A Hybrid Deep Learning Model for Traffic Flow Prediction

Alper Talha Karadeniz[*1], **Yavuz Canbay**[2]

[*1] Samsun University Faculty of Engineering and Natural Sciences Software Department of Engineering, Samsun, Türkiye

[2] Department of Computer Engineering, Faculty of Engineering and Architecture, Sutcu Imam University, Kahramanmaras, Turkey

(Alınış / Received: 04.08.2024, Kabul / Accepted: 01.10.2024, Online Yayınlanma / Published Online: 30.12.2024)

Keywords
traffic flow,
prediction,
deep learning

Abstract: Urbanization has led to increase traffic issues, resulting in economic losses and accelerating environmental pollution, so reducing the quality of life for city residents. Proactively identifying traffic issues and reducing congestion is crucial. Traffic flow prediction involves the examination of historical and present traffic data to predict future traffic conditions. Precise predictions enable individuals to make informed choices on travel routes and methods of transportation. This minimizes travel duration and enhances comfort with traffic conditions. Traditional traffic prediction methods rely on statistical techniques or fundamental time series analysis. Nonetheless, these methods cannot adequately consider for the complex geographical and temporal relationships found in traffic data. Recent advancements in artificial intelligence have enabled the development of complex and data-driven models for prediction. This study introduces a novel hybrid model for predicting traffic flow using deep learning techniques. The proposed model employs a CNN for feature extraction and a BiLSTM for prediction. The effectiveness of the proposed hybrid model was evaluated against other baseline algorithms using a real-world dataset. The experiments indicated that the proposed hybrid model outperformed other models, achieving 5.1834 MAE, 54.4060 MSE, 7.3760 RMSE, and 0.9923 R^2 values. The results demonstrate the potential of deep learning methods in predicting traffic flow and offer direction for future studies.

Trafik Akış Tahmini İçin Hibrit Bir Derin Öğrenme Modeli

Anahtar Kelimeler
trafik akışı,
tahmin,
derin öğrenme

Öz: Kentleşme, trafik sorunlarının artmasına yol açmakta, bu durum yalnızca ekonomik kayıpları beraberinde getirmekle kalmayıp aynı zamanda çevresel bozulmaya da neden olarak kent sakinlerinin yaşam kalitesini tehdit etmektedir. Trafik sorunlarını tespit etmek ve trafik sıkışıklığını proaktif olarak azaltmak son derece önemlidir. Trafik akışı tahmini, gelecekteki trafik koşullarını öngörmek için geçmiş ve güncel trafik verilerinin analiz edilmesiyle yapılır. Kesin tahminler, insanların seyahat rotaları ve ulaşım yöntemleri konusunda bilinçli seçimler yapmasına olanak tanır. Bu durum, yolculuk süresini azaltır ve trafik koşullarından memnuniyet düzeyini artırır. Geleneksel trafik tahmin yöntemleri istatistiksel tekniklere veya basit zaman serisi analizine dayanır. Ancak bu analizler, trafik verilerindeki karmaşık mekânsal ve zamansal bağımlılıkları yakalama konusunda beklenileni vermemektedir. Yapay zekâdaki son gelişmeler, karmaşık ve veriye dayalı tahmin modellerinin oluşturulmasına olanak sağlamıştır. Bu makalede, derin öğrenme tekniklerini kullanarak trafik akışını tahmin etmek için yeni bir hibrit model önerilmektedir. Önerilen modelin performansı, gerçek dünya veri kümesi üzerinde bazı temel algoritmalara göre değerlendirilmiştir. Yapılan deneylerde önerilen hibrit modelin 5,1834 MAE, 54,4060 MSE, 7,3760 RMSE ve 0,9923 R^2 değerleriyle diğer modellerden daha iyi performans gösterdiği görülmüştür.

Sonuçlar, trafik akış tahmini alanında derin öğrenme yöntemlerinin potansiyelini vurgulamakta ve gelecekteki araştırmalar için yol gösterici olduğunu göstermektedir.

1. INTRODUCTION

The effective management of cities depends significantly on the existence of robust transportation systems, which grow more crucial with the acceleration of urbanization. Traffic congestion adversely impacts economic productivity and degrades the quality of life for city residents owing to environmental pollution. Therefore, it is essential to address possible concerns to prevent traffic congestion. There is an increasing interest in the advancement of innovative systems and methodologies for traffic flow prediction [1, 2].

