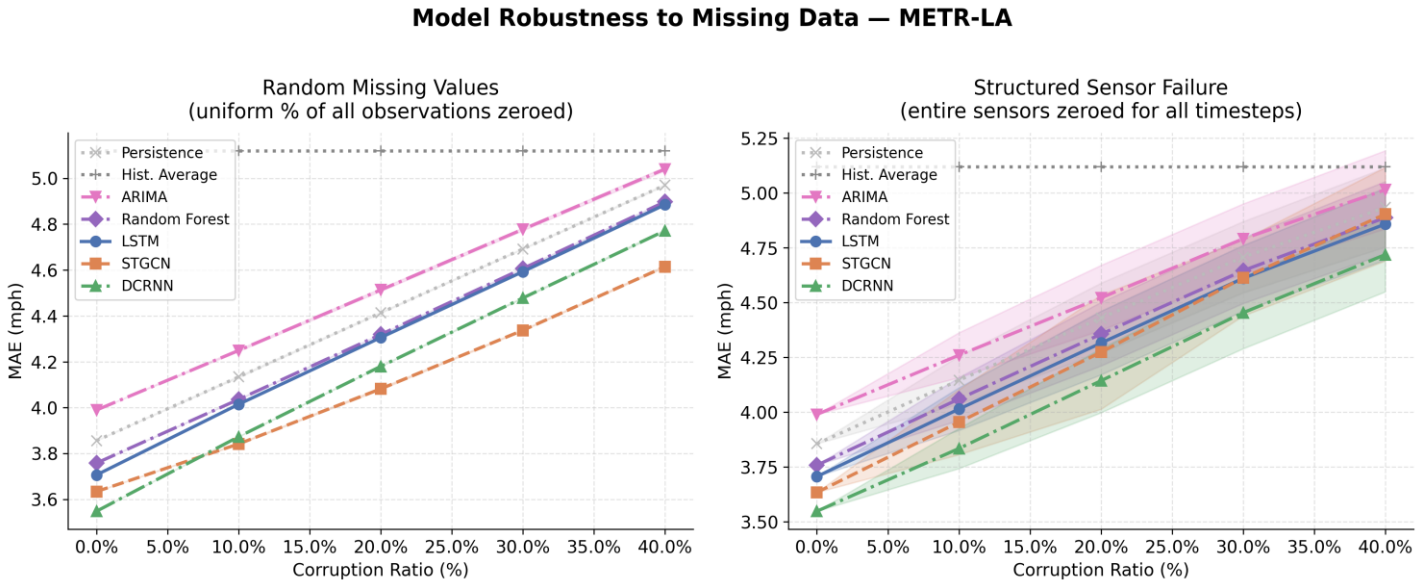

Fig. 9. Robustness degradation curves on METR-LA with ±1σ bands. STGCN maintains a consistently shallower slope than DCRNN across all corruption ratios under random missing. The sensor failure panel shows substantially wider variance bands.

C. Sensor Failure Robustness

Sensor failure represents a fundamentally different corruption structure: rather than scattered missing entries, entire sensors lose all historical readings, creating spatially correlated degradation. The robustness ranking changes substantially under this scenario, revealing a critical interaction between graph architecture and corruption geometry.

On METR-LA at 40% sensor failure, the ranking reverses. LSTM degrades by +31.1% (MAE) and +26.1% (RMSE), outperforming both GNNs: DCRNN degrades by +33.0%/+25.2% and STGCN by +34.9%/+28.5%. STGCN, the most robust model under random missing, becomes the least robust under sensor failure—a rank change of four positions.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

This reversal may be explained by the interaction between graph message passing and spatially correlated corruption. When a sensor fails, its value becomes zero. The graph convolution then aggregates this corrupted zero into the representations of neighboring nodes. Under random missing, corrupted values are spatially scattered, and the convolution effectively performs spatial averaging. Under sensor failure, corruption concentrates in local graph neighborhoods and may be amplified through message passing.

