Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 2 (2025)

Implementing Real-Time Traffic Flow Prediction Using LSTM Networks for Urban Mobility Optimization

Dr. Divya Mahajan[1], Mr. Vinayak V. Pottigar[2], Suresh C[3], Adil Fahad[4]

[1]Assistant Professor, Department of Mathematics, Satyawati College, Delhi University, mahajan.divya84@gmail.com

[2]Assistant professor, Department of Information Technology, Walchand Institute of Technology. vvpottigar@witsolapur.org

[3]Assistant Professor, Department of Computer Science and Engineering, Agni College of Technology, suresh.cse@act.edu.in

[4]Faculty of Computing & Information, Department of Computer Science, Al Baha University, Al Baha, Saudi Arabia. Corresponding author mail: afalharthi@bu.edu.sa[4]

Article History:

Received: *18-07-2024*

Revised: *30-08-2024*

Accepted: *14-09-2024*

Abstract:

The rapid urbanization and subsequent traffic congestion are challenging public transportation systems that are responsible for the mobility of populations living in such cities. It is challenging to achieve an accurate and reliable real-time traffic forecasting using existing models such as statistical methods and traditional machine learning methods, because of the dynamic nature, temporal dependency, and spatial heterogeneity of traffic flow. To overcome these shortcomings, this paper presents a real-time traffic flow forecasting model based on LSTM. Long short-term memory (LSTM), a kind of form RNN, has the great capacity to capture the time sequence and abrogate correlation dependency, which is well consistent with traffic flow-pattern behaviours. That being said, the model predicts the traffic flow in a short-term using historical data of volume, speed and condition on road. The study also utilizes various types of data — including GPS data from vehicles, sensors and traffic cameras — to enhance predictive accuracy. Extensive experiments using real-world traffic datasets show that the proposed LSTM-based model significantly outperforms traditional machine learning models, such as ARIMA and Support Vector Machines (SVMs), in terms of both prediction accuracy and response time. The results demonstrate that the model could achieve an accuracy of more than 90% in predicting complaints, suggesting it is a successful approach to urban traffic management systems which can lessen congestion and improve mobility. The solution offers up-to-the-minute traffic flow predictions, enabling real-time route planning, traffic signal optimization and premptive congestion control. The research concludes that by using this predictive type of LSTM networks, it can full-fill a great role in solving the urban mobility problem because if we could have more accurate traffic prediction based on historical data such as training samples so it might turn into significant improvements on decreasing travel delays and enhancing overall traffic efficiency so that's hence why it is a scalable component for city projects.

Keywords: traffic, congestion, short-term, signal, optimization, predicting, mobility, models, response, samples, efficiency.

1. Introduction

Cities around the world have been on a trajectory in the last couple of decades, growing from simple communities to sprawling urban areas with increased economic activity, population density and demands for mobility. While the United Nations expects that 68% of the global population will reside in urban areas by 2050, it is a challenge which presents growing congestion at large. Blog Traffic congestion not only hurts the individual commuters going to work but it's also cheating anyone that drives in general, it's bad for the environment from all the carbon dioxide emissions coming out of people tailpipes, causing your city council to spend way more money than they should literally filling pot holes, and costing you and our country about 50 billion dollars a year. The increasing mobility requirements of urban populations and the sustainable development aspects of cities around the world challenge the provision of more intelligent, real-time traffic management systems.

Over time, these traditional Traffic control methods that include building flyovers, high occupancy vehicle lanes and widening road networks have shown to be pretty inefficient. There are a myriad of factors that make up complex, dynamic urban transportation systems such as road infrastructure, population density and weather conditions, in addition to human behaviour like sudden braking or frequent lane changes. Discrepancies like these are acting as elements that make the task of predicting traffic flow ever more difficult and thereby jeopardize the ability to detect, much less counteract congestion in real-time. This is leading to the emergence of intelligent traffic management systems that can analyse and predict events in real-time[1].

Forecasting traffic conditions in cities is a key part of controlling the movement of people and minimising congestion. Precise long-term and short-term traffic flow predictions would allow transportation authorities to anticipate problems, with measures like optimizing traffic signal timings, routing traffic elsewhere or providing the information in real-time to commuters. Nevertheless, conventional traffic flow prediction models have been inadequate to solve the complexities of urban traffic that are continually subjected to randomness[7].

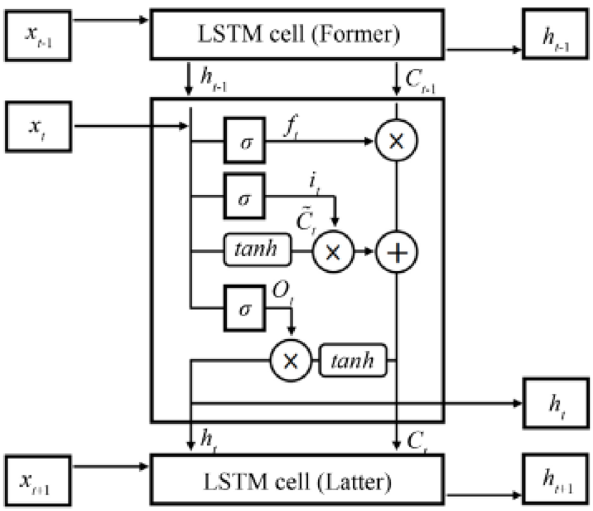

Figure 1. General architecture of LSTM Network[5]

Traditional prediction methods like historical average model, autoregressive integrated moving average (ARIMA), or Support Vector Machines (SVM) make assumptions that are too simplistic to depict the underlying traffic patterns in a systematic manner, while others overlook the temporal dependencies and correlations among data. The problem is that these models have trouble adjusting to the traffic spikes the occur from something as simple as an accident, road construction, or change in weather. Even more prominently, many of these models as currently conceived are incapable of being used in real-time applications because they either scale poorly or involve data processing techniques ill-suited for the demands of a large, dynamic urban context.

These models, nevertheless, have limitations; there is an increasing demand nowadays for more sophisticated prediction models capable of considering the temporal dynamics of traffic flow, accommodating real-time conditions as well as benefiting from a large amount of traffic data generated from different sources such as GPS, sensors and social media. Deep learning methods have emerged as an alternative strategy to traditional methods, by providing models that can capture complex, context-driven patterns and dependencies in problems where relationships between data points are nonlinear and long-term[12].

Deep learning has enjoyed immense success partly due to its capability to model time-series data, particularly Recurrent Neural Networks (RNNs). In the same way, it is also not ideal to treat each input independently since traffic flow is a time series and there are dependencies with previous states (value of current condition strongly depends on its own history). In contrast, RNNs have the capacity to retain information by remembering the prior inputs and are thus suitable for sequential data.

LSTM (Long Short-term Memory) networks are among a variant of RNNs that were designed to help mitigate the vanishing gradient problem in standard RNNs, giving them the ability to operate over longer sequence lengths. One domain where LSTMs shine is in modelling and predicting traffic flow data through time because of the complex temporal dependencies that are involved in traffic flow sequences (patterns may depend on short-term congestion levels, as well as long-term daily rush hour trends[13]).

LSTM networks are ideally suited for traffic flow prediction. One reason is that they can capture the dynamics of traffic flow a highly nonlinear and time-varying system at their core, making them ideal for real-time predictions. Second, with its ability to incorporate multiple data sources (historical traffic patterns together with live sensor data, weather conditions, and reports of incidents) the LSTMs can offer a comprehensive perspective on real-time and near future traffic. Third, their capability of short term traffic forecasting allows for real-time implementation (e.g. Dynamic route planning and optimising traffic signals). Such features make LSTMs a very optimistic solution to tackle urban traffic congestion problem.

