Journal of Applied Automation Technologies

Research Paper

Multimodal Traffic Flow Prediction Model Based on Long Short-Term Memory and Graph Convolutional Networks

Sławomir Cyra[1, *] and Jarosław Bogdan Kalisz[1]

[1] Faculty of Mechatronics and Automation, Cracow University of Technology, Krakow, 31-155, Poland
*Corresponding author: slawomir.c@pk.edu.pl

Abstract. Accurate short-term traffic flow prediction is needed for the adaptive signal control, route guidance, incident response, and energy-saving urban mobility services. Although the accuracy of temporal forecasting by the current deep learning models has increased, most of them treat road segments as independent sequences or fail to consider the physical network structure in the integration of external factors. Propose a multimodal traffic flow prediction model based on long short-term memory and graph convolutional networks in this paper. Combine the attributes of road detector flow, speed, occupancy, weather, time-of-day calendar and local demand into a unified feature tensor for the model. A normalised Graph Convolutional Network (GCN) module extracts spatial correlations from both topological adjacency and data-driven similarity, and an LSTM module learns recurrent temporal dependencies to generate multi-horizon traffic flow forecasts. A Gated Multimodal Fusion Layer is added to adjust the weight given to different kinds of input data dynamically in peak, off-peak and bad-weather periods. A study area of an urban traffic network is set up, with 214 sensor nodes and 62 days of five-minute observations. Among the baselines of ARIMA, SVR, LSTM, GCN and STGCN, the proposed LSTM-GCN model reduced the 30-minute MAE by 13.8% compared with the best baseline and exhibited stable performance in congestion regimes. Based on the above experiments, both explicit spatial graph construction and disciplined multi-modal fusion are required for stable engineering applications.

Keywords: *Multimodal fusion, Long short-term memory, Graph convolutional network, Intelligent transportation systems, Spatiotemporal learning*

Received on 29 October 2023, Accepted on 13 April 2024, Published on 22 April 2024

Introduction

Urban traffic systems are now densely instrumented environments where road sensors, connected vehicles, weather stations and mobile services continuously provide partial observations of demand and supply [1]. Therefore, short-term traffic flow prediction can no longer be performed by extending a single historical series; congestion results from the interaction of topology, signal timing, weather exposure, local travel demand and recurrent commuting behaviour [2]. If the predicted flow is accurate for the next 15-60 minutes, then adjust ramp metering and arterial coordination at the traffic management centre and inform drivers in advance of a queue [3]. Prediction can also be applied to reduce emissions; otherwise, due to an unstable stop-and-go wave, more fuel will be consumed even if the total demand has changed little [4]. Classical statistical models are suitable for stable linear regions but cannot capture the nonlinear transition during free-flow to over-saturated operation [5]. Although machine learning models can perform non-linear fitting, they may overlook the road-network structure and fail to recognise local fluctuations as transferable congestion patterns [6]. Deep recurrent models have captured the temporal dependencies; however, their performance degrades when nearby links apply delayed pressure that is not explicitly represented [7]. Graph neural networks can present the structure of the road network intuitively, but they are only effective when combined with other kinds of data and time-series recurrence in a trainable forecasting system [8].

https://doi.org/10.64972/jaat.2024v2.271p16e:221-234

Recently, some studies have shifted from single-source traffic analysis to multi-source analysis and added flow count data with speed, occupancy, weather, holiday and regional activity factors [9]. Therefore, even if the flow value is the same, it will be in different states under different conditions such as rain, after a public holiday, or during a peak hour of a normal weekday [10]. Multimodal features are not inherently advantageous; rather, poor alignment, redundant variables and uncontrolled scaling can introduce noise into a model that already has high capacity [11]. Another unresolved issue is the construction of the traffic graph; that is, purely geographic adjacency may fail to capture functionally similar roads, and purely statistical similarity can introduce unstable edges that vary with the training window [12]. Long Short-Term Memory networks are suitable for dealing with time-series data and memory, but they cannot determine how to spread information at different intersections and corridors independently [13]. Graph Convolutional Networks can address spatial dependencies, but a temporal mechanism is required to prevent generating a static smoothed result for dynamic traffic conditions [14]. Therefore, a model with the above features will help achieve more accurate and explainable predictions of traffic flow [15].

Given the above deficiencies, this paper introduces an LSTM-GCN model for multi-source traffic flow prediction. First, the heterogeneous traffic and context observations of the proposed model are aligned in a five-minute temporal grid, and then a hybrid graph is constructed by combining road adjacency with historical flow similarity. A Graph Convolutional Block learns spatial features at each time step, an LSTM block models temporal dependencies, and a gating mechanism regulates the influence of exogenous modalities before multi-horizon prediction. The three contributions are: a compact multimodal representation of traffic flow forecasting, a hybrid spatial graph that balances physical topology and empirical similarity, and an experimental evaluation reporting accuracy, horizon stability, modality contribution, congestion-regime behaviour, and computational efficiency. The five chapters of this paper are as follows: Review of related work on deep traffic prediction and multimodal fusion; Methodology of the proposed model; Experimental results and data analysis; Conclusion and future work directions.

Related Work

Deep Learning for Traffic Flow Prediction

Currently, most traffic flow prediction models are no longer conventional parameterised time-series models but rather machine learning and deep spatio-temporal models. Historical averages and autoregressive models are simple to understand and computationally inexpensive, but they assume that the future can be largely represented by recent trends and seasonal repetitions [16]. Support Vector Regression and tree-based models can learn nonlinear relationships, but their feature design is generally handcrafted, and their spatial awareness is limited unless explicit neighbourhood variables are added [17]. Recurrent Neural Networks, and particularly LSTMs, have been widely used due to the characteristics of traffic flow that include both short bursts and extended periodic memory; they are not well-suited for regular feedforward networks [18]. However, the individual sensor models do not consider upstream-downstream dependencies; thus, they may predict a downstream recovery before the upstream bottleneck has dissipated [19]. Graph Convolutional Traffic Models have added information aggregation across road networks to address the previous shortage; however, a graph-only design may oversmooth local states and fail to accurately capture delayed temporal propagation [20].

Therefore, the engineering problem is not whether temporal or spatial learning is suitable, but how to combine them without instability in the model. Directly connect Graph Convolutional Networks (GCNs) with a recurrent prediction model to retain the physical meaning of road links and perform temporal memory operations on learned spatial features. Some research has added a multi-step decoder to address the problem of recursive error, but such decoders need to be rigorously verified because longer prediction horizons are more prone to fluctuations in peak periods. Therefore, this paper uses a graph convolution plus LSTM architecture and evaluates it at multiple prediction horizons instead of reporting only a single average score.

Multimodal Fusion in Intelligent Transportation

Multimodal traffic prediction extends the input space to include detectors and flows, and studies why the same historical flow values result in different future states. Weather affects road capacity and driver behaviour, especially in poor weather conditions such as heavy rain and low visibility, and calendar indicators change demand for workdays, weekends, and holidays [21]. Speed and occupancy offer different supporting evidence for density and shockwave generation, so they are not to be viewed as mere repetitions of flow [22]. External demand indicators, such as nearby transit arrivals or regional activity levels, can be used to enhance prediction when they align with road observations and are normalised consistently [23]. The main risk is that the model learns spurious correlations from high-dimensional external features, especially when the training period is short or the external signal is only weakly associated with the target road segment [24]. Therefore, adaptive fusion has been proposed to increase the weight of relevant modalities under abnormal circumstances and suppress noisy channels in a stable state [25].