STGCN's higher standard deviation under sensor failure provides additional evidence: σ=±0.213 at 40% on METR-LA, compared to σ=±0.004 under random missing—a 53× increase. This indicates that STGCN's performance is highly sensitive to which specific sensors fail. When failed sensors happen to be peripheral nodes with few connections, the impact is contained. When they occupy central positions in the graph topology, the corruption propagates through the spectral convolution to a large portion of the network.

On PEMS-BAY, the pattern partially differs. STGCN maintains comparative robustness under sensor failure (48.3% degradation), outperforming all other models except potentially within the overlap of its wider confidence interval (σ=±0.210). The denser PEMS-BAY graph (7.6 connections/node) may provide sufficient redundant spatial paths to route information around failed sensors, a hypothesis consistent with graph-theoretic notions of vertex connectivity. However, the high standard deviation (σ=±0.364 for RMSE) indicates that this robustness is unreliable and strongly conditioned on failure location. The multi-metric robustness results under 40% sensor-failure corruption are summarized in **Table V**. And the reversal in model robustness rankings between random missing corruption and sensor-failure corruption on METR-LA is summarized in **Table VI**.

Table V: Multi-Metric Robustness Under Sensor Failure (40% Corruption)

Dataset	**Model**	**40% MAE**	**MAE Δ**	**40% RMSE**	**RMSE Δ**	**40% MAPE**	**σ_MAE**
METR-LA	ARIMA	5.014±0.178	+25.7%	9.409±0.240	+15.8%	14.83%	0.178
METR-LA	Persistence	4.934±0.181	+28.0%	9.247±0.245	+18.5%	14.59%	0.181
METR-LA	RF	4.888±0.165	+30.0%	8.912±0.263	+24.8%	15.12%	0.165
METR-LA	LSTM	4.858±0.164	+31.1%	8.858±0.268	+26.1%	15.05%	0.164
METR-LA	DCRNN	4.718±0.169	+33.0%	8.354±0.279	+25.2%	14.53%	0.169
METR-LA	STGCN	4.903±0.213	+34.9%	8.085±0.264	+28.5%	14.42%	0.213
PEMS-BAY	STGCN	3.409±0.210	+48.3%	5.984±0.364	+41.4%	8.19%	0.210
PEMS-BAY	DCRNN	3.189±0.041	+67.4%	6.221±0.114	+54.5%	7.83%	0.041
PEMS-BAY	LSTM	3.398±0.040	+64.1%	6.727±0.115	+49.5%	8.35%	0.040

Table VI: Corruption-Type Reversal Summary (METR-LA at 40%)

Model	**Random Missing Rank**	**Sensor Failure Rank**	**Rank Δ**	**Mechanism**
STGCN	2nd (27.0%)	6th (34.9%)	↓4	Spectral smoothing helps diffuse corruption, amplifies concentrated corruption
LSTM	5th (31.8%)	4th (31.1%)	↑1	No spatial connectivity prevents corruption propagtion
DCRNN	6th (34.5%)	5th (33.0%)	↑1	Diffusion propagates corruption but less than spectral convolution

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

D. Graph Structure Ablation

To investigate the contribution of learned graph topology to model performance, DCRNN and STGCN are evaluated with three adjacency configurations: the learned correlation-based graph, an identity graph (no spatial connections, I_N), and a random symmetric graph with matched density.

The learned graph provides a modest improvement of 1.92% over the identity graph for DCRNN (MAE 3.548 vs. 3.616) and 0.95% for STGCN (3.634 vs. 3.669). The identity-graph DCRNN shows a more pronounced RMSE increase (7.012 vs. 6.672, +5.1%), suggesting that spatial information contributes more to reducing extreme prediction errors than to reducing average error magnitude.

