Journal of Neonatal Surgery
ISSN(Online): 2226-0439
Vol. 14, Issue 29s (2025)
https://www.jneonatalsurg.com

OPEN ACCESS

AI-Powered Road Traffic Flow Prediction

Chandra Bhushan Kumar[1], Suresh Kumar[2]

[1]Netaji Subhas University of Technology, New Delhi, India

Email ID: chandra.kumar.pg23@nsut.ac.in

[2] Netaji Subhas University of Technology, New Delhi, India

Email ID: suresh.kumar@nsut.ac.in

Cite this paper as: Chandra Bhushan Kumar, Suresh Kumar, (2025) AI-Powered Road Traffic Flow Prediction. *Journal of Neonatal Surgery*, 14 (29s), 20-29.

ABSTRACT

These days, many cities have traffic congestion, which is a major problem during certain peak hours and causes inhabitants to experience increased stress, noise, and pollution. Neural networks (NN) along with machine-learning (ML) techniques perform effectively with large parameterized datasets and changing behaviors which causes their adoption for real-world problem solutions over traditional analytical and statistic the developed ML and DL algorithms conduct traffic flow prediction at intersections to serve as a basis for adaptive traffic control at intersections. Two approaches of evolving traffic control could be remote control of traffic light timing or an algorithm based on expected traffic levels that alters the timing. The research develops a comprehensive approach using ML and DL models which enhances prediction accuracy of road traffic flows. The proposed methodology utilizes four models: XGB Regressor, Voting Regressor, CNN-GRU and CNN-LSTM-GRU. The dataset is preprocessed through exploratory data analysis and standardized before training. The XGB Regressor model together with Voting Regressor delivered exceptional predictive results reaching R^2 values of 0.9406 and 0.9412 respectively. Among the models tested CNN-GRU-LSTM delivered the peak forecasting results when tested against the CNN-GRU by having an R^2 score of 0.9713 along with lowest MSE 0.0291 and RMSE 0.1707 and MAE 0.1105. This proved its superiority for traffic flow prediction. The improvements are significant over existing models, including MLP-NN and Stochastic Gradient, which indicates that the proposed ML-DL method can perform effectively for real time management and prediction of traffic.

Keywords: *Intelligent Transportation System (ITS), Traffic flow prediction, Artificial intelligence, machine learning, deep learning.*

1. INTRODUCTION

Vehicular traffic has become a major problem in contemporary society because the fast increase in both the number of people living in cities and the number of vehicles on the road, as well as the technical constraints associated with the timing of traffic signals. The challenge affects health, economic welfare and natural surroundings to a great extent. Fundamentally the infrastructure of any smart city requires an Intelligent Transport System (ITS). ITS may use bigdata information and communication technology to assess road infrastructure in real-time and enhance traffic management [1][2]. The fundamental functionality of this system depends on its traffic flow prediction capabilities. A goal of traffic prediction is to use past data to foretell how a transport network will be impacted by traffic in the future [3]. This information brings value to traffic light and congestion management applications in ITS systems [4]. For instance, it may anticipate potential congestion on the linked road section by calculating the probability of it [5].

Congestion is only one of many transportation problems that ITS hopes to alleviate. Along with the explosion of popularity of AI technologies and their application to increase the accuracy of traffic flow prediction, there has been a similar rapid development in new models and frameworks for traffic flow prediction. A transportation business relies heavily on traffic forecasts[6]. Aside from its importance for the determination of traffic laws and route planning, it may also have a huge effect on the design of road projects and their construction. Additionally, a major problem in congested cities and metropolitan regions is traffic congestion. It must thus be accurately assessed and predicted[7]. Therefore, a trustworthy and effective traffic prediction tool is crucial. Recent efforts to solve complex traffic problems have made use of non-parametric methods, such as ML models [8], in addition to parametric methods, such as temporal and stochastic approaches, for traffic prediction[9]. Predicting traffic numbers using ML algorithms enables not just the incorporation of past data but also the incorporation of several external elements like weather, time of day, and calendar events. In turn, this allows for better forecasting, which aids in making educated judgements about traffic flow management [10]. ML algorithms offer a data-driven method for traffic prediction by drawing on past traffic records to produce reliable projections. A reliable and up-to-date estimate of traffic flows is crucial for government agencies to respond swiftly in traffic management choices [11][12]. However, recent developments in AI have allowed for the merging of ML and DL methods, which greatly improve the accuracy of road traffic flow predictions.