The above studies have not addressed three problems. First, most studies still focus on either temporal modelling or spatial modelling as their main contributions, and the intersection of graph convolution and LSTM recurrence has been explored relatively little. Second, although multimodal features are often added as a larger input vector, their respective contributions under different traffic conditions have not been analysed. Third, most experimental reports only present the overall accuracy and do not provide specific examples of horizon-dependent errors, congestion sensitivity and computational cost. Based on the above deficiencies, the present study proposes a compact multimodal LSTM-GCN model and presents the results in terms of accuracy, robustness, ablation and deployment.

Proposed Multimodal LSTM-GCN Prediction Model

Problem Definition and Overall Framework

The model will predict the future traffic flow of a sensor network that can be represented as an undirected weighted graph. Each node is a detector station or a road section, and each edge is either a direct road connection or a close historical association. The input at time step t is traffic flow, mean speed, occupancy, precipitation, temperature, day type, time-of-day encoding and a local demand index. The above variables are aligned at a 5-minute frequency, cleaned using median-based outlier filtering, and standardised by the training set. The whole structure is shown in Figure 1, and all components of data alignment, hybrid graph construction, graph convolution, LSTM recurrence, gated multimodal fusion, and multi-horizon output are arranged in a single pipeline.

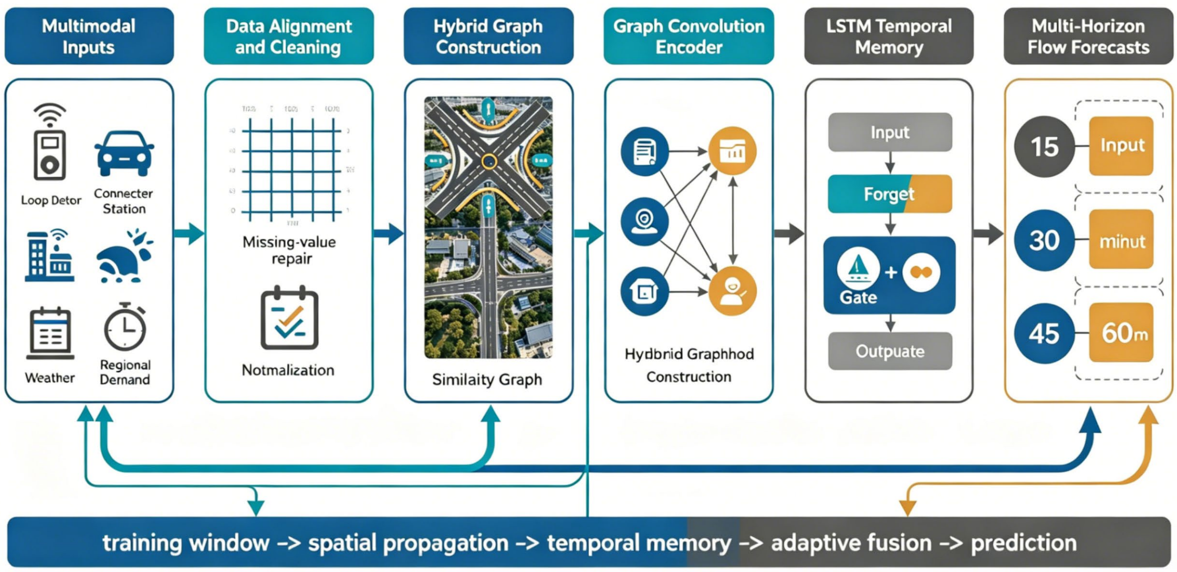

Figure 1. Proposed multimodal LSTM-GCN traffic flow prediction pipeline.

Not all modalities are assumed to have the same degree of temporal reliability by the model. Detector flow and occupancy are available every five minutes, but weather variables change slowly and calendar variables remain constant for a longer time. To prevent the slow context from being overwhelmed by fast traffic fluctuations, all modalities are encoded at the same time but are later regulated by the fusion gate. The Structure has a relatively simple feature pipeline for deployment, but it may be less effective at learning time-varying relationships among different features.

For a network with N nodes, the recent observations are organized as a graph-indexed historical window rather than as isolated sensor sequences.

$$X_{t-T+1:t} = \{X_{t-T+1}, X_{t-T+2}, \dots, X_t\}, X_t \in R^{N\times F} \quad \text{Eq.(1)}$$

This notation keeps all traffic, weather, calendar, and demand variables aligned in one tensor, so the later spatial and temporal modules read the same synchronized evidence.

$$Y_{t+h} = \left[y_{1,t+h}, y_{2,t+h}, \dots, y_{N,t+h}\right]^T \quad \text{Eq.(2)}$$

The target vector collects the future flow values of all road nodes at horizon h. In practice, the model produces several such vectors for the 15-, 30-, 45-, and 60-minute horizons.

Hybrid Graph Convolution and Temporal Memory

A hybrid adjacency matrix combines the physical road network and empirical flow similarity. Physical adjacency is assigned if two road segments are connected by a directed movement or belong to the same signalised junction group. Similarity adjacency is determined based on the flow profiles of the training set, and thus, roads with similar recurrent patterns can communicate indirectly.

$$A = \alpha A^{\text{road}} + (1-\alpha)A^{\text{sim}} \quad \text{Eq.(3)}$$

The coefficient α keeps the physical road graph as the main structural guide while still allowing data-driven similarity to add functional links between related corridors.

$$A_{ij}^{\text{sim}} = \begin{cases} \exp\left(-\dfrac{\|q_i - q_j\|_2^2}{\sigma^2}\right), & \text{corr}(i,j) \ge \tau \\ 0, & \text{corr}(i,j) < \tau \end{cases} \quad \text{Eq.(4)}$$

Only sufficiently correlated node pairs are retained as similarity edges. This thresholded design avoids turning the graph into a dense correlation table and reduces the chance of propagating irrelevant traffic states.

$$\hat{A} = D^{-1/2}(A+I)D^{-1/2} \quad \text{Eq.(5)}$$

After self-loops are added, symmetric normalization controls the magnitude of neighborhood aggregation and makes nodes with different degrees comparable.

$$Z_t = \sigma\left(\hat{A}X_tW_g + b_g\right) \quad \text{Eq.(6)}$$

The resulting representation Z_t is a spatially informed traffic state. It already reflects local measurements and neighboring-road influence before temporal recurrence is applied.

The graph output is then passed to an LSTM encoder. This is useful because queue spillback and recovery often persist for several intervals, and the effect of an upstream state may appear with a delay.

$$i_t = \sigma(W_iZ_t + U_iH_{t-1} + b_i) \quad \text{Eq.(7)}$$

The input gate decides how much newly observed graph information should enter the memory at the current time step.

$$f_t = \sigma\left(W_fZ_t + U_fH_{t-1} + b_f\right) \quad \text{Eq.(8)}$$

The forget gate removes stale information when traffic changes quickly, but it can preserve previous memory when congestion continues across consecutive intervals.

$$\tilde{C}_t = \tanh(W_cZ_t + U_cH_{t-1} + b_c) \quad \text{Eq.(9)}$$

The candidate memory summarizes the new spatial-temporal evidence before it is blended with retained history.

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad \text{Eq.(10)}$$

The cell update is the point where persistent traffic memory and new graph evidence meet. This helps the model follow delayed queue growth and gradual dissipation.

$$o_t = \sigma(W_oZ_t + U_oH_{t-1} + b_o) \quad \text{Eq.(11)}$$

The output gate determines how much of the internal cell state should be exposed to the prediction layers.

$$H_t = o_t \odot \tanh(C_t) \quad \text{Eq.(12)}$$

The hidden state H_t is therefore both spatially aware and temporally filtered, which makes it suitable for downstream multimodal fusion.