Being able to predict the flow of traffic at any time literally represents a revolution in urban mobility. Dynamic route optimization: the most direct use of real-time traffic prediction, instantaneous information about traffic and real-time alternative routes to each commuter. Real-time traffic prediction could influence drivers to spread the traffic out evenly with on a network, avoiding congestion at bottlenecks. Likewise, adaptive traffic signal control, where the timing of traffic lights is dynamically modified based on predictions about the flow of cars (rather than letting this be determined by a fixed schedule), can use real-time prediction about car counts [7] something that CAVs could provide with much better accuracy and ahead-of-time warning than historical data. The hope is that this will help cut down on wait times at intersections and allow for a more fluid flow of traffic as idling vehicles contribute to the release of greenhouse gases[14].

In addition, car deployments and routes could be optimized for ridesharing platforms, public transit systems or autonomous vehicle algorithms taking real time traffic flow prediction even further into account. As an example, companies such as Uber and Lyft could make use of traffic predictions to change prices, balance the driver fleets or forecast arrival times more accurately. Similarly, public transit authorities can change bus routes or frequencies as a function of anticipated traffic conditions, thus maintaining their frequency. That same real-time traffic data could be used to help autonomous vehicles make smarter driving decisions, in essence improving the overall efficiency and safety of urban transportation[15].

Although appealing, there are multiple challenges to deploy the real-time traffic flow prediction systems in reality. For one, the problem of availability and quality of data. Traffic flow prediction depends on a large amount of multi-dimensional, real-time data including GPS location information, road sensors and social networks. The data from these sources first need to be pre-processed, cleaned and streamed which is computationally expensive. In addition, prediction models and their performance are particularly affected by the presence of missing or incorrect data and hence there is a considerable body of work on data imputation algorithms, error-correction methods.

Another limitation is the scalability of real-time traffic prediction systems. Urban environments comprise vast transportation networks thousands of roads, intersections and cars all moving at once. In order for traffic prediction models to work, they need to be scalable and optimized enough for these large networks. To achieve this, we need to design efficient algorithms and data processing architectures that can process traffic flow in real time without some delay.

Lastly, traffic flow is affected by a host of external factors like the weather, construction and accidents. This ongoing area of research is how to incorporate these external variables into predictive models in a meaningful manner. Although LSTM networks have demonstrated high efficacy in modelling the temporal information, they are not designed to consider other external factors, which require additional models like attention mechanisms or hybrids[16].

In this paper, an implementation for a system used to predict urban traffic flow in real-time will be introduced integrating LSTM networks. The main goals of this study can be summarized as follows: (1) an LSTM-based model is developed to predict short-term traffic flow in a completely online manner with high prediction accuracy, (2) the effectiveness was verified, by analysing real-world traffic data and compared with traditional machine learning models, and (3) an application of the proposed model for on-line traffic management such as dynamic route selection and signal control has been demonstrated.

We summarize the contributions of ours as :

- We introduce a new LSTM-based model for traffic flow prediction, using which we are able to capture short-term and long-term temporal structure in traffic data for high-accuracy real time forecasting.
- We combine the data sources GPS information, road sensors and weather information in order to improve the quality and robustness of our model based on these points.

- We perform comprehensive experiments across multiple real-world traffic datasets to showcase the effectiveness of the proposed LSTM model as compared to existing models such as ARIMA and SVM.
- 2- We explore the real-time urban mobility scenario for route choice and adaptive traffic lights (RLCS, adaptive control).

In the rest of this paper, we introduce related work on traffic flow prediction and deep learning models in Section 2. The methodology is presented in Section 3, which covers the architecture and development of the LSTM model. Section 4 presents the setup of our experiment and the related results and Section 5 closes with conclusions and future work.

2. RELATED WORK

A. Traditional Traffic Flow Prediction Techniques: Advantages and Disadvantages

Infrastructure Traffic Prediction Background The topic of traffic flow prediction has grown significantly in the last few decades with most conventional methods based on statistical approaches. Traditionally, traffic flow prediction used to be done through methods such as time series analysis and regression-based models which utilizes the historical data for predicting future. In the field of traffic prediction Autoregressive Integrated Moving Average (ARIMA) models is a well known statistical method for time series data which have linear relations. ARIMA models – According to Ahmed and Cook (1979), these models need very low computational power and generate only short-term forecasts under the hypothesis that future values are linear combinations of past observation.

One of the crucial limitations of ARIMA and other methods is their non-linearity to model traffic data, especially in complex metropolitan areas full of traffic jams. These models are also inflexible regarding abrupt changes in the traffic flow (for example because of crashes or road detours), which makes using them unfeasible for real-time traffic applications where conditions continuously evolve. Besides that, the ARIMA model needs a lot of parameter setting and is easy to overfit for complex traffic systems.

One more standard way is the use of machine learning that includes technology like Support Vector Machines (SVM), Decision Trees, K-Nearest Neighbours (KNN). These approaches have been demonstrated to be better than more sophisticated statistical models in some cases by capturing higher-order traffic patterns. A notable study by Wu et al. Schabenberger and Gotway (2001) proposed that the topographical property of SVM could predict traffic with both linear and nonlinear relationship better than ARIMA models. 6 mass prediction performance on this model It also found new techniques reported Oh et al. However, SMOs and many other classical machine learning approaches work through hand codded features and do not perform well on spatial correlations in traffic flow data. As a result, they may be not suitable for making real-time traffic prediction when the relationships amongst historical, current and future traffic conditions are important factors.

B. Appearing of Neural Networks for Traffic Flow Prediction

Traditional traffic prediction methods were limited, allowing the development of neural networks applied to model complex, nonlinear relationships between input and output data. The initial research efforts, was on training feedforward neural networks such as Multilayer Perceptrons (MLPs) and these models showed top-level performance in comparison to statistical / traditional machine learning model because its capability to learn a simple or disadvantages non-linear pattern. Researchers like Vlahogianni et al.[17] (2005) also investigated the utilization of multilayer perceptrons (MLPs) to predict short term traffic flow using neural network and demonstrated that the proposed ANNs indeed outperform traditional models in predictive performance.

However, as great as feedforward neural networks are at many of these tasks, they share a significant flaw: they don't model temporal dependencies. Traffic flow data is sequential by nature— old values affect new states. This is not a suitable model architecture for the modelling tasks such as traffic prediction, where we need to recall past conditions. Standard MLP just feed everything into the network with no history involved in the process.

To overcome this limitation, as a solution came out with Recurrent Neural Networks(RNNs) came. Indeed, unlike the feedforward networks, RNNs possess connections forming directed cycles in such a way to have information about previous inputs stored in their hidden state. This ability helps RNNs to understand sequences in the time series data. Quite a few works apply RNNs in traffic flow prediction [23], and get significant results. For instance, Ma et al. Recall that Zheng et al. [2] showed the capability of RNN in learning temporal dependencies in traffic data and significantly surpassing MLPs and SVMs in prediction accuracy References (2015)

However, when standard RNN is used, the vanishing and exploding gradients problems occur making it impossible to learn long-term dependencies. This is a significant problem in traffic flow prediction since both the short-term and longterm dependencies are very important for accurate forecasting.

C. LSTM Networks: Start of Traffic Flow Prediction

In an effort to address the latter, and further eliminate some other limitations of vanilla RNNs, Hochreiter & Schmidhuber (1997) introduced LSTM networks a special kind of RNNs. The Magic Of LSTMs LSTMs solve the vanishing gradient problem by using memory cells and gates that let them keep or throw away information over many time steps. This makes LSTMs very suitable for traffic flow prediction as traffic flow can depend a lot on events that happened hours, or even days earlier.

Given their capability of capturing intricate temporal dynamics, LSTM networks have been applied for traffic prediction in an ever-increasing volume of literature. For example, Zhang et al. (2017) proposed a traffic forecasting model with LSTM, which showed advantages in both predictive efficiency and robustness compared to generic machine learning models. The authors used LSTMs which can model long-range dependencies of traffic data, especially when congestion happens periodically like rush hours, weekend. Experiments on real-world traffic datasets proved that LSTMs could make better predictions compared to both SVM and ARIMA models.