Traffic flow prediction employs historical and current traffic data to forecast eventual conditions. Accurate forecasts allow individuals to make informed decisions regarding travel routes and travel methods, leading to decreased travel time and increased satisfaction concerning traffic conditions [3]. The forecast of traffic flow is a complex issue influenced by various factors, including road shape, weather conditions, traffic density, and human behavior. Conventional traffic prediction methods typically emphasize fundamental statistical analysis, which may insufficiently represent the complex spatiotemporal relationships present in traffic data. Recent breakthroughs in deep learning and machine learning can significantly improve traffic flow prediction by enabling the development of complex predictive models based on traffic data [4, 5].

Numerous studies on traffic flow prediction are present in the literature. Hao Peng et al. introduced a methodology for forecasting traffic flow over longer periods via dynamic graphs. The traffic network was illustrated through dynamic traffic flow probability graphs, and spatial information was retrieved by graph convolution. This approach combines Long Short-Term Memory (LSTM) units with graph convolution to capture temporal characteristics. Effective long-term traffic flow estimates were accomplished by applying the approach to urban cycling data from New York City [6]. Jiawei Wang et al. presented a road-based deep learning model that guarantees both reliability and interpretability inside the urban transportation sector. This method also improves traffic speed forecasting on a city scale. To improve the interpretability of traffic flow in the study, the road network was segmented. Subsequently, a Bidirectional Long Short-Term Memory Network (BiLSTM) is utilized to simulate each road, yielding several BiLSTM layers that incorporate temporal information. During the traffic prediction phase, the spatial-temporal characteristics derived from these processes are conveyed to a fully interconnected layer. Finally, the outcomes for each route are aggregated to determine the traffic speed across the network. The experiments confirm the model's efficiency [7].

This study presents a novel hybrid model utilizing deep learning for traffic flow prediction. The proposed model employs CNN algorithm for feature extraction and BiLSTM algorithm for flow prediction. The proposed model was evaluated against multiple deep learning models and machine learning methods. An experimental study was conducted utilizing a publicly available dataset. The algorithms' performances were assessed utilizing established regression metrics such as MSE, MAE, RMSE, and R^2. The findings indicated that the proposed model had the lowest error rates, with 5.1834 MAE, 54.4060 MSE, 7.3760 RMSE, and 0.9923 R^2 values. This work introduces a hybrid model that employs both CNN and BiLSTM for flow prediction.

The paper is organized as following. Section 2 explains the materials and methods utilized in the study. The proposed hybrid model was presented in Section 3. Section 4 shows the outcomes of the performed experimental studies and their results. Finally, Section 5 concludes the paper.

2. MATERIALS and METHODS

2.1. Dataset

Experimental studies were performed using a publicly available dataset from the Caltrans Performance Measurement System (PeMS) [8]. The dataset comprises traffic data from San Bernardino for July 2016. Detectors are located at 170 distinct locations. These detectors capture traffic data every five minutes. The dataset comprises three attributes: flow, occupancy, and speed. The flow variable in the dataset indicates the quantity of cars passing every five minutes. The occupancy variable indicates the time interval utilized by each vehicle. The speed variable represents the average velocity of vehicles passing within a 5-minute interval.

The dataset collection comprises 3,035,520 records from 170 detectors, each containing 17,856 data points. The data is scaled, standardized, and optimized through the use of data science engineering. Figures 1-3 demonstrate the histogram graphs for the variables of flow, occupancy, and speed, respectively [8].

