Proceedings of the 7th International Conference on Computing Methodologies and Communication (ICCMC-2023)
IEEE Xplore Part Number: CFP23K25-ART; ISBN: 978-1-6654-6408-6

Gridlock Prediction using a Gated Recurrent Neural Network for Smart Transport Systems

1st Bajjuri Usha Rani
Department of Computer Science and Engineering
Lakireddy Bali Reddy College of Engineering (Autonomous)
Mylavaram, India
bajjuri.usharani2022@gmail.com

2nd Shaik Sameena Sulthana
Department of Computer Science and Engineering
Lakireddy Bali Reddy College of Engineering (Autonomous)
Mylavaram, India
sameenah557@gmail.com

3rd Prajapati Krishna Kumar
Department of Computer Science and Engineering
Lakireddy Bali Reddy College of Engineering (Autonomous)
Mylavaram, India
kingkrishna4371@gmail.com

4th Ponnuri Bhavana
Department of Computer Science and Engineering
Lakireddy Bali Reddy College of Engineering (Autonomous)
Mylavaram, India
ponnuribhavana19@gmail.com

2023 7th International Conference on Computing Methodologies and Communication (ICCMC) | 978-1-6654-6408-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCMC56507.2023.10084275

***Abstract*--Traffic forecasting is a crucial problem in urban transportation, affecting various aspects of daily life such as mobility, traffic management, and road safety. In recent times, the use of machine learning models, particularly Recurrent Neural Networks (RNNs), has gained increasing popularity for solving this problem. Among RNNs, the Gated Recurrent Unit (GRU) has been widely used due to its good performance and computational efficiency. This paper will investigate the application of GRUs to traffic forecasting. The model is trained on historical traffic data, and its predictions are compared to the actual traffic flow to evaluate its performance. Experimental results demonstrate that GRUs perform well in traffic forecasting, with accuracy comparable to other RNN models, while having a much lower computational cost. The outcomes emphasize the value of good feature engineering and the use of domain-specific knowledge to further enhance the functionality of GRU-based models. In conclusion, this paper provides evidence that GRUs are a promising tool for traffic forecasting and can provide valuable insights into future traffic patterns. The results of this paper have practical implications for urban planners, transportation managers, and policy makers in optimizing the use of transportation resources and improving traffic flow in urban areas.**

KEYWORDS— Traffic Forecasting, Recurrent Neural Network, Deep Learning.

I. INTRODUCTION

The goal of traffic forecasting is to provide accurate and reliable information about traffic flow, volume, and travel time for a specific location, route or network. Traffic forecasting is the technique of anticipating future traffic situations based on previous traffic data and other relevant information. This information can include factors such as road network characteristics, weather conditions, and scheduled events. The goal of traffic forecasting is to improve transportation planning and management, and to inform decisions about infrastructure investments, traffic operations, and emergency response planning. Traffic forecasting can be performed at different spatial and temporal scales, ranging from short-term predictions of traffic conditions in a specific location, to long-term projections of future traffic patterns across a city or region. A wide range of methods can be used for traffic forecasting, including statistical models, machine learning algorithms, and simulation models. The choice of method will depend on the specific problem, available data, and computational resources. Traffic forecasting has many practical applications, including reducing traffic congestion, improving road safety, optimizing transportation systems, and reducing emissions from vehicles. By providing insights into future traffic conditions, traffic forecasting can help decision-makers make informed decisions about transportation investments and operations and can support the development of more sustainable and efficient transportation systems.