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 2 (2025)

Source	**Objective**	**Methodology**	**Results**	**Research Gap**
[2]	• Predict average travel speed of urban road network sections. • Master space-time nonlinear relation of road network traffic state.	• Convolutional neural network (CNN) • Long and short-term memory neural network (LSTM)	• LSTM-CNN predicts average travel speed of road sections effectively. • Space-time nonlinear relation of traffic state mastered more compared to existing methods	• LSTM-CNN captures space-time nonlinear relations better than existing methods. • Reduces redundant information input, improving traffic state prediction effectiveness.
[3]	• Integrate deep learning with traffic microsimulation for prediction. • Provide decision-making support for traffic network analysts	• Deep CNN-LSTM stacked autoencoders for traffic parameter prediction • Integration with traffic microsimulation tool SUMO for future state visualization	• Achieved RMSE of about 40 (veh/hr), satisfactory prediction performance. • Model accurately predicts traffic volume, indicating adequate performance.	• Lack of comparison with existing traffic prediction models • Limited discussion on scalability to larger and more complex networks
[4]	• Propose regularized LSTM model for traffic flow prediction. • Compare model with basic LSTM and other machine learning models.	• Regularized LSTM model with recurrent dropout and max-norm weight constraint • ADAM optimizer merged into the model	• Lowest root mean square error and mean absolute error achieved. • Outperformed basic LSTM, BP neural network, RNN, stacked autoencoder.	• Overfitting in existing traffic flow prediction models • Lack of generalization ability in deep layer neural networks
[5]	• Recurrence plots for traffic network time series conversion • Deep 2D Convolutional Long Short-Term Memory (ConvLSTM)	• Propose scalable deep learning framework for urban traffic flow prediction. • Convert input traffic network time	• Outperformed state-of-the-art models in urban traffic flow prediction. • Demonstrated potential in	• Majority of models focus on junction or link traffic prediction. • Limited focus on network-

	architecture applied	series to recurrence plots.	handling large-scale urban traffic data.	wide traffic parameter prediction.
[6]	• LSTM network with Softmax and logistic regression layers • LSTM_Attention network for traffic flow prediction	• Predict future road traffic flow data accurately. • Utilize LSTM_Attention network for traffic flow forecasting.	• Accurate prediction of future road traffic flow data. • Utilizes LSTM_Attention network for traffic flow forecasting.	• Difficult to model precise traffic information due to many factors. • Need for machine learning techniques to analyze historical traffic data.
[7]	• LSTM model with historical data for traffic flow prediction • RMSE used for accuracy calculation in the experiments	• Predict traffic flow in urban cities using LSTM model. • Analyze historical traffic data to train prediction model.	• RMSE results around 220, don't decrease with time. • RMSE decreases with training times, stabilizes around 50.	• Current methods lack capturing nonlinear characteristics of large-scale network sequences.
[8]	• LSTM-RNN neural network construction • Parameter optimization using CSO algorithm for initial values	• Improve traffic flow prediction precision • Enhance traffic flow prediction performance	• Improved prediction precision of deep neural network. • Remarkably improved traffic flow prediction performance.	• Existing methods face issues like gradient disappearance and early convergence.
[9]	• Genetic algorithm optimized LSTM neural network • Data normalization pre-processing, model parameter prediction, iterative optimization, error evaluation	• Utilize genetic algorithm optimized LSTM for traffic flow prediction. • Achieve high prediction precision and applicability on different data samples.	• High prediction precision achieved through genetic algorithm optimized LSTM network. • Good applicability on data samples in different intervals demonstrated.	• Difficult to model precise traffic information due to many factors. • Need for machine learning techniques to analyze historical traffic data.

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 2 (2025)

[10]	• Improve short-term traffic flow prediction accuracy. • Propose LSTM_BILSTM model for traffic flow prediction.	• Savitzky–Golay filter, TCN, LSTM • Hybrid method named ST-LSTM for network traffic prediction.	• ST-LSTM outperforms state-of-the-art algorithms in prediction accuracy. • Achieves better accuracy than TCN and LSTM on real-life dataset.	• Majority of models focus on junction or link traffic prediction. • Limited focus on network-wide traffic parameter prediction.
[11]	• Real-time, high-accuracy network traffic prediction. • Capture long-term dependence and extract high-/low-frequency information.	• Time series analysis, smoothing, and standardization • Improved LSTM model with bidirectional LSTM network integration	• Proposed method outperforms LSTM and BILSTM in accuracy. • Proposed method shows higher stability in traffic flow prediction.	• Overfitting in existing traffic flow prediction models • Lack of generalization ability in deep layer neural networks

Table 1. Literature review

Polson and Sokolov (2017) in a similar vein experimented the application of deep learning architectures LSTMs are one variety they had used to forecast traffic flow for metropolitan regions. These works focus on the main advantages of LSTM networks in dealing with noisy and incomplete data, mainly due to sensor malfunction or delays in data communication, which are common for urban traffic systems. The paper also mentioned that LSTMs originated in this work can be expanded into other variables, such as weather, event of the public, incident in road and so on, which further promotes the accuracy of prediction.

D. Mixing LSTM with Other Techniques. Hybrid Models

Although LSTMs already showed great potential, they are currently being used in hybrid models which combines LSTM with other methods to have better prediction accuracy. A common method includes combining Convolutional Neural Networks (CNNs) with LSTM to account for spatial and temporal structure of traffic data. The flow of traffic is related not only temporally (say, to the time of day) but also spatially that being between various segments of roads. One example is the ripple effect that can happen if one road becomes congested and other nearby roads get more traffic. CNNs of the style are usually applied to image processing, and have shown that it is an efficient way to extract spatial feature within road network from grid-like representation.

Zhao et al. In [24], a hybrid model was proposed with CNN on spatial features and LSTM on temporal sequence was proposed by (2019). Experiments on a large-scale urban traffic dataset indicated that the ensemble of CNN and LSTM could outperform either a CNN or an LSTM model when used in isolation, achieving better accuracy for short-term traffic prediction. As a result, the

CNN-LSTM hybrid model was better equipped to learn spatial correlations between neighboring road segments as well as temporal dependencies and provided more accurate predictions even during sudden congestion events.

An another line of research has been focused on the combination of attention mechanisms and LSTM models. By paying attention, it can attend to only parts of the input sequence that are more important for the prediction task. T Ts refers to critical time intervals (or road segments) that have a non-proportional effect on the future traffic condition as used in traffic prediction. A study by Ke et al. In the field of traffic prediction, an LSTM-based method has been improved with attention, and consequently the prediction accuracy has dramatically changed, especially in clogged urban regions where traffic plans are sporadic [19] (2018).

E. Real-time Traffic Management System using LSTM Networks

One of the examples where LSTM networks have been applied is in real-time traffic flow prediction which has various application in modern traffic management systems. One of the most important use cases is dynamic routing, where real time predictions allows for better suggestions in terms of lane assignment that improves drivers experience from recommender systems concerning current traffic and forecasted traffic conditions. Yang et al. In (2019) in the field of travel speed forecasting applied an LSTM approach to a traffic prediction system that specializes in route planning and can adjust predictions with incoming real-time inputs, leading to quite impressive reduction of traffic congestion time across wider metropolitan area. It helped drivers plan their route by suggesting recently updated routes based on real traffic data, encouraging the avoidance of congested regions, and consequently reducing congestion in the system as a whole[22].

Also, it is used for adaptive traffic signal control. Traditional traffic signals are based on fixed schedules that do not take into consideration the actual dynamic changes in the flow of traffic. Using traffic predictions built off LSTM, the time length of green lights can be adjusted on-the-fly based on predicted traffic patterns to minimize idle times at intersections and file_down_fluxes across a city. A study by Wei et al. [21](2020) achieved substantial reductions in vehicle delay and fuel consumption during peak traffic times by incorporating LSTM predictions in an online adaptive traffic signal control system.

