

Paper Type: Original Article

Optimizing Urban Traffic Flows Using Advanced Data-Driven Machine Learning Models

Khursheed Mohammed Hussain ^{1,*} 

¹ Engineering and Technology Department, J.B. Hunt Transport, Lowell, Arkansas, United States;
Khursheed.hussain@jbhunt.com

Received: 11 Jan 2025

Revised: 16 Apr 2025

Accepted: 04 Aug 2025

Published: 06 Aug 2025

Abstract

Traffic congestion has lately emerged as a significant worldwide issue due to increasing industrialization and the increased number of vehicles on the road. The control of traffic and congestion is an essential component of urban planning and infrastructure development. Efficient traffic management not only enhances the flow of vehicles but also contributes to reducing fuel consumption, minimizing air pollution, and improving road safety. This study introduces a Long Short-Term Memory (LSTM) model that uses traffic statistics to predict traffic levels. The LSTM design was chosen because of its shown ability to spot complex temporal patterns and long-term correlations in sequential data, which typically go unnoticed by traditional models. Experimental evaluation shows that the proposed LSTM achieved an R^2 score of 93.45%, with a low Mean Absolute Error (5.504), Root Mean Squared Error (6.857), and Mean Absolute Percentage Error (0.174). Comparative analysis indicates that the LSTM outperformed benchmark models such as standard Neural Networks, Random Forests, and Feedforward Neural Networks on the same dataset. The findings highlight the suitability of deep learning for traffic forecasting tasks and provide a foundation for future research that incorporates real-time conditions and hybrid approaches to further improve prediction accuracy. This work supports the development of smarter, data-driven urban traffic management systems.

Keywords: Urban Traffic Flow, Intelligent Traffic Management (ITM), Smart Cities, Traffic Prediction, Machine Learning.

1 | Introduction

Urbanization and the exponential rise in vehicle ownership have dramatically increased the demand on city transportation networks, leading to growing concerns about traffic congestion, delays, road safety, and environmental pollution. With urban populace booming, easing of traffic becomes firstly a utilitarian challenge and secondly, an economic and environmental imperative [1, 2]. The dynamic nature of traffic flows within urban areas has made existing traffic systems, which are more dependent on the use of constant signal designs and preset regulations, obsolete. Urban Traffic Flow management, therefore, stands at the core of smart city development [3]. It denotes the management and optimization of road traffic throughout the road networks and intersections to make transportation smooth, safe, and sustainable [4, 5]. Nevertheless, the extremely dynamic and non-linear character of traffic patterns makes rule-based systems ineffective. This shortcoming highlights the advantage of predictive, intelligent solutions that are able to adjust themselves in real-time to changing traffic conditions [6].



Corresponding Author: hussainyrs@gmail.com



<https://doi.org/10.61356/j.aics.2025.2570>



Licensee **Artificial Intelligence in Cybersecurity**. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0>).

Computational intelligence has turned into a prime research concern in optimizing Urban Traffic flow prediction and detection. Real-time data and sophisticated analytics-enabled Intelligent Traffic Management (ITM) systems can dynamically optimize signal timing and can forecast congestion, as well as assist the passage of emergency vehicles [7]. This system is heavily based on artificial intelligence to process traffic data and analyze anomalies, and make proactive decisions. The main components of the implementation of ITM systems are the incorporation of Data-Driven ML and DL methods [8]. There is significant potential in such AI-based approaches to examining large and multidimensional traffic data. ML models can reveal complicated patterns, categorize traffic states, and make predictions based on data to implement control effectively [9, 10]. The paper has suggested a data-driven ML- and DL-driven framework that can be used to optimize urban traffic patterns. Through the integration of real-time analytics, intelligent modelling, and control components, the suggested technology would improve the effectiveness, safety, and sustainability of contemporary city transportation networks.

1.1 | Motivation and Contribution

The exploding growth of cities and vehicles has made conventional traffic systems outdated, causing severe traffic congestion, safety issues, and long-term sustainability concerns. The dynamic and unpredictable character of contemporary traffic is not accommodated by static, rule-based signal schemes, which leads to delays, greater emissions, and a higher risk of accidents. Intelligent Traffic Management (ITM) systems offer a promising alternative through real-time data processing and adaptive control. However, the complexity of urban networks demands advanced computational intelligence capable of predicting traffic patterns, detecting anomalies, and optimizing flow dynamically. Integrating ML and DL into a unified, data-driven framework is thus essential for enabling efficient, sustainable, and smart urban mobility. This study greatly advances the area of urban traffic studies in several important ways:

- Analyzed a comprehensive Road Traffic Dataset for Great Britain covering motorways, A roads, and minor roads with detailed annual street-level traffic data.
- Developed a comprehensive data pre-processing pipeline handling missing values, duplicate removal, and feature selection for optimal traffic prediction model performance.
- Implemented Min-Max normalization using a mathematical formula to scale traffic data between 0-1, optimizing classifier performance and mitigating outlier effects.
- Designed LSTM neural network architecture with memory blocks and three multiplicative gates to capture temporal dependencies in sequential traffic data.
- Evaluated model accuracy using multiple metrics (R^2 , MAE, RMSE, and MAPE) for thorough performance assessment.

