EVALUATION OF MACHINE LEANING ALGORITHMS FOR PREDICTING THE TRAFFIC CONGESTION

Gurwinder Singh[1], Dr. Dapinder Deep Singh[2], Dr. Gurpreet Singh[3]

[1]Research Scholar, Civil Engineering, Shaheed Bhagat Singh State University, Ferozepur, Punjab, India.

[2]Assistant Professor, Civil Engineering, Shaheed Bhagat Singh State University, Ferozepur, Punjab, India.
[3]Assistant Professor, Civil Engineering, Shaheed Bhagat Singh State University, Ferozepur, Punjab, India.

Abstract: In this study, we have done the evaluation of the machine learning algorithms for predicting the traffic congestion. The XGBoost, LightGBM, Extra Tree, and Histogram Gradient Boosting machine learning algorithms were employed. The dataset was collected from the previous studies. On this collected data, feature engineering and K-mean clustering were applied to prepare it for the ML algorithms. The proposed method was developed in the Python language and used the Google Colab software for simulation purposes. The dataset was split into 70:30 for training and testing the ML algorithms. The simulation evaluation was done using error metrics, namely, Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and determination of the coefficient (R^2). The result indicates that XGBoost outperforms other ML algorithms employed in this study. Finally, the comparative analysis with the previous approach shows that the proposed method achieves a lower error metric value and a higher determination of the coefficient value due to preparing the dataset using feature engineering and k-mean clustering algorithms.

Keywords: Traffic Congestion, K-Mean Clustering, Machine Learning, Prediction, Regression.

1. Introduction

Urbanization efficiently improves the road infrastructure, enhancing connectivity between different cities. However, the incremental increases in vehicle movement on the roads present numerous challenges due to traffic congestion, including high carbon emissions and longer travel times that result in higher fuel consumption [1]. These challenges have negatively impacted the economy of the country. Traffic congestion happens when the traffic demand exceeds the capacity of the road system [2].

In the literature, recurring and non-recurring are the two categories of traffic congestion. The recurring happens when vehicle demand on the road exceeds it for a certain time or space, whereas the non-recurring occurs when, because of accidents, special events, and weather conditions, vehicle movement is slowed down on the roads [3].

To address the issue of traffic congestion, various researchers have employed machine learning-based methods to analyze traffic data, predict congestion, and offer appropriate recommendations to traffic managers, including alternative routes [4]. In previous studies, ML algorithms were used in either a classification or regression approach, depending on the type of traffic congestion data available, which can be in a discrete continuous range, such as vehicle density, or in a categorical form, such as traffic congestion states like low, moderate, or high [5]. The LR, SVM, DT, RF, and NB algorithms were successfully employed in the previous traffic congestion method [1, 6-9]. However, the performance of the ML algorithm is highly influenced by the dataset prepared for training it. Therefore, in this paper, we have presented an improved traffic congestion prediction method that effectively processes the data using feature engineering and the K-mean clustering algorithm. Thereafter, ML algorithms were utilized to predict the traffic congestion. The key contribution is summarized below.

- A method for predicting traffic congestion is developed by utilizing several ML algorithms, like LightGBM, XGBoost, Extra Trees, and Histogram Gradient Boosting.
- A preprocessing method is developed using feature engineering and k-mean clustering to prepare the dataset for ML algorithms.

- The simulation evaluation shows that the proposed method achieves the lowest RMSE value of 616.1863, the MAE value of 12.8026, and the highest R^2 value of 0.9981 for the XGBoost algorithm.

This article is subdivided into five sections. Section 2 presents the previously designed method for predicting traffic congestion. Section 3 defines the ML algorithms used for the proposed approach. Section 4 presents the proposed method for predicting traffic congestion. Section 5 shows the results and analysis. Section 6 presents the conclusion and future scope of this work.