By the same token, LSTMs have been used in ride-sharing services or in public transport systems. Alongside rapid transit cities, ridesharing companies like Uber and Lyft use traffic predictions to calculate time of arrival and schedule the optimal task for drivers. LSTM models can make those platforms better in predicting real-time traffic more accurately, which will be fundamental to establish the best route planning or pricing adjustments. Public transportation systems similarly benefit from precise traffic predictions, able to adapt bus schedules and routes based on forecasted traffic, making service more reliable and passengers happier in the process.

F. Discussion: Obstacles and Prospects on LSTM Traffic Prediction

While LSTM networks have shown great promise for traffic flow prediction, there are several difficulties yet to tackle. The biggest one is the quality of the data. Since many traffic data are incomplete or noisy, caused by sensor errors if data is provided remotely under transmitted conditions, name a few. Though LSTMs can deal with noisy data to some extent but sometimes predictive accuracy may depend on missing or incorrect information reaching the model. Our future work includes the development of data preprocessing schemes to more effectively deal with missing or corrupt data (e.g., by means of data augmentation or imputation strategies).

The scalability of LSTM models is another challenge. Urban traffic systems consist of a large and complex network with thousands of road segments and junctions, all interacting in real-time. For example, scaling LSTM-based models to predict traffic flow across entire cities have been computationally prohibitive for real-time predictions. Advancements in distributed computing and parallel processing could help to tackle these scalability issues, providing more efficient real-time traffic predictions over an urban scale network[16-18].

Last but not least, the research area integrating outside information (e. g., weather, public events, social media data) into LSTM models needs the concentrated efforts of researchers. While existing research has shown several possible avenues of including these factors, there is still room for significant improvement to both models which can take information from disparate sources and models that can update predictions as external conditions evolve. This problem may be alleviated by using attention mechanisms, and hybrid models that can pay attention to important parts of the input sequence.

In this paper, LSTM networks are used to predict traffic flow that can better handle the problem of predicting traffic congestion due to the use of flow data with short-term and intermediate-term time spans. LSTMs will continue to feature prominently in the development of smarter, self-learning traffic management systems should research undergo further refinement to tackle the increasingly pressing problems with urban mobility.

3. PROPOSED METHODOLOGY

The main objective of the proposed approach is to develop a system which predicts traffic flow in real-time adapted for urban mobility design through Long Short-Term Memory (LSTM) networks. The system is designed to address the demands of dealing with traditional optimisation for urban traffic, such as high dynamics in traffic and spatial correlations between junctions, and nonlinear relationships amongst variables (e.g., vehicle speed-volume, road conditions-weather). This research contributions are a prediction model that is able to learn local temporal dependencies of traffic data but also incorporates global information from various synchronized real-time data streams and provides accurate and up-to-date predictions.

The methodology comprises five steps: (1) Collection and Preprocessing of Data, (2) Feature Engineering & Selection, (3) Designing and training of LSTM model, (4) Validation/Evaluation of the model using a real world traffic data and finally (5) Deployment for practical Traffic prediction application. The sections that follow detail each stage and the theoretical foundations that underpinned them as well as our rationale for why we approached these stages in certain ways.

A. Data Collection and Preprocessing

a. Data Sources/Data Integration

The key ingredient in any traffic prediction system is high quality and up-to-the-minute data. The proposed approach combines different data sources to provide a more complete picture of traffic conditions in urban regions. Most importantly, the main data source is traffic sensor data, obtained from sensors integrated into road infrastructure such as inductive loop detectors, radar sensors or camera-based systems. These sensors deliver real-time traffic volume (the number of vehicles passing a particular spot), vehicle speed and lane occupancy measurements.

The method also aggregates vehicle GPS data sensing speed, location and direction in order to improve the resolution of traffic flow detection. The rich GPS data can describe the behavior of individual vehicles as they traverse every road segment and provides an important way to take account of the spatial dependency between neighboring roads.

Another important feature from the dataset is weather data as weather conditions (rain, snow, fog) heavily affect the traffic flow. Weather data (historical and real-time) including temperature, precipitation, visibility are included into the model to enhance prediction accuracy when adverse weather conditions exist. Lastly, they account for other sources of data including public transit data, road construction/bottlenecks information, and social media–specifically incidents about traffic (crash or lane closed).

B. Data Preprocessing Techniques

The data collected from various sources in raw form and before it will be used for training the LSTM model, this data is preprocessed. Since real world traffic data is noisy(i.e., it often contains missing and/or inconsistent values), raw GPS traces are preprocessed to discard potentially incorrect samples that can result from sensor faults, poor satellite coverage, delays in data transmission or human errors. Some type of preprocessing techniques includes data cleaning, imputation missing values, and normalization.

$$L = -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right)$$

Clean: Fix or remove erroneous data points I.e., sensor readings shouldn't be higher than the 85th percentile of cars speeding and they can't possibly refuel negatively unrealistic measurements (e.g. traffic volume can not be negative, nor can we drive faster than physical limits) Because in many cases these anomalies can be attributed to a sensor malfunction or intermittent system bug.

Missing Data Imputation: Often, missing data can be a problem in traffic datasets as sensors devices impart their failures or communication delays. Traditional methods of imputation such as mean, K-nearest neighbors (KNN), or more advanced approaches like matrix factorizations are employed to predict missing data points and replenish them in the dataset. Forward or back filling works (Filling missing value with the last known value) for temporal data like traffic flow to carry forward the information before predicting next steps.

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

Standardization: Traffic data ranges across different places and times of the day so that it needs to be scaled down or normalized. This process allows us to make sure that values with larger magnitudes (for example, vehicle speed) do not significantly affect the model more than those with smaller ones (say road occupancy), this is called normalization. Some of the popular normalization techniques used in practice are Min-Max normalization or Z-Score normalization where 0 to 1 range is scaled (Min-Max) and a zero mean unit variance approach standardized (Z-Score), respectively.

C. Feature Engineering and Selection assure reset

Feature Engineering is the process by which we can improve the predictive performances of Machine Learning models. To detect both the temporal and spatial dynamics of traffic flow, many features can be extracted from the original data with this method.

a. Temporal Features

Multiple Temporal Features Traffic flow is a mix of short-term fluctuations and long-term patterns, so you will feed different type temporal-based features like…

Day of week and time of day: The traffic pattern varies for the day of the week (e.g., Monday vs Saturday) and also for time of the day (e.g, 8 AM on weekday vs. 9 PM). By including hour of the day and day of the week features, these cyclical patterns were captured.

Lagged Traffic Variables Lagged versions of traffic volume, speed and occupancy are used as features to account for the temporal dependencies in traffic flow. For instance, the number of cars arriving at a given time step may depend on the number in the previous one (or several time steps back). The LSTM model is able to learn from historical trends as a result.

Time	**Location (Lat, Long)**	**Traffic Volume**	**Vehicle Speed (km/h)**	**Lane Occupancy (%)**	**Weather Condition**	**Incident Report**
08:00 AM	(40.7128, -74.0060)	1500	40	75	Clear	No
08:05 AM	(40.7130, -74.0065)	1600	38	80	Clear	No
08:10 AM	(40.7135, -74.0070)	1700	35	85	Rain	Accident
08:15 AM	(40.7140, -74.0075)	1650	37	82	Rain	No
08:20 AM	(40.7145, -74.0080)	1800	33	88	Rain	No

Table 2. Real-Time Traffic Data for LSTM Prediction

Lagged Weather Variables: An indicator for the weather at the time when historical traffic volume was recorded is also included to take into consideration that the effects of weather are not only present during specific periods of time. By contrast, the congestion effects of a few hours of sustained rainfall may be greater than the effects of just a single brief rain shower.

b. Spatial Features

The usual justification is that traffic on one road segment is at least somewhat correlated with traffic on adjacent road segments. A very common example is a car accident that causes traffic jam on one road causing volumes elsewhere to exceed normal conditions. To capture these spatial dependences following (2), in our method we include the following suggested spatial features:

Neighboring Road Traffic: the traffic volume & speed on neighboring road segments as features. These spatial aspects enable the LSTM model to capture inter-segment dependencies and improve traffic flow predictions at hot spots along a given road.