1.2 | Justification and Novelty

The justification for using the LSTM model in this study lies in its proven ability to handle sequential and time-dependent data. This makes it especially suitable for modeling the cyclical and fluctuating nature of urban traffic flow. Unlike traditional models that struggle with long-term dependencies, LSTM uses memory cells and gates to retain relevant historical information. These mechanisms help the model learn complex traffic patterns over time. The novelty of this work is its integration of detailed, large-scale street-level traffic data with an LSTM architecture. This produces highly accurate forecasts with a superior R^2 score and low error rates compared to standard Neural Networks, Random Forests, and Feedforward Neural Networks. The approach combines robust preprocessing, careful feature selection, and advanced sequence modelling. This provides a more precise and reliable solution for predicting urban traffic. This research demonstrates the application of DL to real-world traffic datasets. It lays a foundation for future intelligent transportation systems that need high-performance, data-driven prediction tools to manage complex urban mobility challenges.

2 | Literature Review

Several significant research studies on predictive modeling of urban traffic flows have been thoroughly reviewed and analyzed to guide and strengthen the foundation of this study.

Miao and Liao (2025) examine how to improve traffic management in smart cities by combining CNNs with PSO. Developing a predictive model that enhances real-time traffic management, decreases congestion, and increases the accuracy of traffic predictions was the main goal. PSO was used to adjust the CNN hyperparameters and feature selection after traffic data gathered from IoT sensors was analysed to extract pertinent spatial features using CNNs. Important results show that the CNN-PSO model greatly increases the accuracy of traffic predictions, with an average rise from 76.5% to 92.0% in different metropolitan locations. An average of twenty percent less delay in traffic and twenty-five percent more efficient traffic flow were both shown by the model. In real-time, the model was effective in offering timely traffic predictions requiring minimal processing time with reasonable accuracy in the level of prediction errors [11].

Alshardan et al. (2024) design a Traffic Flow Control utilizing a Fuzzy Deep Neural Network with a crayfish optimization algorithm (TFCFDNN-COA) technique in Urban Environments. At first, the TFCFDNN-COA approach includes data pre-processing and a dingo optimizer algorithm (DOA) model for feature selection. The fuzzy deep neural network (FDNN) technique controls traffic flow. Eventually, the crayfish optimization algorithm (COA) model is utilized to fine-tune the best hyperparameter of the FDNN model. A wide range of experimental studies has been completed, and the outcomes have been studied using numerous features. The experimental validation of the TFCFDNN-COA approach portrayed superior MSE, MAE, and MAPE values of 0.0011, 0.0206, and 0.6738, respectively, all observed on Sunday [12].

R G et al. (2024) involves a density-based strategy where the system dynamically adjusts traffic light control, prioritizing high-traffic junctions. This is achieved using the YOLOv8 algorithm for object detection, with the system learning from real-world traffic data. The findings highlight the system's efficiency in alleviating congestion and enhancing traffic management. A comparative analysis of two regression models, RFR and SVR, reveals that RFR attained an accuracy of 86.58%, while SVR slightly surpassed it with an accuracy of 87.84%. This indicates that both models are effective, with SVR showing a minor improvement in both accuracy and error rate. Notably, the error rate in SVR was reduced by 1.26% compared to RFR [13].

Yang, Zhou and Xie (2024) aim to optimize the traditional XGBoost regression model by using the four-vector algorithm to achieve the accurate prediction of road traffic flow. The experimental results show that in terms of fitness curve, when the number of iterations reaches 8, the fitness decreases from 0.045 to 0.039 for the first time. When the number of iterations reaches 18, the fitness further decreases to 0.038 and remains stable. This shows that the model gradually converges during the training process, and the performance is improved. Analysing the training set's prediction line chart reveals that the model's projected value of the traffic flow is quite close to the real value, with an RMSE of 3.0077. Additionally, the model has great generalisability when validated on the test set; its RMSE is 3.5249, demonstrating good prediction ability [14].

Sattarzadeh and Pathirana (2024) enhance both the adaptability and efficiency of traffic control systems. Model integrates a probabilistic graphical framework with DRL, facilitating the extraction of interpretable and actionable insights from complex traffic data. It has developed and validated a model using a simulated environment that reflects real-world traffic conditions, demonstrating substantial improvements in traffic throughput and reduction in average travel times. The proposed model outperforms traditional traffic signal control algorithms. It also offers a robust framework for real-time traffic management, contributing to the development of future smart city infrastructure. The model achieves a 95% success rate, reduces average queue lengths by over 50%, and cuts intersection delays by up to 80%, compared to traditional methods. These improvements highlight its efficiency in optimizing traffic flow and minimizing congestion [15].

Patil, Raorane and Kundale (2023) implemented on embedded controllers through Transfer Learning, enabling deep neural network capabilities even on limited hardware resources. The proposed system comprises a camera system capturing real-time video feeds of roads, which are then processed by a machine learning model to detect vehicle density. The scheduling system constantly adjusts traffic light green timings in response to the observed vehicle density. Furthermore, it conducts an in-depth analysis using various machine learning models to predict traffic patterns. This analysis encompasses a performance comparison of models like KNN, random forests, and decision trees to forecast traffic volume with respective accuracy of 53.43%, 98.88%, and 99.09%. The study also investigates the influence of parameters like time of day, weather and temperature conditions. The primary objective of system is to optimize traffic flow by minimizing waiting times at traffic signals, thereby reducing congestion and enhancing the overall commuting experience for drivers [16].