2. MOTIVATION AND CONTRIBUTIONS

This study is driven by the increasing need for precise and up-to-date traffic flow forecasting in order to alleviate urban congestion, minimise travel delays, and enhance transportation infrastructure. In traffic data, complex spatial-temporal correlations are typically difficult for traditional ML and statistics models to capture. With the rapid advancements in deep learning-based hybrid models offer a promising solution by leveraging feature extraction and sequential learning for improved accuracy. This study aims to develop a robust prediction framework that enhances traffic management efficiency, supports smart city initiatives, and contributes to intelligent transportation systems (ITS). The following research contribution of this work are:

- A novel hybrid DL model called CNN-LSTM-GRU is introduced in the study. Feature extraction is handled by convolutional layers, long-term dependencies are captured by LSTM and sequential learning is accomplished by GRU. Improved traffic flow forecast accuracy is achieved by this model.
- The ML models are optimized using hyperparameter tuning techniques, adjusting parameters like learning rate, tree depth and regularization to achieve optimal predictive performance.
- The study employs advanced data preprocessing techniques, including exploratory data analysis (EDA) missing value imputation, and feature scaling using StandardScaler, ensuring high-quality input for model training.
- The deep learning models incorporate k-fold cross-validation to enhance model generalization and mitigate overfitting, ensuring consistent performance across different traffic conditions.
- The work is useful because it helps to offer real-time traffic direction forecasts to advance smart city developments and transport planning by giving a workable and efficient predictor model.

3. RELATED WORK

The purpose of this part is to summarise the existing literature on traffic flow prediction techniques and to point out how the proposed strategy differs from others.

J. Oyoo et al. (2024), provide a two-layer ensemble ML method for evaluating and forecasting traffic accidents using simulator data. DT, NB, AdaBoost and k-NN are supervised learning methods integrated in the first (base) layer. They used the PSO approach to prioritise our dataset's characteristics in order to streamline the model. When examined against NB, K-NN, DT and AdaBoost the suggested two-layer ensemble model performed best with 88% accuracy, 83% F1 score and 86% AUC. The suggested two-layer ensemble model has potential theoretical and practical applications in evidence-based traffic safety policy development and road safety management to enhance the existing condition of the road network[13].

Y. Pan et al. (2024), to increase an accuracy of a estimations and forecasts produced by the MarkovChainModel (which makes the assumption that it has no memory), use the LSTM model. With a FD-Markov-LSTMmodel, they see a decrease of almost 39% in MAE, 35% in RMSE, and 7.4% in MAPE. Such outcomes prove beyond a reasonable doubt that the FD-Markov-LSTM model is superior to a reference models, allowing for more accurate traffic flow forecasts[14].

Y. Chen et al. (2023), provide a way to forecast ship traffic by using an extreme learning machine with the whale optimisation algorithm. The amount of ship traffic congestion is predicted using fuzzy c-means clustering taking into account a map between congestion and ship traffic flow metrics. The approach takes into account the difficulty of ships' navigation in waterways with heavy traffic as well as the external environmental variability. More specifically, a feasible and practical level of accuracy for projecting ship traffic congestion is 76.04% [15].

A. Navarro-Espinoza et al. (2022), predicting traffic flows is the only emphasis of this effort. The suggested ML and DL models are trained, validated and tested on two datasets that are available to the public. Four out of the six junctions are used to train the ML and DL models in this study. The MLP-NN neural network outperformed the others in terms of both performance (with an R2 and EVscore of 0.93) and training time. GB came in second, while RNNs scored higher but required more time to train. Finally, Stochastic Gradient, Linear Regression and RF. The fact that all of the MLand DL algorithms achieved respectable results suggests that they might work well with smart traffic signal controllers [16].

H. Wang et al. (2021), data is preprocessed using Selenium, OSS, and MessageQueue. Then, three ML algorithms—Linear Regression, DT and SVM—are used to predict the traffic flow. After that, they look at the real-time Beijing traffic statistics on the Baidu map. This paper shows that, out of these three approaches, RF has the best accuracy at 0.719. Additionally, SVM and LR make up the second [17].

Table 1: Summary of above related work for traffic flow prediction using machine learning techniques

References	Methodology	Results	Advantages	Limitations	Future Work
J. Oyoo et al. (2024)	Two-layer ensemble ML model using k-NN, AdaBoost, NB, and DT. PSO for feature selection.	Accuracy: 88%, F1-score: 83%, AUC: 86%	Improved accuracy over individual models, effective feature selection	Complexity due to ensemble learning	Application in real-world road safety management and traffic policy formulation
Y. Pan et al. (2024)	FD-Markov-LSTM model combining Markov chains and LSTM to capture residual time series	MAE reduced by 39%, RMSE by 35%, MAPE by 7.4%	Higher prediction accuracy compared to benchmark models	Assumes memoryless property in Markov model, which may not always be valid	Enhancing estimation models for complex traffic scenarios
Y. Chen et al. (2023)	Whale Optimization Algorithm applied to Extreme Learning Machine (ELM), with Fuzzy C-Means for ship traffic congestion prediction	Prediction accuracy: 76.04%	Considers environmental uncertainty and complex ship navigation conditions	Moderate prediction accuracy	Improving the model to handle dynamic maritime traffic
A. Navarro-Espinoza et al. (2022)	ML and DL models (MLP-NN, GB, RNN, RF, Linear Regression, Stochastic Gradient) for traffic flow prediction	MLP-NN achieved R^2 and EV score of 0.93	MLP-NN had better accuracy with lower training time	Longer training time for RNNs	Integration with smart traffic light controllers
H. Wang et al. (2021)	Preprocessed data using Selenium, OSS, and Message Queue. Used Linear Regression, Decision Tree, and SVM to model traffic flow	Random Forest had highest accuracy (0.719), followed by Logistic Regression and SVM	Comparative analysis of different ML methods for traffic flow modeling	Lower accuracy compared to deep learning models	Application in real-time traffic flow monitoring