Traditional traffic forecasting methods rely on historical data and use statistical models such as regression analysis and time-series analysis to make predictions. The models are trained using historical data and then used to make predictions for future traffic conditions. Advanced traffic forecasting methods use machine learning algorithms and other data sources such as GPS data, traffic sensors, and social media data to make predictions. These methods are designed to handle more complex data and to incorporate more factors that can influence traffic conditions [3]. The machine learning algorithms can learn from the historical data and make predictions about future traffic conditions based on the patterns and relationships they have learned. GPS data from vehicles and mobile devices can be used to estimate traffic volume and speed in real-time. This data can be used to make more accurate predictions about future traffic conditions and to detect and respond to incidents and congestion more quickly. Social media data, such as tweets and posts, can be used to detect and monitor traffic incidents and events. This data can be used to provide real-time information about traffic conditions and to make more accurate predictions about future traffic conditions. These advanced methods can provide more accurate and detailed predictions about traffic conditions, but they also require more data and computational resources, and they can be more complex to implement and maintain.

II. LITERATURE SURVEY

Kera's is a deep learning library that provides a high-level API for creating neural networks. It's written in Python and designed to be user-friendly and modular. Kera's runs on top of TensorFlow or Theano, which are low-level libraries for numerical computation and optimization. Recurrent Neural Networks (RNNs) are a type of neural network used for time series forecasting. RNNs have the ability to remember past information and use that information to make predictions based on the current input. They are useful for estimating and interpreting non-linear functions, which makes them appropriate for time series forecasting. LSTMs and GRUs have the ability to handle time series data effectively and overcome the limitations of RNNs by maintaining the gradient information during backpropagation. LSTMs have a more complex structure than GRUs and use multiple gates to regulate the flow of information in the network, making them more powerful for handling time series data. However, this complexity comes at the cost of more parameters and slower training time compared to GRUs, which have a simpler structure and fewer parameters, making them easier to train and faster to run [1].

DBN (Deep Belief Network) approach for detecting cars by lowering the dimension of the data. The authors attempt to identify motorcycles and cars in this study. In this case, the DBN approach is utilized to reduce data dimension and the SVM idea is used to categorize the data. The recommended approach is used using the UIUC dataset, and the outcomes are contrasted with those of the PCA approach [2].

Proposed a technique for forecasting urban traffic flow that considers multiple factors like road geography and environmental variables. This technique combines two machine learning methods, an artificial neural network and a Gaussian mixture model clustering technique. In this study, an artificial neural network was used to model the relationship between the various factors that influence urban traffic flow, such as road geography and environmental variables, and the traffic flow itself. The Gaussian mixture model clustering technique is a statistical method that is used to identify and separate different groups, or clusters, in a dataset. In this study, the Gaussian mixture model clustering technique was used to divide the data into different clusters based on the traffic flow patterns and trends. The proposed technique uses both of these methods to make predictions about urban traffic flow. The ANN is used to model the relationship between the various factors and the traffic flow, and the Gaussian mixture model clustering technique is used to identify and separate different traffic flow patterns and trends. The combination of these two methods allows the system to make more accurate and detailed predictions about urban traffic flow by taking into consideration a variety of factors. The study's findings revealed that the suggested strategy was more accurate than commonly employed prediction methods. This suggests that the combination of an ANN and a Gaussian mixture model clustering technique can be a powerful tool for forecasting urban traffic flow in a comprehensive and effective manner [8].

III. Existing System

3.1 Temporal spatial correlation

The link between time and place in a dataset is known as temporal-spatial correlation. In the context of traffic forecasting, it refers to the relationship between the traffic conditions at a specific time and location, and how they are influenced by factors such as weather, road conditions, and traffic patterns in the surrounding area. The temporal-spatial correlation is crucial for accurately predicting future traffic patterns, as it allows the model to capture the dynamic relationships between different traffic variables over time and space. For example, traffic congestion on one road at a specific time may affect traffic on neighboring roads [12]. If this correlation is not considered in the traffic forecasting model, the predicted traffic patterns may not be accurate. Similarly, the impact of weather conditions such as rain or snow on traffic patterns can also vary depending on the location and time of day.