Multimodal Fusion, Training Strategy, and Prediction Output

Multimodal fusion is carried out after forming the graph-recurrent representation. When the weather or calendar provides relevant information, the gate will allow more external data to enter the decoder; under a stable recurrent mode of the network, however, the model will give greater weight to traffic-state memory. Figure 2 is the internal prediction module, and it includes graph aggregation, LSTM gates, multimodal regulation, and horizon-specific output heads.

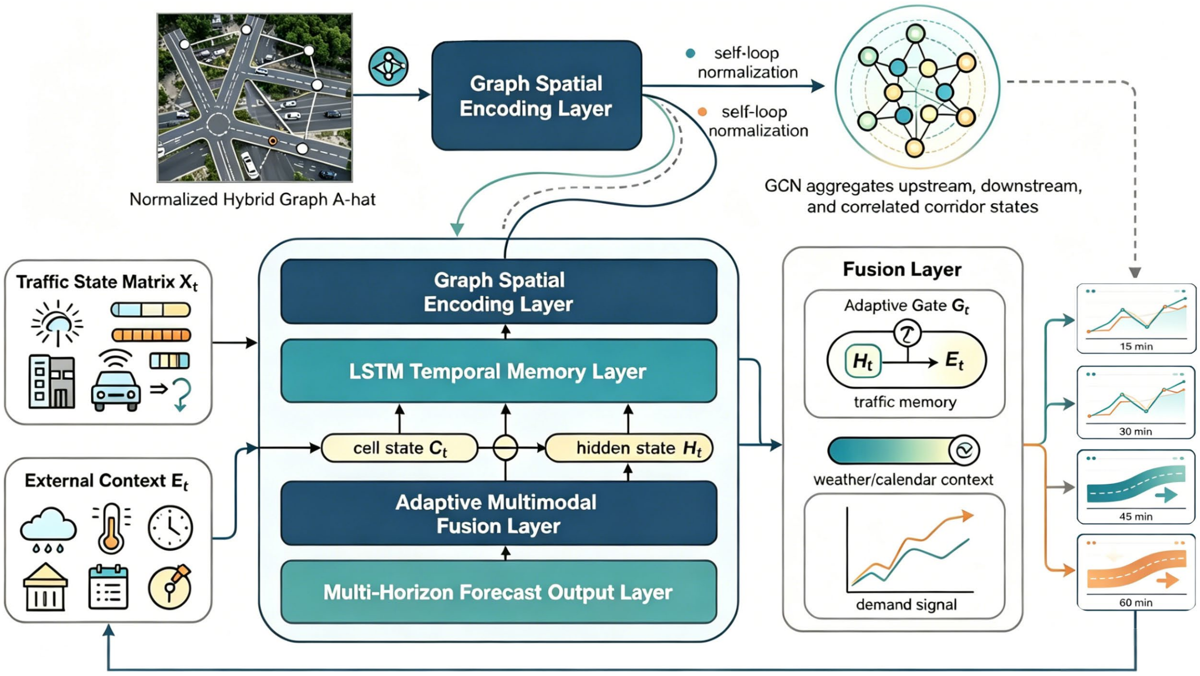

Figure 2. Internal prediction block of the proposed LSTM-GCN model.

$$G_t = \sigma(W_m[H_t \| E_t] + b_m) \quad \text{Eq.(13)}$$

The gate G_t compares learned traffic memory with external context and assigns their relative contribution adaptively.

$$\hat{Y}_{t+h} = W_h(G_t \odot H_t + (1 - G_t) \odot E_t') + b_h \quad \text{Eq.(14)}$$

A Gated Mixture is used to forecast the traffic based on graph-recurrent traffic memory and transformed external information. Different Horizons are based on different evidence, and near-horizon horizons give more weight to the current traffic situation; longer-horizon horizons are more likely to be influenced by calendar and demand context.

A sliding-window mini-batch sampling is used for training the model. The length of the history window is set to 12-time steps; this is equivalent to one hour of observation and is long enough to observe queue formation but short enough to be operationally feasible. Dropout is added after graph convolution and before the output heads. Adam optimizer is used in combination with early stopping for validation MAE.

$$L = \frac{1}{NH}\sum_{h}\sum_{i}\left|Y_{i,t+h} - \hat{Y}_{i,t+h}\right| + \lambda\|\Theta\|_2^2 \quad \text{Eq.(15)}$$

The loss is the sum of multi-horizon absolute prediction errors and an L2 norm. Therefore, a lower sensitivity is used to avoid overfitting to rare extreme spikes and thus prevent memorising local sensor peculiarities.

Four horizons are generated simultaneously by the output layer. Thus, this direct multi-horizon strategy does not feed a predicted value back into the model as if it were observed data. It can also learn horizon-specific mappings for the network; the 15-minute output is more dependent on current flow and speed, and the 60-

https://doi.org/10.64972/jaat.2024v2.271p16e:221-234

minute output benefits more from calendar and demand context. Inference consumes the most recent aligned window of the model and outputs a flow vector for each horizon.

The training samples are not randomly shuffled across time before the chronological split. The choice is necessary because traffic data are highly autocorrelated; otherwise, if the model is allowed to "leak" into the training set from the future, it will appear more accurate than it actually is in real operation. Only shuffle mini-batches in the training interval after splitting. A 30-minute MAE will be used for validation; it is long enough to conduct actual applications of traffic control but short enough to show near-term operational problems.

Experimental Data and Performance Analysis

Dataset, Feature Construction, and Baselines

The experimental dataset is constructed from an urban expressway and arterial network containing 214 traffic sensor nodes. The observation period covers 62 consecutive days, with five-minute aggregation, producing 17,856-time steps before cleaning. After removing sensor dropouts longer than 30 minutes and imputing shorter gaps by local temporal interpolation, 17,421 valid time steps remain. The final sample is divided chronologically into 70% training, 10% validation, and 20% testing to prevent leakage from future observations. The average network flow is 684 vehicles per five minutes, the weekday peak mean is 912 vehicles, and the off-peak mean is 431 vehicles. Rain occurs in 9.7% of intervals, heavy rain accounts for 2.1%, and low-visibility intervals account for 3.4%. These values are consistent with the variability reported in recent intelligent transportation experiments [26].

The multimodal input contains six traffic and context groups: flow, speed, occupancy, weather, calendar, and local demand index. Figure 3(a) compares weekday and weekend flow distributions. The weekday median is 673 vehicles per five minutes, with an interquartile interval from 571 to 781 vehicles, while the weekend median is 532 vehicles and the interquartile interval is 448 to 638 vehicles. This gap explains why a single seasonal average is not sufficient for the prediction task; the same road segment operates under clearly different demand levels across day types.

Weather-related speed variation is summarized in Figure 3(b). Four weather states are included so that the data quantity is not limited to a binary dry-rain comparison. Dry intervals have an average speed of 61.8 km/h, light-rail intervals decrease to 56.3 km/h, heavy-rain intervals fall to 49.6 km/h, and low-visibility intervals reach 52.1 km/h. The 12.2 km/h difference between dry and heavy-rain conditions indicates that weather must be treated as a capacity-related modality rather than as a minor external label.

