Performance Evaluation of Deep Learning Models in Traffic Congestion Forecasting

Pramod Kumar Mishra
AKS University
Sherganj, Satna

Pinki Sharma
AKS University
Sherganj, Satna

Akhilesh A. Waoo
AKS University
Sherganj, Satna

ABSTRACT

Traffic congestion has emerged as one of the major challenges faced by expanding urban environments, resulting in increased travel delays, excessive fuel consumption, and environmental pollution. Accurate traffic forecasting plays an essential role in urban planning and the development of efficient transportation systems. This study applies an experimental approach to evaluate the performance of various deep learning models under different traffic scenarios. The experiments were conducted using the Google Colab platform to assess multiple deep learning architectures, including Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and Graph Neural Networks (GNN). Real-world traffic data consisting of traffic flow, density, speed, and temporal variables were utilized for model training and evaluation. Model performance was measured using standard evaluation metrics including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and coefficient of determination (R^2). Experimental findings indicate that LSTM and GNN models demonstrate superior predictive performance compared with conventional RNN and CNN approaches. The study contributes to the domains of deep learning, time-series forecasting, cloud-based experimentation, and Intelligent Transportation Systems (ITS).

General Terms

Algorithms, Artificial Intelligence, Performance Evaluation, Intelligent Transportation Systems

Keywords

Traffic Congestion, Deep Learning, Google Colab, Time Series Forecasting, Smart Cities, LSTM, GNN, RNN, CNN.

1. INTRODUCTION

1.1 Background

With the increase of vehicles, roadways become overloaded. This causes delays, extra fuel consumption, pollution, and inconvenience. The traffic flow dynamics also vary in time. For example, peak hours, weather conditions, construction works, accidents, and special occasions can impact the flow of cars. Therefore, early prediction of traffic congestion contributes to effective traffic management. Traditional forecasting approaches perform well when used in case of regular traffic data trends, but not for irregular or complex cases.

Deep learning approaches have a great potential because they rely on learning from big data.

Typically, LSTM neural networks are used in traffic congestion prediction as they learn previous traffic dynamics [1].

Also, hybrid models such as GSA-LSTM and PSO-Bi-LSTM are based on applying an optimization approach together with deep learning [2], [3].

GNN models are applicable in cases when traffic flow in road segments depends on each other [4].Recent researches pay attention to uncertain predictions and real-time traffic information [6], [8].However, most of the mentioned models have been tested on small-scale datasets.

They also require intensive computing resources. Thus, there is a necessity of comparative analysis of deep learning algorithms.

In this research, traffic congestion forecasting is carried out via application of deep learning models.

Their efficiency will be estimated based on such criteria as MAE, RMSE, MAPE, and R^2.

1.2 Problem Statement

Even though there have been significant developments in the algorithms of deep learning, selecting an effective model for predicting traffic congestion is quite a difficult decision to make. Presently, most of the scholarly works in this field are focused on individual models, and none of them make comparisons in a controlled environment.

The lack of a unified experimental platform is among the major weaknesses that plague the existing scholarly works. Most of the scholarly works rely on local computational platforms, which affect scalability and reproducibility. In the domain of AI and transportation, reproducibility is one of the most important things.

Moreover, there is very little literature that employs cloud platforms, such as Google Colab, to conduct extensive deep learning tests. Lack of these frameworks makes it impossible to verify the results of different studies.

1.3 Research Objective

The primary objective of the paper is to perform comparative analysis of the deep learning methods applied to the task of traffic congestion forecasting through experiments carried out on an identical platform.

In this paper the work will be done by testing and analyzing some models like RNN, CNN, LSTM, and GNN through the cloud platform in Google Colab. Additionally, the models will be compared in terms of accuracy using different error metrics in order to get a clear view of their actual performance.

1.4 Contributions

The main claim is that a comparative analysis of various deep learning models (RNN, CNN, LSTM, GNN) is done based on a unified framework.

• Experiments on a cloud-based platform (Google Colab) to make them reproducible.

• Analysis based on various performance measures (MAE, RMSE, MAPE, and R2).

• Comparison of temporal and spatial learning abilities of models.

• The most appropriate model to be used in predicting traffic congestion.