By capturing the temporal-spatial correlation, the traffic forecasting model can provide a more comprehensive and accurate prediction of future traffic patterns. Integrating the temporal-spatial correlation into the traffic forecasting model can be challenging due to the complexity of the relationships between different traffic variables. However, by using advanced machine learning techniques such as LSTMs and hierarchical structures, it is possible to develop models that can accurately capture the temporal-spatial correlation and provide reliable short-term traffic forecasts.

For example, in a simple linear regression model, the relationship between the target variable and the input variables can be represented as an equation of the form:

$$y = \beta_0 + \beta_1 + \beta_2 + \cdots + \beta_n + \varepsilon \quad (1)$$

Where y is the target variable, x_1, x_2, ..., xn are the input variables, β_0, β_1, β_2, ..., β_n are the coefficients, and ε is the error term.

3.2 Recurrent Neural Network

RNN is designed to analyze sequential input. Unlike traditional feedforward neural networks, RNNs have a hidden state that can capture information from previous time steps, allowing them to handle sequences of varying length and to maintain context across time. The structure of an RNN consists of repeating blocks called "memory cells" that are connected to themselves and the input data [5]. A new hidden state is produced by the RNN at each time step by processing the current input and the previous hidden state. The hidden state is then used as input to the next time step. This allows the RNN to capture information from previous time steps and maintain context across time.

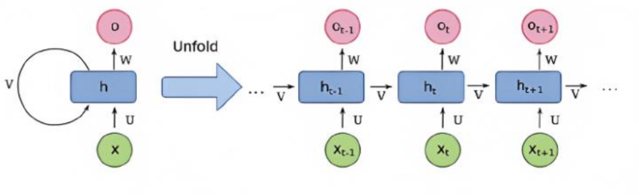

Fig 1. Recurrent Neural Network Architecture [10]

RNNs can be unrolled into a feedforward neural network, where each time step is treated as a separate input. This allows the RNN to be trained using traditional backpropagation algorithms. RNNs have been successfully used for a wide range of tasks, including language translation, sentiment analysis, and time series forecasting. However, RNNs can be computationally expensive and difficult to train, particularly for long sequences [9]. To address these challenges, researchers have developed various variants of RNNs, such as bidirectional RNNs and deep RNNs, that have improved performance and scalability.

3.3 Memory Unit Of LSTM

The LSTM memory unit is the key component of the LSTM network and is designed to handle sequential data. It fixes the vanishing gradient issue in conventional RNNs by enabling the network to keep track of long-term data dependencies [14]. The memory unit is made up of multiple elements including input, forget, and output gates, and a memory cell, which work together to regulate the flow of information in the network and maintain the information needed for making predictions.

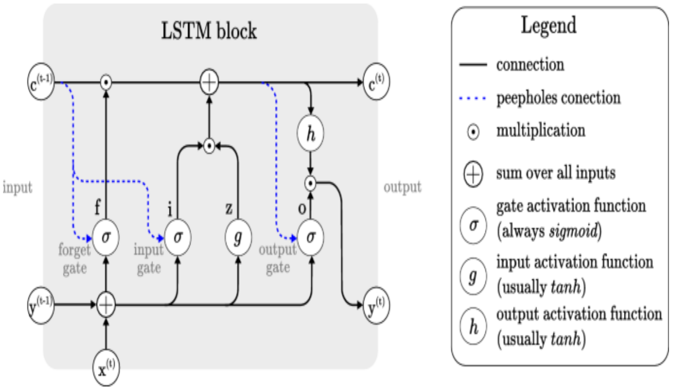

Fig 2. Memory Unit of LSTM [7]

The input gate is responsible for determining how much new information to allow into the memory cell. It uses an activation function such as a sigmoid to generate a value between 0 and 1, which represents the degree to which the new information should be incorporated [11]. The knowledge from the preceding time step that should be ignored, in contrast hand, is decided by the forget gate. The forget gate also uses a sigmoid activation function to determine the amount of information to retain. The memory cell is where the information is stored. It is a simple recurrent unit that stores information for a certain period of time and is updated at each time step. The output gate is responsible for determining the information to be outputted from the memory cell. This information is then passed on to the next LSTM unit in the network.