4. METHODOLOGY

The suggested approach improves the accuracy of traffic flow forecast by combining DL and ML models. The proposed methodology for traffic flow forecast using ML involves four different models: XGB Regressor, Voting Regressor, CNN-GRU and CNN-LSTM-GRU. The Road Traffic Prediction Dataset is preprocessed using exploratory data analysis methods such as info(), describe(), and missing value checks. The data is standardized using StandardScaler, and divide into training and testing. A first model, XGBRegressor, is trained with hyperparameters optimizing boosting rounds, learning rate, tree depth, and regularization to improve accuracy. A second model a Voting Regressor combining Random Forest Regressor and Gradient Boosting Regressor, leverages ensemble learning to enhance prediction performance. An CNN-GRUmodel, which is based on DL uses the Adam optimiser and cross-validation to combine convolutional layers for feature extraction with GRU layers for sequence learning. The Adam optimiser was used to fine-tune a CNN-LSTM-GRU hybrid model, which integrates CNN for feature extraction, LSTM for long-term dependencies, and GRU for sequential learning. Performance evaluation is conducted using R^2, Explained Variance Score, MSE, RMSE and MAE, measures for predicting actual traffic flow trends. These whole process shows in figure 1 and their discussion discussed below:

Data Collection

The research here takes use of the Road Traffic Prediction Dataset, a publicly accessible traffic prediction dataset developed by the Huawei Munich Research Centre. Induction loops are one kind of traffic sensor that contributes to the dataset. It's worth mentioning that there are now just a few public datasets available [18]. The information may be used to make predictions about traffic patterns and adjust the settings for management of stoplights. For short-term forecasting, this dataset is ideal as it includes flow time series data taken from six urban crossings over 56 days. This data shows the total number of cars passing each cross every five minutes [19]. This study makes use of four of the six crossings to represent a four-lane intersection.

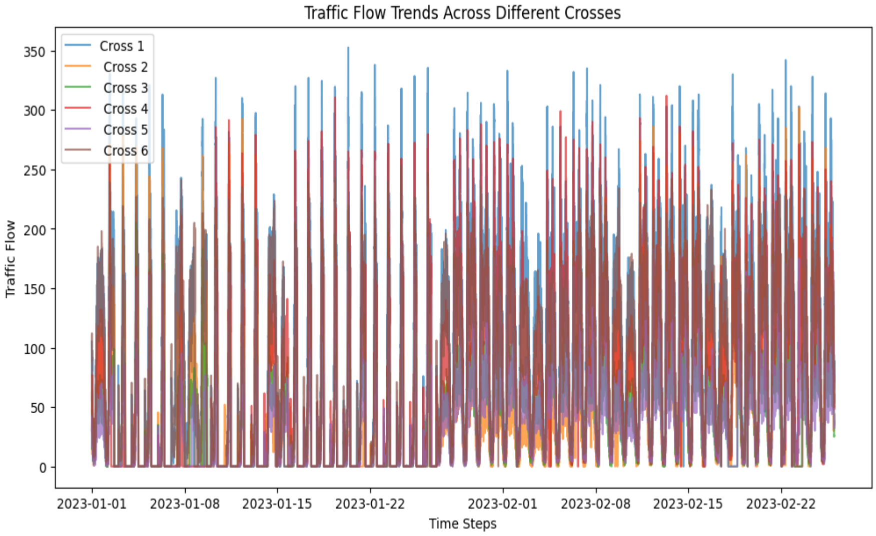

Fig1: Plot Traffic Flow Trends Across Different Crosses

Figure 1 presents a comprehensive visualization of traffic flow trends across six different road crosses over a two-month period, by the beginning of January to late February 2023. The plot provides visual evidence of time-based traffic volume changes that shows unique patterns and traffic distribution differences between different cross sections. The traffic flow at Cross 1 and Cross 4 reaches significantly higher levels than at other locations because these spots face more vehicle congestion. The data enables traffic management organizations alongside urban planners to enhance their optimisation strategies because it reveals recurring traffic peaks occurring mainly during rush hours and select days.