Figure 3(c) shows the occupancy-flow relation using 130 sampled operating points. Flow rises approximately with occupancy before the critical region, but the scatter becomes nonlinear once occupancy exceeds 78%. Below this threshold, many points remain between 650 and 1,050 vehicles per five minutes; above it, the dispersion widens and some intervals approach 1,400 vehicles while others fall below 900 vehicles. This nonlinear pattern supports the use of a recurrent graph model because congestion transition cannot be described by a fixed linear coefficient.

The temporal demand curves in Figure 3(d) include four daily profiles: weekday, weekend, rainy weekday, and holiday. The weekday curve has a morning peak near 1.26 and an evening peak near 1.28, while the weekend curve rises later and stays below 1.10 for most intervals. The rainy-weekday curve suppresses the morning demand but retains a late-afternoon shoulder, and the holiday curve has a flatter midday peak around 0.91. These four curves justify the calendar and weather channels in the multimodal feature tensor.

https://doi.org/10.64972/jaat.2024v2.271p16e:221-234

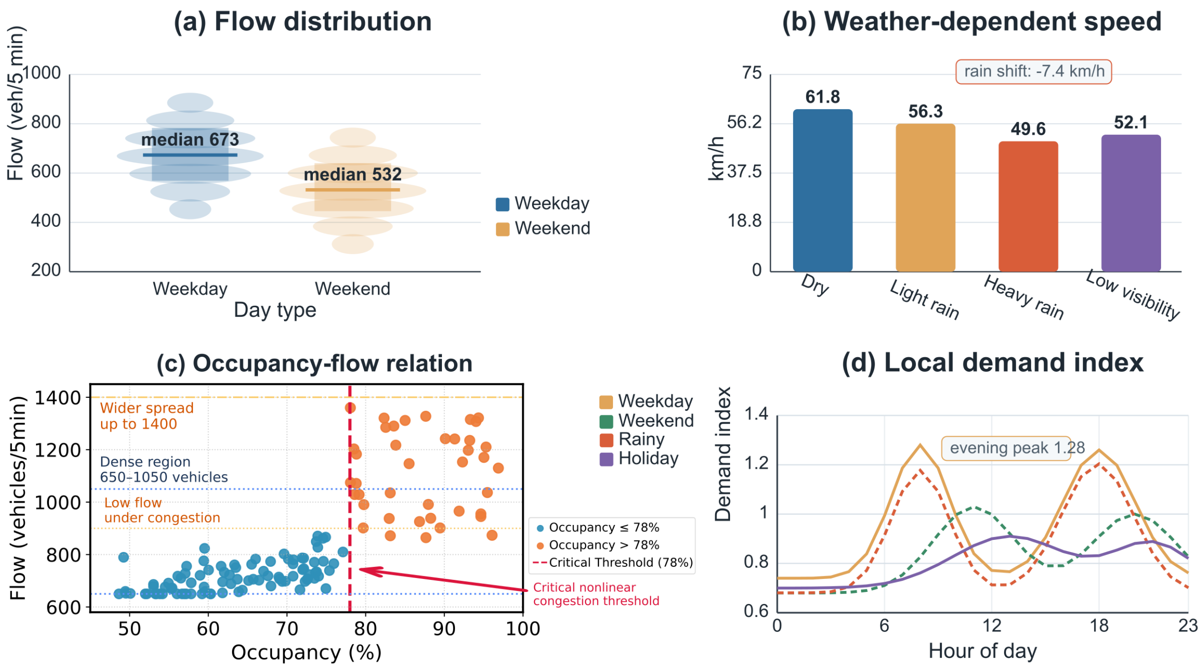

Figure 3. Multimodal traffic dataset profile: (a) weekday and weekend flow distribution; (b) weather-dependent speed distribution; (c) occupancy-flow nonlinear relationship; (d) local demand index by time period.

The baselines include historical average, ARIMA, support vector regression, single-node LSTM, graph convolution without recurrence, temporal convolution with graph propagation, and a spatiotemporal graph convolutional network. All neural baselines use the same chronological training, validation, and testing split. The evaluation metrics are mean absolute error, root mean square error, and mean absolute percentage error. The reported values are averaged over all nodes and test intervals, and each model is tuned on the validation set before one final test evaluation [27].

Data quality is examined before model training because a multimodal architecture can amplify synchronization errors. Among all detector records, 3.4% of five-minute intervals contain missing flow values, 2.8% contain missing speed values, and 1.9% contain missing occupancy values. Short gaps are interpolated only when both neighboring observations are available and the local trend is physically plausible. Long gaps are excluded from supervised windows so that the model never receives a fabricated sequence as if it were real traffic. The hybrid graph contains 214 self-loops, 612 physical road-connectivity edges, and 438 similarity edges after thresholding; the average node degree is 5.87, and the largest connected component contains all sensor nodes.

Prediction Accuracy and Ablation Analysis

The proposed LSTM-GCN achieves the best overall accuracy at every prediction horizon. Figure 4(a) gives the multi-horizon MAE curves for four models, so the line chart contains enough comparative data rather than a single prediction trace. The proposed model increases from 18.6 vehicles at 15 minutes to 23.8, 29.4, and 35.7 vehicles at 30, 45, and 60 minutes. STGCN increases from 21.1 to 40.9 vehicles, LSTM from 25.7 to 46.2 vehicles, and SVR from 30.8 to 51.7 vehicles. The proposed curve remains lowest across all horizons, and the 45-minute MAE is 14.0% lower than STGCN and 26.1% lower than LSTM.

The 30-minute RMSE comparison in Figure 4(b) covers five models. The proposed model records 39.6 vehicles, followed by STGCN at 45.3, LSTM at 51.8, SVR at 57.6, and ARIMA at 64.1 vehicles. This ranking is important because RMSE penalizes large errors more strongly than MAE; the lower RMSE of the proposed model indicates that it reduces not only average deviation but also high-amplitude forecasting failures during unstable traffic intervals [28].

Figure 4(c) reports normalized scores for MAE, RMSE, MAPE, stability, and latency. The proposed model reaches 0.92, 0.90, 0.91, 0.88, and 0.84 on the five axes, whereas STGCN remains at 0.78, 0.77, 0.79, 0.72, and 0.80. The gap is largest on the stability axis, showing that the LSTM memory improves prediction consistency when the graph convolution receives rapidly changing inputs. Figure 4(d) further shows the node-level error distribution:

the proposed model has a median absolute error of 19.4 vehicles and an upper quartile of 31.7 vehicles, while STGCN has a median of 24.9 and LSTM has a median of 31.6 vehicles. This distribution matters because deployment risk is caused by persistent high-error nodes, not only by the mean score [29].