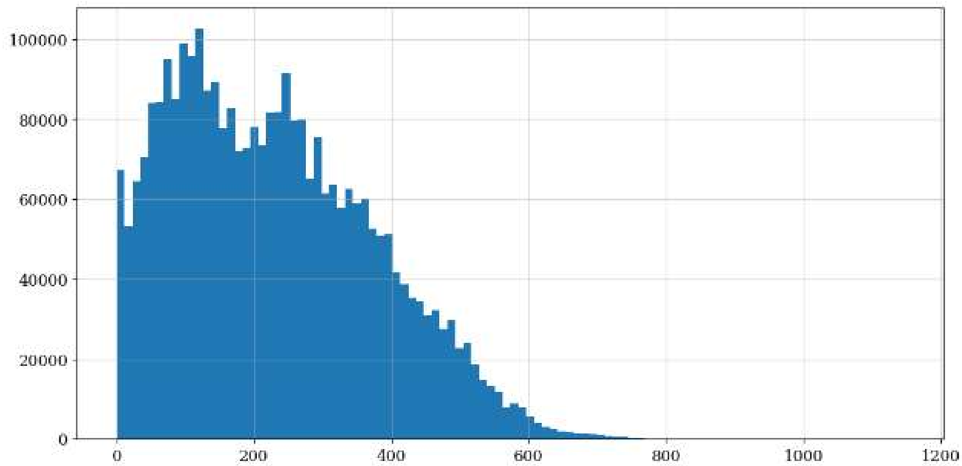

Figure 1. Histogram of flow data

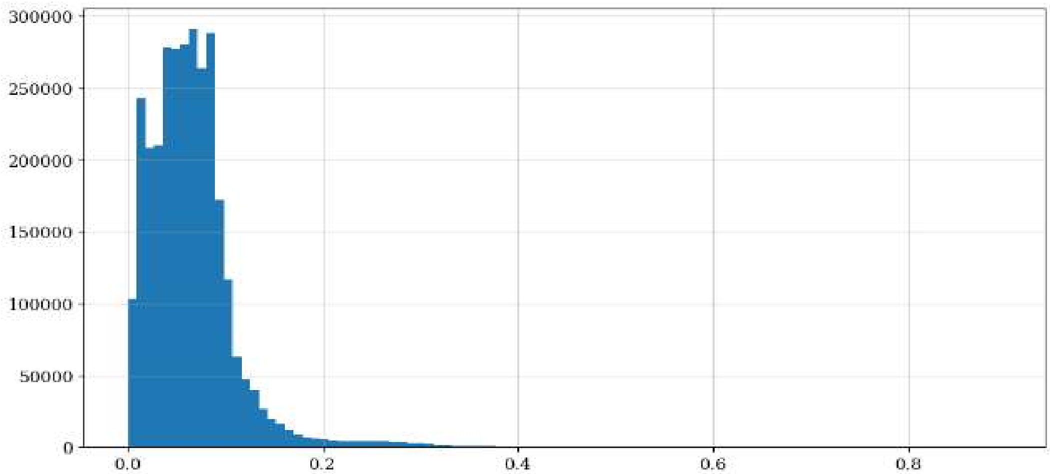

Figure 2. Histogram of occupancy data

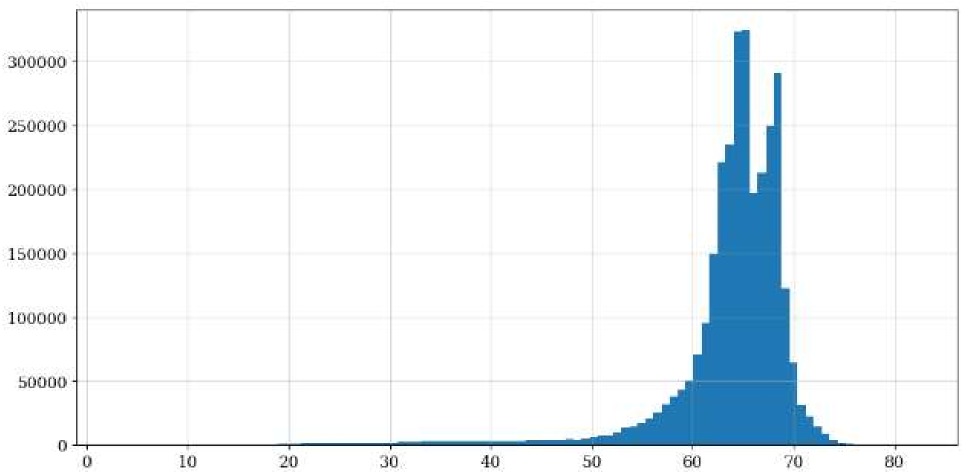

Figure 3 . Histogram of speed data

2.2. CNN

Convolutional Neural Networks (CNNs) have currently attained substantial popularity as a competitive kind of deep learning. These networks are constructed with artificial neural networks and have demonstrated significant success, especially in domains such as image and video processing, pattern recognition, and computer vision applications. CNNs are designed to adaptively acquire spatial hierarchies of features from input data via several layers. A standard CNN design comprises three primary types of layers: convolutional, pooling, and fully connected layers.

Convolutional Layer: This layer performs a mathematical operation known as convolution by applying a sequence of filters (or kernels) to the input data. Each filter identifies particular characteristics in the data, such as edges or textures.

Pooling Layer: This layer minimizes computational load, regulates overfitting, and enhances the model's invariance to minor changes or rotations in the input data.

Fully Connected Layer: fully connected layers combine the features acquired by previous levels to generate predictions.

The combination of these layers makes CNNs remarkable at extracting significant characteristics from large and complex datasets, hence enhancing their efficiency in tasks such as image classification, object recognition, and natural language processing [9, 10].

