Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606

Volume 10 Issue 02
2026

Research Article
https://doi.org/10.56979/1002/2026/1302

Smart Traffic Flow Control System for City Governance Driven by Neural Network Methods

Sundus Munir[1], Maria Tariq[2], Nageen Naeem[2], Muhammad Asif Saleem[3*], and Tahir Alyas[2]

[1]Department of Criminology, Lahore Garrison University, Lahore, Pakistan.
[2]Department of Computer Science, Lahore Garrison University, Lahore, Pakistan.
[3]Department of Information Technology, Lahore Garrison University, Lahore, Pakistan.
[*]Corresponding Author: Muhammad Asif Saleem. Email: asif.saleem@lgu.edu.pk

Received: November 30, 2025 **Accepted:** February 26, 2026

Abstract: Recent advancements in intelligent transportation systems (ITS) like Internet of Things (IoT), machine learning, and deep learning methods are employed for effective traffic flow prediction governing smart traffic control. In this paper, a deep-learning-based multi-neural network method framework is proposed to short-term traffic forecasting which called as Smart Traffic Flow Control System for City Governance driven by Neural Network Methods(STFCS–NNM). With respect to the key difference it makes, the method employs recurrent neural networks (RNNs), namely long short-term memory (LSTM) and gated recurrent unit (GRU) architectures to recognize temporal dependencies and nonlinear correlations within traffic data. The models are trained and checked on real-world traffic data collected from the Caltrans Performance Measurement System (PeMS) aggregated in 5-minute intervals at IoT-enabled sensors. Compared with traditional statistical models such as ARIMA and SARIMA, experimental results show that the proposed models have better prediction accuracy, which validates their effectiveness in the field of traffic flow dynamics modeling and intelligent traffic management.

Keywords: MAE (Mean Absolute Error); Gated Recurrent Unit (GRU); MAPE (Mean Absolute Percentage Error); Traffic Congestion Control (TCC); Long Short-Term Memory (LSTM); RMSE (Root Mean Square Error)

---

1. Introduction

A smart system refers to a system that can sense, actuate, analyze, and control a situation and also make decisions according to the available data in an adaptive and predictive manner. Actions performed are referred to as smart actions. Smart systems have sensors and actuators and can perform operations autonomously [1]. Smart cities use various technologies to provide services to their residents and solve their problems. A smart city can include multiple factors like improved social services, improved traffic control systems, a smart weather forecasting system, smart buildings, sustainability, a smart health care system, and so on [2]. Different Communication technologies and neural networks are combined to make smart systems to provide better use of resources and also a low cost of these services to their residents [3]. Smart TCC is the need of the current era to make life effortless on roads. The number of vehicles on the road is increasing day by day, especially four-wheelers, it needs of the hour to smooth the flow of traffic on roads. This congestion also causes a rise in temperature and noise pollution. Due to all these reasons, the waiting time for travelers has increased, which leads to frustration.

Timely or accurate traffic flow information is needed for government agencies, individual travelers, and business sectors. This information can help travelers make better decisions for travel, reduce carbon emissions, alleviate traffic congestion, and improve the efficiency of traffic operations [4]. The data for traffic flow prediction depends on real-time and historical data, which is collected from various sources such as cameras, radars, Global Positioning System (GPS), social media, crowdsourcing, etc. New emerging traffic sensor technologies and classical traffic sensors are used to collect transportation data. Therefore, transportation management is now considered more data-driven [5][6]. In recent years, deep learning and the machine learning methods have gained a lot of attention in both industrial and academic fields. Deep architecture or multiple-layer architecture is used to extract integral features from data and can discover structure in data [7]. These algorithms do not need any prior knowledge to represent traffic features and also have good performance for the prediction of traffic flow. In this paper, we have addressed traffic flow prediction with neural networks. The drivers can take alternate routes according to the predicted data. Section II represents the related work; Section III explains the proposed methodology, and finally, Section IV represents the conclusion.

