

Evaluating the Performance of LSTM in Traffic Flow Prediction at Different Time Scales

Umuhoza Kibogo Aimee Vanessa¹, Kong Yan¹

¹College of Computer and Software

Nanjing University of Information Science and Technology (NUIST)

Nanjing 210044, China

Abstract — Traffic congestion in smart and major cities has been one of the main problems in traffic management and system guidance. Due to the fast economic growth and the highly increasing number of vehicles, the first challenge is to successfully predict accurate traffic flow information to minimize traffic congestion and traffic accidents. Not long ago, many researchers have started to focus more and more concentration on deep learning techniques, including Recurrent Neural Networks (RNN), especially due to their capacity to learn long-term dependencies of sequence data and capture the nonlinearity nature of traffic flow. This paper has applied three different kinds of recurrent neural network architecture such as simple RNN, Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) by considering different time intervals. The dataset collected from the California department of transportation in the year 2018 and 2019 have been used; however, few missing values have been discovered due to incorrect measurement and equipment errors. In this study, to ensure the data quality to be trained in our models and increase the model performance, the mean method on the same hours has been used to calculate and substitute the missing values. LSTM model is proposed in this study on both short and long time intervals. Two popular metrics, including Mean Absolute Percentage Errors (MAPE) and Root Mean Squared Error (RMSE), have been used to evaluate the prediction efficiency.

Keywords — RNN; LSTM; GRU; Traffic flow prediction

I. INTRODUCTION

The traffic flow congestion and traffic data in modern areas have been blown up in the past years because of the rising number of cars. People get hit in the traffic for many hours, so individual travelers and Intelligent Transportation System (ITS) precise that traffic flow is important for both drivers. With different advanced technologies used these days, electronic devices are being deployed to collect traffic data such as passing vehicle details, including volumes, speed and class at a certain time [1]. However, it is possible to use the detailed reviewed data collected to help transport planners improve existing road networks or construct new ones based on the predicted long-term and short-term traffic flow. All of these are in ITS, the traffic prediction foundational [2, 3].

Although precise traffic prediction is a huge problem to solve, the massive traffic data collected holds missing values or incorrect values for many reasons like equipment errors and incorrect measurement, leading to an inaccurate prediction and poor quality output. One of the best solutions to such imperfections is data preprocessing in which the dataset is prepared and cleaned [4]. Techniques used for traffic forecasting have steadily shifted from statistical models to machine learning intelligence and have been into two major

classes which involved parametric and non-parametric models [5–7]. Furthermore, due to stochastic and nonlinear traffic flow characteristics, parametric linearity method did not provide high efficiency in predicting the next situations and more Researchers started to concentrate on the non-parametric methods which try to learn historical data which is related to the expectation instant and use the information items found to forecast for the future.

Researchers have presented many traffic flow forecasting approaches whereby they made attention to short-term traffic flow prediction but is still observed as a challenge today [8]. According to the literature in parametric models, Autoregressive Integrated Moving Average (ARIMA) is the most commonly used method that supposes that the traffic state is stationary. One weakness of ARIMA is its inherent propensity to focus on the data's mean values from the past sequence. It remains difficult, therefore, to capture a rapidly changing phase [9]. Because of the failure due to nonlinear and the stochastic parametric models which are not able to predict accurately, non-parametric models have been studied and built by more researchers including the Support Vector Regression (SVR) application successfully submitted for the prediction of time series and has shown some disadvantages, such as the lack of standardized means to decide some primary model parameters [9]. Neural networks implementations have become the latest interest in the traffic research field.

The contrast between traditional models and neural networks distinctly present an upper level in predicting accurate traffic information [10]. one of the deep learning models called Recurrent Neural Network (RNN) establish the reputation for dealing with time series via recurrent neural ties; however, Gers et al. in [11,12] show that firstly there are still many problems to be tackled in fashion because RNNs do not train with long time lags in the time series, although this incident is commonly seen in traffic prediction tasks. Secondly, that to learn the processing of the temporal series, RNNs rely on predetermined time lags, but it is not easy to Find in an automated way the optimum time window size. By altering the arrangement of the secret neurons in conventional RNN, Long Short Term Memory (LSTM) has been revolutionarily designed to solve the problem. Wang et al. [13] apply LSTM based approach for the next moment prediction of traffic load in a particular geometric field. In [14], LSTM was applied for traffic speed prediction with remote microwave sensor data. Yongxue Tian and Li pan [8] compared different models including SVM, SAE, FFNN and LSTM RNN and conclude that the LSTM RNN model achieves best results between these

non-parameter models. Li et al. assessed the LSTM and GRU model efficiency to predict traffic flow [15].

