2024 IEEE 27th International Conference on
Intelligent Transportation Systems (ITSC)
September 24- 27, 2024. Edmonton, Canada

2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC) | 979-8-3315-0592-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ITSC58415.2024.10920069

A GAT-BiLSTMA Model for Weather-Aware Prediction of Traffic Speed*

Bikis A. Muhammed[1], Ali R. Hurson[1], Sahra Sedigh Sarvestani[1], and Lasanthi Gamage[2]

Abstract— **This paper presents a method for incorporating the effect of weather conditions in prediction of the average speed of vehicular traffic for each segment of a road network. The proposed approach utilizes two different deep learning methods: graph attention networks and bidirectional long short-term memory with attention layers. The accuracy of predictions is increased by considering the real-world driving distance between road segments, in contrast to the haversine distance used in several existing prediction methods. Categorization of input data as weekend or weekday further increased the prediction accuracy. The proposed approach was validated with two data sets published by the California Department of Transportation, PeMSD4 and PeMSD7. One year of traffic data was supplemented with weather data and used to predict the average traffic speed of each road segment for up to 60 minutes into the future. The method was shown to maintain accuracy over multiple time horizons, scale well with respect to the number of road segments, and outperform existing prediction methods in prediction accuracy.**

I. INTRODUCTION

Countries across the world spend and lose billions of dollars annually on traffic-related issues such as accidents and property damage. This is in addition to injuries, loss of life, and environmental damage due to unnecessary CO_2 emissions [1]. A number of these problems can be proactively mitigated with accurate, timely, and robust traffic forecasting. This in turn requires consideration of respective spatial and temporal aspects of traffic on a larger scale. Given the considerable impact of weather on driving conditions and trends, it is not surprising that prediction methods that consider weather have been found to be more accurate [2].

Deep learning frameworks can recognize complex patterns in past traffic data and have been used to this end in a number of studies on traffic prediction [3]–[8]. For example, long short-term memory (LSTM) networks have been used because of their ability to learn from long-term, dynamic, and complex traffic patterns. Also notable is the graph neural network (GNN) method, which is particularly effective in recognizing complex patterns in spatial data.

While promising, existing GNN-based methods for traffic prediction have a number of limitations. The use of the haversine distance measure instead of actual driving distance is one example. Very few consider the effects of weather, and if they do, their predictions are based on traffic data from a very limited area and short time duration. For example, the methods proposed by Zhao et al. and Ge et al., respectively, were based on two and three months of traffic data [9], [10]. Both studies use the haversine distance measuring technique. Recognizing patterns such as seasonality can increase the accuracy of traffic prediction, but these patterns rarely manifest in a limited scope.

The original research contribution of this paper is a method for accurate prediction of the average speed of traffic on each segment of a road network, with consideration of the effect of weather conditions. As compared to existing methods, the proposed approach considers a longer duration of past traffic data and is able to maintain prediction accuracy for a longer time horizon into the future. Consideration of weather conditions, use of real driving distance, and learning from respective weekend and weekday traffic patterns are notable distinctions of our method. As an enhancement to GNN and LSTM networks, we incorporate respective attention layers to focus on and learn from important spatial and temporal observations. We demonstrate and validate our approach by applying it to traffic data sets from Oakland and Los Angeles, respectively. As a final contribution, we have carried out ablation experiments to confirm that each layer of the proposed approach plays a role in increasing prediction accuracy of the method.

The remainder of this document is organized as follows. In Section II, we present a review of related literature. A detailed description of the proposed method and its parts is presented in Section III. Section IV presents the results of validation and evaluation of the proposed approach. Section V concludes the paper and discusses future extensions to the research.

II. RELATED WORK

Vehicular traffic prediction methodologies can be classified into statistical/probabilistic and machine learning-based methods. Statistical or probabilistic methods usually rely on a single feature for learning patterns and forecasting future events. Machine learning methods, on the other hand, are more comprehensive and rely on multiple features. Figure 1 depicts our taxonomy of related work in vehicular traffic prediction.

A. Statistical and Probabilistic Methods

It was developed to represent patterns in less complex traffic data. These methods are not typically applied to prediction based on more than one feature, and rarely work well with data sets that have complex relationships between

*The authors gratefully acknowledge funding from NSF DUE-1742523.
[1]Bikis A. Muhammed, Ali R. Hurson, and Sahra Sedigh Sarvestani are with the Missouri University of Science and Technology, Rolla, MO 65409, USA {bmpb8, hurson, sedighs}@mst.edu
[2]Lasanthi Gamage is with Webster University, lasanthigamage67@webster.edu