3.4 LSTM Model for Traffic Forecasting

The proposed LSTM network is a temporal-spatial network that will be used to anticipate short-term traffic. It takes historical traffic data as given and uses ODC matrices to extract temporal-spatial connections. The network's cascading structure allows for the division of long-term forecasts into numerous short-term forecasts and the creation of multi-traffic flow estimates in the near future. The network structure is two-dimensional, with the vertical dimension expressing the indices of observation locations and the horizontal dimension reflecting time changes. The network's time delays are governed by the forecast time and the number of layers, which is often no more than eight. The network settings are tweaked in order to reduce the sum of square errors [13]. The short-term traffic forecasting model proposed aims to address the temporal-spatial complexity of traffic data by integrating available technologies such as the internet of vehicles (IOVs), correlation analysis, and Recurrent Neural Networks (RNNs) [6]. The methodology involves several steps starting with the collection of traffic data from various sources. The collected data is then pre-processed to remove any missing or irrelevant information. Correlation analysis is performed to identify relationships and patterns in the traffic data [4]. The model is developed using a hierarchical LSTM network to model the temporal-spatial correlation in the traffic data.

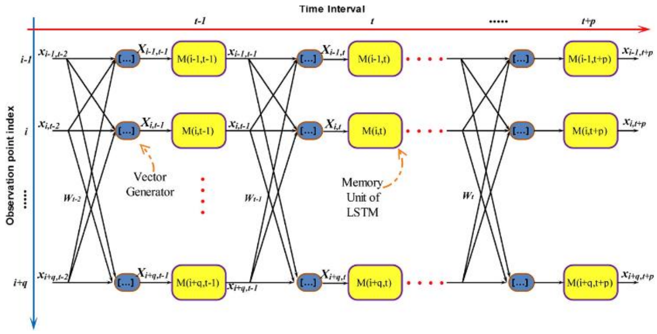

Fig 3. LSTM model for traffic forecasting [13]

The LSTM network is then trained on the pre-processed data to learn the patterns in the traffic data. The model is validated using a separate dataset to measure its accuracy and performance. Finally, the trained and verified model is used for quick traffic prediction in a real-world context. The proposed methodology provides a reliable solution for quick traffic prediction by integrating the available technologies to make use of the temporal-spatial correlation in the traffic data.

IV. Proposed Model

A GRU (Gated Recurrent Unit) is a popular neural network architecture used for time series forecasting, including traffic forecasting. It is a type of Recurrent Neural Network (RNN) designed to process sequential data by capturing the dependencies between time steps. The model employs gates to govern information flow inside the network and to dynamically update the hidden state based on the current time step. An input layer, a GRU surface, and an output units make up the three layers that make up the GRU architecture. At each time step, the input layer receives traffic data and forwards it to the GRU layer. The core element of the model is the GRU layer, which has 2 functions: the updating gate and the resetting gate. The update gate specifies how much of the previous concealed state to preserve, whereas the reset gate defines how much to forget. These gates are used to dynamically update the internal layer depend on the present time step and to record the connections across time steps. The output layer provides the final predictions for traffic volume [15]. In a GRU model for traffic forecasting, the input data typically includes traffic volume, time of day, day of the week, weather conditions, and other relevant factors. The model is trained using historical traffic data to learn patterns and dependencies between the input variables and traffic volume.