Data Preprocessing

Preprocessing is relied upon very heavily by ML and DL models to ensure accuracy and efficiency. A dataset obtained from Zenodo (Road Traffic Prediction Dataset) undergoes several essential preprocessing steps before being fed into predictive models. The first step is the dataset exploration, followed by handling missing values, feature scaling, the dataset splitting, and time series windowing, etc. that improve the model performance. The preprocessing steps include:

1) Dataset Exploration

The info() function together with describe() enable us to analyze both the structure and statistical characteristics of the dataset. The info() function shows a summary of the dataset that includes datatypes and column names as well as details about memory usage. On the other hand, describe() provides important statistical metrics for numerical characteristics, including mean, standard deviation, minimum and maximum values. The identified patterns from the dataset reveal odd data points and patterns alongside information about feature statistics.

2) Handling Missing Values

A model's performance deteriorates when missing values exist thus requiring prior handling before training commences. Using the isnull().sum() function, we check for missing values in the dataset. If any are detected, it can be handled through imputation techniques such as filling with the median, mean, or using forward-fill methods. Data consistency depends on this step as it helps prevent completion errors that could cause biased outcomes in records.

3) Feature Scaling Using StandardScaler

Traffic flow data consists of numerical values with varying ranges, which can affect a learning process of ML and DL models. The Standard Scaler is applied to normalize the dataset. This method speaks the characteristics on a uniform distribution having a stanard deviation of one by making sure that everything considered equally when predicting and speeding the model convergence. This dataset's attributes are transformed using the StandardScaler transformation to have a mean of 0 and a standarddeviation of 1. Equation (1) is used by StandardScaler:

$$X_{scaled} = \frac{X-\mu}{\sigma} \quad (1)$$

Where:

- X_{scaled} is the standardized value,
- XXXis an original datavalue,
- The feature average is denoted by μ,
- "σ" stands for the feature values' standard deviation.

4) Splitting the Dataset

For effective model training and evaluation, a dataset is divide into 75% training data (42days) and 25% testing data (14 days). This partition guarantees that the model has enough historical data to discover patterns before being validated on unseen data. The separation of training and testing sets prevents data leakage and provides an accurate assessment of the model's generalization ability.

5) Time-Series Windowing with Time Steps = 12

Predicting traffic flows is a time-series issue, thus it's important to capture dependencies across time. We choose a 12-step time step since that's how many iterations the model has learnt from to get our future traffic flow value prediction. This approach helps in identifying short-term patterns and fluctuations in traffic conditions, improving predictive performance.

Proposed machine learning and deep learning models for road traffic flow prediction with hyperprameters tunning

Combining classic ML with DL models strengthens and improves road traffic flow prediction. The proposed models include XGB Regressor, Voting Regressor (Random Forest Regressor and Gradient Boosting Regressor), CNN-GRU and CNN-LSTM-GRU. Hyperparameter tweaking is used to optimise each model for optimal performance. The following proposed models are:

1) XGB Regressor Model

The XGB Regressor is an effective ensemble learning model that uses gradient boosting methods to optimise the model repeatedly, leading to improved prediction accuracy [20]. The random_state parameter helps ensure result duplication while the specification of n_jobs = -1 enables the use of all available CPUcores to accelerate processing. XGBRegressor achieves successful fitting of linear and non-linear patterns and minimizes overfitting through these specified hyperparameter settings.

2) Voting Regress Model

Multiple ML models work together through ensemble learning methods in a Voting Regressor so as to produce more accurate predictions [21]. In this study, we use a combination of Random Forest Regressor and Gradient Boosting Regressor. The Random Forest Regressor[22] works by building a network of DT and then average their predictions, which makes it less susceptible to overfitting and noise. Gradient Boosting Regressor[23] is an effective method for time-series forecasting however, as it increases prediction accuracy by progressively fixing the mistakes of less robust models. The voting regressor improves the stability and accuracy of predictions by combining the two models' strengths, providing a fair compromise between the two problems of bias and variance.

3) CNN-GRU-Model

Feature extraction using CNNs and temporal dependency capture using GRUs are both included into the CNN-GRU model [24]. CNNs excel at detecting spatial patterns in traffic information and GRUs demonstrate capability in dealing with sequential patterns which means this architectural design shows promise as a time-series forecasting approach. The model achieves better performance through optimization of multiple adjustable parameters. The CNN-GRU model effectively captures both the short-term fluctuations and the long-term trends in traffic flow data via the combination of geographical feature extraction using CNNs and historical learning using GRUs.