2.3. Multi Layer Perceptron

Multi Layer Perceptron (MLP), a feed-forward artificial neural network, employs backpropagation, a supervised learning approach, for network training. It performs tasks such as classification, regression, and pattern recognition, and is efficient at representing complex input-output interactions. MLP architectures enable the resolution of non-linear problems and the learning and representation of complex data structures [11].

2.4. Random Forest

Random Forest method is typically utilized for regression and classification tasks. To ensure optimal system performance during the training process, it is essential to generate a substantial number of decision trees. The class can be represented as the average prediction of each individual tree for regression tasks and the most frequent class for classification tasks. Each tree in the forest comprises a data sample, and every split in the tree evaluates a random subset of attributes, promoting diversity among trees and reducing the likelihood of overfitting. This method improves the precision and robustness of predictions relative to individual decision trees. This approach is an efficient method for managing large and complex datasets, correcting missing values, and evaluating the importance of variables. This method uses the formula presented in Equation 1 [12]. Here pi represents the relative frequency of the class observed in the data set and c represents the number of classes.

$$Entropy = \sum_{i=1}^{c} - Pi * log2(pi) \tag{1}$$

2.5. Linear Regression

Linear Regression (LR) is a popular regression method utilized to help explain the link between dependent and independent variables. The primary goal is to determine the ideal planar equation by examining the correlation between the independent variables in the dataset and the predicted value of the dependent variable. This method use the formula presented in Equation 2 [13]. In this equation, Y is the dependent variable and X is the independent variable, m is the estimated slope, and b is the estimated intercept.

$$Y = mX + b \tag{2}$$

2.6. Support Vector Regression

Support Vector Regression (SVR) is a known approach for addressing regression problems. The objective of the SVR algorithm is to precisely create a line that encompasses the majority of the data points under an epsilon threshold value. This algorithm seeks to identify the ideal hyperplane that maximizes the margin by focusing exclusively on points located outside the margin. SVR is particularly effective when the dataset comprises numerous dimensions and exhibits complex, non-linear relationships among variables. To mitigate nonlinearity, it employs several kernel functions. The linear SVR formula in Equation 3 is utilized [14].

$$y(X) = \sum_{1}^{n}(a_i - a_i^*). < X_i, X > +b \tag{3}$$

2.7. Long Short-Term Memory

Long Short-Term Memory (LSTM) networks are a specific kind of Recurrent Neural Networks (RNN) developed to tackle the issue of identifying long-term dependencies in sequential input. Traditional RNNs encounter difficulties in keeping information over long sequences, whereas LSTMs address this issue by integrating memory cells that can selectively store and control information. The memory cells are controlled by three key gates: the forget gate, which regulates the information to be discarded; the input gate, which determines the new information to be retained; and the output gate, which determines the information to be transmitted to the subsequent stage. This architecture enables LSTMs to preserve an internal state across time and more effectively capture long-term dependencies in data [15, 16].

2.8. Gated Recurrent Unit

The Gated Recurrent Unit (GRU) algorithm is a version of the RNN model designed to efficiently handle sequential input and reduce issues like vanishing gradients typically faced by conventional RNNs. It belongs to the family of LSTM network variants. The GRU algorithm demonstrates superior training speed compared to LSTMs and necessitates fewer parameters [17].

2.9. Bidirectional Long Short-Term Memory

Bidirectional Long Short-Term Memory (BiLSTM) network is an RNN variant designed to obtain contextual information from both historical and future states, in a sequential manner. LSTM models operate in a single direction. BiLSTMs are composed of two LSTM layers which one is used for processing the input data in the forward direction and the second one is for processing the data in the reverse direction. The bidirectional method, which involves comprehending the context from both perspectives, has a highly beneficial impact on the model's performance [18].

2.10. Performance Metrics

This study employs widely recognized regression metrics to assess the efficiency of the algorithms. The following metrics are summarized below.