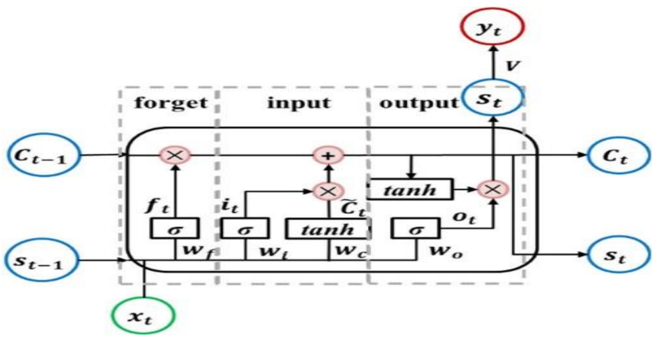

Fig 4. Gated Recurrent Unit Architecture

Once trained, the GRU model can make predictions for future traffic volume based on the current and past inputs.

In summary, the GRU model for traffic forecasting is a powerful tool that can be used to predict traffic volume and make informed decisions about traffic management and congestion.

GRU Algorithm

1. Initialize the input data (x), previous hidden state (h_prev), and the weights for the input data and previous hidden state for the reset gate (W_xr, W_hr), update gate (W_xz, W_hz), and the new candidate hidden state (W_xh, W_hh).
2. Calculate the reset gate:

   reset_gate = sigmoid (W_xr * x + W_hr * h_prev)
3. Calculate the update gate:

   update_gate = sigmoid (W_xz * x + W_hz * h_prev)
4. Calculate the new candidate hidden state:

   candidate_hidden_state = tanh (W_xh * x + (reset_gate * h_prev) * W_hh)
5. Calculate the new hidden state:

   h_new = (1 - update_gate) * h_prev + update_gate * candidate_hidden_state
6. Return the new hidden state:

   return h_new

Where:

x is the input data (current time step),

h_prev is the previous hidden state of the GRU,

W_xz, W_hz, W_xr, W_hr, W_xh, W_hh are the weights for the input data and previous hidden state for the reset gate, update gate, and the new candidate hidden state, respectively, sigmoid and tanh are the activation functions used in the GRU.

4.1 Root Mean Square Error (RMSE)

The root-mean-square error is a common statistic for assessing how far a model's or estimator's predicted statistics diverge from the real numbers seen in a sample or a group (RMSE). The RMSD, also known as the quadratic, mean of the discrepancies between the expected and actual values, is the square root of these discrepancies.

$$RMSE = \sqrt{\frac{\sum_{j=1}^{M} (y_j - \hat{y}_j)^2}{M}} \quad (2)$$

4.2 Coefficient of Determination

A measurement that indicates a prediction performance is called R^2. It is a statistical indication of how closely the regression line in the context of regression matches the actual data. Therefore, it matters if a statistical model is used to test hypotheses or make predictions about the future.

$$R^2 = 1 - \frac{Sum\ Squared\ Regression(SSR)}{Total\ Sum\ Of\ Squares(SST)} \quad (3)$$

The R-squared statistic in a regression model illustrates how much of a dependent variable's variance may be attributed to one or more independent variables.

V. Experimental Results and Analysis

Data used in this study came from the Kaggle website, where machine learning methods were implemented using Python 3 to display results in traffic prediction. In order to compare simply and accurately, two datasets are gathered: One has the date, time, number of cars, and junction; the other contains the same data and is for 2022. The traffic data for 2019 is included in the first one. The extra information can be eliminated by pre-processing the data that was gathered at intervals of 1 to 24 hours in order to anticipate traffic movement at each interval of 1 hour. Date Time, Junction, Vehicles, and ID are the four columns in Dataset.

Table 1. Denotes the R2 and RMSE values.