Thom et al. (2022) designed with ease of connection in mind from the start, despite the fact that their default settings may leave them exposed. Since it is easy to forge IP and MAC addresses, keeping track of IoT devices on a network may be tricky, yet identifying each item is essential for keeping a network secure. Also, ML models need regular updates and tweaks to keep working. They use a hybrid strategy that integrates neural networks with location-sensitive hashing to identify specific devices from their network traffic, sidestepping the need for tedious feature construction and model retraining. This method enables real-time device identification by accurately identifying known devices to within a margin of 98 percent using just one packet sniffed from the network. Furthermore, this method is much superior to others as it does away with feature extraction altogether [17].

While numerous studies have demonstrated the effectiveness of advanced ML techniques, including CNN, fuzzy DNN, XGBoost, and reinforcement learning for traffic flow prediction and intelligent traffic control, several critical research gaps remain. Most existing models are dataset-specific and exhibit limited adaptability to diverse, real-world urban scenarios. Additionally, many approaches involve high computational complexity, which constrains their applicability for real-time deployment in resource-constrained environments. Furthermore, few studies offer a unified framework that combines traffic prediction, optimisation, and adaptability with model interpretability and deployment scalability. These limitations are summarised in Table 1, which outlines key methodologies, datasets used, and the major shortcomings of prior works. To address these gaps, this study proposes a lightweight yet robust deep learning framework capable of maintaining high prediction accuracy, supporting real-time processing, and generalising across varied traffic environments.

Table 1. Overview of Recent Studies on Predictive Modeling of Optimizing Urban Traffic Flows Using Machine learning.

Author	Proposed Work	Dataset	Key Findings	Challenges/recommendations
Miao and Liao (2025)	CNN combined with Particle Swarm Optimization (PSO) for spatial feature extraction and traffic forecasting.	IoT sensor-based traffic data.	Increased prediction accuracy from 76.5% to 92%; reduced traffic delay by 20% and improved flow efficiency by 25%.	Real-time deployment and hyperparameter tuning remain complex.
Alshardan et al. (2024)	Fuzzy DNN with Crayfish Optimization Algorithm (TFCFDNN-COA) and Dingo Optimizer for feature selection.	Urban traffic datasets	Achieved MSE: 0.0011, MAE: 0.0206, MAPE: 0.6738; demonstrated effective flow control.	High computational complexity; focus on system performance and energy efficiency.
R G et al. (2024)	Density-based dynamic signal control using YOLOv8 and comparison of RFR vs SVR models.	Real-time traffic video data	SVR accuracy: 87.84%, RFR: 86.58%; SVR showed 1.26% lower error.	Data annotation and real-time model adaptation are critical.
Yang, Zhou & Xie (2024)	XGBoost optimized with four-vector algorithm for road traffic prediction.	Traffic datasets	RMSE: 3.0077 (train), 3.5249 (test); high consistency with actual values.	Further tuning for larger datasets and real-time prediction is needed.

Sattarzadeh & Pathirana (2024)	Integration of probabilistic graphical models with DRL for real-time traffic control.	Simulated real-world traffic	Achieved 95% success rate; reduced queue lengths by 50% and delays by 80%.	Requires real-world validation and extensive DRL training time.
Patil, Raorane & Kundale (2023)	Transfer Learning on Embedded Controllers for Dynamic Signal Scheduling Using Vehicle Density Analysis	Real-time video data	Accuracy: KNN (53.43%), RF (98.88%), DT (99.09%); improved congestion control.	Hardware limitations in edge devices, as well as environmental parameters like weather, affect performance.
Thom et al. (2022)	Real-time IoT device identification using locality sensitive hashing and neural networks.	Network traffic data	Achieved 98% accuracy using a single packet; eliminated need for complex feature engineering.	IoT network security vulnerabilities and need for continuous model updates.

3 | Research Methodology

For the traffic flow forecasting using machine learning techniques, various steps and phases are required, as depicted in Figure 1. This study's methodology included gathering comprehensive yearly traffic data at the street level, then pre-processing the dataset by loading it, dealing with missing values, removing duplicates, selecting relevant features, and finally using Min-Max Normalisation to scale numerical features from 0 to 1. The cleaned and prepared data were then split into training and testing sets using an 80/20 ratio. The sequential patterns in the traffic flow data were captured by an LSTM model, which successfully learnt temporal relationships by using its memory blocks and gates. At last, R^2 , MAE, RMSE, and MAPE, which are typical regression metrics, were used to evaluate the model's performance and determine the correctness and dependability of the predictions.