Two competing interpretations exist for the modest ablation impact, and the current experiments cannot distinguish between them. Under the temporal dominance hypothesis, the recurrent (DCGRU) and convolutional (gated CNN) temporal modules capture most of the predictive signal, rendering spatial information supplementary. Under the weak graph hypothesis, the correlation-based graph at ε=0.3 is too sparse (2.2 connections/node, 48.3% isolated nodes) to provide substantial spatial information in the first place. The sparsity pattern of the learned METR-LA adjacency matrix at $\varepsilon = 0.3$ is visualized in **Fig. 10.** A definitive test would require evaluating performance across multiple graph construction methods and sparsity levels with adequate statistical power.

The random-graph variant achieves performance comparable to the learned graph for both models (DCRNN: 3.532 vs. 3.548; STGCN: 3.630 vs. 3.634). This observation is based on a single training run per configuration; we explicitly note that drawing conclusions about the relative value of learned versus random topology would require multiple training seeds and statistical significance testing. What can be stated conservatively is that, within the experimental configuration tested, graph topology choice did not produce differences exceeding 2%. The graph-structure ablation results for DCRNN and STGCN on METR-LA are summarized in **Table VII**, while the effect of the adjacency-threshold parameter on graph sparsity and connectivity is presented in **Table VIII.** The corresponding influence of the sparsity threshold on graph density, connectivity, node-degree distribution, spectral properties, and isolated-node percentage is illustrated in **Fig. 11.**

Table VII: Graph Structure Ablation on METR-LA

Model	**Graph**	**Overall MAE**	**RMSE**	**MAPE (%)**	**Δ MAE vs. Learned**
DCRNN	Learned	3.548	6.672	9.95	—
DCRNN	Identity	3.616	7.012	10.25	+1.92%
DCRNN	Random	3.532	6.584	9.92	−0.45%
STGCN	Learned	3.634	6.291	9.75	—
STGCN	Identity	3.669	—	—	+0.95%
STGCN	Random	3.630	—	—	−0.11%

Table VIII: Graph Sparsity Analysis (METR-LA)

ε Threshold	**Edges**	**Avg. Conn./Node**	**Isolated Nodes (%)**	**Spectral Gap**
0.1	424	0.83	33.8%	0.0252
0.2	308	0.75	39.6%	0.0111
0.3 (used)	240	0.67	48.3%	0.0034
0.5	148	0.50	59.9%	0.0069