Layer (type)	**Output Shape**	**Param #**
conv1d (Conv1D)	(None, 10, 64)	1,216
max_pooling1d (MaxPooling1D)	(None, 5, 64)	0
conv1d_1 (Conv1D)	(None, 3, 32)	6,176
max_pooling1d_1 (MaxPooling1D)	(None, 1, 32)	0

gru (GRU)	(None, 1, 50)	12,600
gru_1 (GRU)	(None, 50)	15,300
dense (Dense)	(None, 6)	306
Total params: 35,598 (139.05 KB) **Trainable params**: 35,598 (139.05 KB) **Non-trainable params**: 0 (0.00 B)		

4. *CNN-LSTM-GRU Model*

An enhancement to the CNN-GRU architecture, the CNN-LSTM-GRU model makes use of LSTM units, which are very effective for learning sequential data's long-term relationships. Complex time-series prediction is made easy with our hybrid model's CNN layers extracting crucial geographical information from traffic data, GRUs handling short-term dependencies, and LSTMs capturing long-term patterns. A learning rate is used to optimise the model using the Adam optimiser, which guarantees efficient and steady learning. Like the CNN-GRU model 5-fold cross-validation is used to validate performance across different traffic data distributions.[25][26][27][28][29][30]

Performance measures

The 'y' test was first enhanced with an inverse scaler to evaluate ML and DL algorithms' performance. Afterwards, we used the scikit-learn library's measures, which include R2, explained variance score (EVS), MSE, RMSE and MAE. They are defined as:

1) *Mean Squared Error (MSE)*

MSE is more sensitive to outliers and emphasises bigger mistakes, which might be important for predicting stock prices. It calculates as equ.2.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (2)$$

2) *Root Mean Square Error (RMSE)*

Due to its greater interpretability in financial applications, RMSE is often used as an error measure in the original scale of stock prices. It calculates as equ.3.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (3)$$

3) *Mean Absolute Error (MAE)*

MAE is a simple metric for evaluating accuracy as it takes into account both big and little mistakes equally when measuring the average magnitude of prediction errors. Equation 4 provides the MAE.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (4)$$

where y_i is an actualvalue, $\hat{y}_i$ is a predicting value, and n is a sum of all the datapoints.

4) *R-squared (R²)*

A measure of the accuracy with which a model's predictions correspond to the actualvalues is R²) which is also called the coefficient of determination. As a whole, it stands for the extent to which the independent variables may explain the dependent variable's variation. R² ranges from 0 to 1, where equ.5:

$$R^2 = (y, \hat{y}) = -\frac{\sum_{i=1}^{n}(y-\hat{y})^2]}{\sum_{i=1}^{n}(y-\bar{y})^2} \quad (5)$$

5) *Explained Variance Score (EVS)*

The EVS demonstrates how well the model predictions explain the volatility patterns of the target variable. The EVS indicates the capability of a model to explain data variability in predicting target variables. The formula for EVS is equ.6:

$$EVS = (y, \hat{y}) = -\frac{Var[y-\hat{y}]}{Var[y]} \quad (6)$$

Evaluate the proposed forecasting methods for road traffic by testing their prediction accuracy using these selected performance metrics.

Results analyis and discussion

This section delineates the experimental results of models proposed for forecasting the flow of traffic on roadways applying machine learning and deep learning techniques. The study's results additionally made use of Jupyter Notebook and the Python language to improve understand and code reusability. The data visualization used Seaborn, matplotlib libraries and model implementation used Kera's, Tenser flow and Scikit-learn data preprocessing with NumPy and pandas.

Model	R^2	EVS	MSE	RMSE	MAE
XGBRegressor	0.9406	0.9410	0.0492	0.2218	0.1497
Voting Regressor	0.9412	0.9413	0.0487	0.2207	0.1483
CNN-GRU Model	0.9593	0.9620	0.0413	0.2031	0.1405
CNN-GRU-LSTM	0.9713	0.9713	0.0291	0.1707	0.1105

Table II presents a performance metricsof a proposed ML-DL models for road traffic flow prediction.

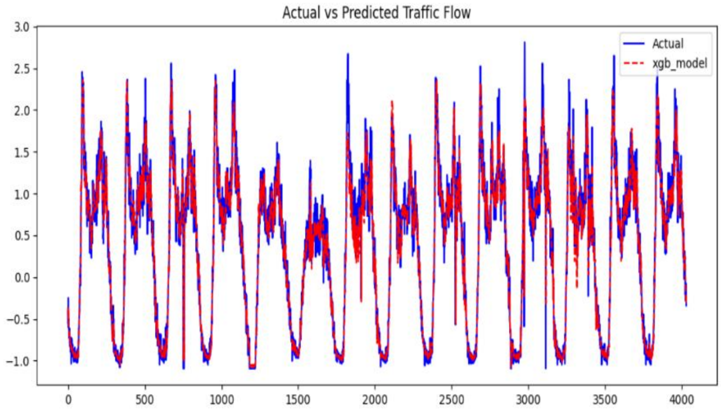

Fig2: Actual vs Predicted Traffic Flow XGB_model

The figure 2 compares actual (blue) and predicted (red) traffic flow using an XG Boost model. An x-axis displays time-steps and y-axis illustrate the traffic flow values. Both actual and predicted lines demonstrate periodic patterns with similar peaks and troughs, indicating the model captures traffic flow dynamics well. Mean absolute error appears low, as the red and blue lines closely overlap throughout the time series, suggesting the XGB model provides accurate traffic flow predictions across different time intervals.