2. Literature Review

Traffic congestion control is a recent and problematic issue nowadays, as the number of vehicles is increasing on roads day by day. Various applications are available to handle these, such as data analysis and wireless sensor networks, but these technologies have their own cost for installation. Infrared sensors can be used to provide early warnings to drivers so they can choose alternate routes to prevent congestion [8]. Saturation occurs due to the less efficient systems for public transportation. It includes multiple factors, such as a rise in population, a rise in the number of vehicles on roads, etc. Another foremost reason is the immense export of vehicles [5]. In smart cities literature, safe and smooth traffic flow is the main concern to develop smart traffic congestion management system. This intelligent framework can be developed by integrating various communication technologies and the Internet of Things. The smart transportation system has many applications, such as reducing greenhouse gas emissions, reducing pollution, increasing road security and safety, traffic flow monitoring, saturation detection, weather conditions of the route, minimum travel time, alternate routing, fuel consumption efficiency, noise monitoring, air pollution detection, emergency management, etc. Scalable Enhanced Road Side Unit (SERSU) was used for the framework of climate change data and the management of versatile traffic control. The modules of (SERSU) were placed at various intervals on roadsides that caught the sensor motions of another module [9].

Traditional solutions are used to monitor transportation systems through various sources, such as speed trackers and CCTV cameras, which are used to monitor vehicle speed, human tracking, pollution check, classical traffic control lights, etc. These conventional solutions cannot ensure the traffic flow because of vehicular density on roads [10]. Nowadays, Automobile industries are not mechanical, but rather they are electronic and automated. These automobiles are fully equipped with multiple kinds of electronic control units, e.g., braking systems, rearview cameras, power steering, etc. In the current era of technology, a combination of mobile internet and cars has made a huge impact on recent research trends. Integration of GPS apps in mobile phones makes it very easy for drivers to follow a route by navigation. Apart from that, this smartphone technology makes it very easy for drivers to use multiple electronic devices for different purposes, like listening to the radio, sending text messages, surfing the internet, navigation, watching videos, etc. Smartphones can be considered a complete package for all of these utilities. Due to their portability and various uses, it is highly used in the automobile industry [11].

For the past few years, researchers have been working on the prediction of short-term traffic flow. The previous methodologies used models such as the support vector regression model, time series model, Kalman filter model, and hybrid combination model [12]. In recent years, artificial intelligence and the deep learning neural networks have been used with the improvement of the computing power of computers. Neural networks have self-learning abilities to learn the features of traffic flow prediction. In 1970, the Autoregressive Integrated Moving Average Model (ARIMA) was used to predict the flow of traffic of urban expressway traffic [13]. CNN network based on a single hidden layer integrated with terror freed-back used to predict the traffic velocity[14]. Many researchers use different approaches to perform prediction on traffic flow control, like Bayesian Network, Naïve I and II models, SARIMA, ARIMA, ARMA, SVR, KNN, and ANN[15][16][17].

Contributions of the Study

The key contributions of this study are as follows:

- In the context of an Intelligent Transportation System (ITS), a unified framework is proposed for traffic flow prediction based on IoT-generated data.

- A comparison of deep learning models, with a specific focus on Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), against conventional statistical techniques like ARIMA and SARIMA based on consistent experimental conditions.
- The proposed models outperform the evaluation of realistic data from the PeMS dataset, thus validating their practical efficacy.

3. Materials and Methods

Our proposed (STFCS-NNM) system model uses LSTM and GRU for intelligent traffic congestion prediction. The system model is represented (STFCS-NNM) in **Figure 1.** Herein, firstly, the dataset is downloaded from the California Performance Measurement System (PeMS) website. The downloaded dataset is in raw form, which needs to be pre-processed. In the Pre-processing layer, outliers and missing values were handled, and performed. Afterwards, the data was passed to the training layer. In the training layer, LSTM and GRU recurrent neural networks were used to train the dataset. The performance layer is used to evaluate the overall performance of the trained model in terms of MAPE, MAE, RMSE, R2, and Variance Score.