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

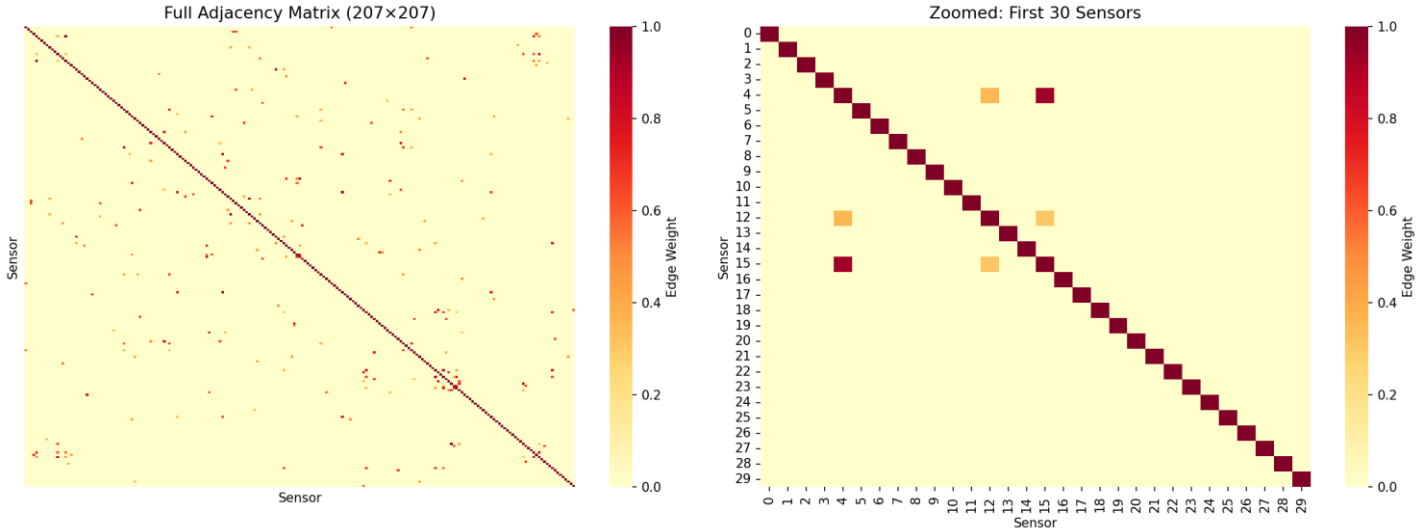

Fig. 10. Learned adjacency matrix heatmap (METR-LA, ε=0.3). The extreme sparsity is visible—most entries are zero. Nearly half the nodes have no spatial connections.