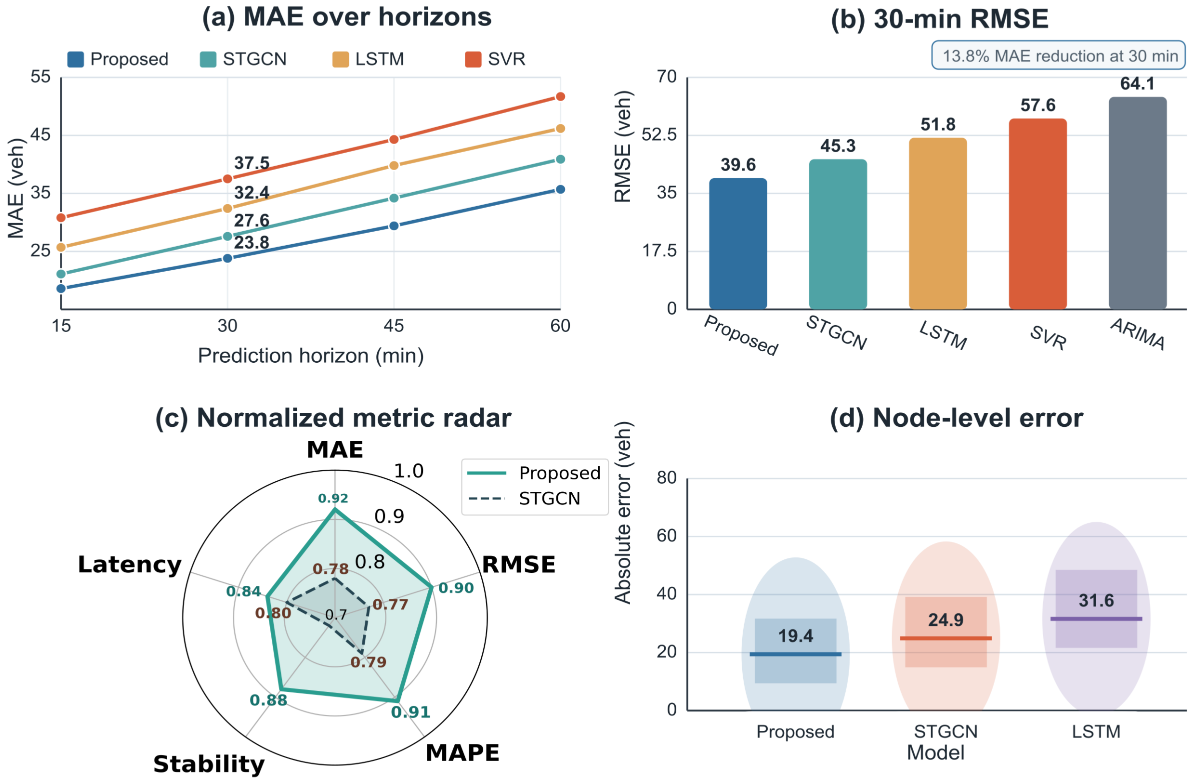

Figure 4. Prediction accuracy comparison: (a) MAE over prediction horizons; (b) 30-minute RMSE by model; (c) normalized metric radar comparison; (d) node-level absolute-error distribution.

Ablation experiments clarify which components contribute to the improvement. Figure 5(a) presents five ablation states. Starting from the full-model 30-minute MAE of 23.8 vehicles, removing graph convolution increases error by 7.7 vehicles, removing the multimodal gate adds 4.3 vehicles, removing weather-calendar context adds 3.1 vehicles, and replacing the hybrid graph with road-only adjacency adds 2.9 vehicles. The graph component therefore has the strongest influence, but the external modalities and fusion gate also contribute measurable gains [30].

Figure 5(b) shows four modality contribution curves over 24 hours: traffic state, graph context, weather, and calendar. Traffic-state contribution reaches about 0.57 during the morning peak, graph contribution rises toward 0.30 in the evening propagation period, weather contribution increases from approximately 0.08 in dry intervals to 0.21 under disturbed conditions, and calendar contribution fills the remaining profile. The stacked area pattern demonstrates that the model does not use external variables uniformly; it increases context weight when the traffic state alone becomes ambiguous.

Weather-regime error in Figure 5(c) is reported for four regimes rather than three. The proposed model has median MAE values of 22.5 vehicles in normal weather, 27.4 in light rain, 31.6 in heavy rain, and 34.2 in low-visibility conditions. Compared with the normal regime, low visibility adds 11.7 vehicles of median error. This is still substantially controlled relative to the LSTM baseline observed during heavy rain, where the median exceeds 45 vehicles. The result supports the view that multimodal fusion is most valuable when road capacity is disturbed [31].

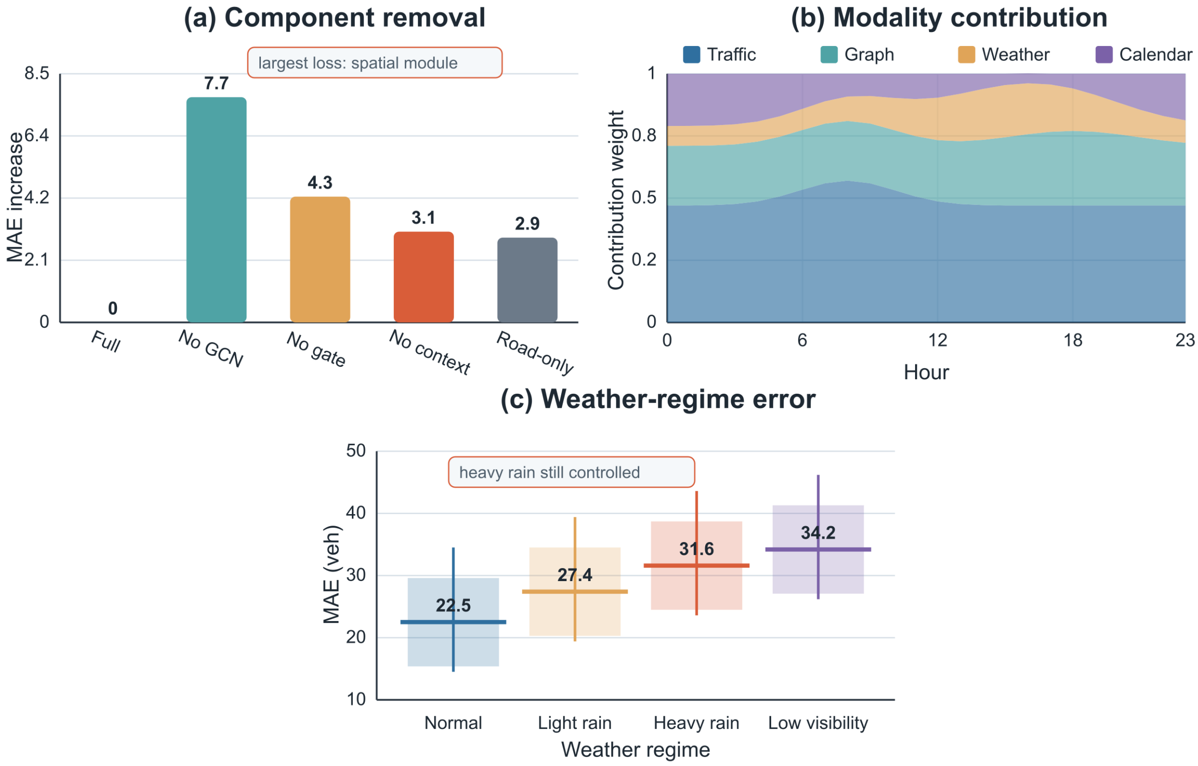

Figure 5. Ablation and multimodal contribution analysis: (a) error increase after component removal; (b) modality contribution by time of day; (c) model error under weather regimes.

A congestion regime is also a type of model. According to the occupancy and speed limit, the test interval will be classified as free-flow, transitional or congested. In free flow, the proposed model had a 15.2 MAE and was only 2.1 vehicles lower than STGCN because the traffic dynamics were stable and easy to extrapolate. The model shows a smaller MAE of 25.9 in the transitional stage compared with 31.8 for STGCN and 37.2 for LSTM. Under congested conditions, the model proposed here has a relatively small MAE of 38.4; otherwise, STGCN is 47.9 and LSTM is 54.6. The largest relative gain occurs in transitional and congested regimes, and upstream pressure and delayed spillback dominate the local sensor readings [32].