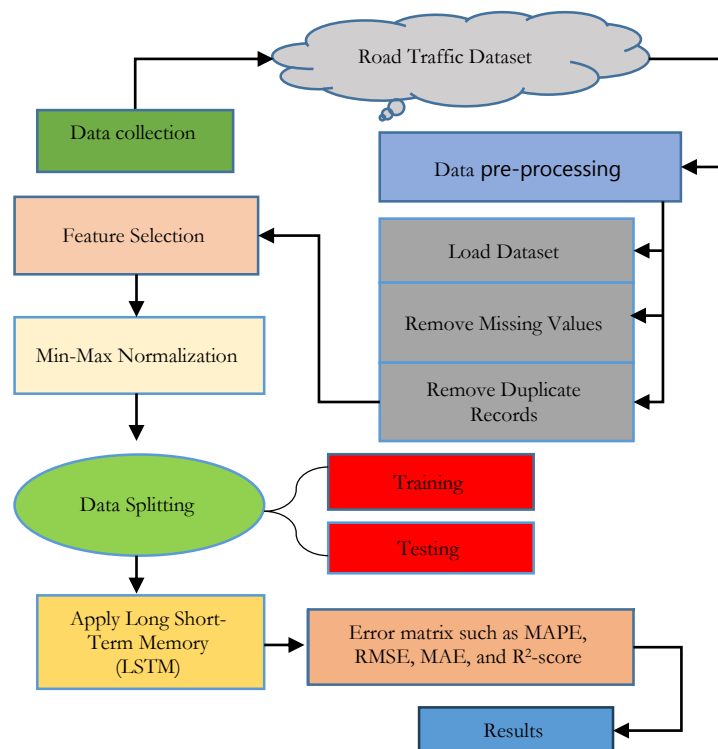


Figure 1. Proposed Flowchart for Urban Traffic Flows Forecasting

The following is a comprehensive explanation of each step outlined in the proposed flowchart for predictive modelling of urban traffic flow.

3.1 | Data Collection

The Road Traffic Dataset¹ Provides detailed annual street-level traffic data for Great Britain's motorways, A roads, and some minor roads, combining vehicle counts and distances traveled. It uses roadside manual counts, automatic traffic counters, and road length data, with quality flags indicating estimate reliability. While national and regional Figures are robust, local estimates should be used carefully. This dataset is useful for traffic analysis, planning, and prediction at national, regional, and local levels. Some visuals of datasets are provided below:

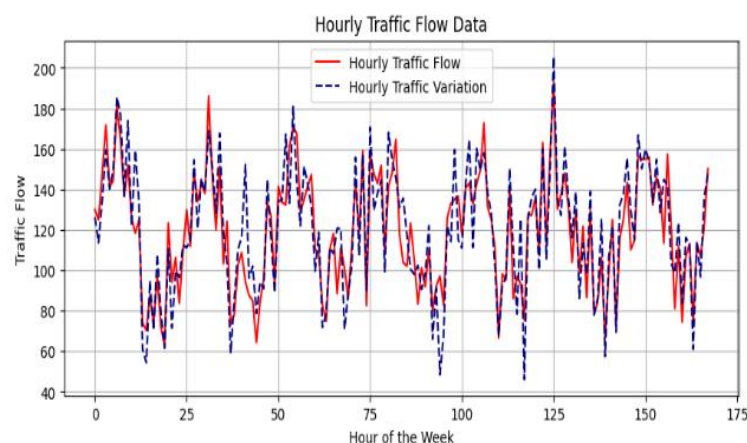


Figure 2. Plot for Hourly Traffic Flow.

Figure 2 displays hourly traffic flow data over 175 hours, showing cyclical patterns with peaks of 180-200 vehicles during rush hours and valleys of 40-60 vehicles during off-peak times. The close alignment between actual traffic flow (blue line) and traffic variation (red dashed line) demonstrates effective pattern recognition, with regular 24-hour cycles reflecting typical urban commuting behaviors and occasional variations indicating external factors like weather or incidents affecting traffic dynamics.

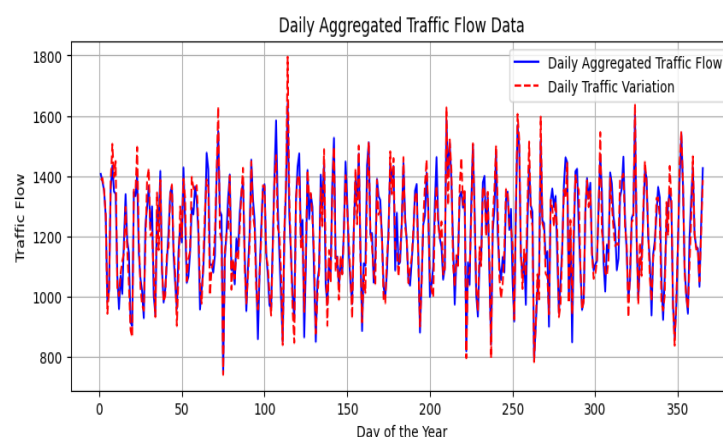


Figure 3. Time series Plot for daily Aggregated traffic Flow.