R^2: is utilized in regression analysis to assess the level to which the model accurately represents the data by evaluating the error rate. A robust model fit is signified by values approaching 1 [19]. Equation 4 presents the formula for the metric.

$$R^2 = correlation(actual_i, predicted_i)^2 \tag{4}$$

MAE: is the average of the absolute differences between the expected and actual values [20]. Equation 5 indicates the formula for the MAE metric.

$$MAE = \frac{\sum_{i=1}^{N}|predicted_i - actual_i|}{N} \tag{5}$$

MSE: denotes the average of the squared differences between the expected and actual values. It measures the magnitude of errors and maximizes the impact for significant errors [20]. Equation 6 is the formula for the MSE metric.

$$MSE = \sum_{i=1}^{N}(predicted_i - actual_i)^2/N \tag{6}$$

RMSE: quantifies the magnitude of errors between actual and predicted values, calculated as the square root of MSE, as stated in Equation 7 [20, 21].

$$RMSE = \sqrt{\sum_{i=1}^{N}(predicted_i - actual_i)^2/N} \tag{7}$$

2.11. The Proposed Hybrid Model

This study presents a novel hybrid deep learning model that integrates CNN and BiLSTM architectures. The CNN model is used for feature extraction, while the BiLSTM model is applied for predictions. Figure 4 illustrates the detailed presentation of the proposed model.

In Figure 4, the CNN model processes the flow data using a sliding window approach to extract features. The dataset undergoes one-dimensional convolution with 64 filters and a kernel size of 2. The feature maps from the previous phase are subjected to max pooling with a pool size of 2. Finally, the CNN model is finalized by incorporating a flatten layer, which reformats the data for input into the BiLSTM layer. Two BiLSTM layers are employed in the BiLSTM section, succeeded by two dense layers for prediction purposes. The predicted flow data has been obtained.

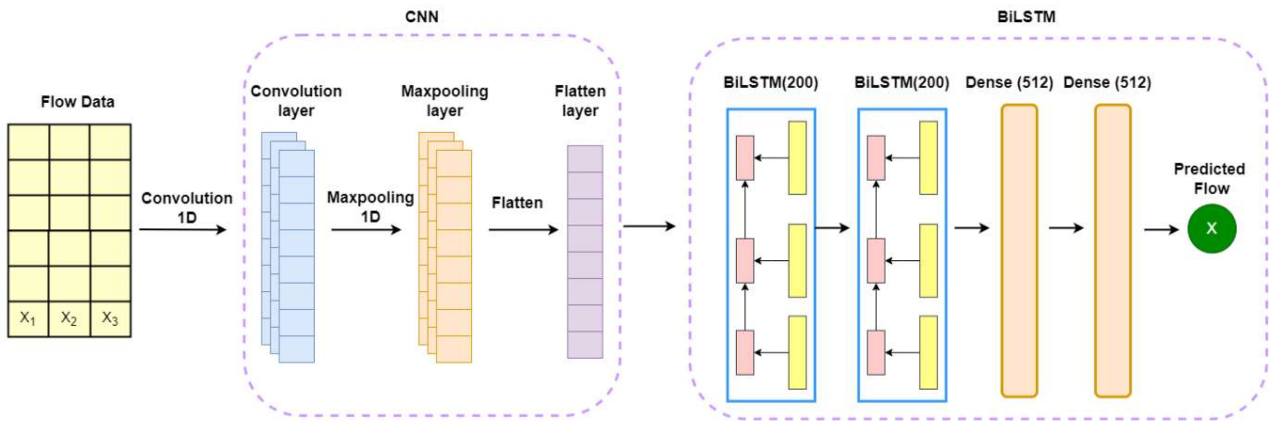

Figure 4. The proposed model

Sliding window method facilitates the transformation of time series tasks into supervised learning tasks. The time series is generally divided into short periods, aiming to predict the target value for each segment [22]. The sliding window technique, illustrated in Figure 5, converts the time series into data points including an output value and one or more input attributes. This transformation can anticipate future values by utilizing historical time series data. Figure 6 illustrates the architectural representation of the proposed model.