2. LITERATURE REVIEW

The latest progress in the domain of deep learning has considerably enhanced the performance of congestion prediction models in terms of their effectiveness. There have been several models utilized in this regard, which include Recurrent Neural Networks, Convolutional Neural Networks, LSTM networks, and GNN models. Even though LSTM networks are efficient in capturing the time-dependent nature of the data, there have been indications that GNN models can efficiently capture the spatial relations among traffic networks. Table 1 provides a comparative analysis of recent studies in traffic congestion prediction, emphasizing the applied models, experimental datasets, key findings, and research limitations to establish the motivation for the present study.

Table 1. Summary of Recent Studies on Traffic Congestion Prediction

Author(s) & Year	Method Used	Dataset	Key Contribution	Limitations
Dai et al. (2023)	Ensemble machine learning with decomposition and LSTM	Short-term traffic flow data	This study used an ensemble approach to manage complex traffic patterns..	It may be difficult to apply in small studies or by beginners.
Naheliya et al. (2023)	GSA-LSTM hybrid model	Short-term traffic flow dataset	This study combined LSTM with the Gravitational Search Algorithm	The model can improve accuracy, but it needs extra computation and careful parameter setting.
Chen et al. (2023)	CNN-based Model	PEMS Dataset	CNN model is used to extract local temporal patterns from traffic data.	Poor performance for long-term dependencies
Wang et al. (2024)	RNN, LSTM	Urban Traffic Dataset	Comparative study demonstrating LSTM's superior performance over RNN	Limited scalability due to local system usage
Fu et al. (2024)	Bayesian deep learning	Urban interrupted traffic flow data	This study used Bayesian deep learning for urban traffic prediction.	The model is more suitable for urban signal-controlled traffic and may need changes for highway traffic.
Zhao et al. (2024)	Attention-based LSTM	Large-scale traffic dataset	Improved temporal feature learning with an attention mechanism	High computational cost
Li et al. (2025)	Graph Neural Network (GNN)	METR-LA, PEMS-BAY	successfully captured the spatial connections between traffic nodes	Requires graph structure design
Zhou et al. (2025)	Spatio-temporal GNN	Smart city traffic data	For increased accuracy, spatial and temporal learning were combined.	Complex implementation

2.1 Critical Analysis

It can be seen from the literature review that the DL models have significantly improved traffic prediction's accuracy as compared to traditional models. It is also evident that the LSTM models have been widely acknowledged for their capability of learning temporal dependencies while GNNs have comparatively recently been able to learn spatial dependencies.

However, most existing studies suffer from various limitations, which include a lack of comparison of the performance of various models, a lack of reproducibility due to cloud-based computing, inability to combine temporal and spatial learning, and high complexity of hybrid and attention-based models.

2.2 Research Gap

- Not many studies offer a consistent comparison between RNN, CNN, LSTM and GNN.
- There is limited research that uses Google Colab or other cloud platforms to reproducibly experiment.
- Absence of standard evaluation structures among datasets.
- Lack of emphasis on the accuracy-computational efficiency tradeoff

3. METHODOLOGY

This research suggests a systematic and replicable approach to the assessment of the deep learning model performance on traffic congestion prediction. The methodology will provide a fair comparison across several models due to consistency in the data processing, model training, and assessment.

3.1 Research Framework

The general architecture of the research is built on the principle of a sequential pipeline for processing time series traffic data and predicting its values. It can be described as a process consisting of the following steps:

Data Gathering → preprocessing → Training the model → Prediction → Evaluation.

3.2 Problem Formulation

The task of predicting traffic congestion is framed as a time series prediction problem, whereby the future traffic pattern is estimated from past observations. Assuming that the traffic pattern sequence is $X = \{x1, x2, x3, \dots, xt\}$, the goal is to estimate future traffic status using a function $f(x1, xt - 1, \dots, xt - n)$, where $f(\cdot)$ is the mapping learned by the deep learning algorithms like RNN, CNN, LSTM, and GNN..

3.3 Dataset Description

The dataset used in this research project is based on traffic data that have been acquired from Bangalore City, and it represents the traffic conditions in the real world. It is made up of time series data collected at fixed intervals and contains relevant variables vehicle speed, traffic flow, and timestamp. Such as traffic volume, speed, traffic flow, and timestamp. Other calculated variables include hour of the day and day of the week. The dataset has been preprocessed and normalized before model training can be done as shown in Table 2.