In this paper, we evaluate and propose LSTM model, which has been compared with GRU and Simple RNN, all known to have the same RNN architectures. The best model for short term traffic prediction is compared with four different time frame sections of 1hour to 4 hours predicted results in the future. The rest of this paper is organized as follows: Section II presents the LSTM description; section III show the experimental setup, section IV indicates the results and lastly section V highlights conclusion and the future work.

II. DESCRIPTION OF LSTM

A. Overview of LSTM

The most robust and well-known sub-class of RNN is called LSTM. Both are artificial neural networks designed to recognize patterns in data sequences such as numerical time series data, stock markets and government agencies. LSTM is a special kind of RNN that can learn long-term dependencies; a memory cell is the core concept behind the LSTM architecture that can hold its condition over a long period, which controls the movement of information out of the cell. The standard LSTM consists of one input layer, one hidden recurring layer, a layer with memory block as the basic unit, and one output layer. The memory block comprises a self-connected memory cells with Temporal state memorization, and three adaptive, multiplicative ones Gating units: the input, output and gates to control the forgotten gates The flow of information inside the block. The three additional gates provide access to the Continuous analogues of operations on the block to write, read and reset. Multiplicative gates can learn how to open and close, so over long periods, LSTM memory cells can store and access information. Mitigating the question of the vanishing gradient. An example of the LSTM memory block is given in Fig 1.

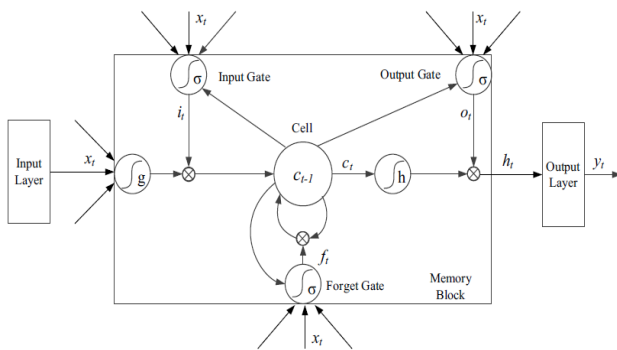


Fig 1. LSTM RNN architecture

The historical traffic data is denoted as $x = (x_1, x_2, \dots, x_T)$, the hidden state of the memory cell is represented as $h = (h_1, h_2, \dots, h_T)$ and the traffic data predicted as $y = (y_1, y_2, \dots, Y_T)$. The networks of LSTM do the computations as follows:

$$h_t = H(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \quad (1)$$

$$y_t = W_{hy}h_t + b_y \quad (2)$$

The W term denotes weight matrices (e.g. W_{xh} is the input-hidden weight matrix), b term denote bias vectors (e.g. b_h is

hidden bias vector). H is the hidden layer function, which is performed by the following composite function:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (3)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (4)$$

$$c_t = f_t c_{t-1} + i_t g(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (5)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (6)$$

$$h_t = o h(c_t) \quad (7)$$

In the below equations, $\sigma(x)$ is the stated regular logistic sigmoid function in Eq.(8), $g(x)$ and $h(x)$ are the function transformations of $\sigma(x)$ whose respectively range is $[-2, 2]$ and $[-1, 1]$. Therefore, i_t , f_t , o_t and c_t represent the input gate, forget gate, output gate, and the vectors of cell activation, each of them has the same size as the hidden vector h .