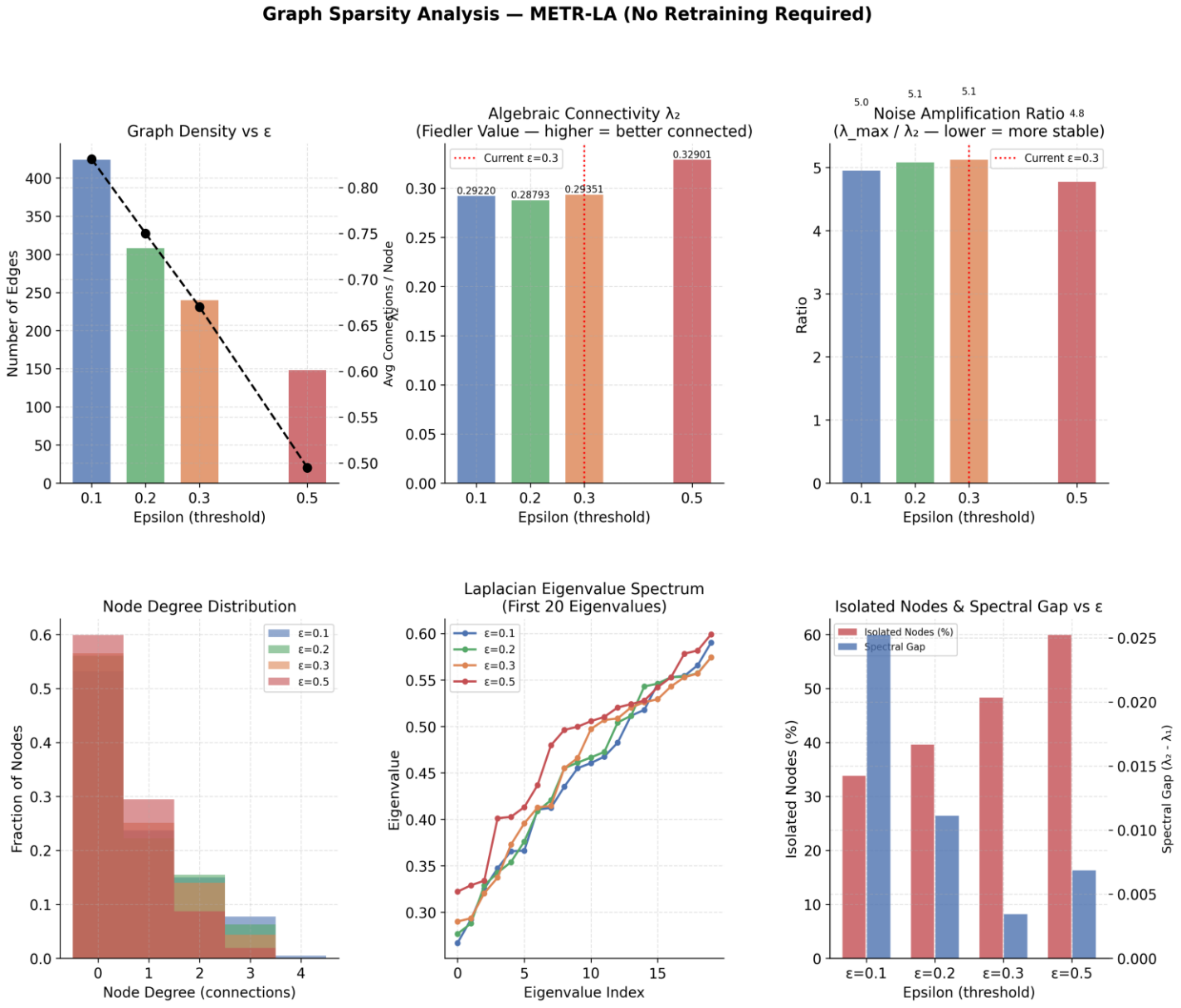

Fig. 11. Effect of sparsity threshold ε on graph properties and model performance. Higher ε removes edges monotonically while performance remains relatively stable, consistent with the marginal ablation impact.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

DISCUSSION

The experimental results reveal a fundamental tension between clean-data accuracy and robustness to input corruption that has direct implications for deploying graph neural networks in intelligent transportation systems.

Accuracy-robustness trade-off: DCRNN achieves the highest clean accuracy yet exhibits the highest sensitivity to input corruption. The autoregressive decoder that may provide DCRNN's clean-data advantage becomes a potential liability under corruption, as degraded encoder states propagate through the sequential decoding chain. The RMSE data is consistent with this interpretation: DCRNN's disproportionate RMSE growth under corruption suggests more frequent extreme mispredictions, not merely higher average errors.

Corruption geometry matters: The reversal in robustness ranking between random missing and sensor failure demonstrates that robustness is not a single-dimensional property. This suggests that robustness evaluations using only one corruption type may produce misleading conclusions about deployment readiness. Practical deployment implications. In real-world traffic networks, both types of corruption occur. A conservative deployment strategy would use DCRNN for clean or lightly corrupted inputs while maintaining an LSTM fallback for regions experiencing equipment failures. Corruption-aware training strategies such as input dropout could be investigated to improve GNN robustness without sacrificing clean accuracy.

Graph topology value: Under the configuration tested (ε=0.3), learned graph topology provides marginal benefit over identity and random graphs. Future work should investigate denser graph construction methods, alternative proximity measures, or adaptive graph learning.

Key Findings

Finding 1: Clean accuracy and robustness are inversely related.

DCRNN achieves the best clean MAE (3.548 METR-LA, 1.905 PEMS-BAY) but the highest degradation under 40% random missing (+34.5%, +70.3%). Clean benchmark accuracy alone is insufficient for deployment-oriented model selection.

Finding 2: STGCN is the most robust GNN under random missing corruption.

STGCN degrades by only 27.0% (METR-LA) and 39.1% (PEMS-BAY), the lowest among all learned models across all three metrics (MAE, RMSE, MAPE).

Finding 3: The robustness ranking reverses under sensor failure.

STGCN drops from rank 2 to rank 6 on METR-LA; LSTM rises from rank 5 to rank 4. The 53× increase in standard deviation (0.004→0.213) indicates strong dependence on failure location, suggesting graph message passing may propagate spatially correlated failures.

Finding 4: RMSE and MAPE provide independent corroboration. DCRNN's RMSE (+56.4%) and MAPE (+92.5%) degradation on PEMS-BAY substantially exceed its MAE degradation, indicating more frequent extreme mispredictions concentrated during lower-speed conditions.

Finding 5: Learned graph topology provides marginal benefit.

The correlation-based graph improves DCRNN by 1.92% and STGCN by 0.95% over identity graphs. These single-run observations cannot establish statistical significance.