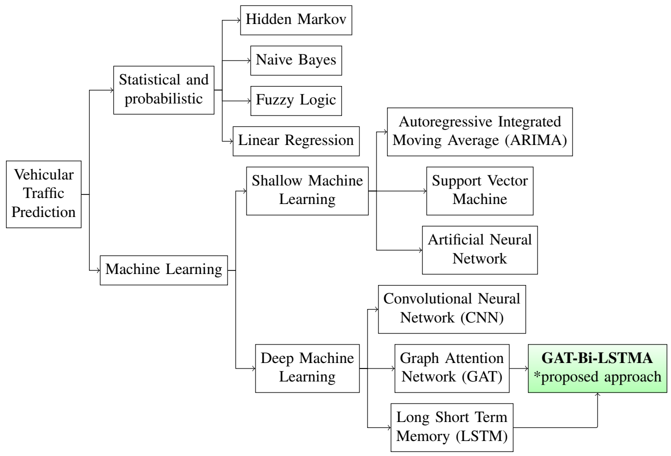

Fig. 1. Taxonomy of Related Literature

data points [10]. Notable studies have applied Markov [11], naive Bayes [12], fuzzy logic [13], and linear regression [14] techniques to traffic prediction.

B. Machine Learning Methods

Machine learning methods are capable of considering non-linear relationships between data points, such as spatial and temporal interdependencies [15]. These techniques can be further sub-classified into shallow and deep learning methods.

Shallow machine learning models have been used in traffic prediction to enable consideration of more complex relationships between attributes such as seasonality of traffic patterns. Notable examples have applied the autoregressive integrated moving average (ARIMA) [16], support vector machines [17], and artificial neural networks [18].

1) Deep Machine Learning Methods: Statistical, probabilistic, or shallow machine learning methods are rarely effective for predictions that involve time series, non-linear relationships, or longer time dependencies [19]. Deep learning models, on the other hand, can identify non-linear, complex, and long-term relationships in a data set. Long short-term memory networks [20]–[22] and convolutional neural networks (CNN) [20], [22], [23] are examples of deep learning methods that have been used for traffic prediction. Also notable are graph neural/attention networks, which can consider the interdependence between traffic sensors in road traffic networks [6]–[8], [21].

The closest study to our proposed approach is work by Wei et al., where a spatio-temporal causal graph attention network (STCGAT) has been utilized for traffic prediction [9]. This study considers three months of traffic data as input, in contrast to the one-year duration of our input data. Weather conditions are not considered in [9], nor is real driving distance between points.

Table I compares our proposed approach with the closest studies identified in related literature.

III. Methodology

Our method predicts the average vehicle speed for each road segment of a given road network, based on data from past traffic on the same network and with consideration of weather conditions, which can take one of six values: partially cloudy, overcast, clear, rain, rain and partially cloudy, and rain and overcast. Six predictions are generated for each road segment - one for each potential weather condition. In conjunction with the weather forecast, our method enables a more accurate prediction of traffic. Figure 2 depicts a high-level overview of our proposed approach, where given n past observations: $\{x_{t-n}, ..., x_{t-3}, x_{t-2}, x_{t-1}\}$, the goal is to predict m future values: $\{x_t, x_{t+1}, x_{t+2}, x_{t+3}, ..., x_{t+m}\}$. In this notation, x_t represents a tuple that includes the average speed and weather conditions for a given sensor at time t.

The remainder of this section articulates the specifics of our approach.

A. Representation of Road Network

We represent the road network as a graph, where each node represents a roadside traffic sensor that records the average traffic speed. Two nodes are connected by an edge if the real driving distance between the corresponding sensors is less than a specified threshold. We used multiple threshold values, and the threshold values were selected based on whether the values did not create a sparse spatial adjacency matrix value after trial and error. The weight of the edge is the real driving distance between the endpoints, defined as the shortest route that can be driven on contiguous road segments.

The use of real driving distance is an important distinction of our method. Both of the two methods closest to the proposed approach, [9] and [10], use the haversine distance, which represents the shortest distance between two points along the surface of a sphere, and is calculated using the latitudes and longitudes. Outside trivial distances, the haversine route is unlikely to exist in reality as a driving route. Figure 3 illustrates the difference between the two measures.

The real driving distance was computed using the Bing Distance Matrix API [24]. The piece-wise function proposed by [7] and shown in Equation 1 was used to construct a weighted adjacency matrix, $\mathbf{W}$, representing the road network graph.

$$w_{ij} = \begin{cases} e^{\frac{-d_{ij}^2}{\sigma^2}} & \text{if } i \neq j \text{ and } e^{\frac{-d_{ij}^2}{\sigma^2}} \geq \epsilon \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

In Equation 1, d_{ij} represents the distance between sensors i and j and σ and ϵ are threshold values used to ensure sparsity of the adjacency matrix by creating edges only between nodes that are in close physical proximity.