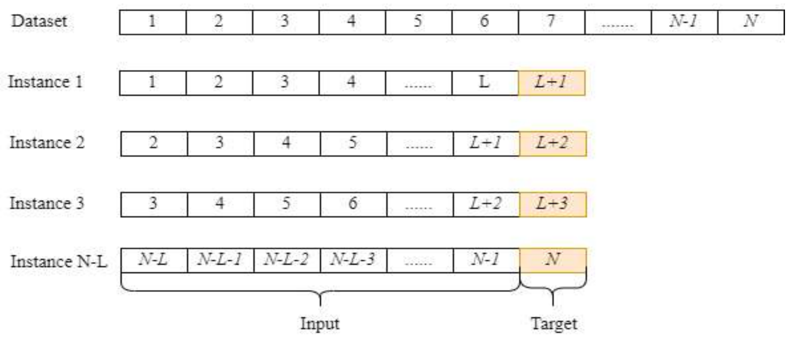

Figure 5. Sliding window method

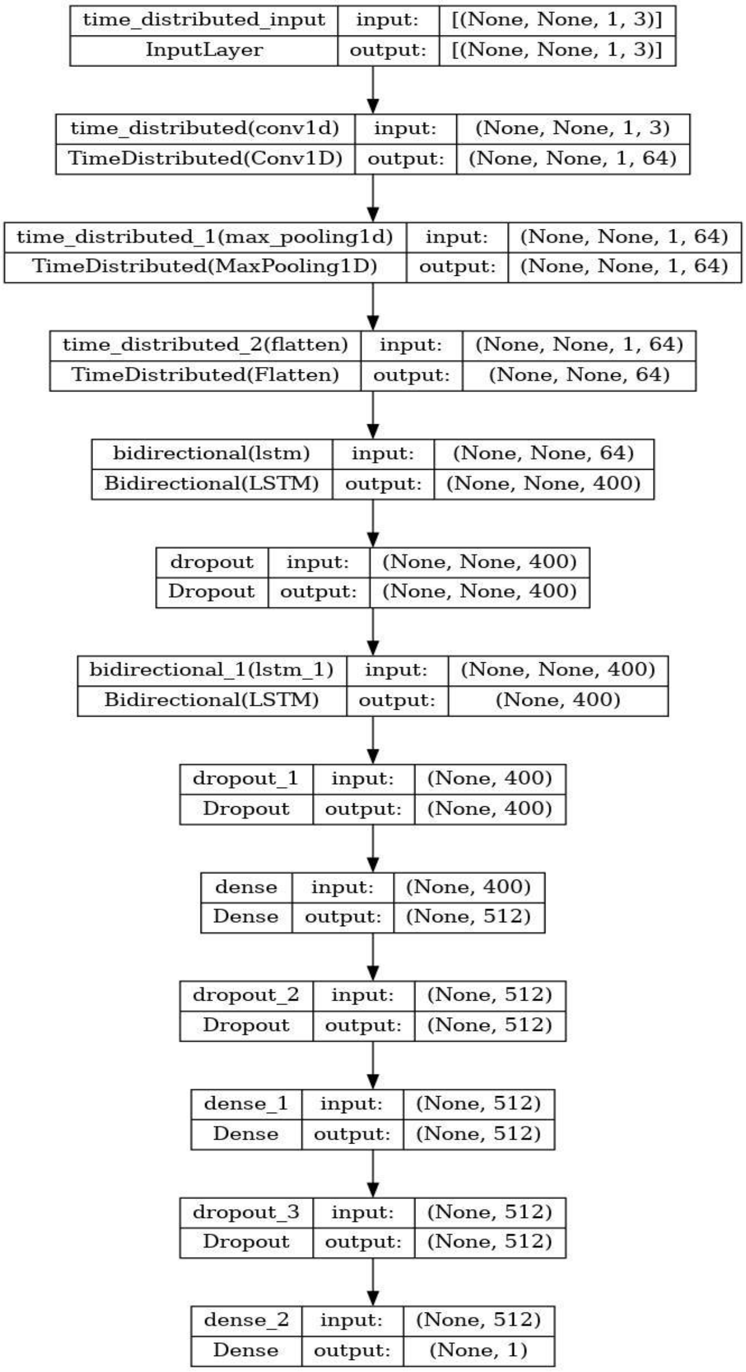

Figure 6. The architecture of the proposed model

3. RESULTS

In this study, a hybrid deep learning model for traffic flow prediction was proposed. One station was selected from a total of 170 stations in the PEMS08 data set. Experimental studies were performed on the flow variable of the chosen station utilizing 10 distinct algorithms. The dataset is partitioned into 80% for training and 20% for testing. Furthermore, 20% of the training data is utilized for validation purposes. The overall data count is 17,856, and the window size is 3.

The number of neurons in the CNNs was set to 64, with a kernel size and pool size of 2, and the activation function utilized was ReLU. The neuron count for LSTM, GRU, and BiLSTM was established at 200, with a layer count of 2. Table 1 displays the hyper parameters employed in model training.

Table 1. Hyper parameters used in model training

Hyper parameter	Method
Loss Function	MSE
Optimizer	Adam
Learning Rate	0.0001
Epoch	100
Batch Size	50

As a result of the experimental studies, the algorithms were compared with MAE, MSE, R^2 and RMSE performance metrics and comparison results were shown in Table 2 and presented with graphics in Figure 7.