Table 2. Description of Dataset Fields

S. No.	Feature Name	Description	Data Type
1	Timestamp	Date and time of traffic observation	DateTime
2	Traffic Volume	Number of vehicles passing a location	Numeric
3	Traffic Speed	Average speed of vehicles	Numeric
4	Traffic Flow	Vehicles per unit time	Numeric
5	Density	Number of vehicles per road segment	Numeric
6	Hour	Hour of the day	Integer
7	Day	Day of the week	Integer
8	Month	Month of the year	Integer
9	Year	Year of observation	Integer
10	Day of Month	Day of the month	Integer
11	Weekday/Weekend	Indicates weekday or weekend	Categorical
12	Holiday	Indicates whether it is a holiday	Binary
13	Weather Type	Weather condition (clear, rainy, etc.)	Categorical
14	Temperature	Atmospheric temperature	Numeric
15	Humidity	Humidity level	Numeric
16	Road Condition	Road condition (dry, wet, congested)	Categorical

3.4 Data Preprocessing

Preprocessing of data is done to enhance the quality of data and successful training of the model. The missing values are taken care of through forward-fill and interpolation, to ensure continuation in the data.

The values are scaled to the range [0,1] by using min-max normalization. A sliding window technique is used to transform the time-series data into supervised learning format. The data is separated into training and testing sets in the ratio 80:20.

These measures help to make sure that the input data is uniform and fits the deep learning models

3.5 Feature Engineering

The feature engineering is implemented to increase predictive performance of the models. To capture the temporal dependencies, lag features are created to capture the past traffic conditions. Periodic traffic patterns are represented by the temporal features like hour of the day and the day of the week. External conditions like weather conditions are also included where necessary to enhance model robustness.

3.6 Model Development

Four deep learning models are used to carry out a detailed assessment. Recurrent Neural Network is a sequential data modeling baselines model. One-dimensional convolution is used to implement Convolutional Neural Network and extract local temporal features.

Long Short-Term Memory the Long Short-Term Memory is constructed to handle the long-term dependencies in memory cells. Graph Neural Network models transportation networks between traffic nodes in the form of graphs. This choice allows examining the time and space features of traffic data.

3.7 Model Training

The same experimental conditions are used to train all models, in order to compare them fairly. The training is by a batch of 32 and Adam optimizer and Mean Squared Error loss function. The convergence behavior is used to determine the number of epochs. The hyperparameter optimization is conducted to optimize the model performance and all the models are tested on unseen test data.

3.8 Experimental Platform

Experiments will be conducted in the Google Colab platform, which offers an environment suitable for conducting deep learning experiments. It also offers GPU acceleration capabilities, which make it possible to train models fast. This makes sure that experiments are easily reproducible due to ease of sharing the experimental environment. Table 3 summarizes the experimental environment, model training parameters, and computational settings adopted to maintain consistency and enable reliable performance evaluation across all deep learning models.

Table 3. Experimental Configuration and Model Training Settings

Parameter	Value
Platform	Google Colab
GPU	Tesla T4 / Runtime
Train-Test Split	80:20
Batch Size	32
Optimizer	Adam
Loss Function	MSE
Epochs	XX
Learning Rate	XX

3.9 Evaluation Metrics

In terms of evaluation, the Mean Absolute Error, Root Mean Square Error, Mean Absolute Percentage Error, and R^2 score will be used. Such metrics will make it possible to assess the effectiveness and robustness of models.

3.10 Comparative Analysis

Comparison will be conducted fairly as all the models will undergo evaluation in the same environment. All the models will use the same set of data preprocessed similarly.

4. RESULTS AND ANALYSIS

Performance assessment for all the suggested neural networks was done on the basis of MAE, RMSE, MAPE, and R2 score. All neural networks models were subjected to the same experimental settings when training and testing for unbiased comparison results.The quantitative results show that the The model based on LSTM obtains the smallest MAE, RMSE, and MAPE and the largest R-squared value, thus exhibiting a better ability of prediction. It is reasonable because of its capability to detect and predict traffic data with their temporal dependency patterns.