2. Related Work

This section presents the evaluation of the ML algorithms utilized in predicting the traffic congestion in the existing approaches. The evaluation of these approaches was based on the dataset, the preprocessing methods used, the ML algorithms employed, and the simulation results. M. Hassan and A. Arabiat [1] collected the dataset from the Greater Amman Municipality and employed several ML algorithms (NB, SGD, LR, DT, RF, MLP, and a fuzzy unordered rule induction algorithm) for traffic congestion prediction. They have evaluated their method in the Weka tool. The results indicate that the fuzzy unordered rule induction algorithm achieves the highest accuracy value over other ML algorithms. Gupta et al. [6] utilized the Kaggle dataset of traffic congestion and employed LR, GB, RF, DT, and SVR algorithms to predict the traffic congestion. In their work, the dataset was prepared using the K-mean clustering algorithm and split into training and testing ratios of 70:30. The K-mean clustering algorithm was applied to sort the dataset so ML algorithms could easily learn its features. The simulation evaluation shows that the GB algorithm achieves the lowest RMSE value of 620.84 and MAE value of 470.741 and the highest R^2 value of 0.92 over the other algorithms. Das et al. [7] used the CNN architecture for vehicle detection and adaptively controlling the traffic signal lights to overcome the traffic congestion. In their work, they collected the dataset from the cameras. They have claimed that they have reduced the 75% waiting time. Jha et al. [8] presented a traffic forecasting and hot-spot identification using the ANN algorithm to overcome the traffic congestion. They have collected real-time data on Maryland freeway corridors and shown that traffic congestion is higher in the morning frame of 6 a.m.–9 a.m. and in the afternoon frame of 4 p.m.–7 p.m.

In the previous studies, several authors utilized the ML algorithms for predicting the traffic congestion [1, 6-8]. However, the preprocessing method used to prepare the dataset significantly influences the performance of the ML algorithms. Previous approaches employed traditional feature engineering and the K-mean clustering algorithm [6]. In this study, appropriate feature engineering approaches (the IQR method, generating lag features, and cycling encoding) with k-mean clustering were applied, and ML algorithms were utilized for predicting the traffic congestion.

3. Machine Learning Algorithms

This section presents the machine learning algorithms utilized in this study to predict traffic congestion. The detailed description of these algorithms is given below.

- LightGBM: It is one of the most advanced and mature algorithms. It is a light-weight algorithm that is based on a gradient boosting strategy. It uses less memory, resulting in higher accuracy and faster training. Thus, it is suitable for regression and classification problems [9]. It is useful for handling large datasets with high-dimensional features.
- XGBoost: It is an improved version of GBDT (Gradient Boosting Decision Tree). This algorithm is tree-based and offers a good balance between computational complexity and performance. It offers high performance, computational efficiency, and scalability [10].
- Extra Trees: This algorithm is like RF (Random Forest). This creates predictions by generating a wide range of decision trees [11]. This model helps to reduce variance and improve generalization.
- HistGradientBoosting: This is suitable for large datasets as it helps to improve training efficiency. This is due to the histogram binning techniques, and as a result, it is suitable for both classification and regression problems [12].

4. Proposed Traffic Congestion Prediction Method

This section presents the proposed traffic congestion prediction method developed using machine learning algorithms. The main novelty of the presented method is that the feature engineering and K-means clustering algorithms were employed to enhance the performance of the ML algorithms for predicting the traffic congestion. Figure 1 shows the flowchart of the proposed method.