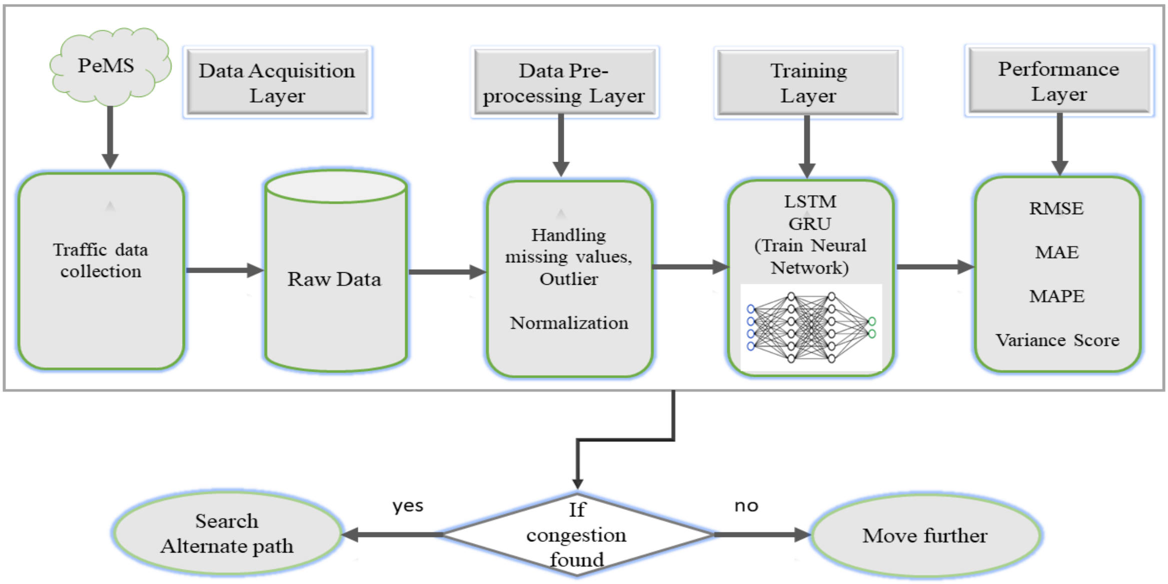

Figure 2. Proposed (STFCS-NNM) System Model

Here, LSTM and GRU deep learning models are presented to predict traffic flow congestion, which are famous deep learning models. The input parameters are taken as input to the neural network, and the outcome is the prediction of traffic flow. We have used four lanes as input data of traffic flow, and the number of hidden layers was four, and the hidden units were 5. Table 1 represents the output and input parameters for the proposed model.

Table 1. Input, Output Parameters for Proposed Model

Parameter Names	Description
Hidden Layer	04
Hidden Unit	05
Input 1	Lane 1 Flow
Input 2	Lane 2 Flow
Input 3	Lane 3 Flow
Input 4	Lane 4 Flow
Output	Congestion

3.1. Long Short-Term Memory

LSTM comes under the umbrella of a Recurrent Neural Network (RNN), which is based on deep learning and is broadly used in problems such as speech recognition, sequence prediction problems, language modeling, etc. The basic working flow of the LSTM is that it takes a sequence of vectors as input and processes it one by one, while processing it passes information from the previous hidden state to the next step of the sequence. Hidden states access the neural network memory that holds information on the previous data that the network has seen before. The hidden state is calculated as follows: first, the input from the previous hidden state is combined to form a vector, which has information on the current input from the previous inputs. The vector that goes to the tanh activation in the output is the new hidden state or the memory of the network. The tanh activation is used to help regulate the values flowing through the network. The tanh function squishes values to always be between -1 and 1. The tanh function ensures that the values remain between -1 and 1, thus adjusting the neural network output.