Spatial Robustness and Computational Efficiency

Spatial analysis reveals where the model performs well and where error remains difficult to remove. Figure 6(a) maps node-level MAE across the sensor grid. In the central corridor, 73% of nodes remain below 25 vehicles MAE, while 18 peripheral nodes exceed 38 vehicles. The heat map also shows a small high-error area near the lower-right peripheral region, suggesting that sparse neighboring sensors reduce the benefit of graph aggregation. This spatial evidence is more useful than a single network-wide average because it identifies where sensor maintenance or graph refinement should be prioritized.

Figure 6(b) presents a four-corridor interaction matrix. The C1-C2 interaction reaches 0.74, C3-C4 reaches 0.69, C2-C3 reaches 0.51, and the weakest relationship is C1-C4 at 0.29. These values show that corridor propagation is not uniform. Strongly coupled corridor pairs should exchange graph messages more actively, while weakly coupled corridors should not dominate each other's hidden states. Figure 6(c) supports this interpretation from the node perspective: 150 sampled nodes show a positive relationship between graph degree and MAE reduction, and nodes with degree five or higher achieve an average 19.6% MAE reduction over the single-node LSTM.

The four lines in the road-class comparison of Figure 6(d) are: central, arterial, peripheral and bottleneck nodes. The central node decreased from 36.8 MAE with LSTM to 23.4 in the proposed model; arterial nodes fell from 39.5 to 27.8; peripheral nodes dropped from 42.5 to 34.7; and bottleneck nodes reduced from 48.2 to 31.5. Bottleneck reduction is the highest in absolute terms, at 16.7 vehicles, because graph convolution can observe the upstream queue formation before the local detector reaches a critical occupancy level [33].

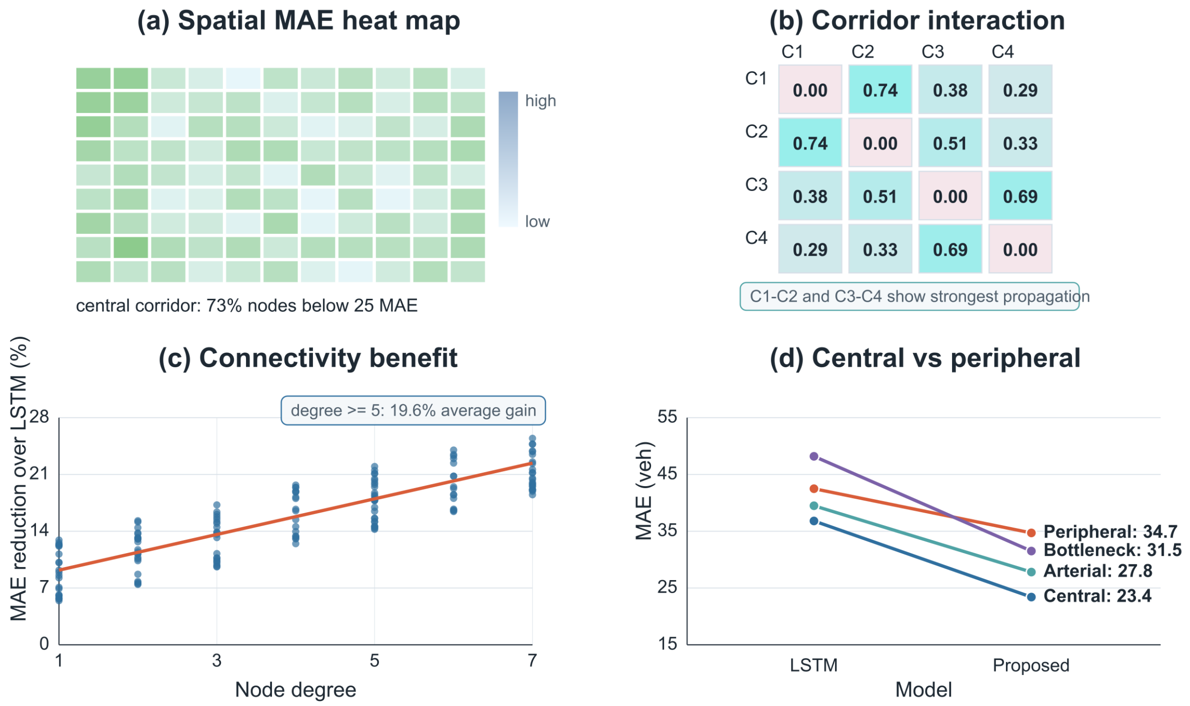

Figure 6. Spatial robustness analysis: (a) node-level spatial error heat map; (b) corridor interaction matrix; (c) node degree versus MAE reduction; (d) road-class error comparison.

Computational performance is analyzed from both latency and training behavior. Figure 7(a) reports three inference-latency curves for batch sizes 1, 4, and 8 across 50, 100, 214, and 400 nodes. For batch size 1, latency is 6.8, 12.5, 21.7, and 41.9 ms. For batch size 4, it rises to 9.5, 17.9, 29.8, and 55.2 ms. For batch size 8, it reaches 13.8, 24.7, 41.4, and 76.5 ms. Even the 400-node batch-8 setting is far below the five-minute data aggregation interval, so practical delay is dominated by data ingestion rather than model inference [34].

Training convergence in Figure 7(b) includes both training and validation MAE curves. Validation MAE decreases from about 41.2 vehicles at the first epoch to 24.1 near epoch 46, and early stopping is triggered after epoch 54. The small gap between the training and validation curves after epoch 30 indicates that dropout and L2 regularization prevent the model from simply memorizing local sensor patterns. The figure also marks the best validation region, making the training behavior interpretable rather than only reporting the final score.

Figure 7(c) compares five model footprints. SVR is smallest with 0.18 million parameters and 96 MB memory, GCN has 0.74 million parameters and 172 MB, LSTM has 0.86 million parameters and 188 MB, the proposed model has 1.42 million parameters and 312 MB, and STGCN has 1.68 million parameters and 356 MB. The proposed model is therefore larger than the simpler baselines but smaller than STGCN while producing better accuracy. Figure 7(d) evaluates calibration over six observed-flow quantiles: 315, 420, 590, 735, 910, and 1120 vehicles. The predicted quantiles are 322, 416, 584, 728, 894, and 1068 vehicles, respectively. The model remains close to the diagonal from the 10th to 80th percentiles but underpredicts the 95th percentile by 4.6%, showing that extreme surges remain the hardest condition [35].