Figure 3 displays two time series over 365 days, representing a full year. The blue solid line shows the total traffic flow recorded each day. The red dashed line illustrates the daily fluctuations or variability in traffic. Both lines exhibit a strong periodic pattern, likely indicating weekly or seasonal cycles in traffic flow, with peaks and troughs occurring consistently throughout the year. The Daily Aggregated Traffic Flow typically falls between about 900 - 1600 units, whilst the Daily Traffic Variation will often display the same

¹ <https://www.kaggle.com/datasets/arashnic/road-traffic-dataset>

characteristics as the aggregated flow but with a greater degree of larger/frequent fluctuations, which, in this case, could indicate more unstable traffic patterns.

3.2 | Data Pre-Processing

Cleaning and preparing raw traffic for accuracy and usability is the foundation of data pre-processing [18]. This involves the detection and treatment of missing values and, removal of duplicates. It also entails developing new meaningful characteristics, including vehicle per kilometre ratios or cyclic year encodings to capture the trends. Besides, numerical data should be normalised or scaled. These procedures ensure that the data is dependable, consistent, and prepared for efficient modelling or analysis. The key preprocessing steps included the following:

- **Load Dataset:** loading the data and taking a close look at its structure so that you know what you are dealing with. This involves checking the row and column counts, column names, and column types, as well as previewing records.
- **Handle Missing Value:** Common methods are deleting, imputation (the missing values are replaced with a calculated or predicted value), and flagging. The nature of the data and the level and nature of missingness determine the best strategy.
- **Remove Duplicate Records:** It's essential to verify whether any rows are identical copies of others and remove them to maintain unique records. This step ensures that the same traffic counts are not accidentally included more than once, which would otherwise inflate total vehicle numbers or distort averages.

3.3 | Feature Selection

Feature selection is a method for extracting useful information from a dataset by determining which variables are most important and keeping just those. For traffic data, this means carefully choosing columns that have a clear relationship with the target variable, for example, keeping year, link_length_km, and cars_and_taxis if they significantly affect overall traffic patterns, while removing redundant or irrelevant fields that add noise or redundancy. Good feature selection improves model accuracy, reduces computational cost, and makes the results easier to interpret by focusing only on the attributes that genuinely help explain or predict traffic behavior.

3.4 | Min-Max Normalisation

The data were normalised by limiting the values to a range of 0 to 1, using the min-max approach. Their goal in doing this was to improve the classifiers' performance and lessen the impact of extreme cases. The following Eq. (1) was used to accomplish the normalisation:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

where X stands for the feature's initial value, X' is the normalized value, X_{min} is the feature's lowest possible value, and X_{max} is the highest amount of the same.

3.5 | Data Splitting

Data splitting involves separating a dataset into smaller pieces, usually for the purpose of training, validating, and testing ML models. Split the data in half, with 80% going into training and 20% into testing. Ensuring there's enough data for both training and testing the model's effectiveness is important.

3.6 | Proposed Long Short-Term Memory (LSTM) Model

An example of a layered network that takes prior outputs as inputs for subsequent layers is the LSTM classifier, which is a kind of RNN. LSTM is able to process sequences of data rather than simply individual

points because of its feedback links [19]. An LSTM node consists of a cell, an Input Gate, an Output Gate, and a Forget Gate. Information flows through the cell, which is in charge of long-term value storage, according to the three gates. Three multiplicative gates are included in each of the recurrently connected memory blocks that make up LSTM layers. To guarantee the temporary data is utilised for a certain amount of time, gates execute a continuous write, read, and reset operation. The input of the unit, x_t, h_{t-1}, c_{t-1} and the output of the unit, h_t, c_t are updated as follows in Eqs. (2) to (7):

Gates:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (4)$$

Input Transform:

$$g_t = \tanh(W_g x_t + U_g h_{t-1} + b_g) \quad (5)$$

State update:

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (6)$$

$$h_t = o_t \odot \tanh(c_t) \quad (7)$$

The logistic sigmoid function is represented by σ in the above equations, while element-wise multiplication is denoted by $\odot$. The LSTM unit has an InputGate i_t , a ForgetGate f_t , an OutputGate o_t , a HiddenUnit h_t and a MemoryCell c_t at each TimeStep t . There is an additional bias denoted by b , and the learnt parameters are W and U . The input gate seems to regulate the amount of data written to each unit, the forget gate the amount of data wiped from the memory cell, and the output gate the amount of data shown from the internal memory.

3.7 | Evaluation Metrics

Model evaluation enables comprehension of model performance and simplifies the explanation and presentation of the models' output; it is an essential part of building ML projects. The goal is to show how closely the predicted values match the actual values, since it might be difficult to predict a regression model's exact value. This research evaluated the models using four performance evaluation criteria: R², MAE, RMSE, and MAPE.

3.7.1 | R-Squared

A high R² score suggests that the data were well-fitted by the regression model. R² may have values ranging from 0 to 1, where higher values indicate a better fit between the model and the data. A flawless prediction of the response data is indicated by an R² value of 1. In contrast, a model that fails to explain any variability in the response data around its mean has an R² value of 0. Eq. (8) may be used to determine R²:

$$R^2 = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (8)$$

3.7.2 | Mean Absolute Error (MAE)

A common metric for assessing the accuracy of a prediction model is the MAE. Without taking the direction of the mistakes into account, it calculates the average magnitude of the forecasts. A lower MAE number indicates that the performance is better. To get the MAE, use the Eq. (9).