A common LSTM mainly consists of three gates through which data comes from outside. The input gate is responsible for taking input from the output of the LSTM cell in the last iteration. Forget gate chooses when to forget the output findings and finds out the time lag for the input sequence. The output gate is responsible for taking all the calculated results and generating output. Gates learn what info is relevant to keep or ignore during training, and these gates contain sigmoid activation. The Linear Regression layer is operated on the output layer of the LSTM to calculate the various performance measures. The general structure of LSTM is presented in **Figure 3.**

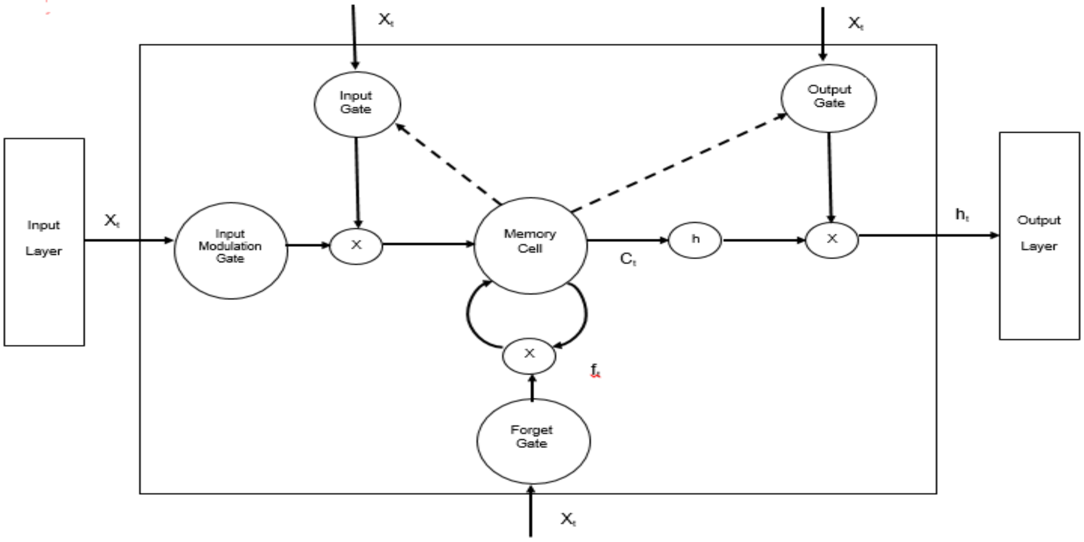

Figure 4. General Structure of LSTM

3.2. Mathematical model of LSTM

The mathematical model of LSTM can be represented as the following equation, where the input flow sequence can be represented as $y = y_1, y_2, \ldots\ldots y_t$ and hidden state can be denoted as $h = h_1, h_2, \ldots\ldots h_t$ and output sequence as $z = z_1, z_2, \ldots\ldots z_t$ The LSTM can be calculated by the following equations.

$$z_t = W_{hz}h_t + b_z \quad (1)$$

$$h_t = H\ (W_{yh}y_t + W_{hh}W_{t-1} + b_h) \quad (2)$$

Where b signifies bias vector, W represents weight matrices, e.g. W_{yh} denotes the input hidden matrix, and the hidden layers functions are denoted by H. These states are computed in the following mathematical equations:

$$i_t = \sigma\ (W_{yi}y_t + W_{hi}h_{t-1} + W_{ai}a_{t-1}\ b_i) \quad (3)$$

$$f_t = \sigma\ (W_{yf}y_t + W_{hf}h_{t-1} + W_{af}a_{t-1}\ b_f \quad (4)$$

$$a_t = f_t a_{t-1} + i_t g\ (W_{ya}y_t + W_{ha}h_{t-1} + b_a \quad (5)$$

$$o_t = \sigma\ (W_{yo}y_t + W_{ho}h_{t-1} + W_{ao}a_t\ b_o \quad (6)$$

$$h_t = o_t\ h\ (a_t) \quad (7)$$

The objective function that is used to minimize the square of error can be computed as follows

$$e_t = \sum_{t=1}^{n}(y_t - p_t)^2 \quad (8)$$

Where y_t denotes the input data flow and p_t denotes predicted data flow.

3.3. Gated Recurrent Unit

GRU is the type of RNN that maintains the structure of the network simpler, significantly increases speed, and also keeps the RNN prediction performance. By inheriting the advantages of RNN, it models long-distance dependent information effectively and automatically learns features of the data. GRU is very much like LSTM, but very simple to implement and compute. It was proposed in 2014, and it's a powerful recurrent neural network [18]. The general structure of GRU is based on two gates: one is a reset gate, and the other is an update gate. The Reset gate works in the same way as the forget gate works in LSTM, and the update gate is responsible for determining the amount of previous information that should pass to the next state. GRU saves a lot of time in training as it has one less gate as compared to LSTM, which reduces some matrix multiplication.