In contrast, the GNN model has relatively poor prediction performance in comparison with LSTM, CNN, and RNN models due to higher values of errors and a lower R-square. CNN and RNN exhibit rather decent results, although the former detects only local temporal dependency and the latter performs worse than LSTM in dealing with long-term patterns as shown in Table 4.

Table 4. Comparative Performance of Deep Learning Models Based on MAE, RMSE, MAPE, and R²

Model	MAE	RMSE	R²	MAPE
RNN	3.414838821	4.876071783	0.956049006	5.868523768
CNN	3.889800278	5.332688728	0.947432056	6.281685655
LSTM	3.314152598	4.642455290	0.960159574	5.575886170
NN	14.388888360	17.884813300	0.408714354	18.925594330

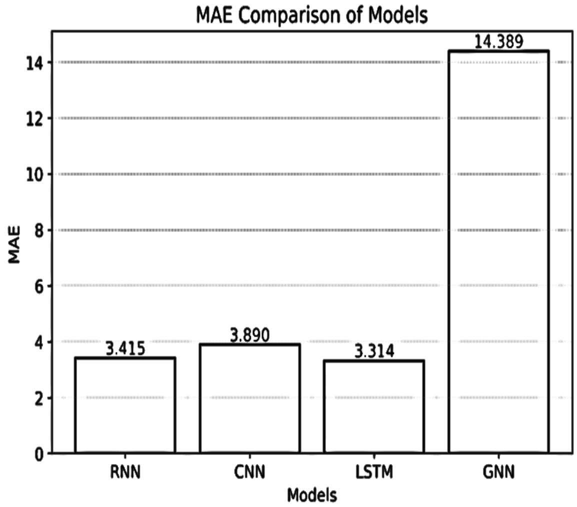

Fig 1: MAE Comparison of Models

Figure 1 presents the comparison of Mean Absolute Error (MAE) among the evaluated deep learning models. The LSTM model achieved the lowest MAE value (3.314), indicating superior prediction accuracy and better capability to capture temporal dependencies in traffic data. The RNN and CNN models showed comparable performance with slightly higher error values, whereas the GNN model recorded the highest MAE (14.389), indicating reduced prediction effectiveness under the current experimental configuration.

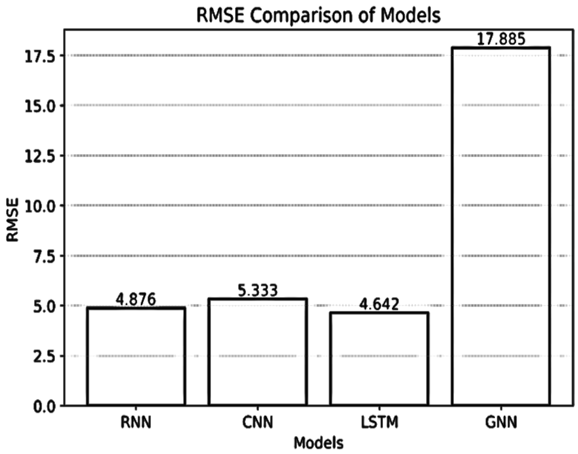

Fig 2: RMSE Comparison of Models

Figure 2 illustrates the Root Mean Square Error (RMSE) values for all evaluated models. Among the considered approaches, LSTM achieved the minimum RMSE value (4.642), demonstrating improved robustness and lower prediction deviation from actual traffic conditions. RNN and CNN exhibited moderate performance, while GNN showed substantially higher RMSE values, suggesting weaker generalization capability for this dataset.