B. Input Data

The fundamental inputs to our method are the adjacency matrix described above and data representing past traffic and weather condition observations. Our experiments were conducted using the Performance Measurement System

TABLE I

COMPARISON OF TRAFFIC PREDICTION METHODS

Method	Max duration of input data	Consideration of weather	Distance measure	Future time horizon	Attention
STCGAT [9]	2 to 3 months	Yes	Haversine	5 to 60 min	Spatial
T-GCN [10]	3 months	No	Haversine	5 to 30 min	None
Proposed approach	1 year	Yes	Real-world	5 to 60 min	Spatial and Temporal

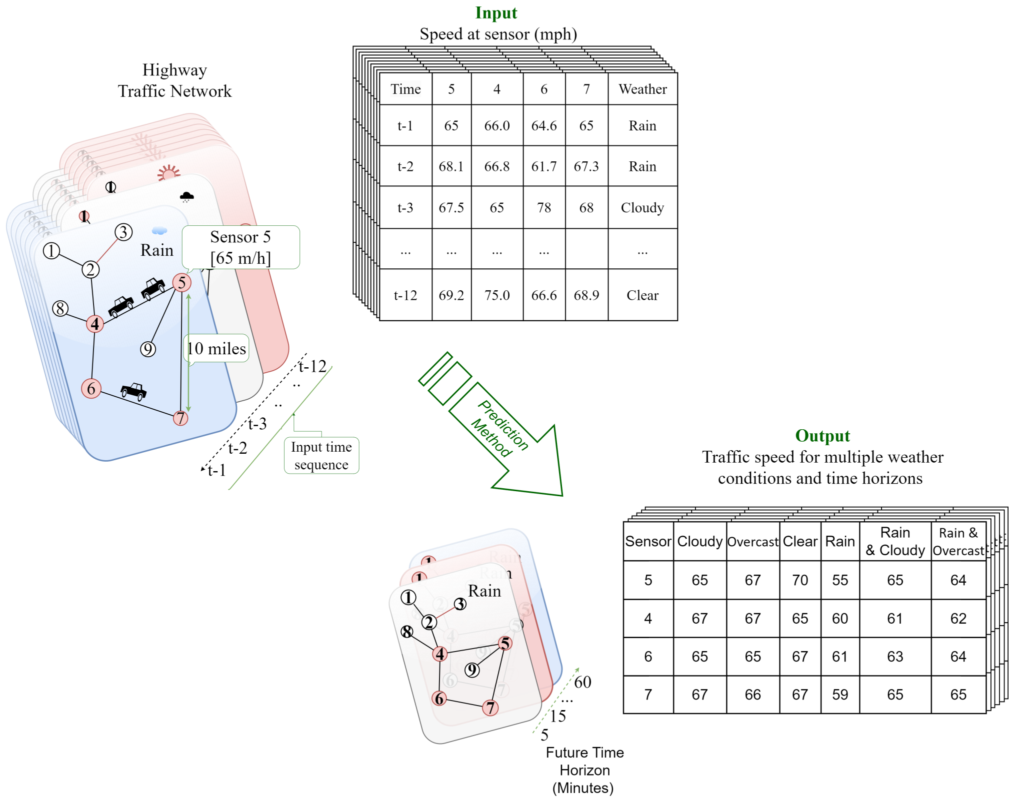

Fig. 2. High-Level View of Traffic Prediction

Fig. 3. Haversine vs. Real Driving Distance

(PeMS) data set published by the California Department of Transportation (Caltrans) [25]. More specifically, our input data is composed of the average traffic speed observed during the year 2022 by respective roadside sensors installed on major highways in California Districts 4 (Oakland, 55.8 square miles) and 7 (Los Angeles, 468.7 square miles). In the remainder of this paper, we refer to the Oakland data as PeMSD4 and the Los Angeles data as PeMSD7. The weather conditions at each observation were determined using the Visual Crossing Weather API [26]. The traffic and weather data for each location were fused during pre-processing.

C. Prediction

The proposed approach operates on discrete time slots of five minutes each. Given the average traffic speed and weather conditions for each node of the road network for the past twelve time slots (60 minutes), we predict the average traffic speed of each node under each of the six weather conditions.

of traffic sensors that are in proximity. Our addition of a GAT aims to give priority to sensors that have significant influence on a given sensor's traffic data.

The general structure of our proposed prediction method is depicted in Figure 4. Three stages are involved, data pre-processing, processing of spatial information, and learning from temporal patterns, respectively. A graph attention layer (GAT) is a GNN-based method that we apply directly to the road network graph to focus on the most important nodes of the graph. The LSTM layer learns from temporal patterns of traffic, and it is used to predict multiple future time horizons, i.e., 5, 15, 30, 45, and 60 minutes into the future. The remainder of this section articulates additional detail about each stage of the prediction.