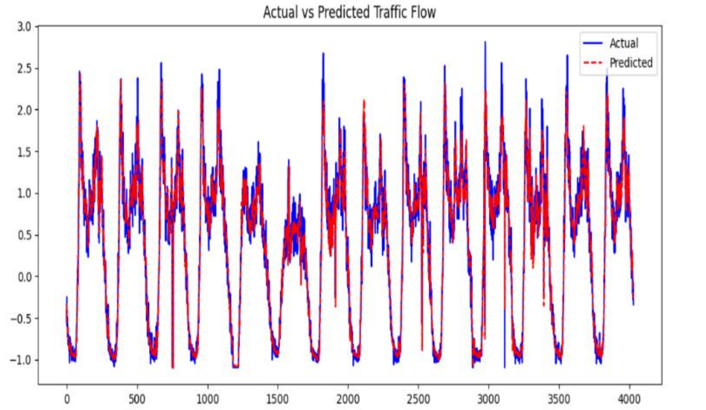

Fig3: Actual vs Predicted Traffic Flow Voting Regress_model

The figure 3 displays actual (blue) and predicted (red) traffic flow using a Voting Regressor model. An x-axis illustrates timesteps and a y-axis displays traffic flow values. The prediction line closely tracks the actual traffic flow, revealing similar periodic patterns with peaks and troughs. Visual analysis suggests high correlation among actual and forecasted values with minimal deviation. A Voting Regressor model demonstrates strong performance in capturing traffic flow dynamics, maintaining consistency across different time intervals with low mean absolute error.

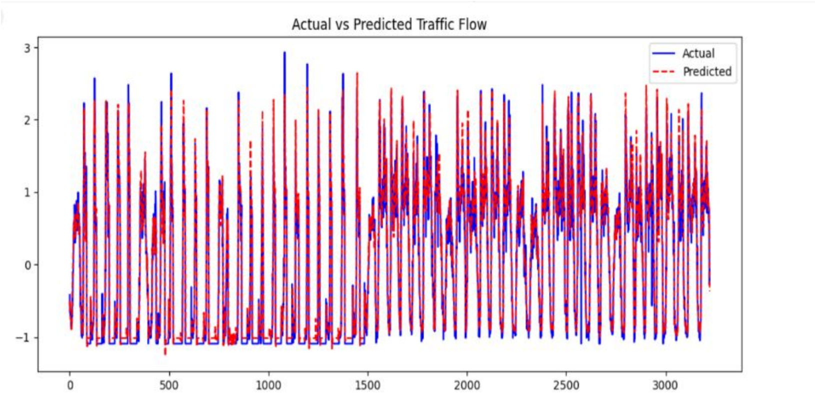

Fig4: Actual vs Predicted Traffic Flow CNN-GRU_model

The figure 4 shows actual (blue) and predicted (red) traffic flow using a CNN-GRU model. The plot reveals high similarity between actual and predicted values across time steps, with both lines exhibiting complex, oscillating patterns. The prediction line closely tracks the actual traffic flow, demonstrating the model's effectiveness in capturing traffic dynamics. Slight variations exist, but overall, the CNN-GRU model appears to provide accurate traffic flow predictions with minimal divergence between actual and predicted values.

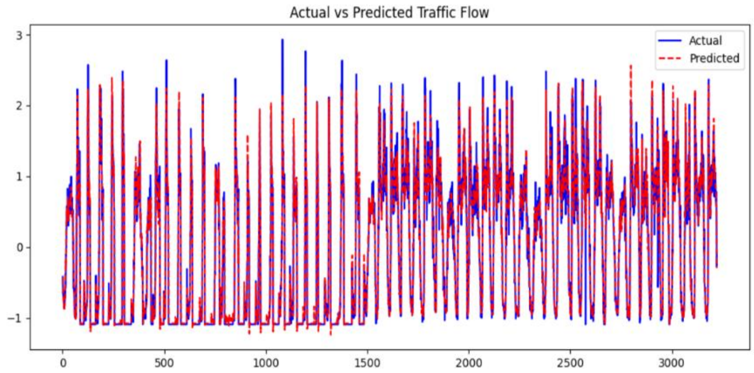

Fig5: Actual vs Predicted Traffic Flow CNN-LSTM-GRU_model

The figure 5 displays actual (blue) and predicted (red) traffic flow using a CNN-LSTM-GRU model. Complex oscillating patterns are accurately captured by a forecasting line that closely follows the actual traffic flow. The model's exceptional ability to predict traffic flow dynamics with minimal divergence between actual and predicted values is demonstrated by the similar peaks and troughs observed in both lines across time.