Overall, the experiments demonstrate that evaluating traffic forecasting models solely on clean benchmark datasets may overestimate their suitability for deployment, emphasizing the importance of robustness-aware evaluation under realistic sensor degradation scenarios.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Limitations And Future Work

Although the proposed benchmark provides a comprehensive evaluation of robustness under realistic sensor degradation, several limitations should be acknowledged. First, the experiments are conducted on two widely used public datasets, METR-LA and PEMS-BAY, and therefore the conclusions may not directly generalize to other traffic networks with different spatial characteristics or sensing infrastructures. Second, the robustness evaluation considers two corruption mechanisms—random missing values and complete sensor failures—but real-world deployments may also experience burst errors, communication delays, calibration drift, biased measurements, or dynamically changing road topologies. Third, the graph ablation study is based on single training runs for each graph configuration; consequently, while the observed differences are informative, statistical significance cannot be established without additional multi-seed experiments. Fourth, all models were trained exclusively on clean data, meaning the observed degradation reflects worst-case robustness without any corruption-aware adaptation.

Future research can extend this benchmark in several directions. Modern architectures such as Graph WaveNet, AGCRN, ASTGCN variants, GMAN, graph-attention models, and transformer-based spatio-temporal models should be evaluated under the same corruption protocol to determine whether recent advances improve robustness as well as clean-data accuracy [4]-[6], [20], [21]. In addition, adaptive graph learning, uncertainty-aware forecasting, corruption-aware data augmentation, masked reconstruction objectives, graph convolutional baselines, adversarial robustness analysis, and joint imputation-forecasting frameworks represent promising approaches for increasing resilience to degraded sensor inputs [11], [22]. Finally, evaluating robustness under more realistic operational scenarios, including evolving graph topology caused by road closures or network disruptions, would further improve the practical relevance of robustness-oriented traffic forecasting benchmarks [24].

CONCLUSION

This paper presented a robustness-oriented benchmark of seven traffic forecasting models on the METR-LA and PEMS-BAY datasets, comparing traditional statistical methods, deep learning models, and spatio-temporal graph neural networks under both clean and degraded sensing conditions. While DCRNN achieved the highest forecasting accuracy on clean data, the experiments demonstrated that clean-data accuracy alone does not guarantee robustness under realistic sensor degradation. STGCN consistently exhibited greater resilience to randomly missing inputs, whereas sensor failure altered the robustness ranking and, on METR-LA, allowed the non-graph LSTM model to outperform both graph-based models. These findings suggest that graph-based message passing, while beneficial for exploiting spatial dependencies, may also propagate corrupted information when sensor failures occur.

The graph ablation study further indicated that, under the evaluated configuration, learned graph topology contributed only marginal improvements over simpler alternatives, suggesting that temporal learning played a dominant role in the observed predictive performance. Although these observations require additional multi-seed validation to establish statistical significance, they highlight the importance of carefully evaluating the contribution of graph construction in spatio-temporal forecasting systems. Overall, the results demonstrate that robustness should be considered alongside predictive accuracy when selecting forecasting models for Intelligent Transportation Systems. Evaluating models solely on clean benchmark datasets may overestimate their suitability for real-world deployment, where missing data, sensor failures, and communication disruptions are unavoidable. Future traffic forecasting research should therefore emphasize robustness-aware evaluation and corruption-resilient model design in addition to achieving state-of-the-art accuracy.

ACKNOWLEDGMENT

We want to thank the people who made the METR-LA and PEMS-BAY datasets. We also want to thank the people who made the public benchmark repository and the open-source research community for giving us datasets and other things that helped us with this study.

Funding

The authors declare that no specific funding was received for this research from any funding agency in the public, commercial, or not-for-profit sectors.

Data Availability

The datasets used in this study, METR-LA and PEMS-BAY, are publicly available through the Zenodo repository and can be accessed at **Zenodo Data Link** . These datasets were used to conduct all experiments and support the findings reported in this study.

Code Availability

The source code developed for this study, including data preprocessing and graph construction modules, implementations of all evaluated forecasting models, training and evaluation pipelines, robustness assessment framework, graph ablation and sparsity analysis scripts, experimental configurations, and scripts for generating the figures and tables presented in this paper, is publicly available at: **Github Repository** . The repository also includes documentation and reproducibility instructions for reproducing the reported experiments.