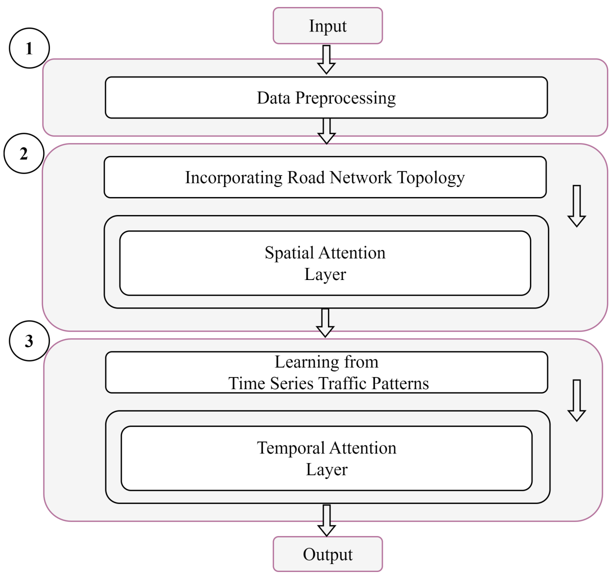

Fig. 4. Overview of the Proposed Approach

1) Data Pre-Processing: Data pre-processing includes normalization, feature extraction, and fusing traffic and weather data sets. We have used time series imputation and z-score normalization techniques to update missing values and normalize the data set to a mean of zero and a standard deviation of one. The former is to mitigate the effect of outliers [7]. Feature extraction includes removing unnecessary traffic sensor information from the data set. In addition, the respective traffic and weather data set were merged based on date and time. The data sets were split into respective subsets for training (60%), validation (20%), and testing (20%).

2) Incorporating Road Network Topology: Most GNN-based road traffic prediction approaches use a weighted adjacency matrix to capture spatial information between neighboring nodes or traffic sensors. To compute the distance between traffic sensors and build a representative adjacency matrix, we studied both Euclidean and non-Euclidean measures of distance, such as the haversine distance formula; the taxicab python module, which tries to use both driving and walking distances between two locations; Google distance matrix API; and Bing Distance Matrix API. The haversine distance computation method was faster than other methods; however, it computes a distance that may not be drivable.

Motivated from convolutional neural networks, a graph convolutional network (GCN) aggregates traffic information of traffic sensors that are in proximity. Our addition of a GAT aims to give priority to sensors that have significant influence on a given sensor's traffic data.

3) Bi-LSTM Layer: One advantage of using the LSTM layer for time series prediction is that it can capture long-term patterns in time series. To improve the efficiency, we utilize a bidirectional LSTM (Bi-LSTM) layer, where the input is processed in both the forward direction and the backward direction, allowing inferences to be made from data points both before and after a given point in time. The predicted output is the average of these two outputs.

4) Time Series Attention Layer: Some outputs of the Bi-LSTM model layer may be insignificant, i.e., have little or no effect on the predicted value. Duplicates may exist among the outputs. We have utilized a time series attention layer in an effort to focus on important values of the time series instead.

IV. Evaluation and Discussion of Results

We have evaluated our proposed prediction methodology using multiple criteria, including the future time horizons, and number of road segments, among others. The results showed consistent accuracy. The aim of the assessment is twofold - to determine whether incorporating a feature will improve accuracy or not (e.g., categorization of weekend vs. weekday data) and to assess the accuracy of the model with different settings (e.g., number of road segments). Moreover, the assessment revealed that our prediction technique was not affected by sudden changes in the traffic state due to inclement weather, rush hour congestion, special events, or holidays.

Our model was implemented using the TensorFlow Python framework, using the Adam optimizer with a learning rate of 0.01 and mean-square error as a performance measure for training. Unlike previously proposed traffic prediction methods, we have used one-year of traffic and weather data for training and testing. We randomly sampled traffic sensors and formed a group of 232 traffic sensors that are in proximity to both PeMSD4 and PeMSD7 data sets for most of the evaluation criteria. The duration of input sequences was kept to 12 or 60 minutes, and the output sequences to 3 or 15 minutes. The CPU and GPU environments shown in Table II were used during both training and evaluation.

TABLE II
Experimental Environments

System	Windows 11
CPU	Intel(R) Core(TM) i7-1185G7 @ 3.00GHz
GPU	A100 Nvidia Tesla T4 (Google Colab PRO)

A. Measures of Prediction Accuracy

The accuracy of our method was assessed using three different measures: mean absolute error (MAE), root-mean-square error (RMSE), and mean absolute percentile error (MAPE). These measures are computed as in Equations 2, 3, and 4, respectively.

$$\text{mean absolute error (MAE)} = \frac{1}{n}\sum_{t=1}^{n}|\tilde{y} - y| \quad (2)$$

$$\text{root mean square error (RMSE)} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(\tilde{y} - y)^2} \quad (3)$$

$$\text{mean absolute percentage error (MAPE)} = \frac{100}{n}\sum_{t=1}^{n}\frac{|y - \tilde{y}|}{y} \quad (4)$$

MAE and MAPE measure the average magnitude of errors, while RMSE computes the standard deviation of the magnitude of errors. MAE and MAPE are less sensitive to outliers and scale-independent, but they penalize positive errors rather than negative errors [27]. Moreover, MAE and MAPE are also easily influenced by small time series observations - this has been visible in some of our testing accuracy results below.

B. Influence of Haversine Distance Measure

We have selected 250 road traffic sensors (each of which corresponds to one road segment) and built a distance matrix using both haversine and real-world driving distance measurement techniques. Similar threshold values were applied to construct an adjacency matrix for each measure. During exploration, the distance matrix built by the haversine distance measure resulted in an extremely sparse adjacency matrix for the PeMSD4 data set. In contrast, the adjacency matrix for PeMSD7 is very dense.