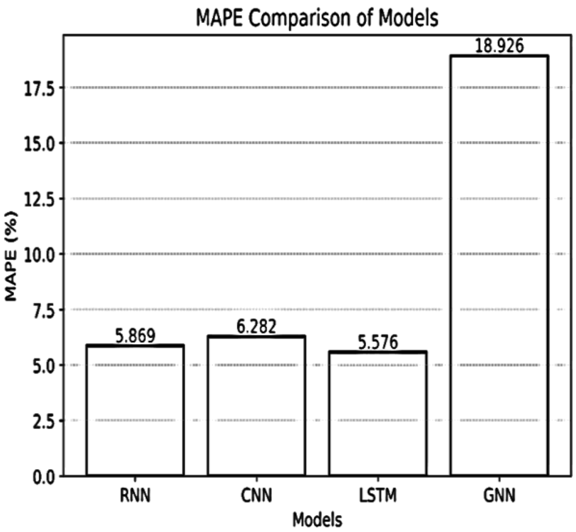

Fig 3: MAPE Comparison of Models

Figure 3 shows the Mean Absolute Percentage Error (MAPE) comparison across the deep learning models. The LSTM model achieved the lowest MAPE value (5.576%), indicating higher prediction reliability and lower percentage deviation from actual observations. RNN and CNN produced acceptable prediction performance, whereas GNN exhibited significantly larger percentage error, reflecting limitations in capturing traffic patterns accurately.

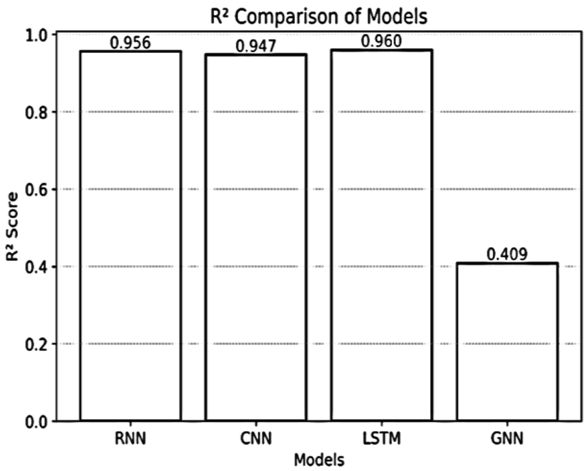

Fig 4: R^2 Score Comparison of Models

Figure 4 compares the coefficient of determination (R^2) values for all models. LSTM obtained the highest R^2 score (0.960), indicating the strongest explanatory capability and best fit to the observed traffic data. RNN and CNN also demonstrated strong predictive performance with high R^2 values above 0.94, while GNN recorded a substantially lower score (0.409), suggesting weaker alignment with actual traffic trends under the selected experimental settings.

The findings from Table 3 and the graphs below illustrate that the LSTM model is the best-performing model of the models assessed. The LSTM model exhibits low errors and high R^2 values, suggesting that the model is consistent with the actual traffic values.

Comparison between the models using graphs also reflects the same findings, as illustrated above. Based on the graph analysis, it can be seen that LSTM outperforms RNN and CNN, whereas GNN shows poor results.

5. DISCUSSION

The results gained from the experiments shed light on the performance of various deep learning architectures used for traffic congestion prediction. It was found that the quality of predictions depends heavily on the capacity of capturing spatial and temporal dependencies in traffic data.

The high accuracy provided by the LSTM model is due to its internal memory system which makes it possible to memorize long-range dependencies in sequences. Patterns in traffic data depend strongly on time since traffic congestion is affected by past traffic flows, especially during peak hours and traffic cycles. Therefore, LSTM performs well when predicting traffic patterns.

Consequently, the task that comes up for GNNs is modeling the spatial relationships among traffic nodes. However, any question about congestion in one place necessarily includes information about the level of congestion in other nearby places. Nevertheless, at the moment, the use of graphs and data sets makes it impossible to adequately capture spatial dependencies. In terms of efficiency, CNN is average since it allows extracting local features through convolutional operations. However, due to its inability to model long-term dependencies, CNN has some inefficiencies. As for RNN, it is not as efficient as LSTM due to the same reasons.

Efficient use of Google Colab as a platform for experimenting is yet another notable discovery. Through this approach, there can be assured replicability and reduction of any hardware limitations.

6. LIMITATIONS

There is a restriction on the usage of the specific data set, which can affect the applicability of the findings in different traffic scenarios. Other elements, including weather conditions, accidents, and road conditions, cannot be fully taken into account due to the lack of integration into the forecasting mechanism.

Graph Neural Network relies on predefined graphs, limiting the model from considering all the possible changes in space. The computational restrictions prevent thorough hyperparameter tuning and experimenting.

7. CONCLUSION

This paper is a comparative study of the deep learning model of traffic congestion prediction within a single cloud-based experimental setup. The models compared are RNN, CNN, LSTM and GNN and their performance is considered based on MAE, RMSE, MAPE and R 2 score.