- Data Collection: In this study, traffic congestion data was collected from the open source Kaggle database [13]. The dataset contains traffic observations collected from multiple traffic monitoring stations. The primary attributes include count_point_id (traffic sensor identifier), count_date (observation date), hour (time of observation), road_type, direction_of_travel, link_length_km, link_length_miles, and volume. The volume attribute represents traffic flow and was used as the target variable for congestion prediction. The dataset provides both spatial and temporal traffic information, making it suitable for machine learning-based traffic forecasting applications.
- Pre-Processing Approaches: The preprocessing stage consists of label encoding categorical attributes, removing outliers using the IQR method, extracting temporal features from the date attribute, generating lag and rolling statistical features, and applying cyclical encoding to hourly information. Subsequently, K-means clustering is employed to generate cluster labels that are incorporated into the dataset as additional predictive features. These preprocessing steps improve data quality and enable the ML algorithms to capture both temporal and behavioural traffic patterns.
- Random Data Splitting: In this study, the prepared dataset was split into training and testing ratios. This split was done randomly to train and test the performance of the ML algorithms.
- Traffic Congestion Prediction using the ML Algorithms: The dataset collected from the previous studies contains the target dataset variable in a discrete continuous range. Therefore, the ML algorithms were under the regression approach. The machine learning algorithms learn the relationship between traffic characteristics and traffic volume by analysing historical traffic observations. During training, the models identify complex nonlinear patterns and dependencies among the input features. Once they’re trained, the models start predicting future traffic volume values from unseen (test) data. XGBoost, LightGBM, Extra Trees, and HistGradientBoosting were chosen because they’ve shown solid ability with nonlinear regression tasks, and they can manage large-scale traffic datasets well, too.