3.4. Mathematical model of GRU

$$R_t = \sigma\,(W_r .\ [\hbar_{t-1}, x_t]) \tag{9}$$

$$z_t = \sigma\,(W_z .\ [\hbar_{t-1}, x_t]) \tag{10}$$

$$\hbar_t = (1 - z_t) * \hbar_{t-1} + z_t * \widehat{\hbar}_t \tag{11}$$

$$\widehat{\hbar}_t = tanh\,(\,W.[r_t * \hbar_{t-1}, x_t]) \tag{12}$$

3.5. Dataset

The dataset used for the experiment is taken from PeMs (Caltrans Performance Measurement System) database [19-20], which is a publicly available traffic database provided by the California Department of Transportation, and a deep architecture model is applied to the collected data. This real-time traffic data is collected from over 15000 individual detectors every 30s and these detectors are deployed in California statewide in freeway systems. The data detected by each detector station is aggregated into 5-minute intervals each, and we have used 5-minute traffic flow for the study. The dataset includes multiple features such as speed, occupancy, traffic flow, and timestamp information. In this paper, we used five weeks of data on traffic flow for experimental purposes, collected on weekdays in the year 2017. The details of the selected time periods are as follows: Week 1 starts from (28th of August to September 3rd), Week 2 from (4th to 10th September), Week 3 (11th to 17th September), Week 4 (18th to 24 September), and Week 5 (Sep 25th to October 3rd). The data set time interval is 5 min, and each week consists of 1,753 samples, resulting in a total of 8,765 samples used in this study. Although the data is collected through IoT-based sensors, the proposed model is tested in an offline manner, meaning it works using historical datasets rather than with real-time systems.

4. **Results**

This section covers the necessary details of implementation along with the results. We used the proposed (STFCS-NNM) as our data is time-series data in which GRU and LSTM are implemented, and compared the performance with the ARIMA, SRIMA, and ARMA models. We also compare our model (STFCS-NNM) performance with LSTM and GRU, work done previously [21]. We evaluate regression results by using four indicators of performance: MAPE (Mean Absolute Percentage Error), Variance Score, RMSE (Root Mean Square Error), and MAE (Mean Absolute Error) [21]. The first indicator, MAPE, is the absolute average value that gives a percentage error value, which ranks the model prediction result. The MAPE is calculated as mentioned in Equation 1.

$$MAPE = \left[\frac{100\%}{total-number} \sum_{k=1}^{total-number} \left(TF_k - T\widehat{F}_k\right)/TF_k\right] \tag{13}$$

The second indicator is score variance used for the variability measure mentioned in Equation 2.

$$VAR(TF_m) = \left[\frac{1}{total-number} \sum_{k=1}^{total-number} \left(TF_{km} - T\widehat{F}_k\right)^2\right] \tag{14}$$

The third indicator is root mean square error; another name is "Standard Error", used to compute the ratio. The ratio is between observed and true values. Equation 3 is used for the calculation of RMSE.

$$RMSE = \left[\frac{1}{total-number} \sum_{k=1}^{total-number} \left(TF_k - T\widehat{F}_k\right)^2\right] 1/2 \tag{15}$$

The last indicator, MAE, is the absolute error average, calculated as mentioned in Equation 4. In each equation TF_k is the actual traffic flow data set value and $T\widehat{F}_k$ is the predicted traffic flow value.

$$MAE = \left[\frac{1}{total-number}\sum_{k=1}^{total-number}\left(TF_k - \hat{TF}_k\right)\right] \quad (16)$$