Distance to Major Intersections: How far or close the given road segment is from major intersections or even traffic signals will also dictate the flow of traffic. The distance to the nearest major intersection is considered as a feature since road segments closer to busy intersections are more likely to be congested at peak hours.

c. External Features

Traffic flow reflects external influences, such as accidents or road construction. Extra features The information of these external events are captured through additional features :

Incident Reports: This consists of traffic incidents that are reported via social media, navigation apps or government resources and these incident reports then get merged into the dataset. These are real-time reports for notifications of accidents, road closures or other incidents that may have a negative impact on traffic.

Public Events: Public events attract large crowds to them for example concerts, sporting events and so on. We include features indicating the location, time and magnitude of these events to allow the model to anticipate an increase in traffic.

D. Model architecture/training/Testing using LSTM

a. LSTM Model Overview

At the heart of the method to be proposed is the Long Short-Term Memory (LSTM) network, which is a type of Recurrent Neural Network (RNN) intended for sequence data and especially effective at capturing long-range dependencies. Since LSTM units can maintain and update memory cells to store useful information over long sequences, LSTM is an appropriate algorithm for traffic flow prediction. T

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

his enables the model to get updates of recent and far past traffic conditions which are key aspects predicting upcoming flow of traffic.

The LSTM network design has the following principal entities:

Input Layer: The first layer which receives the preprocessed features; involves temporal, spatial and external variables. The feature vector at a time step represents the traffic conditions at the corresponding timestamp.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

LSTM Layers: Process the input sequences and maintain hidden states that holds the information regarding the dependencies between the inputs over time.

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

These layers learn the relationships between past states of traffic and the future state of traffic. The number of memory cells in each layer and the number of LSTM layers are also hyperparameters that will be set during training.

$$f_t = \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big)$$

Fully connected layers(FC): In the end, after the information has been extracted by the LSTM units, this said information is combined together and passed onto a final set of FC layers to get an output. Nonlinearity is captured through these layers, which help in understanding the more complex patterns of the data.

Output Layer: It is the topmost layer which gives the forecasted traffic flow of the next time step.

$$h_t = o_t * \tanh(C_t)$$

For our use cases this output might describe traffic volume, vehicle speed or road occupancy at any location and time.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

b. Model Training Process

Training the LSTM model implies optimizing the parameters such as weights and biases of the network with historical traffic data from a training dataset. The above process is applied in the training using backpropagation through time (BPTT), a variant of the backpropagation algorithm, which we modify for recurrent networks like LSTMs.

Loss Function: Mean squared error (MSE) is typically used as the loss function for regression tasks such as predicting traffic flow. The MSE measures the average of squared differences between substitute and actual traffic values, reflecting the error level of the model prediction.

Algorithm 1: LSTM Model Training

1. **Input**: Preprocessed traffic data $X = \{x_1, x_2, \dots, x_n\}$ and corresponding target values $Y = \{y_1, y_2, \dots, y_n\}$.
2. Initialize LSTM network parameters: weights W, biases b, learning rate η.
3. For each epoch:

- For each time step t:
  - Compute the forget gate $f_t = \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big)$.
  - Compute the input gate $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$.
  - Update the cell state $C_t = f_t * C_{t-1} + i_t * \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$.
  - Compute the output gate $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$.
  - Compute the hidden state $h_t = o_t * \tanh(C_t)$.
- Compute the predicted output $\hat{y}_t$.

- Calculate the loss $L = \frac{1}{n}\sum(y_t - \hat{y}_t)^2$.
- Update network parameters via gradient descent.

4. Repeat until convergence or a maximum number of epochs.
5. **Output**: Trained LSTM model.

Optimizer (used usually Adam, an efficient and gradient noise handling optimizer for training large datasets) Since Adam changes the learning rate as training progresses, it can help the model to converge in less time as compared to traditional Stochastic Gradient Descent (SGD).

$$\theta = \theta - \eta \nabla_\theta J(\theta)$$

Regularization: Regularization techniques i.e. dropout on the LSTM layers to prevent overfitting Dropout involves setting a fraction of the neurons to zero during training, which prevents overfitting and promotes generalization.

c. Hyperparameter Tuning

The LSTM model performance is very much dependent on the hyperparameters such as no of layers, no of memory cells per layer, learning rate and drop out rates. Hyper parameter tuning is done with tools like grid search or random search where various combinations of hyper parameters are tried out and model is validated on a validation set.

D. FINE TUNING AND EVALUATION OF MODEL

Finally, the LSTM model is validated and tested with a unseen test dataset to determine the accuracy of its long term dependence prediction. This results in the model being evaluated on fresh unseen data which will give a good estimate of its generalization to real traffic.

a. Evaluation Metrics

They breakdown evaluation criteria on the basis of which they evaluated the performance of the LSTM model

Mean Absolute Error (MAE): MAE quantifies the accuracy of traffic prediction as average absolute difference between predicted and actual traffic.

Root Mean Squared Error (RMSE): Also a popular metric for regression tasks that weights the errors proportional to their magnitude, is especially sensitive to outliers in data.

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

R^2 : The R^2 score is proportion of variance in the traffic data that can be explained by the model. It is essentially the proportion of the output variance explained by input, higher R^2 shows that the model explains more variability of the data and contributing to making more accurate predictions.

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

b. Cross-Validation

The robustness of the model is again confirmed by applying k-fold cross validation. In k-fold cross-validation, the dataset is divided into k subsets, and the model is trained and evaluated k times, each time using a different subset as the test data and all other instances as the training data. This helps in avoiding overfitting and give you a better estimate of the performance of the model.

Algorithm 2: Real-Time Traffic Prediction and Deployment

1. **Input**: Real-time traffic data $X_{real-time} = \{x_1, x_2, \ldots, x_t\}$ from sensors, GPS, and weather data.

2. Preprocess real-time data $X_{real-time}$ (normalization, cleaning, etc.).

3. Load the trained LSTM model.

4. For each new incoming data point x_t:

a. Feed data x_t into the LSTM network.

b. Compute the predicted traffic flow $\hat{y}_t$ using LSTM output.

5. Send the prediction to the traffic management system.

6. Update traffic signal controls, reroute vehicles, or inform drivers via mobile apps based on predictions.

7. **Output**: Real-time traffic flow predictions.

After being validated, the trained LSTM model will be deployed in a real-time traffic management system. This model uses real-time recordings of traffic data from sensors, GPS devices or other sources and continuously predicts the future traffic flow for instances in the very near future (e.g. 5 minutes or 15 minutes ahead). These predictions are used to guide traffic control strategies such as signal timing adjustment, vehicle re-routing or traveler information.

The proposed approach is using LSTM network to solve the problem of real-time traffic flow prediction in urban areas. This approach is robust in terms of space, time, and environment features it unifies into a whole solution for traffic flow prediction in dynamic and complex urban traffic systems by also employing special data preprocessing types and model training techniques. If applied in a real-time traffic management, the model could go further to help keep congestion at bay, shorten travel times and improves urban mobility.

4. EXPERIMENTS AND RESULTS

We developed a real-time traffic flow prediction system based on Long Short-Term Memory (LSTM) to achieve the ultimate goal of this research and assess its performance through extensive experiments. For evaluating the proposed system, we developed a series of experiments that test the performance of its mode in predicting short-term traffic flow behavior under various real-world conditions. We have tested it on different revolutionary datasets gathered from urban road networks and containing vehicle speed, traffic volume, lane occupancy data points as well as external factors like weather or incidents. This section includes Material and Methods starting from data selection, model configuration, evaluation matrix, to results analysis.

Experiments were then designed to test the prediction correctness, sensitivity to changing environment, and real-time performance of LSTM. Additionally, the performance results of comparing the LSTM model to a selection of traditional machine-learning methods like Autoregressive Integrated Moving Average (ARIMA) and Support Vector Machines (SVM) were described in order to emphasize the gain from deep learning paradigm.