SNO	Name	R^2 *value*	RMSE
0	Average R2 and Sum RMSE	0.940047	0.26937
1	Average R2 and Sum RMSE	0.853600	0.34101
2	Average R2 and Sum RMSE	0.748499	0.64017
3	Average R2 and Sum RMSE	0.487431	0.71723
4	Average R2 and Sum RMSE	0.757394	0.66489

Traffic Forecasting for four junctions as follows:

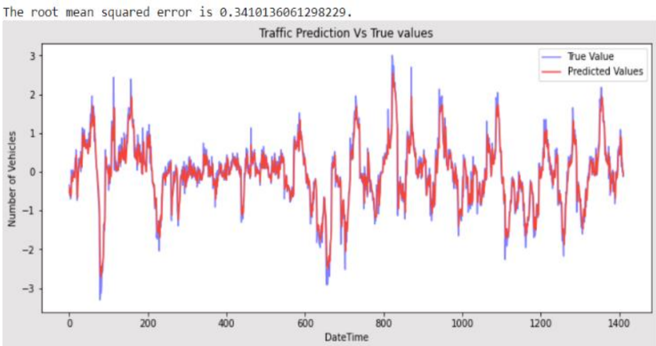

Fig 5. The figure represents the traffic forecast for Junction 1.

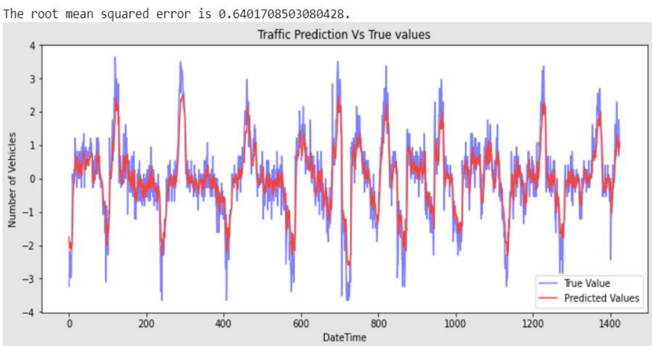

Fig 6. The figure represents the traffic forecast for Junction 2.

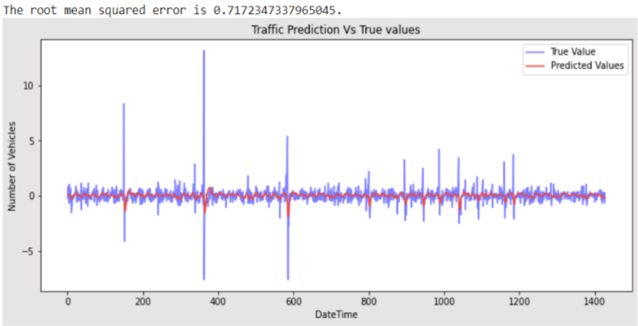

Fig 7. The figure represents the traffic forecast for Junction 3.

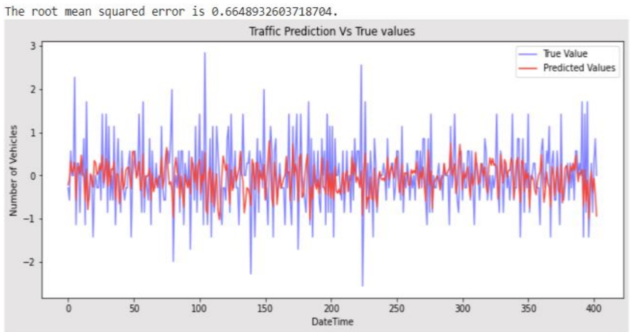

Fig 8. The figure represents the traffic forecast for Junction 4.

VI. CONCLUSION

This paper provides evidence that GRUs, a type of Recurrent Neural Network (RNN), are a promising tool for traffic forecasting in urban areas. The experiments on historical traffic data and found that GRUs performed well in predicting future traffic patterns, with accuracy comparable to other RNN models while having a lower computational cost. The results of this study have practical implications for various stakeholders in urban transportation, such as urban planners, transportation managers, and policy makers. The use of Gated Recurrent Unit (GRU) models for traffic forecasting in urban areas offers several advantages. Firstly, GRUs have been proven to be effective in predicting future traffic patterns based on historical data. This can provide valuable insights for urban planners, transportation managers, and policy makers to optimize transportation resources and improve traffic flow. Secondly, GRUs have a simpler architecture and faster training time compared to other Recurrent Neural Network (RNN) models. This makes them a more efficient solution for large-scale traffic forecasting problems. Furthermore, GRUs are meant to solve the vanishing gradient problem, which is typical in regular RNNs, allowing them to successfully capture long-term relationships in data. This is especially relevant in traffic forecasting, where trends and patterns across time are vital. Finally, using GRUs for traffic forecasting improves accuracy, computational efficiency, and the capacity to capture long-term relationships in data.