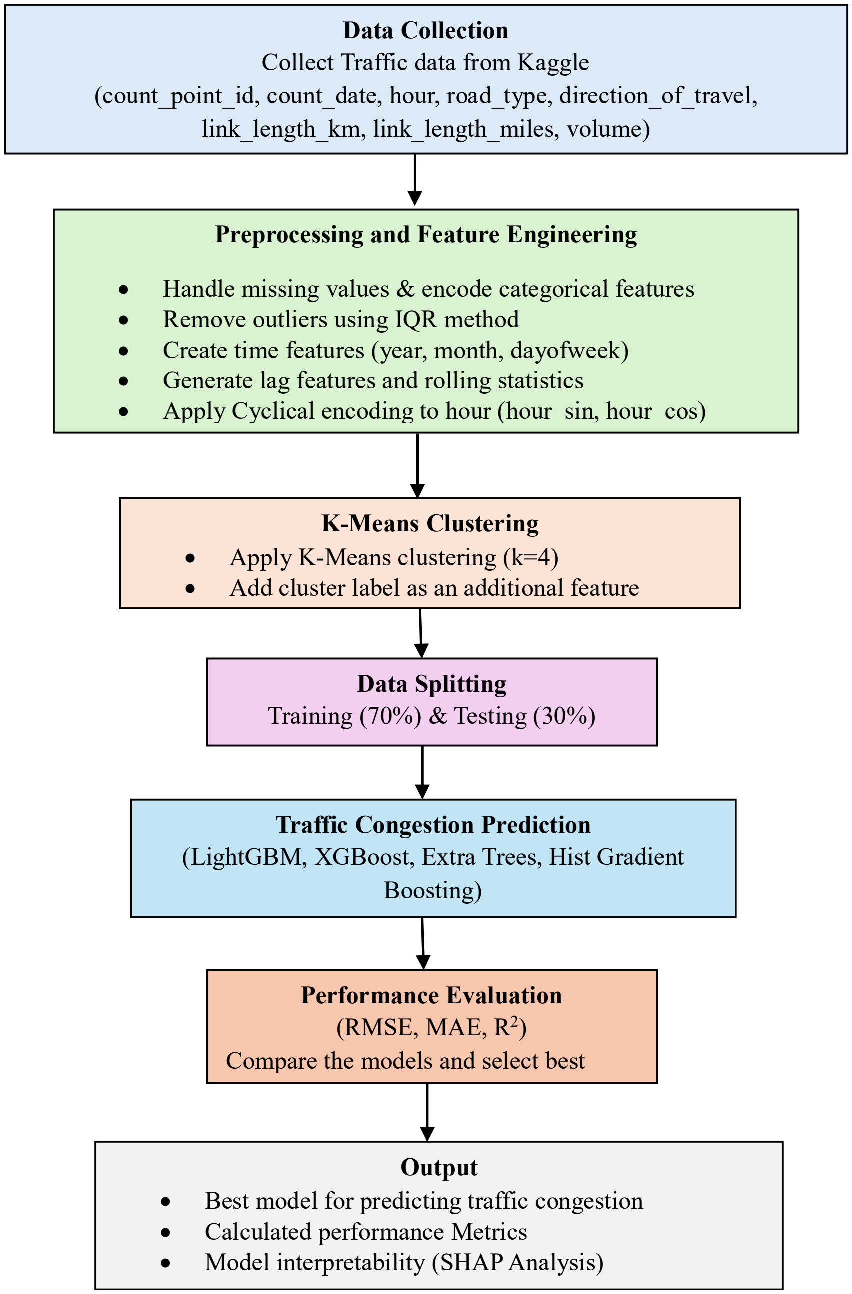


Figure 1. Flowchart of the Proposed Traffic Congestion Prediction Method

- Performance Evaluation: Due to the regression approach, the proposed method was evaluated using the error metrics (RMSE and MAE) and determination of coefficient (R^2) parameters [14]. Eq. (1-3) defines how these parameters are calculated.

$$\text{RMSE} = \sqrt{\frac{1}{\square}\sum_{\square=1}^{\square}|\square_{\square} - \widehat{\square}_{\square}|^2} \quad (1)$$

$$\text{MAE} = \frac{1}{\square}\sum_{\square=1}^{\square}|\square_{\square} - \widehat{\square}_{\square}| \quad (2)$$

$$\square^2 = 1 - \frac{\sum(\square_\square - \widehat{\square}_\square)^2}{\sum(\square_\square - \overline{\square})^2} \qquad (3)$$

In the above Eqs., $\square_\square$, $\widehat{\square}_\square$represent the actual and predicted value, n: total number of samples, and $\overline{\square}$ denotes the mean value of actual value.

Table 1 shows a pseudocode for predicting traffic congestion using the proposed approach.

Table 1. Pseudocode of the Proposed Traffic Congestion Prediction Method

Input: Clean dataset (traffic_selected.csv)
Output: Prediction of Traffic Congestion
1. Load dataset into dataframe df 2. Handle categorical variables: Apply Label Encoding to road_type and direction_of_travel 3. Remove outliers using IQR method and convert count_date to datetime 4. Extract features: year, month, dayofweek and create segment_id using: count_point_id + direction_of_travel 5. Generate lag features: lag_1, lag_2, lag_3 (grouped by segment and date) 6. Compute rolling statistics: rolling_mean_3, rolling_std_3 7. Apply cyclical encoding: a. hour_sin = sin(2π * hour / 24) b. hour_cos = cos(2π * hour / 24) 8. Remove missing values and apply K-Means clustering for k = {2, 3, 4} and add cluster labels as new features 9. Split dataset into training (70%) and testing (30%) 10. Train models: LightGBM, XGBoost, Extra Trees, and HistGradientBoosting 11. Evaluate models using: RMSE, MAE, R^2 and select best-performing model

5. Results and Analysis

This section presents the simulation results of the proposed method for predicting traffic congestion and a comparative analysis with previous approaches. The proposed method was developed in the Python language and used the Google Colab software for simulation purposes. Table 2 shows the parameter configuration of the proposed method used for simulation purposes.

Table 2. Parameter Configuration of the Proposed Method

Parameter	Value
LightGBM Regressor	n_estimators=1000, learning_rate=0.03, num_leaves=128, max_depth=10, subsample=0.8, colsample_bytree=0.8, random_state=42
XGBoost Regressor	n_estimators=1000, learning_rate=0.03, max_depth=10,