Figure 5 and Table III compare the prediction accuracy achieved with the two distance measures for the PeMSD4 data set. Table IV compares the results for PeMSD7. The real-world driving distance yields more accurate predictions in both cases.

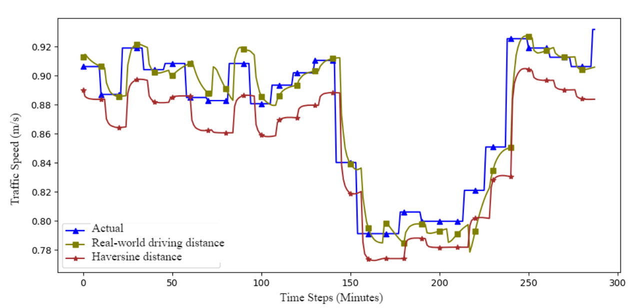

Fig. 5. Prediction Results for Different Distance Measures (PeMSD4)

TABLE III
PREDICTION ACCURACY VS DISTANCE MEASURES (PEMSD4)

PeMSD4		Prediction Accuracy		
Type	Edges	MAE	RMSE	MAPE
Haversine	1,371	0.003	0.052	4.62%
Real-world	33,204	0.003	0.052	4.87%

TABLE IV
PREDICTION ACCURACY VS DISTANCE MEASURES (PEMSD7)

PeMSD7		Prediction Accuracy		
Type	Edges	MAE	RMSE	MAPE
Haversine	42,158	0.003	0.057	6.84%
Real-world	30,229	0.003	0.056	6.00%

C. Influence of Categorization of Weekday/Weekends

Daytime road traffic patterns on weekdays and weekends are completely different from each other. To visualize this difference, we split the PEMSD7 data set into respective weekday and weekend data sets. As shown in Figure 6, traffic speed on weekdays is generally slower than on the weekend. The local and absolute minimum of the weekday traffic line correspond to the morning and evening rush hours, which do not appear in the weekend traffic speed line.

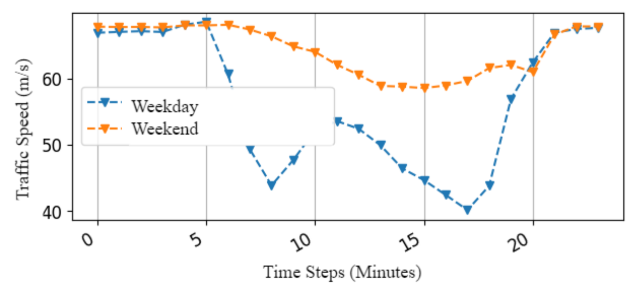

Fig. 6. Traffic Speed (m/h) on Weekday vs Weekend (PeMSD7)

To increase the accuracy of predictions, we added respective category labels to weekend and weekday traffic data. Table V and Figure 7 demonstrate that categorization yielded the intended effect and significantly increased prediction accuracy.

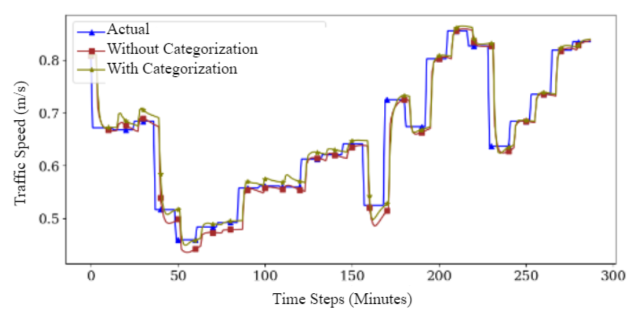

Fig. 7. Prediction Results with/without Categorization of Days (PeMSD4)

D. Effect of Future Time Horizon

The future time horizon refers to the duration for which values are predicted. The accuracy of the proposed approach was evaluated for multiple future time horizons, specifically, 5, 15, 30, 45, and 60 minutes, while maintaining the duration of the input at 60 minutes. The results depicted in Figure 8 and tabulated in Table VI demonstrate that reasonable accuracy is maintained as far as 60 minutes into the future.

TABLE V

Prediction Accuracy with/without Categorization

	PeMSD4			PeMSD7		
Categorization	MAE	RMSE	MAPE	MAE	RMSE	MAPE
Without	0.003	0.052	38.00%	0.011	0.109	55.00%
With	0.002	0.049	5.00%	0.004	0.061	15.80%

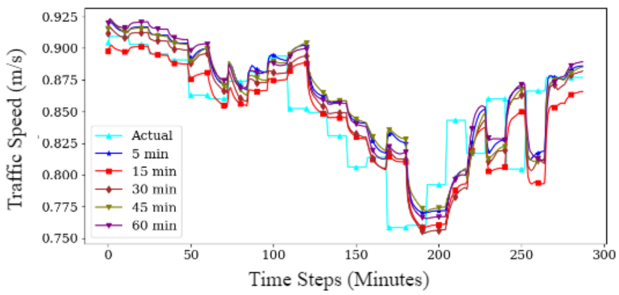

Fig. 8. Prediction Results for Different Time Horizons (PeMSD4)

E. Scalability of The Proposed Model

To evaluate the scalability of the proposed approach, we assessed the prediction accuracy for an increasing number of road segments. Table VII and Figure 9 attest to scalability of the model.