Let’s our data sequence for traffic flow is denoted as $\{TF\ time\ k\}$, where $k = 1,2,3,\ldots,\text{end} - \text{variable}$, and $\text{time} = 1,2,3,\ldots\text{end} - \text{time}$. Our problem objective is to predict the traffic flow congestion at time + (time interval, where delta is the horizon of prediction. In previous research, the scale of aggregation, time is in minutes, like {3, 5, 10, 15}. We used 5-minute intervals in our model input. Our proposed model's effectiveness is evaluated by four different indicators of performance. The dataset was divided into 80% data for training and 20% data for testing purposes. We implemented our model in Python and used the Keras Library. In this research, the performance of the proposed (STFCS-NNM) model is measured by using various statistical parameters. Performance is measured based on MAPE (Mean Absolute Percentage Error), Variance Score, RMSE (Root Mean Square Error), and MAE (Mean Absolute Error). Hyperparameters were chosen using configurations reported in the literature for LSTM and GRU-based models. The models were trained with a fixed architecture and standard training settings, using the same configuration for all experiments.

Table 2 presents the performance comparison between GRU and LSTM models using various evaluation metrics. The GRU model performs better with lower RMSE (10.36) and MAE (7.69) values over LSTM (RMSE=11.48, MAE=8.55). Whereas LSTM displays a slightly lower MAPE, GRU leads in overall accuracy at 93%, while LSTM attains 92%. Moreover, GRU gets better values of R^2 (0.934) and variance score (0.934), which confirms a greater fit and generalization of the model. The GRU model achieves improved predictive performance, with lower MAE and RMSE values and higher accuracy and R^2 compared to LSTM. The GRU demonstrates superior performance across most evaluation metrics, despite LSTM showing slightly lower MAPE.

Table 2. Performance comparison of LSTM and GRU models for traffic flow prediction

Metric	**LSTM**	**GRU**
MAPE (%)	19.52	24.70
MAE	8.55	7.69
RMSE	11.48	10.36
R^2 Score	0.919	0.934
Variance Score	0.925	0.934
Accuracy(%)	92	**93**

The resulting plot of LSTM and GRU, along with the original data are shown in Figure 3. Afterward, the results of our proposed (STFCS-NNM) system model are compared with the previous work done by another researcher for the effectiveness of our study.

The resulting plot of LSTM and GRU, along with the original data are shown in **Figure 5:**

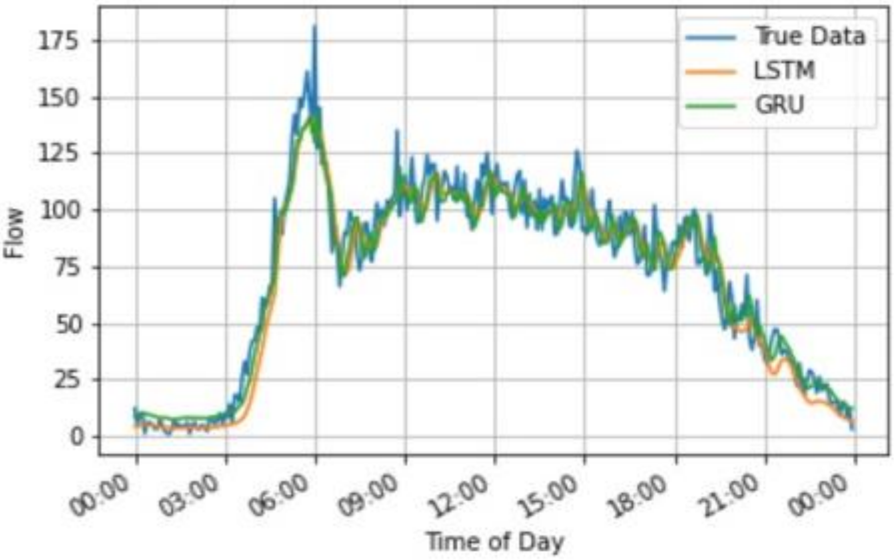

Figure 6. LSTM and GRU result values

Table 3 shows that our proposed (STFCS-NNM) model results are more accurate, which makes our model more feasible in traffic flow congestion prediction. Our proposed (STFCS-NNM) model gave better results as compared to the previous approach [19-20].