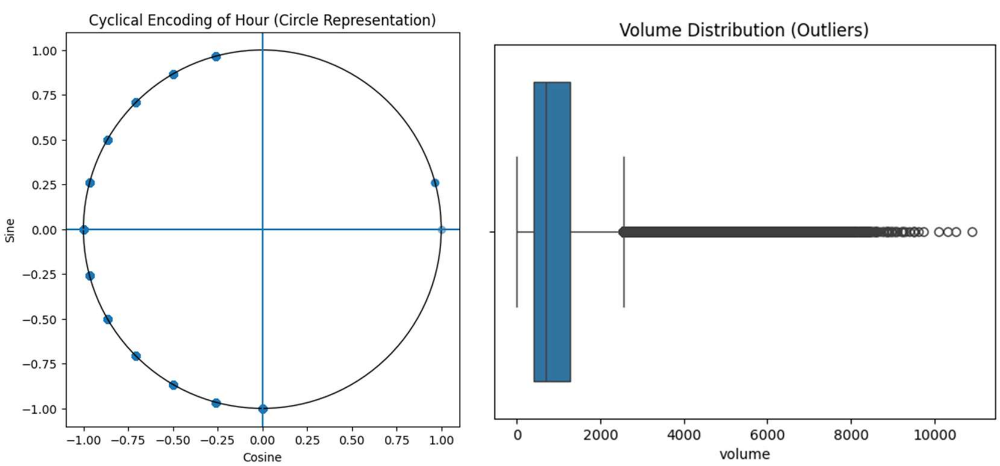
	subsample=0.8, colsample_bytree=0.8
ExtraTrees Regressor	n_estimators=300, max_depth=12, random_state=42, n_jobs=-1
HistGradientBoosting Regressor	learning_rate=0.05, max_depth=10, max_iter=1000, random_state=42

5.1 Simulation Results

This section describes the results of the proposed approach.

5.1.1 Feature Engineering and IQR

The feature engineering significantly improved the quality of the dataset by capturing temporal traffic dependencies and cyclic traffic behaviour. The generated lag features enabled the models to learn traffic patterns from previous time intervals, while rolling statistics represented short-term traffic trends.



(a) (b)

Figure 2 (a) Cyclical encoding of hour (b) Box plot for outlier detection

The cyclical encoding of hourly information using sine and cosine transformations effectively captured daily traffic periodicity (Figure 2(a)). These engineered features enhanced the learning capability of all machine learning models and contributed to the improved prediction accuracy achieved by the proposed approach. Also, the IQR approach is used to remove the outliers (Figure 2(b)). By getting rid of the extreme observations, the method reduces noise and improves it reliability and accuracy for traffic congestion prediction.

5.1.2 Evaluation of K-Means Clustering

K-Means clustering was used to find some hidden patterns inside traffic data and then group the observations that share similar traits.