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - y_i^P| \quad (9)$$

The actual value is denoted by y_i , the projected value by y_i^P , And the number of observations is known as n .

3.7.3 | RMSE (Root Mean Squared Error)

The square root of MSE is represented by this metric. The degree to which the model's predictions differ from the actual values is measured by RMSE. Lower RMSE values show better model performance. The RMSE Eq. (10) is:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y_i^P)^2} \quad (10)$$

3.7.4 | Mean Absolute Percentage Error (MAPE)

MAPE calculates mistakes as a percentage, particularly the average percentage difference between the dataset's intended objectives and forecasts. The MAE returned as a percentage is another way to look at MAPE Eq. (11).

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left(\frac{(y_i - y_i^P)^2}{y_i} \right) / * 100 \quad (11)$$

These metrics together give information about the model's accuracy and efficacy in predicting the target variable.

4 | Results and Discussion

An evaluation of the outcomes is carried out in this section. To assess proposed models, use the metrics described above, which are compiled in Table 2. Python 3.11.5 is the programming language to utilise for creating traffic flow forecasting models. A single GPU node with an Intel Xeon E-2146G CPU running at 3.50 GHz, 32 GB of RAM, and an NVIDIA Tesla V100-PCIE 16 GB GPU made up the experimental setting used in this work. Due to the high computing needs of DL models, the GPU is used for rapid model training. The propose model achieves a high R^2 score of 93.45%, indicating that it can effectively capture the variance in traffic flow patterns. Furthermore, the MAE of 5.504 and the RMSE of 6.857 reflect low levels of prediction error, suggesting that the model provides accurate and consistent forecasts. Additionally, the MAPE of 0.174 confirms the model's robustness in maintaining reliable prediction accuracy across different traffic volume levels, making it suitable for real-world urban traffic management applications.

Table 2. Experiment Results of Proposed Models for Urban Traffic Flows Prediction on the Traffic Dataset.

Performance Matrix	Long Short-Term Memory (LSTM)
R^2	93.45
MAE	5.504
RMSE	6.857
MAPE	0.174

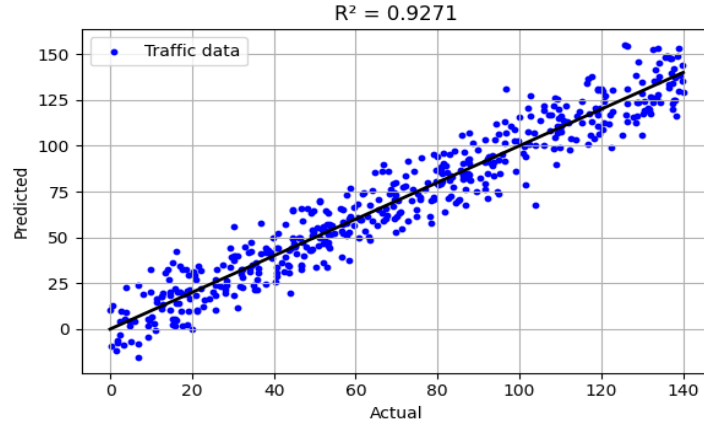


Figure 4. Actual and Predicted Traffic Flow for LSTM.

Figure 4 illustrates that the forecasted values closely follow the actual values, aligning well along the diagonal reference line, indicating strong model performance. The model successfully captures temporal relationships in traffic data and has a good prediction accuracy, as shown by the R^2 score of 0.9271. The blue data points, labeled as “Traffic data,” demonstrate a tight clustering around the regression line, signifying that the LSTM model is capable of modelling complex traffic flow patterns with minimal deviation. The success of deep learning methods in predicting traffic flows in real time is shown by this performance.

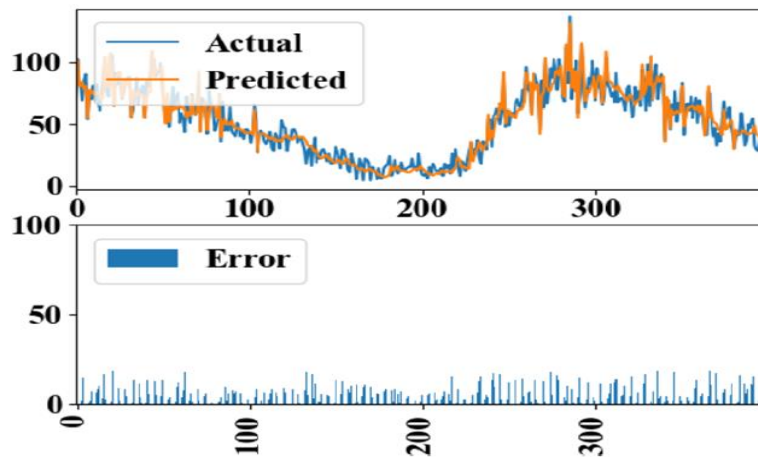


Figure 5. Actual vs. Predicted Values with Error Distribution for LSTM.