Author Contributions

S.N.D. and K.I.G. contributed equally to the conception, methodology, implementation, experiments, data analysis, visualization, and manuscript writing. A.M.M. reviewed the manuscript and provided technical feedback. A.K. reviewed the manuscript and supervised the overall work. All authors read and approved the final manuscript.

Competing Interests

The authors declare no competing interests.

REFERENCES

1. Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in Proc. Int. Conf. Learn. Represent. (ICLR), Vancouver, BC, Canada, 2018.
2. B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in Proc. 27th Int. Joint Conf. Artif. Intell. (IJCAI), Stockholm, Sweden, 2018, pp. 3634–3640.
3. M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in Proc. Advances Neural Inf. Process. Syst. (NeurIPS), vol. 29, 2016, pp. 3844–3852.
4. Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph WaveNet for deep spatial-temporal graph modeling," in Proc. 28th Int. Joint Conf. Artif. Intell. (IJCAI), Macao, China, 2019, pp. 1907–1913.
5. L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," in Proc. Advances Neural Inf. Process. Syst. (NeurIPS), vol. 33, 2020, pp. 17804–17815.
6. S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in Proc. AAAI Conf. Artif. Intell., vol. 33, no. 1, 2019, pp. 922–929.
7. W. Jiang and J. Luo, "Graph neural network for traffic forecasting: A survey," Expert Syst. Appl., vol. 207, p. 117921, 2022.
8. S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.
9. Z. Cui, K. Henrickson, R. Ke, and Y. Wang, "Graph Markov network for traffic forecasting with missing data," Transp. Res. Part C, Emerg. Technol., vol. 117, p. 102671, 2020.
10. J. Zuo, K. Zeitouni, Y. Taher, and S. Garcia-Rodriguez, "Graph convolutional networks for traffic forecasting with missing values," Data Mining Knowl. Discovery, vol. 37, pp. 913–947, 2023.

11. T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in Proc. Int. Conf. Learn. Represent. (ICLR), Toulon, France, 2017.
12. L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.
13. G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, Time Series Analysis: Forecasting and Control, 5th ed. Hoboken, NJ, USA: Wiley, 2015.
14. K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP), Doha, Qatar, 2014, pp. 1724–1734.
15. Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in Proc. 34th Int. Conf. Mach. Learn. (ICML), Sydney, Australia, 2017, pp. 933–941.
16. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in Proc. 3rd Int. Conf. Learn. Represent. (ICLR), San Diego, CA, USA, 2015.
17. E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," Transp. Res. Part C, Emerg. Technol., vol. 43, pp. 3–19, 2014.
18. C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, "Freeway performance measurement system: Mining loop detector data," Transp. Res. Rec., vol. 1748, no. 1, pp. 96–102, 2001.
19. B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results," J. Transp. Eng., vol. 129, no. 6, pp. 664–672, 2003.
20. P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in Proc. Int. Conf. Learn. Represent. (ICLR), Vancouver, BC, Canada, 2018.
21. C. Zheng, X. Fan, C. Wang, and J. Qi, "GMAN: A graph multi-attention network for traffic prediction," in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 1, 2020, pp. 1234–1241.
22. D. Zügner, A. Akbarnejad, and S. Günnemann, "Adversarial attacks on neural networks for graph data," in Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, London, UK, 2018, pp. 2847–2856.
23. D. A. Tedjopurnomo, Z. Bao, B. Zheng, F. M. Choudhury, and A. K. Qin, "A survey on modern deep neural network for traffic prediction: Trends, methods and challenges," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 4, pp. 1544–1561, 2022.
24. Y. Zhang, T. Cheng, and Y. Ren, "A graph deep learning method for short-term traffic forecasting on large road networks," Comput.-Aided Civil Infrastruct. Eng., vol. 34, no. 10, pp. 877–896, 2019.
25. Y. Li, R. Yu, C. Shahabi, and Y. Liu, “Diffusion convolutional recurrent neural network: Data-driven traffic forecasting,” in Proc. Int. Conf. Learn. Represent. (ICLR), 2018.