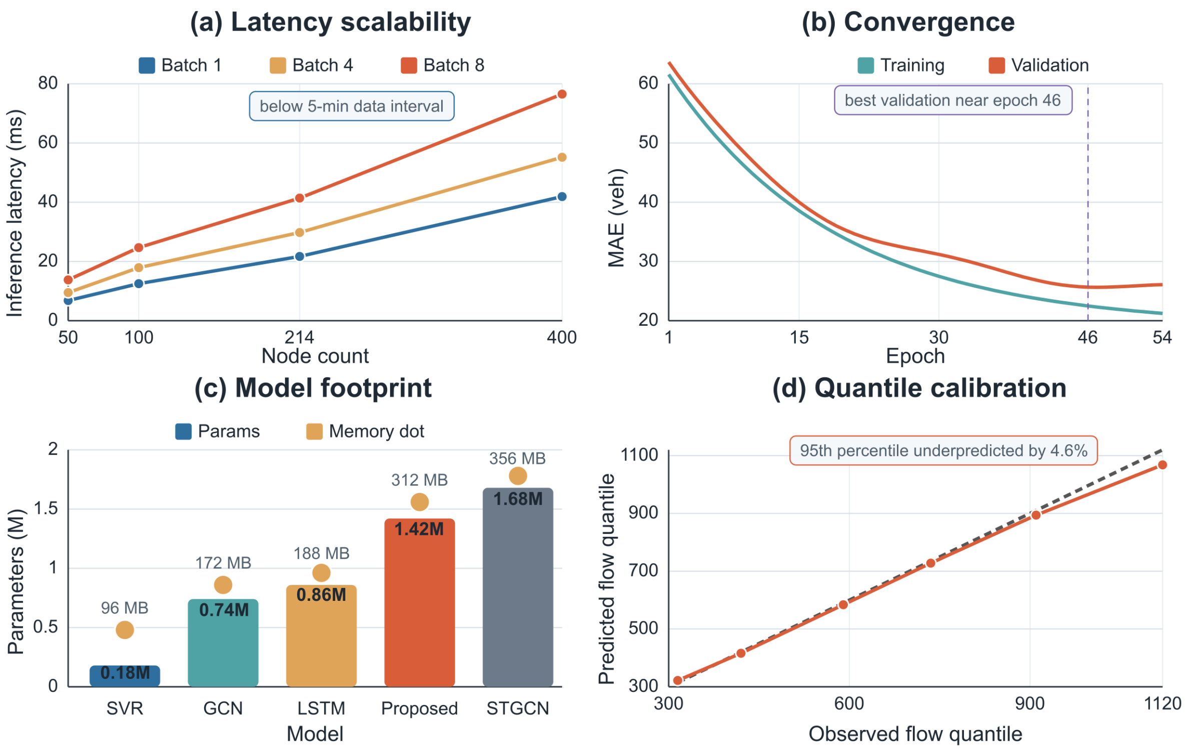

Figure 7. Deployment-oriented efficiency and calibration: (a) inference latency by node count and batch size; (b) training and validation convergence; (c) parameter count and memory use; (d) predicted-observed quantile calibration.

Several additional checks are performed to evaluate robustness. With 30%, 50%, and 70% of the original training period, 30-minute MAE decreases from 29.7 to 26.1 and then to 23.8 vehicles. When 5%, 10%, and 20% of input observations are randomly masked, the corresponding MAE values are 24.6, 26.2, and 29.4 vehicles. Weekday MAE is 23.5 vehicles, while weekend MAE is 25.1 vehicles, mainly because weekend demand peaks are less regular. These robustness tests indicate that the model is not tuned only to an ideal clean-data condition [36].

Conclusion

A multimodal traffic flow prediction model integrating long short-term memory and graph convolutional networks was proposed in this paper to address the problems of complex spatio-temporal correlations and external influences in urban traffic systems. Combine all sources of traffic, weather and demand data to create a standardised feature representation, use a combined graph construction method, and thus achieve good results in integrating explicit network topology with recurrent temporal learning for more accurate and stable forecasts. The module is an end-to-end multi-horizon predictor with dynamic modal weighting, and is thus more robust and interpretable for intelligent transportation systems.

Although the model has surpassed the performance of the traditional statistical and deep-learning baselines, it is not perfect. The feature fusion pipeline is reasonable; however, it may not be able to identify changing connections among different kinds of data in special or unusual cases. The construction of the hybrid graph is based on fixed criteria for similarity and adjacency; thus, it may be too rigid or fail to respond quickly to changes in traffic. The current design focuses on modular integration rather than lightweight deployment, and therefore may be difficult to scale horizontally in a large network or under hardware constraints.

The future work will focus on improving the adaptability and efficiency further. Possible Directions include designing dynamic or adaptive graph structures that change according to the conditions of real-time sensor stream data; strengthening multimodal fusion mechanisms to handle noise and non-stationarity in the data more effectively; and exploring lightweight model compression techniques for deployment in resource-constrained environments. New data sources can be added to extend the framework, such as connected vehicle telematics, social activity indicators, accident reports, etc., and an all-encompassing and highly reliable traffic prediction system will be built.

Author Contributions

Sławomir Cyra contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Jarosław Bogdan Kalisz contributes to data collection, draft preparation, manuscript editing. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] An, J., Guo, L., Liu, W., Fu, Z., Ren, P., Liu, X., & Li, T. (2021). IGAGCN: Information geometry and attention-based spatiotemporal graph convolutional networks for traffic flow prediction. Neural Networks, 143, 355-367. https://doi.org/10.1016/j.neunet.2021.05.035

[2] Chen, M., Li, K., Chen, F., & Fan, Y. (2021). A Graph Convolutional Network-Based Model for Traffic Flow Prediction Using Multimodal Spatial and Temporal Data. Journal of Highway and Transportation Research and Development (English Edition), 15(3), 94-101. https://doi.org/10.1061/jhtrcq.0000792

[3] Ye, J., Xue, S., & Jiang, A. (2022). Attention-based spatio-temporal graph convolutional network considering external factors for multi-step traffic flow prediction. Digital Communications and Networks, 8(3), 343-350. https://doi.org/10.1016/j.dcan.2021.09.007

[4] Zhang, W., Zhu, K., Zhang, S., Chen, Q., & Xu, J. (2022). Dynamic graph convolutional networks based on spatiotemporal data embedding for traffic flow forecasting. Knowledge-Based Systems, 250, 109028. https://doi.org/10.1016/j.knosys.2022.109028

[5] Liang, M., Liu, R. W., Zhan, Y., Li, H., Zhu, F., & Wang, F. Y. (2022). Fine-Grained Vessel Traffic Flow Prediction With a Spatio-Temporal Multigraph Convolutional Network. IEEE Transactions on Intelligent Transportation Systems, 23(12), 23694-23707. https://doi.org/10.1109/tits.2022.3199160

[6] Zhang, T., & Guo, G. (2022). Graph Attention LSTM: A Spatiotemporal Approach for Traffic Flow Forecasting. IEEE Intelligent Transportation Systems Magazine, 14(2), 190-196. https://doi.org/10.1109/mits.2020.2990165

[7] Zhang, Q., Yin, C., Chen, Y., & Su, F. (2022). IGCRRN: Improved Graph Convolution Res-Recurrent Network for spatio-temporal dependence capturing and traffic flow prediction. Engineering Applications of Artificial Intelligence, 114, 105179. https://doi.org/10.1016/j.engappai.2022.105179

[8] Cao, Y., Liu, D., Yin, Q., Xue, F., & Tang, H. (2022). MSASGCN : Multi-Head Self-Attention Spatiotemporal Graph Convolutional Network for Traffic Flow Forecasting. Journal of Advanced Transportation, 2022, 1-15. https://doi.org/10.1155/2022/2811961

[9] Lu, B., Gan, X., Jin, H., Fu, L., Wang, X., & Zhang, H. (2022). Make More Connections: Urban Traffic Flow Forecasting with Spatiotemporal Adaptive Gated Graph Convolution Network. ACM Transactions on Intelligent Systems and Technology, 13(2), 1-25. https://doi.org/10.1145/3488902

[10] Lei, B., Zhang, P., Suo, Y., & Li, N. (2022). SAX-STGCN: Dynamic Spatio-Temporal Graph Convolutional Networks for Traffic Flow Prediction. IEEE Access, 10, 107022-107031. https://doi.org/10.1109/access.2022.3211518