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (8)$$

$$g(x) = \frac{4}{1 + e^{-x}} - 2 \quad (9)$$

$$h(x) = \frac{2}{1 + e^{-x}} - 1 \quad (10)$$

The following equation written below contribute to reducing the sum of square errors:

$$e_t = \sum (y_t - p_t)^2 \quad (11)$$

III. EXPERIMENTS

A. Data Description and Experimental Design

The dataset collected in this research is downloaded from the California Department of transportation (Caltrans) performance measurement system (PeMS). It is one of the Foremost Widely used database in traffic flow data. We used The freeway SR237-W data obtained in a real-time from individual sensors across the freeway system, located in Santa Clara County, the city of Sunnyvale in California. The data was collected from 1st January 2018 to 31st December 2019 with an update frequency of the 30s and then aggregated for each detector station into 5min the minimum interval, 1hour, daily or weekly. The whole sample points in the dataset we used include 12000 samples, of which 80% was used for training, and the remaining 20% was used for testing. The raw dataset used in our experiment is divided each day with 1-hour of an interval. However, the traffic flow data has an apparent one-day cycle composed of 24-hours, whereby the workdays patterns are very different from the holidays and weekends patterns. As seen in the literature, the trend of eliminating weekends is very prevalent [16-18], in this paper working days are the only chosen as shown in Fig 2, and the two peak hours are 8 AM and 5 PM whereby around 2500 vehicles can pass in only 1-hour.

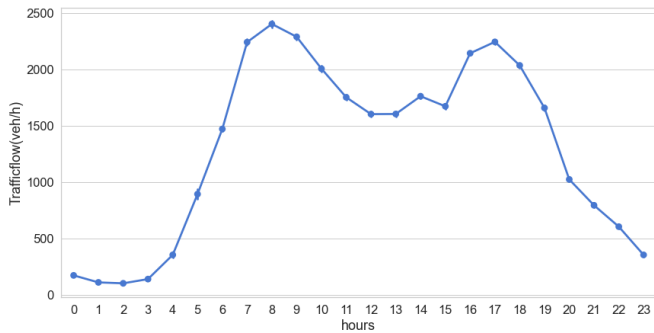


Fig 2. Time series of hourly traffic flow

In our experiment, all the algorithms have been implemented in python using tensor flow as backend and Keras library. Our model was built through the following steps, which focuses on deep learning data preprocessing to achieve accurate predictions results:

- Step1: The first initial and important step is to obtain a relevant and latest dataset that we have downloaded on the California Department of Transportation (Caltrans) performance measurement system (PeMS).
- Step2: We imported all the necessary python libraries, and the dataset gathered.
- Step3: The next step of our preprocessing is to identify and substitute the missing values in our dataset which occupy a very small part of the whole data, therefore to ensure an accurate result, missing records have been replaced by the historical average mean value of the same past hour.
- Step4: Normalization is an important step that allows scaling the data within a range of 0 and 1 while training and performing data analysis. If the data is very wide, the comparison of the statistics will be difficult.
- Step5: We split our dataset into two different datasets (training and test) 80% and 20% respectively and determined the model input and output values.
- Step6: The next step is to build a model by establishing all the parameters, including the number of layers and neurons.
- Step7: Now, the LSTM model can be trained, and the results will be analyzed before changing the parameters.

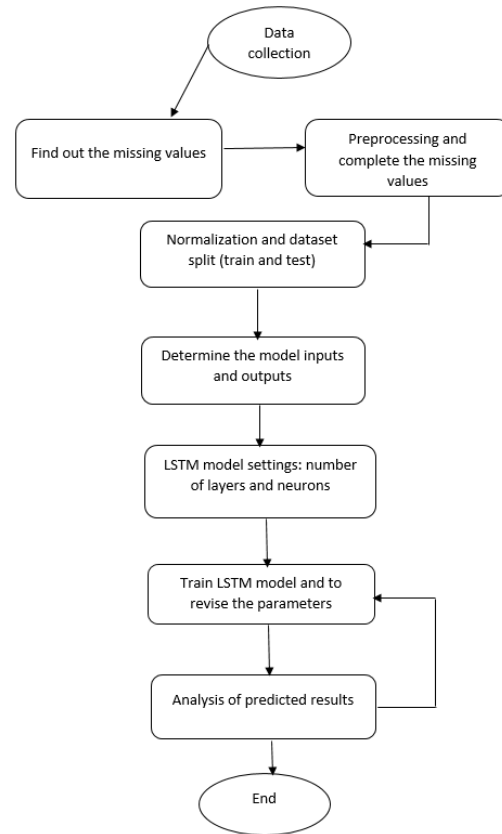


Fig 3. Flowchart of short term traffic flow prediction based on LSTM.