Table 2. Experimental test results of algorithms

Algorithm	MAE	MSE	R^2	RMSE
CNN-BiLSTM	5.183448	54.406092	0.992332	7.376049
CNN-LSTM	5.280843	55.569123	0.992168	7.454470
CNN-GRU	5.205485	55.322532	0.992202	7.437912
BiLSTM	5.309702	57.210446	0.991936	7.563759
LSTM	5.314796	57.698882	0.991868	7.595978
GRU	5.273546	56.565239	0.992027	7.520987
SVR	9.189161	136.057386	0.980823	11.664364
LR	5.355579	58.528052	0.991751	7.650363
RF	6.032785	66.877764	0.990574	8.177883
MLP	5.726425	65.254140	0.990803	8.078003

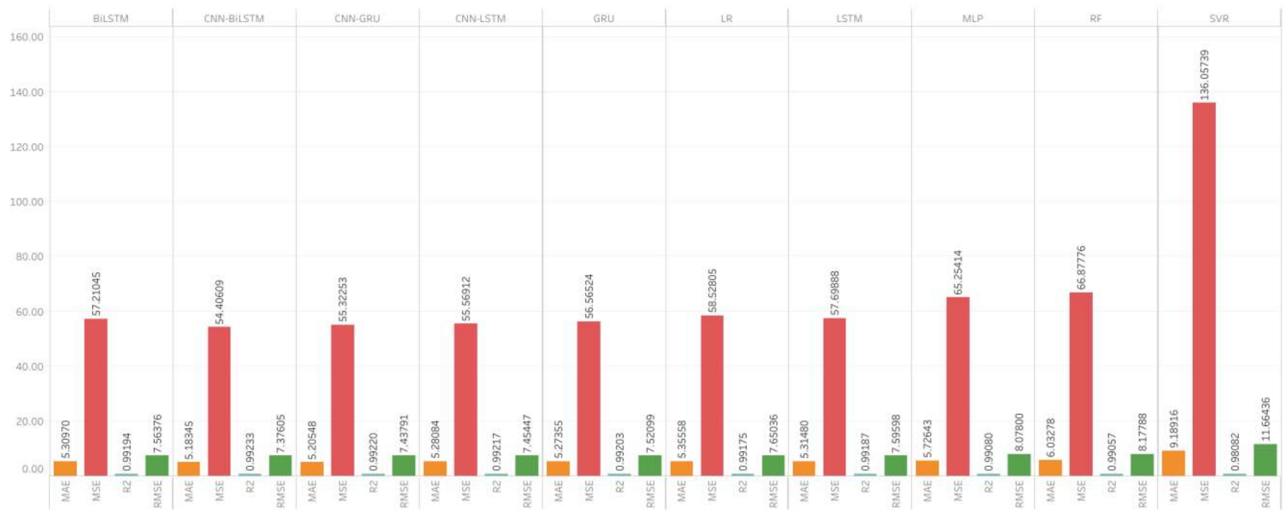

Figure 7. Graphical representation of the experimental results

Table 2 and Figure 7 demonstrate that the CNN-BiLSTM algorithm yields the most favorable results, with an MAE of 5.1834, MSE of 54.4060, RMSE of 7.3760, and R^2 value of 0.9923.

4. DISCUSSION AND CONCLUSION

The rapid increase in urbanization has weakened the quality of life for city residents and increased traffic issues. This situation has resulted in financial loss and increased environmental pollution. Efficiently controlling and reducing traffic congestion is essential for the sustainability of urban transportation systems. To attain more dependable and efficient traffic flow predicts, it is essential to utilize both historical and current traffic data. In this context, deep learning and machine learning methods provide efficient solutions to diverse issues. In recent years, these methods have been extensively utilized in the design of traffic prediction models.

This work presents a novel hybrid model based on deep learning for traffic flow prediction. A real-world dataset obtained from the Caltrans Performance Measurement System (PeMS) was utilized for performance evaluation. The efficiency of the proposed model was assessed utilizing established regression metrics and compared with various deep learning and machine learning algorithms, including BiLSTM, LSTM, GRU, SVR, LR, RF, and MLP. The evaluations are made using the metrics of MAE, MSE, RMSE, and R^2. The CNN-BiLSTM hybrid model in our experiment demonstrated optimal performance, with an MAE of 5.1834, MSE of 54.4060, RMSE of 7.3760, and R2 of 0.9923. These findings highlight the effectiveness of deep learning methods in the design of traffic prediction models.

In future efforts, we intend to create several hybrid versions of this model and introduce novel solutions that incorporate all stations within the dataset. The model's generalization capabilities can be further evaluated using larger datasets. Additionally, methods that integrate comprehensive evaluations of time series data and spatial characteristics can be developed. These improvements possess the capacity to enhance model performance and provide sophisticated solutions for traffic management and planning.