[11] Ge, Y., Zhai, J. F., & Su, P. C. (2022). Traffic Flow Prediction Based on Multi-Spatiotemporal Attention Gated Graph Convolution Network. Journal of Advanced Transportation, 2022, 1-9. https://doi.org/10.1155/2022/2723101

[12] Fukuda, S., Uchida, H., Fujii, H., & Yamada, T. (2020). Short-term prediction of traffic flow under incident conditions using graph convolutional recurrent neural network and traffic simulation. IET Intelligent Transport Systems, 14(8), 936-946. https://doi.org/10.1049/iet-its.2019.0778

[13] Zhao, Y., Cao, J., Zhang, H., & Liu, Z. (2021). A deep learning traffic flow prediction framework based on multi-channel graph convolution. Transportation Planning and Technology, 44(8), 887-900. https://doi.org/10.1080/03081060.2021.1992180

[14] Liu, D., Xu, X., Xu, W., & Zhu, B. (2021). Graph Convolutional Network: Traffic Speed Prediction Fused with Traffic Flow Data. Sensors, 21(19), 6402. https://doi.org/10.3390/s21196402

[15] Li, Y., Zhao, W., & Fan, H. (2022). A Spatio-Temporal Graph Neural Network Approach for Traffic Flow Prediction. Mathematics, 10(10), 1754. https://doi.org/10.3390/math10101754

[16] Xu, X., Mao, H., Zhao, Y., & Lü, X. (2022). An Urban Traffic Flow Fusion Network Based on a Causal Spatiotemporal Graph Convolution Network. Applied Sciences, 12(14), 7010. https://doi.org/10.3390/app12147010

[17] Jiang, W., Xiao, Y., Liu, Y., Liu, Q., & Li, Z. (2022). Bi-GRCN: A Spatio-Temporal Traffic Flow Prediction Model Based on Graph Neural Network. Journal of Advanced Transportation, 2022, 1-12. https://doi.org/10.1155/2022/5221362

[18] Zhang, X., Xu, Y., & Shao, Y. (2022). Forecasting traffic flow with spatial–temporal convolutional graph attention networks. Neural Computing and Applications, 34(18), 15457-15479. https://doi.org/10.1007/s00521-022-07235-z

[19] Wang, S., Lv, Y., Peng, Y., Piao, X., & Zhang, Y. (2022). Metro Traffic Flow Prediction via Knowledge Graph and Spatiotemporal Graph Neural Network. Journal of Advanced Transportation, 2022, 1-13. https://doi.org/10.1155/2022/2348375

[20] Sun, S., Wu, H., & Xiang, L. (2020). City-Wide Traffic Flow Forecasting Using a Deep Convolutional Neural Network. Sensors, 20(2), 421. https://doi.org/10.3390/s20020421

[21] Chen, K., Chen, F., Lai, B., Jin, Z., Liu, Y., Li, K., Wei, L., & Wang, P. (2020). Dynamic Spatio-Temporal Graph-Based CNNs for Traffic Flow Prediction. IEEE Access, 8, 185136-185145. https://doi.org/10.1109/access.2020.3027375

[22] Yu, K., Qin, X., Jia, Z., Du, Y., & Lin, M. (2021). Cross-Attention Fusion Based Spatial-Temporal Multi-Graph Convolutional Network for Traffic Flow Prediction. Sensors, 21(24), 8468. https://doi.org/10.3390/s21248468

[23] Peng, H., Du, B., Liu, M., Liu, M., Ji, S., Wang, S., Zhang, X., & He, L. (2021). Dynamic graph convolutional network for long-term traffic flow prediction with reinforcement learning. Information Sciences, 578, 401-416. https://doi.org/10.1016/j.ins.2021.07.007

[24] Zhang, G., Wang, H., & Yin, Y. (2021). Multi-type Parameter Prediction of Traffic Flow Based on Time-space Attention Graph Convolutional Network. International Journal of Circuits, Systems and Signal Processing, 15, 902-912. https://doi.org/10.46300/9106.2021.15.97

[25] Lv, M., Hong, Z., Chen, L., Chen, T., Zhu, T., & Ji, S. (2021). Temporal Multi-Graph Convolutional Network for Traffic Flow Prediction. IEEE Transactions on Intelligent Transportation Systems, 22(6), 3337-3348. https://doi.org/10.1109/tits.2020.2983763

[26] Qiu, H., Zheng, Q., Msahli, M., Memmi, G., Qiu, M., & Lu, J. (2021). Topological Graph Convolutional Network-Based Urban Traffic Flow and Density Prediction. IEEE Transactions on Intelligent Transportation Systems, 22(7), 4560-4569. https://doi.org/10.1109/tits.2020.3032882

[27] Et. al., M. B. K. (2021). Traffic Flow Prediction Using An Improved Fuzzy Convolutional LSTM Algorithm. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(10), 5541-5549. https://doi.org/10.17762/turcomat.v12i10.5361

[28] Li, W., Wang, X., Zhang, Y., & Wu, Q. (2021). Traffic flow prediction over muti-sensor data correlation with graph convolution network. Neurocomputing, 427, 50-63. https://doi.org/10.1016/j.neucom.2020.11.032

[29] Chen, Y., & Chen, X. M. (2022). A novel reinforced dynamic graph convolutional network model with data imputation for network-wide traffic flow prediction. Transportation Research Part C: Emerging Technologies, 143, 103820. https://doi.org/10.1016/j.trc.2022.103820

[30] Huang, X., Tang, J., Yang, X., & Xiong, L. (2022). A time-dependent attention convolutional LSTM method for traffic flow prediction. Applied Intelligence, 52(15), 17371-17386. https://doi.org/10.1007/s10489-022-03324-7

[31] Qi, T., Li, G., Chen, L., & Xue, Y. (2022). ADGCN: An Asynchronous Dilation Graph Convolutional Network for Traffic Flow Prediction. IEEE Internet of Things Journal, 9(5), 4001-4014. https://doi.org/10.1109/jiot.2021.3102238

[32] Zhao, S., & Li, X. (2022). An Attention Encoder-Decoder Dual Graph Convolutional Network with Time Series Correlation for Multi-Step Traffic Flow Prediction. Journal of Advanced Transportation, 2022, 1-17. https://doi.org/10.1155/2022/7682274

[33] Zhao, S., Xing, S., & Mao, G. (2022). An Attention and Wavelet Based Spatial-Temporal Graph Neural Network for Traffic Flow and Speed Prediction. Mathematics, 10(19), 3507. https://doi.org/10.3390/math10193507

[34] Liao, L., Hu, Z., Zheng, Y., Bi, S., Zou, F., Qiu, H., & Zhang, M. (2022). An improved dynamic Chebyshev graph convolution network for traffic flow prediction with spatial-temporal attention. Applied Intelligence, 52(14), 16104-16116. https://doi.org/10.1007/s10489-021-03022-w

[35] Wang, Y., Jing, C., Xu, S., & Guo, T. (2022). Attention based spatiotemporal graph attention networks for traffic flow forecasting. Information Sciences, 607, 869-883. https://doi.org/10.1016/j.ins.2022.05.127

[36] Zhang, C., Zhou, H. Y., Qiu, Q., Jian, Z., Zhu, D., Cheng, C., He, L., & Liu, G. (2022). Augmented Multi-Component Recurrent Graph Convolutional Network for Traffic Flow Forecasting. ISPRS International Journal of Geo-Information, 11(2), 88. https://doi.org/10.3390/ijgi11020088