For our experiment, we consider only the traffic flow data as the prediction input without taking other variables into account, including road accidents data, atmospheric conditions or other basic traffic flow parameters like speed and density. Some primary optimal parameters of the proposed model in short term traffic prediction are detailed in the following table I including the size of the input layer, the number of hidden layers and the hidden units in each hidden layer, the number of epochs, the activation function, the batch size and the output layer size.

TABLE I. LSTM model parameters

Optimal Parameters	Values
Input size	1
Hidden layers	2
Hidden units	8,16,32,64,128,256
Batch size	4,8,16,32,64,128
Output size	1
The architecture of the model	Input layer→LSTM layer →LSTM layer→Dropout layer→Fully connected layer→Output layer
Epoch	500
optimizer	Adam
Learning rate	0.01
Dropout	0.2
Loss function	Mean_Squared_Error
Activation function	Tanh

B. Index of Performance

In this research two popular metrics have been used to evaluate the accuracy of the short term traffic flow, including both Root Mean Squared Error (RMSE) which is a common way to calculate a model's Error in quantitative data prediction, and Mean Absolute Percentage Errors (MAPE) which tests the prediction accuracy of a forecasting system usually presented in percentage. Following Eq (12) and Eq (13) represent MAPE and RMSE calculations.

$$MAPE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i} \tag{12}$$

$$RMSE(y, \hat{y}) = \left[\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \right]^{\frac{1}{2}} \tag{13}$$

Thus y is the traffic flow observed value, and \hat{y} is the predicted traffic flow value.

IV. RESULTS

The predictions results obtained show that most of the variations are identified reasonably good, in terms of forecasting accuracy and reliability, a comparative performance analysis of three forecasting models including Simple RNN, GRU and LSTM is presented and some valuable results have been interpreted:

1. As the time interval increases from one hour to the fourth hour, short term traffic flow prediction efficiency increases accordingly to all the above three different models.
2. As the best model to predict the short term traffic flow, both GRU and LSTM have shown closer accurate results than Simple RNN. However, by considering the prediction stability, LSTM outperforms GRU.
3. The LSTM takes advantage of its capability to update the input in its memory continuously. This enables the model to learn for a long time in memory the pattern, trend, and fluctuation in the dataset during the training.

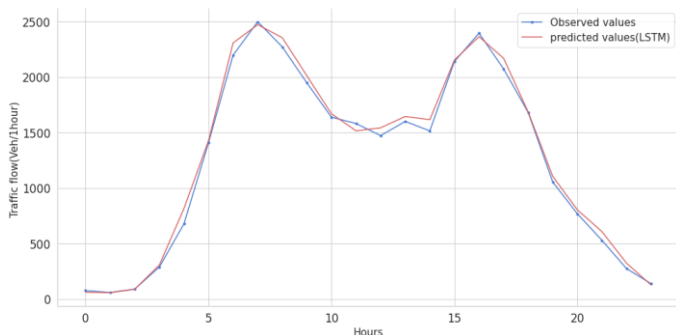


Fig 4. Comparison of observed and predicted traffic flow

Fig 4 illustrates the comparison of the observed and predicted traffic flow values of vehicles per hour. The results show some correlations between them, which means that our model tried to capture the real values. The peak hours shown on our graph is at seven in the morning and four in the afternoon, therefore at midnight (zero hour) to the 4th hour early in the morning

and from 10 to 11th hour in the night the traffic flow is very low which means that the number of vehicles cannot cause the traffic flow.

A. Model Validation

Other types of RNN prediction models have been selected to compare and validate the efficiency of the proposed model LSTM, such as GRU and Simple RNN. The researcher, Cho et al.in 2014, suggested that GRU has gating units that control the data flow within the unit while Simple RNN and LSTM calculates a weighted sum of the inputs and applies tanh as a nonlinear function [19]. All of the prediction models chosen have both the same architecture and the same prediction model process. According to a different time interval, the average results of RMSE and MAPE values of the three prediction models are summarized in table II and table III.