Figure 5: actual and predicted curves mostly align well, indicating that the model has captured the trend of the data reasonably accurately despite some fluctuations and deviations, especially in regions with higher variance around the middle to the later time steps (approximately after index 200). The lower subplot displays the error distribution as a bar plot, revealing that while most prediction errors remain relatively low, there are occasional spikes indicating points where the model struggled to match the true values precisely. Overall, this visualization demonstrates that the model performs well in general but has room for improvement in reducing prediction variance and handling sudden changes more smoothly.

4.1 | Comparative Analysis

This section evaluates the suggested model's performance in comparison to a few cutting-edge baseline ML models. A comparison of several prediction models used to improve urban traffic flows using the traffic information is shown in Table 3. The LSTM model shows the most significant results with the biggest R^2 value of 93.45 and the smallest MAE of 5.504, implying a more accurate and less estimating error. Neural Network (NN) model ranks closely with a high R^2 of 92 and an MAE of 7.2, depicting high predictive abilities. The RF model has a rather high R^2 value of 0.90, yet a significantly higher MAE value

of 81.47, implying more inaccurate predictions despite explaining a large percentage of data variance. Feedforward Neural Network (FNN) outputs an R^2 of 75, and a significantly low MAE of 0.45, suggesting irregular or overfitting. The LSTM model is, overall, the most balanced and successful approach with which one can make secure assumptions about the flow of traffic in a city.

Table 3. Comparison of different models for optimising urban Traffic Flows using the Traffic Dataset.

Models	R^2	MAE
FNN [20]	75	0.45
RF[21]	90	81.47
NN[7]	92	7.2
LSTM	93.45	5.504

The LSTM model proposed is very useful in the prediction of traffic flow, based on its outstanding performance measured against other competing models such as Neural Networks, Random Forest and Feedforward Neural Networks in several evaluation metrics and its very low prediction errors as compared to other methods. The fact that the model was capable of reliably forecasting and duplicating observed values demonstrates that it can accommodate traffic observations with complex temporal dependencies, and is precise over a broad range of traffic volumes, thus it is applicable both during peak and off-peak conditions. This high accuracy, low error rates, and local temporal pattern recognition makes the LSTM model specifically applicable to real-world urban traffic management, allowing traffic authorities to make informed decisions toward traffic optimization, congestion management, and infrastructure planning and is computationally efficient enough to be used in practice through standard hardware resources, offering the most balanced and effective approach in all measures of predictive reliability in varied traffic conditions.

5 | Conclusion and Future Study

The Real-Time Traffic Prediction and Optimization Program is strong and algorithm-based. It uses weather APIs, traffic sensors and GPS data in real time. This data is processed by sophisticated ML algorithms that properly forecast the flow of traffic. Based on predictions, the system optimises routes to ease urban congestion. This study introduces an LSTM model designed to predict urban traffic patterns in Great Britain using comprehensive street-level traffic data. The proposed model achieved a high R^2 score of 93.45%, along with a low MAE of 5.504, RMSE of 6.857, and MAPE of 0.174, outperforming benchmark models including Neural Network, Random Forest, and Feedforward Neural Network. These results highlight the LSTM's superior capability to capture temporal trends and cyclical patterns in traffic data. However, the research is limited by its dependence on a single dataset and its inability to fully capture unpredictable factors such as accidents or extreme weather events. Future work will focus on integrating real-time data sources, employing hybrid DL frameworks, and validating the model across diverse urban environments to enhance its resilience, adaptability, and practical utility for smart and sustainable urban traffic management systems.

Author Contribution

All work, conceptualization, methodology, analysis, software, writing, and validation were carried out by Khursheed Mohammed Hussain. The author has read and approved the final manuscript. All terms were described in the CT file.

Funding

This research has no funding source

Data Availability

The data (Road Traffic Dataset) was open from the Kaggle and is available at: <https://www.kaggle.com/datasets/arashnic/road-traffic-dataset>.

Conflicts of Interest

The authors declare no conflict of interest.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