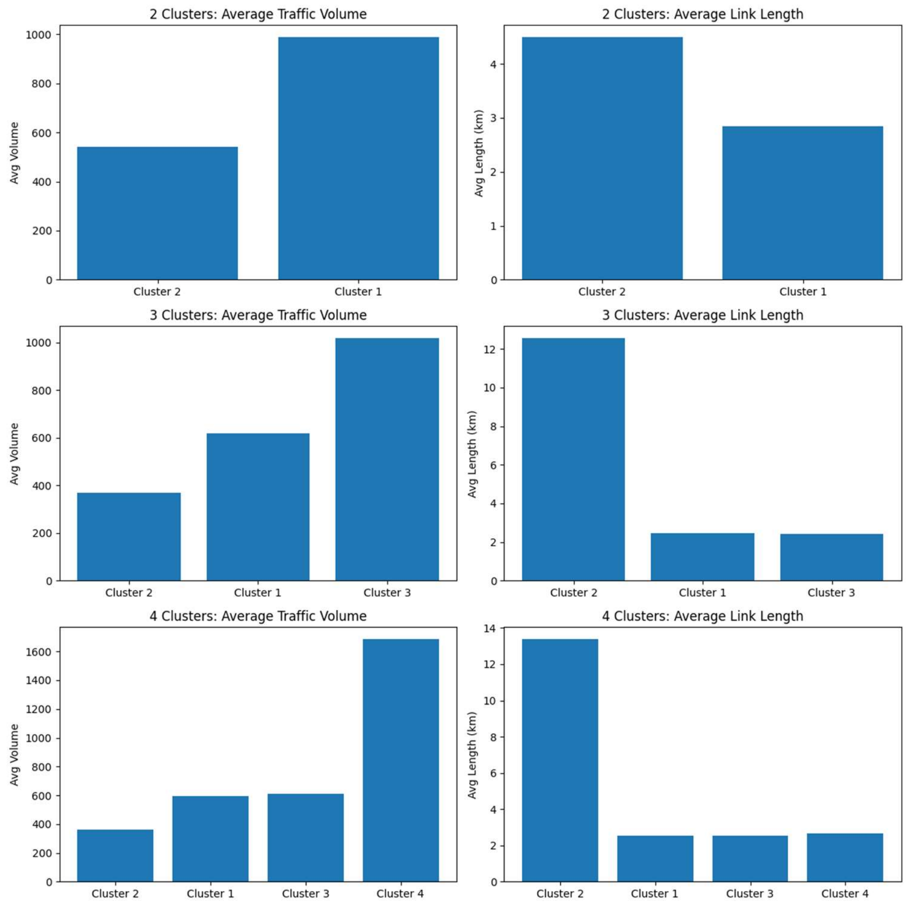


Figure 3. K-Mean Clustering for the Proposed Method

The algorithm kind of splits the whole dataset into clusters while minimizing the distance between the data points and the related cluster centroids. Here, the clustering was run with traffic related features like traffic volume, hour, and road segment length (Figure 3). Different cluster configurations (k = 2, 3, and 4) were examined, and the generated cluster labels were added as additional input features for the ML models. The clustering results revealed distinct traffic behavior patterns, where clusters with higher traffic volumes were generally associated with longer road segments. By incorporating cluster information, the proposed method improved the representation of traffic conditions and enhanced prediction accuracy.

Table 3. Evaluation of the Proposed Traffic Congestion Method

ML Algorithms	RMSE	MAE	R^2
XGBoost	616.1863	12.8026	0.9980
LightGBM	768.86425	16.3929	0.9975
HistGB	1008.4117	19.1634	0.9968
ExtreTree	7960.0045	60.4702	0.9750

Table 3 and Figures 4–6 demonstrate the prediction performance of the evaluated ML algorithms. Among all models, XGBoost achieved the best performance with an RMSE of 616.1863, an MAE of 12.8026, and an R^2 value of 0.9980. LightGBM obtained the second-best results, followed by HistGradientBoosting. Although Extra Trees achieved a relatively high R^2 value, it produced significantly larger RMSE and MAE values, indicating poorer prediction accuracy. Overall, the results confirm that XGBoost is the most suitable algorithm for traffic congestion prediction using the proposed preprocessing and clustering framework.

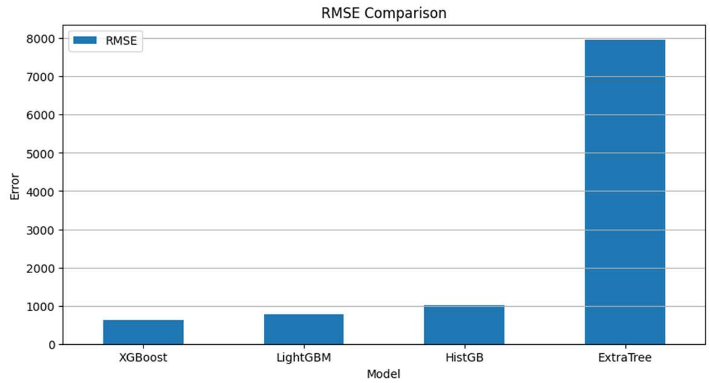


Figure 4. Comparative Analysis based on RMSE Parameter for Different ML Algorithms