REFERENCES

[1] S. A. Sayed, Y. Abdel-Hamid, and H. A. Hefny, “Artificial intelligence-based traffic flow prediction: a comprehensive review,” J. Electr. Syst. Inf. Technol., vol. 10, no. 1, p. 13, 2023.
[2] N. A. M. Razali, N. Shamsaimon, K. K. Ishak, S. Ramli, M. F. M. Amran, and S. Sukardi, “Gap, techniques and evaluation: traffic flow prediction using machine learning and deep learning,” J. Big Data, vol. 8, pp. 1–25, 2021.
[3] Z. İ. E. Cicek and Z. K. Ozturk, “Short term traffic flow forecasting using artificial neural networks,” Sigma, vol. 9, no. 4, pp. 405–414, 2018.
[4] S. Goswami and A. Kumar, “Traffic Flow Prediction Using Deep Learning Techniques,” in International Conference on Computing Science, Communication and Security, Springer, 2022, pp. 198–213.
[5] A. Utku, “Deep Learning Based an Efficient Hybrid Model for Urban Traffic Prediction,” Bilişim Teknol. Derg., vol. 16, no. 2, pp. 107–117, 2023.
[6] 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, vol.
[7] J. Wang, R. Chen, and Z. He, “Traffic speed prediction for urban transportation network: A path based deep learning approach,” Transp. Res. Part C Emerg. Technol., vol. 100, pp. 372–385, 2019.
[8] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, “Attention based spatial-temporal graph convolutional networks for traffic flow forecasting,” in Proceedings of the AAAI conference on artificial intelligence, 2019, pp. 922–929.
[9] A. T. Karadeniz, Y. Çelik, and E. Başaran, “Classification of walnut varieties obtained from walnut leaf images by the recommended residual block based CNN model,” Eur. Food Res. Technol., pp. 1–12, 2022.
[10] A. T. Karadeniz, E. Başaran, Y. Çelik, and “Identification of walnut variety from the leaves using deep learning algorithms,” Bitlis Eren University Journal of Science, pp. 531-543, 2023.
[11] J. Isabona et al., “Development of a multilayer perceptron neural network for optimal predictive modeling in urban microcellular radio environments,” Appl. Sci., vol. 12, no. 11, p. 5713, 2022.
[12] G. Louppe, “Understanding random forests: From theory to practice,” arXiv Prepr. arXiv1407.7502, 2014.
[13] K. H. Zou, K. Tuncali, and S. G. Silverman, “Correlation and simple linear regression,” Radiology, vol. 227, no. 3, pp. 617–628, 2003.
[14] M. Awad, R. Khanna, M. Awad, and R. Khanna, “Support vector regression,” Effic. Learn. Mach. Theor. concepts, Appl. Eng. Syst. Des., pp. 67–80, 2015.
[15] B. Yang, S. Sun, J. Li, X. Lin, and Y. Tian, “Traffic flow prediction using LSTM with feature enhancement,” Neurocomputing, vol. 332, pp. 320–327, 2019.
[16] Y. Kaya, Z. Yiner, M. Kaya, and F. Kuncan, “A new approach to COVID-19 detection from X-ray images using angle transformation with GoogleNet and LSTM,” Meas. Sci. Technol., vol. 33, no. 12, p. 124011, 2022.

[17] R. Dey and F. M. Salem, “Gate-variants of gated recurrent unit (GRU) neural networks,” in 2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS), IEEE, 2017, pp. 1597–1600.
[18] Y. Hou et al., “Deep feature mining via the attention-based bidirectional long short term memory graph convolutional neural network for human motor imagery recognition,” Front. Bioeng. Biotechnol., vol. 9, p. 706229, 2022.
[19] J. Gao, “R-Squared (R^2)–How much variation is explained?,” Res. Methods Med. Heal. Sci., p. 26320843231186400, 2023.
[20] D. Chicco, M. J. Warrens, and G. Jurman, “The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation,” Peerj Comput. Sci., vol. 7, p. e623, 2021.
[21] P. Canbay and H. Tas, “Prediction of Heating and Cooling Loads of Buildings by Artificial Intelligence,” Int. J. Pure Appl. Sci., vol. 8, no. 2, pp. 478–489, 2022.
[22] A. Utku “Deep Learning Based an Efficient Hybrid Model for Urban Traffic Prediction,” Journal of Information Technologies, vol. 16, no. 2, pp. 107-117, 2023.