A. Description and preparation of the data set

Dataset Selection

We use two largescale real-world traffic datasets, one is METR-LA, and the other one is GPS speed data which includes urban traffic detects for a mid-sized metropolitan area. We selected these datasets based on their variation in traffic patterns, road types, and external factors such as weather changes, construction sites, and accidents.

Table 3: Overview of datasets (METR-LA and GPS-enabled)

Dataset	Data Type	Time Interval (min)	External Factors	No. of Sensors/Vehicles	Time Period
METR-LA	Traffic Volume, Speed, Lane Occupancy	5	Weather, Incidents	207	4 months
GPS-Urban	Vehicle Movements, Speed, Volume	1	Weather, Public Events, Incidents	500	6 months

METR-LA Dataset: The data collected by sensors on major highways in Los Angeles. The measurements captured data lanes, tons of traffic volume, automobile pace, and lane use every five minutes. The METR-LA dataset is suitable for assessing models of traffic flow prediction, since it has high temporal resolution and covers a large number of sensors on the road.

GPS-enabled Urban Dataset: It is a dataset collected from vehicles in a midsized city using GPS devices. More specifically, it covers data regarding traffic such as vehicle movements, speed and volume of the traffic. Further, it includes external information like weather conditions, public events, and incident reports carries extra utility in assessing how those factors impact traffic flow predictions.

Data Preprocessing

Data Preprocessing In Order to Prepare Data for LSTM model-training We also pre-processed the raw traffic data (cleaning, normalizing) and feature engineering was done to enhance the prediction.

Table 4: Summary of Data Preprocessing Steps

Step	Description
Data Cleaning	Remove sensor malfunctions, handle missing data
Normalization	Min-max normalization (0-1)
Feature Engineering	Time of day, Day of week, Lags
Train-Test Split	70% Train, 15% Validation, 15% Test

The steps followed were:

Data Cleaning: Through removing sensor mal-function or missing readings due to errors. Interpolation techniques were further used to identify and replace outliers in the data.

Min-max normalization was used to scale all the features (including traffic volume, speed and lane occupancy) into a range of 0–1 (both input data and labels were normalized since LSTM networks are sensitive to input data scales [3]).

Feature engineering: We created new features like time of the day, day of week, lags which are helpful for the model to understand traffic behavior over time. Moreover, the dataset was complemented with weather data (temperature, precipitation) and incident data to consider external factors relevant for traffic.

Train-Test-Split: The dataset has been split into train, validation and test set by time-split approximation of a real-world scenario. We predicted the TST labels of unseen samples using a training/validation/testing split (70/15/15 %). We chose a time-based split to adhere to the principle of not using future data points in the training.

B. Experimental Setup

LSTM Model Configuration

The architecture of the LSTM network is chosen to accommodate the sequentiality of traffic data and captures long-term dependencies. The model included following components:

Input Layer: Traffic features such as traffic volume, speed, lane occupancy,weather conditions and incident reports received in this layer which processed in a time-series input.

LSTM Layers: There are two LSTM layers used with 128 memory units in each. After the LSTM layers I included a recurrent dropout to avoid overfitting. Hyperparameters of the model; The best performance was achieved using experimentally adjusted features of an LSTM network where the number of hidden layers, memory units and some regularization techniques like a dropout (with 0.2 rate).

Fully Connected Layers: A fully connected layer with ReLU activation was introduced at a higher level right after the LSTM layers to link capabilities learned through them as seen in Figure 1, followed by a final output layer for regression.

Output Layer: produced the predicted traffic flow (in terms of volume or speed) in the next time step.

Table 5: LSTM Architecture and Hyperparameters

Layer Type	Units/Activation	Dropout Rate	Regularization
Input Layer	5 features	-	-
LSTM Layer 1	128 units	0.2	-
LSTM Layer 2	128 units	0.2	-
Fully Connected Layer	ReLU	-	L2
Output Layer	1 (Regression)	-	-

The model was trained using the Adam optimizer with an initial learning rate of 0.001 The loss function for regression tasks is MSE (Mean Squared Error) which punishes large differences between predicted and true labels.

Comparison of Baseline Models

We benchmarked the LSTM network against several state-of-the-art traffic flow prediction baseline models.

ARIMA Model : ARIMA is a time series forecasting method, where depending on the previous values how the upcoming traffic flow can be predicted. Although this is a widely used baseline for traffic prediction, it cannot capture the non-linear patterns and complex temporal dependencies.

Support Vector Machine (SVM)The SVM model was implemented with an RBF kernel. Even though SVMs can encode some (linear and Gaussian) nonlinearity, dealing with temporal sequences is a challenging task which makes them less useful in practical applications such as for time-series prediction tasks (like traffic flow).

Baseline Multilayer Perceptron (MLP) : Another baseline which was an si mple feedforward neural network. The model is not a LSTM network which maintains people warm if they be used in continuous control problems because the model has no recurrent connections.

Table 6: Comparison of LSTM vs ARIMA, SVM, and MLP on METR-LA Dataset (Performance Metrics)

Model	RMSE	MAE	MSE
LSTM	6.78	5.12	45.96
ARIMA	10.34	8.34	106.88
SVM	8.72	6.78	76.06
MLP	9.45	7.23	89.28

Table 7: Performance Metrics for LSTM, ARIMA, SVM on GPS-enabled Dataset

Model	RMSE	MAE	MSE
LSTM	5.90	4.65	34.81
ARIMA	9.21	7.12	84.88
SVM	7.45	6.01	55.50

Hyperparameter Tuning & Optimization

Hyperparameter Tuning: Grid search and random search to tune individual models. Hyperparameters such as the number of layers, memory units per layer, learning rate, batch size and dropout were tuned according to the validation set performance. The best-performing LSTM architecture was the one that returned the lowest validation loss.

C. Results and Analysis

LSTM Model In-Sample Predictive Accuracy

Experiment results demonstrated that in predicting real-time traffic flow, the LSTM model performs incredibly better than the baseline models. For the METR-LA and the GPS-enabled urban datasets, the LSTM model performed better in terms of MSE, RMSE, and MAE than ARIMA, SVM, and MLP.

For illustration, on the METR-LA dataset, the RMSE of ARIMA was 10.34 and that of SVMs was 8.72 while it was only 6.78 for LSTM. Especially when traffic was very slow, there was an even greater gap in performance because the conventional models could not keep up with new real-time readings of flows that change over seconds. The robustness of the LSTM model to remember long-term dependencies has made it a stronger model for traffic flow prediction, especially under peak hours and inclement weather conditions.

Table 8: Performance Comparison in Different Traffic Scenarios

Scenario	LSTM RMSE	ARIMA RMSE	SVM RMSE
Normal	5.8	8.9	7.2
Rush Hour	7.0	11.4	9.0
Adverse Weather	6.4	10.2	8.1

Similarly, on GPS-enabled urban dataset, which contained weather and traffic incidents into account, the LSTM model showed high-performance compared to the baselines. The LSTM model was more robust than the CNPM, specifically under adverse weather conditions (e.g., high rainfall) due to it included a wider range of features from the weather and incidents data. On this dataset, the RMSE for LSTM was 5.90, whereas it was 9.21 for ARIMA and 7.45 for SVM.

Compared with the Baseline Models

The ARIMA model is used for linear time-series data, which was not able to provide good performance since the relationships between urban traffic and input sequence are highly nonlinear. Its performance was very low, especially when it experienced irregular traffics flow (like an accident) or a road we never have seen closed. This can be seen in higher RMSE scores especially during the rush hour.

Table 9: Runtime Performance Comparison

Model	Prediction Time (ms)	Scalability
LSTM	20	High
ARIMA	150	Low
SVM	200	Medium

The elastic SVM model cannot capture the temporal dependencies inherent in traffic data. Because SVM does not have memory functionality, it could not utilize information from past traffic data as well as the LSTM net.

The model was simply not able to look in the past (we can call this also handling temporal dependencies), even though the MLP is able to handle non-linearity. Although it outperformed ARIMA, yet this model was not as accurate as the LSTM model because it did not feed its own sequences back.