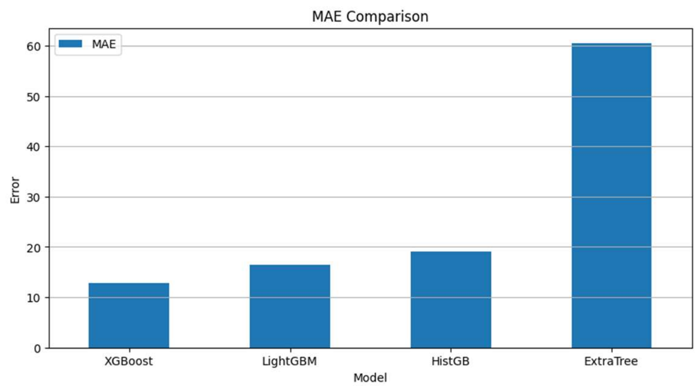


Figure 5. Comparative Analysis based on MAE Parameter for Different ML Algorithms

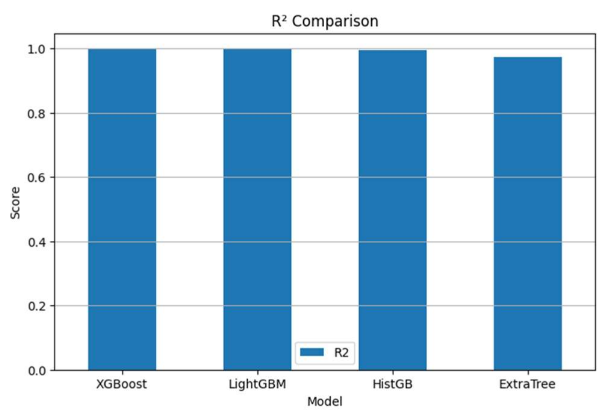


Figure 5. Comparative Analysis based on R^2 Parameter for Different ML Algorithms

5.2 Comparative Analysis

This section presents the comparative analysis with the previous study [6] that utilized the same dataset and was split into 70:30 ratios for training and testing the performance of the ML algorithms. Table 4 shows the comparative analysis. The result indicates that the proposed method achieves the lowest RMSE and MAE values and a higher value of the determination coefficient (R^2) due to the feature engineering and K-means clustering algorithms used in the preprocessing step to prepare the dataset for ML algorithms.

Table 4. Comparative Analysis

Methods	Parameters	RMSE	MAE	R^2

Gupta et al. [6]	LR	2219.92	1463.44	0.016
	GB	620.84	470.741	0.92
	RF	632.07	480.20	0.920
	DT	640.80	482.81	0.91
	SVR	795.05	301.46	0.713
Proposed Method	XGBoost	616.1863	12.8026	0.9980
	LightGBM	768.86425	16.3929	0.9975
	HistGB	1008.4117	19.1634	0.9968
	ExtreTree	7960.0045	60.4702	0.9750

6. Conclusion and Future Scope of the Work

This study presents a method for predicting traffic congestion based on machine learning algorithms. The main finding of the proposed method is that preprocessing the dataset using feature engineering and k-means clustering algorithms helps train the optimal features for the machine learning algorithms for predicting traffic congestion. The result indicates that the XGBoost algorithm accomplishes the lowest value of error metrics, whereas the ExtraTree algorithm achieves the highest value. The limitation of the designed approach is that it uses the same dataset for both training and testing purposes. In the future, the performance of the presented method will be enhanced by hyper-tuning the parameters of the machine learning algorithms through optimization approaches and by hybridizing several machine learning algorithms using an ensemble learning approach.

References

[1] M. Hassan and A. Arabiat, "An evaluation of multiple classifiers for traffic congestion prediction in Jordan," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 36, no. 1, p. 461, Jul. 2024, doi: 10.11591/ijeecs.v36.i1.pp461-468. Available: https://doi.org/10.11591/ijeecs.v36.i1.pp461-468

[2] RM Yasir, M. Asad, N. Nower, and M. Shoyaib, "Traffic congestion prediction using machine learning techniques," arXiv preprint arXiv:2206.10983, June 2022.