F. Improved Accuracy with Consideration of Weather

Weather conditions have a significant and direct impact on vehicular traffic speed [2]. For instance, Figure 10 shows that mean traffic speed on rainy days of the year is lower than clear or sunny days. Categorical weather conditions such as partially cloudy, overcast, clear, rain, and rain & partially cloudy, and rain & overcast were considered for evaluation.

To incorporate the effect of weather conditions, we have augmented the traffic data set with the weather condition attribute with during data processing, so the GAT layer considers the weather information during aggregation of spatial traffic data. Table VIII and Figure 11 confirm that consideration of weather improves prediction accuracy. It is worth noting that the gains in accuracy may not be significant, despite the increased complexity of pre-processing. However, consideration of weather will make the prediction model more robust to sudden changes to traffic due to weather conditions.

TABLE VI

Prediction Accuracy vs Time Horizon

	PeMSD4			PeMSD7		
Horizon	MAE	RMSE	MAPE	MAE	RMSE	MAPE
5 min	0.001	0.032	2.70%	0.001	0.036	3.40%
15 min	0.002	0.050	3.90%	0.004	0.065	8.10%
30 min	0.005	0.068	6.20%	0.007	0.085	9.60%
45 min	0.007	0.083	11.00%	0.014	0.117	2.11%
60 min	0.009	0.096	10.10%	0.017	0.130	1.95%

TABLE VII

Prediction Accuracy vs Numbers of Road Segments

	PeMSD4			PeMSD7		
Segments	MAE	RMSE	MAPE	MAE	RMSE	MAPE
100	0.003	0.054	5.20%	0.003	0.053	5.96%
250	0.003	0.052	4.61%	0.003	0.051	5.66%
465	0.003	0.051	4.14%	0.003	0.054	8.54%

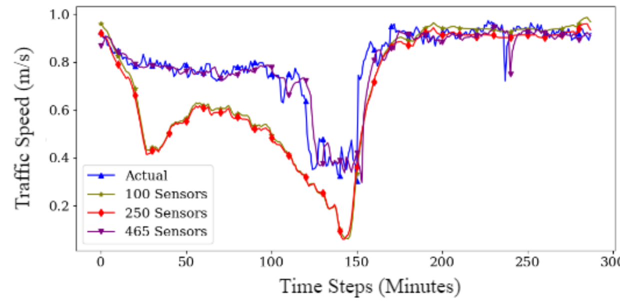

Fig. 9. Prediction Results for Different Numbers of Road Segments (PeMSD7)

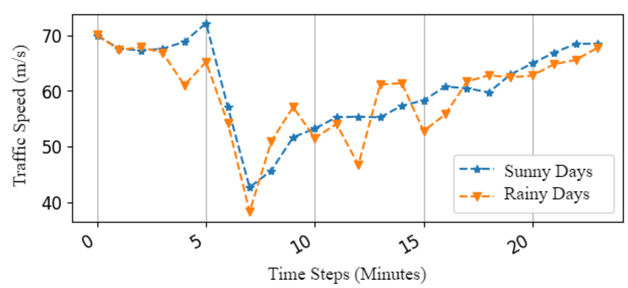

Fig. 10. Effect of Rain on Traffic Speed (PeMSD7)

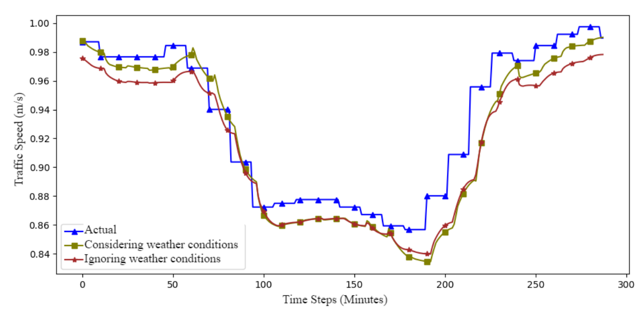

Fig. 11. Prediction Results with/without Consideration of Weather Conditions

G. Comparison with Other Prediction Methods

Among GNN-based traffic prediction methods, the spatio-temporal causal graph attention network (STCGAT) [9] is the closest to our proposed approach. In this section, we compare our results to those of this method and other alternatives

TABLE VIII

PREDICTION ACCURACY WITH AND WITHOUT CONSIDERATION OF WEATHER

	PeMSD4			PeMSD7		
Type	MAE	RMSE	MAPE	MAE	RMSE	MAPE
With weather	0.24	0.48	1.32%	0.18	0.43	1.22%
Without weather	0.27	0.53	3.10%	0.25	0.50	1.90%

from related literature, specifically reinforcement learning-designed LSTM (RL-LSTM), diffusion convolutional recurrent neural network (DCRNN), attention based spatial-temporal graph convolutional networks (ASTGCN), and spatial-temporal fusion graph neural networks (STFGNN).