REFERENCES

[1] Vlahogianni, E. I., M. G. Karlaftis, and J. C. Golias. Optimized and Meta-Optimized Neural Networks for Short Term Traffic Flow Prediction: A Genetic Approach. Transportation Research Part C: Emerging Technologies, Vol. 13, No. 3, 2005, pp. 211–234

[2] Big data-driven machine learning-enabled traffic flow prediction Fanhui Kong1 Jian Li1 Bin Jiang2 Tianyuan Zhang3 Houbing Song3, 2018.

[3] Bao G, Zeng Z, Shen Y. Region stability analysis and tracking control of memristive recurrent neural network. Neural Netw. 2017;5(1):74-89.

[4] D. M. S. Arsa, G. Jati, M. Soleh& W. Jatmiko(2017), "Vehicle Detection using Dimensionality Reduction based on Deep Belief Network for Intelligent Transportation System," 6th IEEE International Conference on Advanced Logistics and Transport (ICALT), pp. 199-204.

[5] Identification Traffic Flow Prediction Parameters Anuchit Ratanaparadorn Department of Industrial Engineering, Kasetsart University, Thailand, Anuchit Ratanaparadorn, Sasivim ol Meeampol,Thaneerat Siripachana, Pornthe ,Anussornnitisarn, 19-21 2013, zadar, Croatia, international conference.

[6] Gillala Rekha, V. Krishna Reddy, and Amit Kumar Tyagi, "KDOS - Kernel Density based Over Sampling - A Solution to Skewed Class Distribution", 2020, Journal of Information Assurance and Security (JIAS), Vol. 15 Issue 2, p44-52. 9p.

[7] https://www.researchgate.net/figure/Architecture-of-a-typical-vanilla-LSTM-block_fig1_340493274.

[8] L. Kanuru, A. K. Tyagi, A. S. U, T. F. Fernandez, N. Sreenath and S. Mishra, "Prediction of Pesticides and Fertilizers using Machine Learning and Internet of Things," 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021, pp. 1-6, doi: 10.1109/ICCCI50826.2021.9402536.

[9] ZhongjianLv et al"LC_RNN: A Deep Learning Model for Traffic Speed Prediction" Proc.Proceedings of the Twenty-Seventh International Joint Conference on Artifical Intelligence (IJCAI-18),pp 3470-3479.

[10] https://medium.com/analytics-vidhya/rnns-lstms-cnns-transformers-and-bert-be003df3492b

[11] Felix Kunde Alexander Hartenstein Stephan Pieper Petra Sauer, Traffic prediction using a Deep Learning paradigm, CEUR-WS.org, 2017.

[12] Azzouni A, Pujolle G. A long short-term memory recurrent neural network framework for network traffic matrix prediction.Comput Sci. 2017;3(6):18-27

[13] https://www.semanticscholar.org/paper/LSTM-network%3A-a-deep-learning-approach-for-traffic-Zhao-Chen/eb8b18fa6a2c42b2c33395633508609e4ec9dcc5.

[14] Sun, P., Aljeri, N., & Boukerche, A. (2020). Machine learning-based models for real-time TFP in vehicular networks. IEEE Network, 34(3), 178-185.

[15] Diao, Z., Zhang, D., Wang, X., Xie, K., He, S., Lu, X., & Li, Y. (2018). A hybrid model for short term traffic volume prediction in massive transportation systems. IEEE Transactions on Intelligent Transportation Systems, 20(3), 935-946.