The findings indicate that LSTM has the highest overall performance since it is able to identify long-term temporal relationships in traffic data. Even though GNN is conceptually powerful in terms of spatial modelling, the existing one does not perform better than the temporal models in this dataset scenario. CNN and RNN are still viable baselines, which are less efficient to model complex traffic patterns.

Google Colab is a good platform to use in the implementation of deep learning experiments because it is reproducible, scalable, and easy to implement. Altogether, it is possible to state that the LSTM is the most effective model to predict traffic congestion in the specified experimental conditions.

8. FUTURE WORK

To make it easier for researchers to capture both time and space, they can focus their efforts in the near future on developing hybrid models which combine LSTMs with Graph Neural Networks (GNNs). The prediction accuracy could then be increased by including other factors or variables into the models; such as weather, traffic accidents, and real time IoT sensor data. The performance of the models could also potentially be increased using more complex models based upon transformers and/or attention mechanisms. Future studies should investigate increasing the robustness and generalizability of the proposed models by investigating the use of larger and more diverse datasets. It would also be beneficial to explore deploying these models in a real time environment within a Smart City setting, so that Intelligent Traffic Management Systems (ITMS) and Decision Making Systems (DMS) could utilize this predictive capability..

9. ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to AKS University, Satna, for providing academic support and necessary facilities for conducting this research. Appreciation is also extended to the developers and providers of Google Colab for offering a cloud-based computational environment that facilitated the implementation and evaluation of deep learning models. Finally, the authors acknowledge the availability of publicly accessible traffic datasets and related research resources that contributed to the successful completion of this study.

10. REFERENCES

[1] Chen, Y., Guo, J., Xu, H., Huang, J., & Su, L. (2023). Improved long short-term memory-based periodic traffic volume prediction method.IEEE Access, 11, 103502–103510. doi:10.1109/ACCESS.2023.3305398

[2] Naheliya, B., Redhu, P., & Kumar, K. (2023). A hybrid deep learning method for short-term traffic flow forecasting: GSA-LSTM. Indian Journal of Science and Technology, 16(46), 4358–4368. doi:10.17485/IJST/v16i46.2520

[3] Bharti, Redhu, P., & Kumar, K. (2023). Short-term traffic flow prediction based on optimized deep learning neural network: PSO-Bi-LSTM. Physica A: Statistical Mechanics and its Applications, 625, Article 129001. doi:10.1016/j.physa.2023.129001

[4] Jiang, W., Luo, J., He, M., & Gu, W. (2023). Graph neural network for traffic forecasting: The research progress. ISPRS International Journal of Geo-Information, 12(3), Article 100. Doi: 10.3390/ijgi12030100

[5] Afandizadeh, S., Abdolahi, S., & Mirzahossein, H. (2024). Deep learning algorithms for traffic forecasting: A comprehensive review and comparison with classical ones. Journal of Advanced Transportation, 2024, Article 9981657. doi:10.1155/2024/9981657

[6] Wang, Y., Ke, S., An, C., Lu, Z., & Xia, J. (2024). A hybrid framework combining LSTM NN and BNN for short-term traffic flow prediction and uncertainty quantification. KSCE Journal of Civil Engineering, 28(1), 363–374. Doi: 10.1007/s12205-023-2457-y

[7] Song, Y., Luo, R., Zhou, T., Zhou, C., & Su, R. (2024). Graph attention informer for long-term traffic flow prediction under the impact of sports events. Sensors, 24(15), Article 4796. Doi: 10.3390/s24154796

[8] Fu, F., Zheng, X., & Wu, Y. (2024). Urban traffic flow prediction based on Bayesian deep learning. Sustainability, 16(5), Article 1818. Doi: 10.3390/su16051818

[9] Guo, C., Zhang, Y., & Li, X. (2024). MVHS-LSTM: The comprehensive traffic flow prediction model based on multiple variables heuristic selection long short-term memory. Applied Sciences, 14(7), Article 2959. Doi: 10.3390/app14072959.

[10] Dai, G., Tang, J., & Luo, W. (2023). Short-term traffic flow prediction: An ensemble machine learning approach. Alexandria Engineering Journal, 74, 467–480. doi:10.1016/j.aej.2023.05.015