The result comparison with previously implemented approaches is shown in **Table 3.**

Table 4. Proposed Model (STFCS-NNM) Contrast with literature

Study	Model	MAE	MSE
X. X. et. al [21]	ARIMA	19.18	841.01
	LSTM	18.13	710.05
	GRU	17.21	668.93
Proposed (STFCS-NNM)	LSTM	8.55	11.48
	GRU	7.69	10.36

The proposed (STFCS-NNM) system model with LSTM and GRU has better results as compared to previously implemented ARIMA, LSTM, and GRU. GRU has superior performance across most evaluation metrics as compared to LSTM.

Thus, the proposed STFCS-NNM framework has practical contributions for intelligent traffic management system. The forecasted traffic flow can be used to optimize the timings of these signals where adaptive control is performed based on congestion levels in real-time/ near real-time. Moreover, the model has also the potential to be used by route guidance systems in finding congested routes and guiding drivers through alternative paths to alleviate travel delays. Forecasting results can also be used by traffic authorities to plan for future conditions, thereby aiding mitigation of congestion as well as assisting urban traffic planning in smart cities.

5. Discussion

The proposed STFCS-NNM framework is computationally efficient. LSTM & GRU models are comparatively lightweight deep learning architectures requiring relatively fewer compute resources compared to more complex models like the Transformer-based approaches. The inference process is fast when trained, rendering the proposed models suitable for real-time traffic flow prediction. In a smart city deployment scenario, the system can process incoming traffic data generated by IOT and provide predictions in time for better decision-making during traffic control. Yet, this computational cost of the training phase is potentially higher than that of other approaches and could be performed offline, whereas a subsequently deployed system requires only efficient real-time inference. Although the results are promising, it only performs an offline evaluation of the framework on historical traffic data, which may limit its application in real-time scenarios. Moreover, the experiments are performed on a part of PeMS dataset in a limited extent. Nevertheless, the framework has scalability potential to be extended to larger datasets, further sensors, and potentially real-time traffic scenarios with adequate integration into systems.

6. Conclusion

In smart city governance, the prediction of traffic flow is a critical problem in Intelligent Traffic Systems (ITS) in the current era. We have proposed a deep learning-based neural network for the prediction of traffic congestion. LSTM and GRU recurrent neural networks are used to predict the 5-minute flow traffic on a PeMS dataset and compared with previous approaches. As compared to previous approaches, our proposed model gave better accuracy and performance. In future work, more deep learning algorithms can be used to predict the traffic flow on different datasets to observe their effectiveness.

Funding: No specific funding for this research
Data Availability Statement: Data will be available on request
Acknowledgments: Thanks to all authors
Conflicts of Interest: The authors declare no conflict of interest.