Discussion To suggested models do in comparison to current models for predicting traffic flows on roads. The suggested approach improves upon previous methods for predicting traffic flows on roads by combining ML and DL models; this results in more accurate and resilient predictions. Improved prediction performance is shown by better R^2 values and lower RMSE and MAE when ensemble learning in the VotingRegressor is combined with feature extraction using CNN-GRU and CNN-LSTM-GRU models. The complex architectures, especially in the CNN-GRU-LSTM model, demand significant computational resources and are sensitive to data quality and volume.

5. CONCLUSION

In this work successfully integrates ML and DL models to forecast road traffic flow with enhanced accuracy. The proposed models including XGB Regressor, Voting Regressor, CNN-GRU and CNN-GRU-LSTM demonstrate significant improvements over traditional models. The CNN-GRU-LSTM model demonstrates the best performance in predicting traffic flow dynamics because it produces an R^2 value of 0.9713 along with an EVS value of 0.9713 and the most accurate results shown through an MSE of 0.0291, RMSE of 0.1707 and MAE of 0.1105. Analysis between predicted and actual traffic patterns through multiple models confirms the effectiveness of this approach because both streams align closely together. The feasibility of DL and ML integration for the purpose of predicting traffic flows with considerable accuracy and reliability offers crucial insights to traffic management and urban planning which this study provides. Future research could include improving traffic flow forecast models in terms of accuracy and resilience by expanding from the variables used in this study to include weather, road construction and public events that could affect traffic pattern. Investigating how reinforcement learning and real time data stream can be used to improve the effectiveness and flexibility of models in dynamic traffic management would benefit and improve the model. It would be valuable to increase the dataset size by adding traffic data from other locations and to assess how good the models generalize.

REFERENCES

[1] M. Ahmed, S. Masood, M. Ahmad, and A. A. Abd El-Latif, "Intelligent Driver Drowsiness Detection for Traffic Safety Based on Multi CNN Deep Model and Facial Subsampling," *IEEE Trans. Intell. Transp. Syst.*, 2022, doi: 10.1109/TITS.2021.3134222.

[2] Rahul Anand and Smita Sankhe, "Traffic Prediction for Intelligent Transportation Systems using Machine Learning," *Int. J. Mod. Trends Sci. Technol.*, 2022, doi: 10.46501/ijmtst0807041.

[3] A. Jedhe and S. Chopade, "Traffic Prediction for Intelligent Transportation System using Machine Learning," *Int. J. Res. Appl. Sci. Eng. Technol.*, 2022, doi: 10.22214/ijraset.2022.46604.

[4] A. Boukerche and J. Wang, "Machine Learning-based traffic prediction models for Intelligent Transportation Systems," 2020. doi: 10.1016/j.comnet.2020.107530.

[5] A. Boukerche, Y. Tao, and P. Sun, "Artificial intelligence-based vehicular traffic flow prediction methods for supporting intelligent transportation systems," 2020. doi: 10.1016/j.comnet.2020.107484.

[6] S. A. Sayed, Y. Abdel-Hamid, and H. A. Hefny, "Artificial intelligence-based traffic flow prediction: a comprehensive review," *J. Electr. Syst. Inf. Technol.*, 2023, doi: 10.1186/s43067-023-00081-6.

[7] 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*, 2021, doi: 10.1186/s40537-021-00542-7.

[8] I. Moumen, J. Abouchabaka, and N. Rafalia, "Adaptive traffic lights based on traffic flow prediction using machine learning models," *Int. J. Electr. Comput. Eng.*, 2023, doi: 10.11591/ijece.v13i5.pp5813-5823.

[9] M. M. Ahsan, M. A. P. Mahmud, P. K. Saha, K. D. Gupta, and Z. Siddique, "Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance," *Technologies*, 2021, doi: 0.3390/technologies9030052.

[10] Y. Xing, X. Ban, and C. Guo, "Probabilistic Forecasting of Traffic Flow Using Multikernel Based Extreme Learning Machine," *Sci. Program.*, 2017, doi: 10.1155/2017/2073680.

[11] M. Berlotti, S. Di Grande, and S. Cavalieri, "Proposal of a Machine Learning Approach for Traffic Flow Prediction," *Sensors*, vol. 24, no. 7, p. 2348, Apr. 2024, doi: 10.3390/s24072348.

[12] B. Liu, T. Zhang, and W. Hu, "Intelligent Traffic Flow Prediction and Analysis Based on Internet of Things and Big Data," *Comput. Intell. Neurosci.*, 2022, doi: 10.1155/2022/6420799.