To match their experimental settings/parameters and make our research consistent, we have used only the PeMSD4 data set. We have partitioned the data set using a sliding window of one hour, which means each hour of past traffic data is used to predict the next hour of traffic, with some overlapping inputs. The prediction method has a hidden layer size of 64, a multi-head attention size of 3, and an Adam optimizer with a learning rate of 0.001. Table IX compares the accuracy of our method to the aforementioned methods from related literature for the PeMSD4 data set and shows that our method is more accurate.

TABLE IX

COMPARISON OF PREDICTION ACCURACY FOR PEMSD4

	PeMSD4		
Model	MAE	RMSE	MAPE
RL-LSTM	25.01	41.42	16.18%
DCRNN	21.22	33.44	14.17%
ASTGCN	22.03	34.99	14.59%
STFGNN	20.18	32.41	13.94%
STCGAT	19.21	31.12	12.36%
GAT-Bi-LSTMA	**10.45**	**32.32**	**10.3%**

H. Ablation Experiments

An ablation experiment aims to ascertain the effect of each layer of a computational model, by removing one layer at a time. Our ablation experiment included the following scenarios:

- Without bidirectional LSTM (unidirectional LSTM, with a single LSTM layer to capture long-term temporal patterns)
- Without time series attention
- Without GAT (using a graph convolution layer for processing spatial information)
- Without multiple attention heads (using a single attention (self-attention) layer during spatial information aggregation)

Table X summarizes the results of the ablation experiment for PeMSD4 and attests to the contribution of each layer has contributed to the overall accuracy of the method.

TABLE X

RESULTS OF ABLATION EXPERIMENTS ON PEMSD4

	PeMSD4		
Model	MAE	RMSE	MAPE
without bidirectional LSTM	0.24	0.5	30.2%
without time series attention	0.28	0.53	22.5%
without graph attention network	0.29	0.53	34.7%
without multiple attention heads	0.26	0.51	16.4%
GAT-Bi-LSTMA	**0.19**	**0.43**	**9.7%**

V. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a machine learning model for prediction of the average speed of road traffic. The model considers weather conditions and weekday/weekend traffic patterns, and utilizes real-world driving distance instead of the more common (and less accurate) haversine distance. In the first two layers of the proposed model, we use a GAT to select only important neighboring nodes during aggregation of spatial data. This is followed by a bidirectional long short-term memory (Bi-LSTM) layer for learning from temporal traffic patterns from both input to output and output to input directions. In addition, we have used the time series attention layer to focus on only significant time series observations. We evaluated our prediction method on two PeMS data sets for multiple sensors and future time horizons, and the method showed consistently accurate results.

The proposed method has two limitations. The first limitation is that the model was trained on a single geographical area. The second limitation is the difference in granularity between the traffic and weather data sets.

In future work, we will extend the model to predict other attributes of road traffic, including congestion and occupancy. We also plan to refine the model by tuning the hyperparameters.

REFERENCES

[1] N. Kumar and M. Raubal, "Applications of deep learning in congestion detection, prediction and alleviation: A survey," *Transportation Research Part C: Emerging Technologies*, vol. 133, no. 103432, Dec. 2021.

[2] A. Nigam and S. Srivastava, "Hybrid deep learning models for traffic stream variables prediction during rainfall," *Multimodal Transportation*, vol. 2, no. 1, p. 100052, 2023.

[3] A. Derrow-Pinion, J. She, D. Wong, O. Lange, T. Hester, L. Perez, M. Nunkesser, S. Lee, X. Guo, B. Wiltshire, P. W. Battaglia, V. Gupta, A. L. asnd Zhongwen Xu, A. Sanchez-Gonzalez, Y. Li, and P. Veličković, "ETA prediction with graph neural networks in google maps," in *Proc 30th ACM Intl Conf on Information and Knowledge Management (CIKM2021)*, Virtual Event, Queensland, Australia, Aug. 2021, p. 3767–3776.

[4] Y. Fan, M. Ju, C. Zhang, and Y. Ye, "Heterogeneous temporal graph neural network," in *Proc SIAM International Conference on Data Mining (SDM)*, Oct. 2021, p. 657–665.

[5] W. Luo, H. Zhang, X. Yang, L. Bo, X. Yang, Z. Li, X. Qie, and J. Ye, "Dynamic heterogeneous graph neural network for real-time event prediction," in *26th ACM SIGKDD Intl Conf on Knowledge Discovery & Data Mining*, Virtual Event, CA, USA, Aug. 2020, p. 3213–3223.