References

1. C. González García, D. Meana Llorián, C. Pelayo G-Bustelo, and J. M. Cueva-Lovelle, "A review about Smart Objects, Sensors, and Actuators," Int. J. Interact. Multimed. Artif. Intell., vol. 4, no. 3, p. 7, 2017, doi: 10.9781/ijimai.2017.431.
2. U. Kingdom, "Preprint Traffic Management and Forecasting System Based on 3D GIS."
3. M. I. Pramanik, R. Y. K. Lau, H. Demirkan, and M. A. K. Azad, "Smart health: Big data enabled health paradigm within smart cities," Expert Syst. Appl., vol. 87, pp. 370–383, 2017, doi: 10.1016/j.eswa.2017.06.027.
4. N. Zhang, F. Y. Wang, F. Zhu, D. Zhao, and S. Tang, "DynaCAS: Computational experiments and decision support for ITS," IEEE Intell. Syst., vol. 23, no. 6, pp. 19–23, 2008, doi: 10.1109/mis.2008.101.
5. J. Zhang, F. Y. Wang, K. Wang, W. H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," IEEE Trans. Intell. Transp. Syst., vol. 12, no. 4, pp. 1624–1639, 2011, doi: 10.1109/TITS.2011.2158001.
6. C. L. Philip Chen and C. Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on Big Data," Inf. Sci. (Ny)., vol. 275, pp. 314–347, 2014, doi: 10.1016/j.ins.2014.01.015.
7. P. Luo, "Learning deep architectures via generalized whitened neural networks," 34th Int. Conf. Mach. Learn. ICML 2017, vol. 5, pp. 3500–3508, 2017.
8. Y. Joshi, A. Joshi, N. Tayade, P. Shinde, S. Rokade, and A. Professor, "IoT Based Smart Traffic Density Alarming Indicator," Int. Res. J. Eng. Technol., vol. 3, no. 10, pp. 1086–1089, 2016, [Online]. Available: www.irjet.net.
9. A. Al-Dweik, R. Muresan, M. Mayhew, and M. Lieberman, "IoT-based multifunctional Scalable real-time Enhanced Road Side Unit for Intelligent Transportation Systems," Can. Conf. Electr. Comput. Eng., pp. 1–6, 2017, doi: 10.1109/CCECE.2017.7946618.
10. J. Islam and Y. Zhang, "A Novel Deep Learning Based Multi-class Classification Method for Alzheimer's Disease Detection Using Brain MRI Data," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 10654 LNAI, no. November, pp. 213–222, 2017, doi: 10.1007/978-3-319-70772-3_20.
11. Y. Wang, H. Ou, and J. Zhang, "Realization of Intelligent Traffic Control Management System Based on Internet of Things RFID Technology," no. Emcs, pp. 405–409, 2016, doi: 10.2991/icsnce-16.2016.30.
12. Y. Lee, H. Jeon, and K. Sohn, "Predicting Short-Term Traffic Speed Using a Deep Neural Network to Accommodate Citywide Spatio-Temporal Correlations," IEEE Trans. Intell. Transp. Syst., vol. 22, pp. 1–14, Feb. 2020, doi: 10.1109/TITS.2020.2970754.
13. B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results," J. Transp. Eng., vol. 129, no. 6, pp. 664–672, 2003, doi: 10.1061/(ASCE)0733-947X(2003)129:6(664).
14. J. Wang, Q. Gu, J. Wu, G. Liu, and Z. Xiong, "Traffic speed prediction and congestion source exploration: A deep learning method," Proc. - IEEE Int. Conf. Data Mining, ICDM, vol. 0, pp. 499–508, 2016, doi: 10.1109/ICDM.2016.0061.
15. D. Mambo, M. Kimathi, and D. Seck, "Simulation of Traffic Congestion at Unsignalised Intersections using a Microscopic Traffic Flow Model," Int. J. Appl. Math. Sci., vol. 9, pp. 197–211, Jan. 2016.
16. W. Zhou, W. Wang, X. Hua, and Y. Zhang, "Real-Time Traffic Flow Forecasting via a Novel Method Combining Periodic-Trend Decomposition," Sustain., vol. 12, no. 15, pp. 1–22, 2020, doi: 10.3390/SU12155891.
17. B. Williams, P. Durvasula, and D. Brown, "Urban Freeway Traffic Flow Prediction: Application of Seasonal Autoregressive Integrated Moving Average and Exponential Smoothing Models," Transp. Res. Rec., vol. 1644, pp. 132–141, Jan. 1998, doi: 10.3141/1644-14.
18. K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," EMNLP 2014 - 2014 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf., pp. 1724–1734, 2014, doi: 10.3115/v1/d14-1179.
19. Aljaidi, M., Samara, G., Singla, M. K., Alsarhan, A., Hassan, M., Safaraliev, M., ... & Tavlintsev, A. (2024). A particle swarm optimizer-based optimization approach for locating electric vehicles charging stations in smart cities. *International Journal of Hydrogen Energy*, *87*, 1047-1055.
20. Al-Na'amneh, Q., Aljaidi, M., Nasayreh, A., Gharaibeh, H., Al Mamlook, R. E., Jaradat, A. S., ... & Samara, G. (2024). Enhancing IoT device security: CNN-SVM hybrid approach for real-time detection of DoS and DDoS attacks. Journal of Intelligent Systems, 33(1), 20230150.
21. C. Ma, Y. Li, R. He, F. Wu, B. Qi, and Q. Ye, "Route optimisation models and algorithms for hazardous materials transportation under different environments," Int. J. Bio-Inspired Comput., vol. 5, no. 4, pp. 252–265, 2013, doi: 10.1504/IJBIC.2013.055473.