References

- [1] Patel, N. (2021). Sustainable Smart Cities: Leveraging IoT and Data Analytics for Energy Efficiency And Urban Development. *Journal of Emerging Technologies and Innovative Research*, 8(3), 313–219.
- [2] Garg, S. (2021). Next-Gen Smart City Operations with AIOps & IoT: A Comprehensive look at Optimizing Urban Infrastructure. *Journal of Advances in Developmental Research (IJADR)*, 12(1), 1–9.
- [3] S, V., & P, D. B. P. (2024). A Comprehensive Review on Real-Time Traffic Signal Coordination Using Machine Learning: Optimizing Urban Traffic Flow for Efficiency and Sustainability. *International Journal for Research in Applied Science and Engineering Technology*. <https://doi.org/10.22214/ijraset.2024.58160>
- [4] Ata, A., Khan, M. A., Abbas, S., Ahmad, G., & Fatima, A. (2019). Modelling Smart Road Traffic Congestion Control System Using Machine Learning Techniques. *Neural Network World*, 29(2), 99–110. <https://doi.org/10.14311/NNW.2019.29.008>
- [5] Salaymeh, A., Schwiebert, L., & Remias, S. (2021). *Multi-Agent Reinforcement Learning for Optimizing Traffic Signal Timing*. <https://doi.org/10.5121/csit.2021.110102>
- [6] Sagili, S. R., & Kinsman, T. B. (2024). Drive Dash: Vehicle Crash Insights Reporting System. *2024 International Conference on Intelligent Systems and Advanced Applications (ICISAA)*, 1–6. <https://doi.org/10.1109/ICISAA62385.2024.10828724>
- [7] Rohilla, C. (2024). *Study of Designing Road Traffic Control System Using Machine Learning and Implementation Will Be in Python*. 15(3), 365–376. <https://doi.org/10.17605/OSF.IO/2P46A>
- [8] Navarro-Espinoza, A., López-Bonilla, O. R., García-Guerrero, E. E., Tlelo-Cuautle, E., López-Mancilla, D., Hernández-Mejía, C., & Inzunza-González, E. (2022). Traffic Flow Prediction for Smart Traffic Lights Using Machine Learning Algorithms. *Technologies*, 10(1), 5. <https://doi.org/10.3390/technologies10010005>
- [9] Prajapati, N. (2025). The Role of Machine Learning in Big Data Analytics: Tools, Techniques, and Applications. *ESP Journal of Engineering & Technology Advancements*, 5(2), 16–22. <https://doi.org/10.56472/25832646/JETA-V5I2P103>
- [10] Majumder, R. Q. (2025). Machine Learning for Predictive Analytics: Trends and Future Directions. *International Journal of Innovative Science and Research Technology*, 10(4), 3557–3564. <https://doi.org/10.38124/ijisrt/25apr1899>
- [11] Miao, Z., & Liao, Q. (2025). IoT-Based Traffic Prediction for Smart Cities. *IEEE Access*, 13, 52369–52384. <https://doi.org/10.1109/ACCESS.2025.3552276>
- [12] Alshardan, A., Maashi, M., Alanazi, S. M., Darem, A. A., Mahgoub, H., Alliheedi, M. A., & Alshammeri, M. (2024). Mathematical Modeling and Simulation of Traffic Flow Control in Urban Environments Using Fuzzy Deep Neural Network With Optimization Algorithm. *IEEE Access*, 12, 158322–158332. <https://doi.org/10.1109/ACCESS.2024.3483846>
- [13] R G, P., J, G. H., V, S. B., M B, S. K., R G, P. A., M S, V., & Ramesh, P. (2024). Traffic Light Control System and Traffic Prediction Using Machine Learning with SVR and RFR. *2024 4th International Conference on Mobile Networks and Wireless Communications (ICMNWC)*, 1–7. <https://doi.org/10.1109/ICMNWC63764.2024.10872340>
- [14] Yang, Y., Zhou, D., & Xie, Z. (2024). Optimization of XGBoost Algorithm for Road Traffic Flow Regression Prediction Based on Four-Vector Algorithm. *2024 5th International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, 479–483. <https://doi.org/10.1109/ICAICE63571.2024.10863881>
- [15] Sattarzadeh, A. R., & Pathirana, P. N. (2024). Enhancing Adaptive Traffic Control Systems with Deep Reinforcement Learning and Graphical Models. *2024 IEEE International Conference on Future Machine Learning and Data Science (FMLDS)*, 39–44. <https://doi.org/10.1109/FMLDS63805.2024.00017>
- [16] Patil, A., Raorane, A., & Kundale, J. (2023). Enhancing Traffic Management with Deep Learning Based Vehicle Detection and Scheduling Systems. *2023 International Conference on Modeling, Simulation & Intelligent Computing (MoSICom)*, 223–227. <https://doi.org/10.1109/MoSICom59118.2023.10458787>
- [17] Thom, J., Thom, N., Sengupta, S., & Hand, E. (2022). Smart Recon: Network Traffic Fingerprinting for IoT Device Identification. *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, 72–79.

- <https://doi.org/10.1109/CCWC54503.2022.9720739>
- [18] Malali, N. (2025). *AI-Powered Data Preprocessing and Transformation Platform for Autonomous Data Cleaning, Advanced Fea.*
- [19] Nokhwal, S., Chilakalapudi, P., Donekal, P., Nokhwal, S., Pahune, S., & Chaudhary, A. (2024). Accelerating Neural Network Training: A Brief Review. *2024 8th International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence (ISMSTI)*, 31–35. <https://doi.org/10.1145/3665065.3665071>
- [20] Atassi, R., & Sharma, A. (2023). Intelligent Traffic Management using IoT and Machine Learning. *Journal of Intelligent Systems and Internet of Things*, 8(2), 08–20. <https://doi.org/10.54216/JISIoT.080201>
- [21] Rasulkhamedov, M., Tashmetov, T., & Tashmetov, K. (2024). *Forecasting Traffic Flow Using Machine Learning Algorithms*. 14. <https://doi.org/10.3390/engproc2024070014>

Disclaimer/Publisher's Note: The perspectives, opinions, and data shared in all publications are the sole responsibility of the individual authors and contributors, and do not necessarily reflect the views of Sciences Force or the editorial team. Sciences Force and the editorial team disclaim any liability for potential harm to individuals or property resulting from the ideas, methods, instructions, or products referenced in the content.