[3] U. Jilani, M. Asif, M. Y. I. Zia, M. Rashid, S. Shams, and P. Otero, "A Systematic Review on Urban Road traffic Congestion," *Wireless Personal Communications*, vol. 140, no. 1–2, pp. 81–109, Aug. 2023, doi: 10.1007/s11277-023-10700-0. Available: https://doi.org/10.1007/s11277-023-10700-0

[4] A. Mystakidis, P. Koukaras, and C. Tjortjis, "Advances in Traffic Congestion Prediction: An Overview of Emerging Techniques and methods," *Smart Cities*, vol. 8, no. 1, p. 25, Feb. 2025, doi: 10.3390/smartcities8010025. Available: https://doi.org/10.3390/smartcities8010025

[5] G. Tsironis, "Regression and classification," in *Understanding complex systems*, 2025, pp. 21–38. doi: 10.1007/978-3-031-81946-9_2. Available: https://doi.org/10.1007/978-3-031-81946-9_2

[6] S. Gupta, A. Kumar, and A. Kumar, "Analysis of Traffic Flow Congestion by Integrating Supervised Machine Learning with K-mean Clustering," *SN Computer Science*, vol. 6, no. 3, Mar. 2025, doi: 10.1007/s42979-025-03761-4. Available: https://doi.org/10.1007/s42979-025-03761-4

[7] T. Das, I. Chatterjee, and S. Mondal, "Traffic congestion prediction using machine learning algorithm," *Cureus Journal of Computer Science.*, Jan. 2025, doi: 10.7759/s44389-024-01981-y. Available: https://doi.org/10.7759/s44389-024-01981-y

[8] M. K. Jha, R. Jaiswal, D. S. K. Varma, S. Rankavat, A. K. Bachu, and P. K. Jha, "A machine learning approach to traffic congestion hotspot identification and prediction," *Future Transportation*, vol. 5, no. 4, p. 161, Nov. 2025, doi: 10.3390/futuretransp5040161. Available: https://doi.org/10.3390/futuretransp5040161

[9] W. Cheng, J.-L. Li, H.-C. Xiao, and L.-N. Ji, "Combination predicting model of traffic congestion index in weekdays based on LightGBM-GRU," *Scientific Reports*, vol. 12, no. 1, Art. no. 2912, Feb. 2022, doi: 10.1038/s41598-022-06975-1.

[10] B. Lartey, A. Homaifar, A. Girma, A. Karimoddini, and D. Opoku, "XGBoost: A Tree-Based Approach for Traffic Volume Prediction," in Proceedings of the 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Melbourne, Australia, Oct. 17–20, 2021, pp. 3642–3647, doi: 10.1109/SMC52423.2021.9658959.

[11] C. Chaoura, H. Lazar, and Z. Jarir, “Predictive System of Traffic Congestion Based on Machine Learning,” in Proc. 2022 International Conference on Wireless Networks and Mobile Communications (WINCOM), 2022, doi: 10.1109/WINCOM55661.2022.9966448.

[12] S. Deng, “Research on Traffic Prediction Based on Machine Learning,” in Proceedings of the 3rd International Conference on Mechatronics and Smart Systems, 2025, doi: 10.54254/2755-2721/135/2025.21218.

[13] Möbius, “Road Trafic Dataset,” *Kaggle.com*, 2018. https://www.kaggle.com/datasets/arashnic/road-trafic-dataset

[14] T. Matue, A. A. Akinyelu, and M. Mase, “Comparative evaluation of recurrent and convolutional neural network architectures for traffic flow prediction at non-signalised intersections,” *Discover Artificial Intelligence*, May 2026, doi: 10.1007/s44163-026-01091-9. Available: https://doi.org/10.1007/s44163-026-01091-9