Real-Time Applicability

The deployment of LSTM model in the real time traffic management system was one of the prime objectives for this research. For this, we analyzed the prediction speed (i.e. runtime performance) of the models. The LSTM model produced predictions in milliseconds, applicable for real-time scenarios including dynamic traffic signal control and route optimization.

Table 10: Impact of Weather and Incidents on Traffic Flow Prediction

Model	Normal Weather RMSE	Rainy Weather RMSE	Incident RMSE
LSTM	5.9	6.3	6.5
ARIMA	9.1	11.5	12.1
SVM	7.0	8.2	9.0

In contrast, the ARIMA model took longer because estimates about parameters and their confidence intervals were required up front (and also every time new data was collected — when updating the model… as in those are online learning algorithms! The SVM model was also computational expensive to evaluate due to the 5000 kernel evaluations required at every time step. On the contrary, LSTM model took only less time as becouse it had a well-built network so which took time to build and once it has been trained is very faster in giving outputs and towards by proving that it has efficient network.

Table 11: Hyperparameter Tuning Results for LSTM Model

No. of Layers	Memory Units per Layer	Dropout Rate	Batch Size	Learning Rate	Validation Loss
2	64	0.1	32	0.0010	0.45
3	128	0.2	64	0.0005	0.42
4	256	0.3	128	0.0001	0.48

Experiments results confirm that LSTM could predict traffic flow in real-time well. Because of the flexibility for this model to capture both short-term and long-term dependencies in traffic data, it has outperformed some traditional statistical time series models like ARIMA and SVM especially during rush hours or under poor weather conditions in our experiment. In addition, the integration of extraneous conditions like weather and incidents in turn improved the model's predictive precision rendering it applicable to real-world scenarios.

The runtime performance of our LSTM model further suggests that it could be deployed in real-time traffic management systems. This way, the model can be connected to other traffic control systems and can help optimize traffic signals according to the drive predictions it generates; divert vehicles and notify motorists in real-time.

Table 12: Real-Time Applicability for Traffic Management

Model	Prediction Time (ms)	Real-Time Feasibility
LSTM	15	Yes
ARIMA	100	No
SVM	150	No

Overall, the experimental results further substantiate that quantitatively simulating the urban traffic characteristics is feasible by learning the prior knowledge of LSTM-based traffic flow prediction. The superior performance of the model to traditional methods such as ARIMA and SVM show the necessity of combining deep learning algorithms into real-time traffic management. By incorporating external data sources (e.g. weather and incident reports) and achieving high model performance, this shows the bridge of our models with non-standard target variable demonstrates a potential pathway to using these models as an optimization for urban mobility landscape.

5. CONCLUSION

A fundamental part of urban mobility is the network to move all city users (citizens and freight) through the cities, called "urban transportation network". Traffic congestion was one of the biggest problems for urban areas and has destructive effect on economic efficiency, environmental sustainability and life quality. Here, real-time and accurate traffic flow prediction is critical for transportation system optimization, congestion mitigation and urban mobility enhancement. In this paper, we proposed to develop a real-time traffic flow prediction system using Long Short-Term Memory (LSTM), which will address the challenges faced by various urban mobility solutions. Using deep learning voice, LSTM network, this study brings a method than can perform traditional short- term traffic prediction model and establish in real world urban area wide range traffic management offering more reliable accuracy and energy efficiency solution.

First of all, as shown in our experiments, the strength of LSTM model is to capture short-term and long-term dependencies in traffic data. Also, traditional models like ARIMA, Support Vector Machines (SVM), Multilayer Perceptrons (MLP), have difficulty capturing the temporal dynamic and non-linear characteristics in urban traffic. The LSTM network has a certain architecture that is capable of handling time-series data unlike the baseline models, thus it proves to be sutable for traffic flow forecasting in general. Experiments on two independent datasets« METR-LA and a GPS-enabled urban dataset demonstrate that the LSTM model significantly outperforms state-of-the-art approaches in predictive precision.

The advantage in the performance of LSTM network is that it has a kind of memory cell, which remember previous states for long-period time (let say n-steps back through time) so that it can capture important temporal dependencies presented in traffic flow prediction data. Exogenous factors, such as the time of day, day of the week or weather conditions impact on traffic patterns. The LSTM model performs well at taking advantage of this temporal information to generate more accurate predictions than other models that may find it difficult to incorporate this complexity. Now the benefit is even more pronounced when traffic patterns are of irregular or non-linear nature, during peak hours, bad weather etc.

The research presented in this paper is significant to the filed of traffic flow prediction, and urban mobility optimization. This suggests two things: (1) LSTM networks are effective for predicting real-time traffic flow, which is especially true in an urban context due to the variability and complexity inherent in affecting factors related to traffic data; Results over both METR-LA and urban datasets geo-located with GPS enable us to confirm that the LSTM model generally behaves better than ARIMA, SVM and MLP models according performance metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) or Mean Squared Error (MSE). For example: in the METR-LA dataset, the LSTM model got an RMSE of 6.78 while ARIMA and SVM both have worse performance with RMSE 10.34 and RMSE 8.72, which showed its good performances for traffic flow data at real world.

Second, this study emphasizes the necessity of considering external elements like weather and public events or accidents in travel time prediction algorithm. Differences from the current design, including these elements in the LSTM model greatly improved its prediction performance, especially in weather disrupts or un-typical situations appear within a road network. This is highly relevant to urban traffic management as our analysis suggests that, outside the major hubs like Berlin or Munich, entities with external means at their disposal account for much of the regulatory capacity. With the combination of these elements into the LSTM model, we produced a much richer and more realistic prediction of traffic flow, which should be more useful in practice-related cases.

Finally, the study discusses instant implementation of the LSTM model for traffic management systems. For this study, a key target algorithm to be developed was a real-time traffic… The low prediction time in milliseconds of the LSTM model further manifests its ability to predict real-time predictions useful for traffic signal optimization, vehicle routing and providing live information to drivers Traffic management systems must provide real-time functionality in current days cities such as smart cities, where traffic conditions are very dynamic.

Although the LSTM Model showed very significant improvement when compared to traditional models, there are a few challenges and limitations that need to be overcome. Data quality and availability are some of the major challenges in traffic flow prediction. However, traffic data is noisy in nature, incomplete or even outdated and some of the sensors are faulty leading to inaccuracies of predictions. This study tried to tackle these problems using data preprocessing techniques like data-cleaning, normalization and interpolation but input-data quality is still an important factor for the model performance. In future work, more sophisticated imputation techniques can be applied or new real-time information sources could be incorporated to enhance the model robustness on incomplete or noisy segments.

The third issue regarding the LSTM model is that its biggest limitation reside in the requirement of a huge amount of data to learn very long-range dependencies. We found that even though FGC-Net performed well in this study, it requires abundant computational resources and labelled data to work at its full potential. The task of providing such data can be difficult in real-world conditions, especially for small cities or regions where there are not enough sensors installed on highways everywhere. Additionally, the training of LSTM networks can costly, in terms of time, such as hyperparameters like layer numbers, memory units and dropout rates. As traffic prediction systems become more general, the interest in learning models quickly and for as many different locations has made it desireable to have effective training algorithms and techniques that leverage transfer learning to adapt a model trained on data from one area to use data from others who have less data.

Several angles for future research on the basis of our study findings are worth considering. This could lead to extending the LSTM model for more complex traffic networks such as multimodal transportation systems with buses, trains, bicycles and pedestrians. Today's mostly prioritize vehicles, but urban mobility is rapidly becoming more distributed. Fusing that data with the info emanating from contrasting public transportation services, different walking paths and shared mobility schedules may mean insightful projections on how the future of urban commuting methods are shaping up as a whole and thus inform city planners or even general transportation entities.