[13] J. O. Oyoo, J. S. Wekesa, and K. O. Ogada, "Predicting Road Traffic Collisions Using a Two-Layer Ensemble Machine Learning Algorithm," *Appl. Syst. Innov.*, vol. 7, no. 2, p. 25, Mar. 2024, doi: 10.3390/asi7020025.

[14] Y. A. Pan, J. Guo, Y. Chen, Q. Cheng, W. Li, and Y. Liu, "A fundamental diagram based hybrid framework for traffic flow estimation and prediction by combining a Markovian model with deep learning," *Expert Syst. Appl.*, 2024, doi: 10.1016/j.eswa.2023.122219.

[15] Y. Chen, M. Huang, K. Song, and T. Wang, "Prediction of Ship Traffic Flow and Congestion Based on Extreme Learning Machine with Whale Optimization Algorithm and Fuzzy c-Means Clustering," *J. Adv. Transp.*, 2023, doi: 10.1155/2023/7175863.

[16] A. Navarro-Espinoza *et al.*, "Traffic Flow Prediction for Smart Traffic Lights Using Machine Learning Algorithms," *Technologies*, 2022, doi: 10.3390/technologies10010005.

[17] H. Wang, X. Wei, J. Yao, and Y. Zhang, "Traffic Flow Prediction Using Machine Learning Methods," in *Proceedings - 2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence, MLBDBI 2021*, 2021. doi: 10.1109/MLBDBI54094.2021.00014.

[18] Y. Hou, J. Chen, and S. Wen, “The effect of the dataset on evaluating urban traffic prediction,” *Alexandria Eng. J.*, 2021, doi: 10.1016/j.aej.2020.09.038.

[19] C. Chen, Y. Wang, L. Li, J. Hu, and Z. Zhang, “The retrieval of intra-day trend and its influence on traffic prediction,” *Transp. Res. Part C Emerg. Technol.*, 2012, doi: 10.1016/j.trc.2011.12.006.

[20] C. Yuan, Y. Li, H. Huang, S. Wang, Z. Sun, and Y. Li, “Using traffic flow characteristics to predict real-time conflict risk: A novel method for trajectory data analysis,” *Anal. Methods Accid. Res.*, 2022, doi: 10.1016/j.amar.2022.100217.

[21] A. Abdulboriy and J. S. Shin, “An Incremental Majority Voting Approach for Intrusion Detection System Based on Machine Learning,” *IEEE Access*, 2024, doi: 10.1109/ACCESS.2024.3361041.

[22] A. Ilghami Kkhosroshahi, “Comparing Regression Techniques for Temperature Downscaling in Different Climate Classifications †,” *Eng. Proc.*, 2023, doi: 10.3390/ASEC2023-15256.

[23] R. Mary Chittilappilly, S. Suresh, and S. Shanmugam, “A Comparative Analysis of Optimizing Medical Insurance Prediction Using Genetic Algorithm and Other Machine Learning Algorithms,” in *Proceedings of the 2nd IEEE International Conference on Advances in Computing, Communication and Applied Informatics, ACCAI 2023*, 2023. doi: 10.1109/ACCAI58221.2023.10199979.

[24] X. Xing, X. Li, and Y. Zhai, “GRU-CNN Neural Network Method for Regional Traffic Congestion Prediction Serving Traffic Diversion Demand,” *Wirel. Commun. Mob. Comput.*, 2022, doi: 10.1155/2022/8164105.

[25] S. K. Anil Sharma, "Machine learning and ontology-based novel semantic document indexing for information retrieval," p. 108940. doi:https://doi.org/10.1016/j.cie.2022.108940.

[26] S. K. Anand and S. Kumar, "Ontology-based soft computing and machine learning model for efficent retrieval," knowl. Inf. Syst., vol. 66, no. 2, pp. 1371-1402, doi: 10.1007/s10115-023-01990-8.

[27] A. Sharma and S. Kumar, "Semantic Web-Based Information Retrieval Models: A Systematic Survey," Commun. Comput. Inf. Sci., vol. 1230 CCIS, no. May 2020, pp. 204-222, 2020, doi: 10.1007/978-981-15-5830-6_18.

[28] A. Dwivedi, S. Kumar, A. Dwivedi, and M. Singh, "Cancellable Biometrices for Security and Privacy Enforcement on Semantic Web," Int. J. Comput.Appl., vol. 21, no. 8, pp. 1-8, 2010, doi: 10.5120/2535-3460.

[29] S. Kumar, N. Kumar, R. K. Prajapati, M. Singh, and A. De, "Realization of Threats and Countermeasure in Semantic Web Services," Int. J. Comput. Theory Eng., vol. 2, no. 6, pp. 919--924, 2010, doi:10.7763/ijcte.2010.v2.264.

[30] S. Kumar, N. Kumar, M. Singh, and A. De, "A Rule-Based Approach for Extraction of Link-Context from Anchor-Text Structure *," pp. 261-262,2013.