[6] Z. Shao, Z. Zhang, F. Wang, and Y. Xu, "Pre-training enhanced spatial-temporal graph neural network for multivariate time series forecasting," in *Proc of the 28th ACM SIGKDD Conf on Knowledge Discovery and Data Mining*, New York, NY, United States, Aug. 2022, pp. 567–1577.

[7] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proc Intl Joint Conf on Artificial Intelligence*, Jul. 2018, pp. 3634–3640.

[8] W. Zhong, Q. Suo, X. Jia, A. Zhang, and L. Su, "Heterogeneous spatio-temporal graph convolution network for traffic forecasting with missing values," in *Proc IEEE Intl Conf on Distributed Computing Systems (ICDCS)*, DC, USA, Oct. 2021, pp. 707–717.

[9] W. Zhao, S. Zhang, B. Zhou, and B. Wang, "A spatio-temporal causal graph attention network for traffic flow prediction in intelligent transportation systems," *PeerJ. Computer science*, vol. 9, no. 8, pp. 1–9, 2023.

[10] L. Ge, H. Li, J. Liu, and A. Zhou, "Temporal graph convolutional networks for traffic speed prediction considering external factors," in *Proc IEEE Intl Conf on Mobile Data Management (MDM)*, vol. 2019-June, Hong Kong, China, 2019, pp. 234–242.

[11] J. F. Zaki, A. Ali-Eldin, S. E. Hussein, S. F. Saraya, and F. F. Areed, "Traffic congestion prediction based on Hidden Markov Models and contrast measure," *Ain Shams Engineering J*, vol. 11, pp. 535–551, Sep. 2020.

[12] G. Wang and J. Kime, "The prediction of traffic congestion and incident on urban road networks using Naive Bayes classifier," in *Australasian Transport Research Forum*, Melbourne, Australia, Nov. 2016, pp. 1–14.

[13] T. Iokibe, N. Mochizuki, and T. Kimura, "Traffic prediction method by fuzzy logic," in *Proc IEEE Intl Conf on Fuzzy Systems*, vol. 2, San Francisco, USA, Aug. 1993, pp. 673 – 678.

[14] H. Sun, H. X. Liu, H. Xiao, R. R. He, and B. Ran, "Use of local linear regression model for short-term traffic forecasting," *J of the Transportation Research Record*, vol. 1836, no. 1, pp. 143–150, 2003.

[15] Z. Lu, W. Lv, Y. Cao, Z. Xie, H. Peng, and B. Du, "LSTM variants meet graph neural networks for road speed prediction," *Neurocomputing*, vol. 400, pp. 34–45, 2020.

[16] T. Alghamdi, K. Elgazzar, M. Bayoumi, T. Sharaf, and S. Shah, "Forecasting traffic congestion using arima modeling," in *Proc Intl Wireless Communications & Mobile Computing Conf (IWCMC)*, Tangier, Morocco, Jul. 2019, pp. 1227–1232.

[17] X. Gu1, T. Li, Y. Wang, L. Zhang, Y. Wang, and J. Yao, "Traffic fatalities prediction using support vector machine with hybrid particle swarm optimization," *J of Algorithms and Computational Technology*, vol. 12, pp. 20–29, 2018.

[18] B. Sharma, S. Kumar, P. Tiwari, P. Yadav, and M. I. Nezhurina, "Ann based short-term traffic flow forecasting in undivided two lane highway," *J of Big Data*, vol. 5, no. 48, Dec. 2018.

[19] A. A. Kashyap, S. Raviraj, A. Devarakonda, S. R. N. K, S. K. V, and S. J. Bhat, "Traffic flow prediction models – A review of deep learning techniques," *Cogent Engineering*, vol. 9, no. 1, p. 2010510, 2022.

[20] Z. Yuan, X. Zhou, and T. Yang, "Hetero-ConvLSTM: A Deep Learning Approach to Traffic Accident Prediction on Heterogeneous Spatio-Temporal Data," in *Proc 24th ACM SIGKDD Intl Conf on Knowledge Discovery & Data Mining*, Jul. 2018.

[21] F. Hou, Y. Zhang, X. Fu, L. Jiaoa, and W. Zheng, "The prediction of multistep traffic flow based on AST-GCN-LSTM," *J of Advanced Transportation*, p. 1–10, Dec. 2021.

[22] Z. Zhao, Z. Li, F. Li, and Y. Liu, "CNN-LSTM based traffic prediction using spatial-temporal features," *J of Physics: Conf Series*, vol. 2037, no. 1, p. 012065, Sep. 2021.

[23] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang, "Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction," *Sensors for Transportation*, Jan. 2017.

[24] "Bing Distance Matrix API," https://www.microsoft.com/en-us/maps/bing-maps, 2023, accessed: 2024-06-01.

[25] California Department of Transportation, "CalTrans Performance Measurement System (PeMS)," https://pems.dot.ca.gov/, 2023, accessed: 2024-06-01.

[26] "Visual Crossing," https://www.visualcrossing.com/weather-data, 2023, accessed: 2024-06-01.

[27] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *Intl J of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006.