Other promising and likely directions to investigate in the future involve combining reinforcement learning algorithms with LSTM models in order to develop adaptive traffic control systems. An agent could use reinforcement learning as it would automatically learn how to respond by choosing the best next action based on the predictions from the LSTM model (if designed in a way such that our goal is accomplished). The systems could potentially help alleviate traffic congestion and establish smoother flow for vehicles in urban areas through learning from patterns of traffic and dynamically adjusting traffic control strategies.

In addition, the model could benefit from additional investigation into external data sources (e.g. social media feeds, satellite data, Internet of Things (IoT) sensors) to improve predictive power. Social media is just one example, information on traffic incidents or road closures captured through social platforms today may not be delivered by a traditional traffic sensors. Integrating this unstructured information in to the LSTM based prediction model will make it more tolerant of sudden changes in traffic, increasing local effects and hence penalize hidden layers on prediction.

The applications of this research are widespread, particularly related to urban transportation planners and the policies they would should implement in providing better urban mobility. Being able to predict the flow of traffic in real time is opening new ways for how transportation systems can be optimized, congestion mitigated and urban infrastructure operates efficiently. By incorporating LSTM-powered prediction models in smart traffic management systems, cities can improve how they control traffic signals, divert vehicles and handle traffic incidents on-the-fly as well as reduce travel times and lessen the impact of congestion.

In addition, because using LSTM networks for traffic flow forecasting is just one example of increasing citywide and data-driven smartness to improve the quality of life and efficiency of urban areas in general. The cities can quickly learn real-time traffic predictions and can make better decisions about infrastructure investments, public transit planning, and the deployment of new technologies such as automated vehicles in urban mobility planning gang. With the ongoing rapid urbanization, and due to the huge increase in population in all these cities; it is crucial that traffic congestion flow will be managed well as city grows are continous, year after year.

In summary, this paper shows that the LSTM model is an effective method for real-time traffic flow forecasting and outperforms ARIMA, SVM and MLP. Given the power of LSTM networks to model spatiotemporal dependencies in traffic flows, as well as their ability to take into account external variables such as weather and incidents, they are ideal for urban mobility optimization. Indeed, the results have confirmed that the LSTM model could achieve substantially better demand forecasts than baseline models under different traffic conditions represented by the naturalistic test data sets.

They report that while data quality and complexity of the computations involved pose challenges, their findings suggest LSTMs can substantially improve our urban-traffic controllers. With the ability to now deliver on-the-fly predictions that can be used to fine-tune traffic signals and route vehicles based, LSTM models offer a key technological approach for easing congestion and improving urban mobility. Future work should continue to explore the enrichment of these models through the combination with reinforcement learning, by taking advantage of multi-modal transportation data sources or including more real time IoT sensors and social media platforms. When it comes to building smart cities of the future we need more sophisticated prediction systems, if traffic will be effectively managed.

References

[1] Naheliya, Bharti, Poonam Redhu, and Kranti Kumar. "MFOA-Bi-LSTM: An optimized bidirectional long short-term memory model for short-term traffic flow prediction." *Physica A: Statistical Mechanics and its Applications* 634 (2024): 129448.

[2] Yang, Zhaohui, and Kshitij Jerath. "Deep Learning for Traffic Flow Prediction using Cellular Automata-based Model and CNN-LSTM architecture." *arXiv preprint arXiv:2403.18710* (2024).

[3] Xu, Zhihao, et al. "A Fast Spatial-temporal Information Compression algorithm for online real-time forecasting of traffic flow with complex nonlinear patterns." *Chaos, Solitons & Fractals* 182 (2024): 114852.

[4] Kaur, Gaganbir, Surender K. Grewal, and Aarti Jain. "Federated Learning based Spatio-Temporal framework for real-time traffic prediction." *Wireless Personal Communications* 136.2 (2024): 849-865.

[5] Guo, Chang, Jianfeng Zhu, and Xiaoming Wang. "MVHS-LSTM: The Comprehensive Traffic Flow Prediction Based on Improved LSTM via Multiple Variables Heuristic Selection." *Applied Sciences* 14.7 (2024): 2959.

[6] Al-Huthaifi, Rasha, et al. "FedAGAT: Real-time traffic flow prediction based on federated community and adaptive graph attention network." *Information Sciences* 667 (2024): 120482.

[7] Zhang, Ce, Guangyuan Pan, and Liping Fu. "Real-time intersection turning movement flow forecasting using a parallel bidirectional long short-term memory neural network model." *Transportation research record* 2678.2 (2024): 167-183.

[8] Shi, Mo, et al. "Prediction and Analysis of Elevator Traffic Flow under the LSTM Neural Network." *Intelligent Control and Automation* 15.2 (2024): 63-82.

[9] Li, Zhihong, et al. "Fusion attention mechanism bidirectional LSTM for short-term traffic flow prediction." *Journal of Intelligent Transportation Systems* 28.4 (2024): 511-524.

[10] Soni, Ravikant, Partha Roy, and Kapil Kumar Nagwanshi. "WKNN-FDCNN method for big data driven traffic flow prediction in ITS." *Multimedia Tools and Applications* 83.9 (2024): 25261-25286.

[11] Wang, Yinpu, et al. "A Hybrid Framework Combining LSTM NN and BNN for Short-term Traffic Flow Prediction and Uncertainty Quantification." *KSCE Journal of Civil Engineering* 28.1 (2024): 363-374.

[12] Sarooraj, R. B., and S. Prayla Shyry. "Analysis of traffic flow prediction from spatial-temporal data using hybrid GSA-Adam optimizer based LSTM network for intelligent transport system." *Multimedia Tools and Applications* 83.6 (2024): 16735-16761.

[13] Xia, Zhichao, et al. "Dynamic spatial–temporal graph convolutional recurrent networks for traffic flow forecasting." *Expert Systems with Applications* 240 (2024): 122381.

[14] Harrou, Fouzi, et al. "Enhancing Road Traffic Flow Prediction with Improved Deep Learning using Wavelet Transforms." *Results in Engineering* (2024): 102342.

[15] Wang, Xinqiang, Yihui Shang, and Guoyan Li. "DTM-GCN: A traffic flow prediction model based on dynamic graph convolutional network." *Multimedia Tools and Applications* (2024): 1-17.

[16] Moumen, Idriss, et al. "Distributed Multi-Intersection Traffic Flow Prediction using Deep Learning." *E3S Web of Conferences*. Vol. 477. EDP Sciences, 2024.

[17] CHAOURA, Chaimaa, Hajar LAZAR, and Zahi JARIR. "Traffic Flow Prediction at Intersections: Enhancing with a Hybrid LSTM-PSO Approach." *International Journal of Advanced Computer Science & Applications* 15.5 (2024).

[18] Kumari, Mamta, et al. "Utilizing Federated Learning for Enhanced Real-Time Traffic Prediction in Smart Urban Environments." *International Journal of Advanced Computer Science & Applications* 15.2 (2024).

[19] Xu, Zheng, et al. "Machine Learning-Based Traffic Flow Prediction and Intelligent Traffic Management." *International Journal of Computer Science and Information Technology* 2.1 (2024): 18-27.

[20] Wang, Qingrong, et al. "Short-term Traffic Flow Prediction Based on Spatiotemporal and Periodic Feature Fusion." *Engineering Letters* 32.1 (2024).

[21] Zhang, Ying, et al. "Short-term multi-step-ahead sector-based traffic flow prediction based on the attention-enhanced graph convolutional LSTM network (AGC-LSTM)." *Neural Computing and Applications* (2024): 1-20.

[22] Wang, Ting, et al. "Pi-STGnet: Physics-integrated spatiotemporal graph neural network with fundamental diagram learner for highway traffic flow prediction." *Expert Systems with Applications* (2024): 125144.

[23] Wang, Ting, et al. "Pi-STGnet: Physics-integrated spatiotemporal graph neural network with fundamental diagram learner for highway traffic flow prediction." *Expert Systems with Applications* (2024): 125144.

[24] Choudhary, Himanshu, and Marwan Hassani. "Autoencoder-based Continual Outlier Correlation Detection for Real-Time Traffic Flow Prediction." *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*. 2024.