TABLE II. Prediction performance (RMSE)

Predicted time	Models		
	Simple RNN	GRU	LSTM
1-Hour	159.79	154.66	149.52
2-Hours	308.98	304.77	314.41
3-Hours	469.29	463.89	436.92
4-Hours	583.89	592.25	584.63

The RMSE values presented in table II show that GRU has calculated a low difference of observed and predicted error values on the second hour. All the RNN architecture models' prediction performance can be found that the two metrics RMSE and MAPE are close, particularly in table III the MAPE of LSTM and GRU whose values are 11.04% and 11.70% respectively. The traffic flow prediction of the 4th hour has outperformed all the hours. When the percentage is getting smaller, previous data into the models may help achieve greater prediction accuracy to re-train again to the next hour. Therefore, this proves that LSTM and GRU are capable of learning and memorizing long term dependencies.

TABLE III. Prediction performance (MAPE)

Predicted time	Models		
	Simple RNN	GRU	LSTM
1-Hour	15.61%	14.60%	13.28%
2-Hours	15.59%	14.72%	12.92%
3-Hours	15.09%	13.04%	12.03%
4-Hours	13.76%	11.70%	11.04%

V. CONCLUSION

In this paper, three different RNN architectures, including Simple RNN, GRU, and LSTM, have been applied to predict traffic flow within a short and long time interval. Few missing values in the dataset collected on Caltrans PEMS from 2018 and 2019 have been substituted by the same missing hour's mean method to ensure the preprocessed data quality. LSTM outperform GRU and RNN in our study. However, GRU has shown closer results to our proposed method, especially on the fourth-hour traffic flow prediction where MAPE of LSTM is 11.04%, and GRU presented 11.70%. In This study, traffic flow has been considered the only input. Other factors such as the vehicles speed and the weather conditions will be

considered to improve the RNN models prediction performance in future work.

ACKNOWLEDGMENT

The authors would like to acknowledge the Nanjing University of Information Science and Technology for providing a conducive research environment.

REFERENCES

- [1] M. Jiber, I. Lamouik, Y. Ali, and M. A. Sabri, "Traffic flow prediction using neural network," 2018 Int. Conf. Intell. Syst. Comput. Vision, ISCV, 2018, vol. 2018-May, pp. 1–4, 2018, doi: 10.1109/ISACV.2018.8354066.
- [2] X. Chen and R. Chen, "A Review on Traffic Prediction Methods for Intelligent Transportation System in Smart Cities," Proc. - 2019 12th Int. Congr. Image Signal Process. Biomed. Eng. Informatics, CISP-BMEI, 2019, no. 5, 2019, doi: 10.1109/CISP-BMEI48845.2019.8965742.
- [3] E. Bolshinsky and R. Freidman, "Traffic Flow Forecast Survey," Tech. Inst. Technol. Report.–15, pp.1–15, 2012, [Online]. Available: <http://nwww.cs.technion.ac.il/users/wwwb/CGI-bin/tr-get.cgi/2012/CS/CS-2012-06.pdf>.
- [4] S. Zhang, C. Zhang, and Q. Yang, "Data preparation for data mining," Appl. Artif. Intell., vol. 17, no. 5–6, pp. 375–381, 2003, doi: 10.1080/713827180.
- [5] S. Oh, Y. J. Byon, K. Jang, and H. Yeo, "Short-term Travel-time Prediction on Highway: A Review of the Data-driven Approach," Transp. Rev., vol. 35, no. 1, pp. 4–32, 2015, doi: 10.1080/01441647.2014.992496.
- [6] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," Transp. Res. Part C Emerg. Technol., vol. 43, no. February 2018, pp. 3–19, 2014, doi: 10.1016/j.trc.2014.01.005.
- [7] L. Lin, Q. Wang, and A. Sadek, "Short-term forecasting of traffic volume," Transp. Res. Rec., vol. c. No. 2392, pp. 40–47, 2013, doi: 10.3141/2392-05.
- [8] Y. Tian and L. Pan, "Predicting Short-term Traffic Flow by Long Short-Term Memory Recurrent Neural Network," 2015, doi: 10.1109/SmartCity.2015.63.
- [9] W. Hong, "Application of seasonal SVR with a chaotic immune algorithm in traffic flow forecasting," pp. 583–593, 2012, doi: 10.1007/s00521-010-0456-7.
- [10] P. Poonia and V. K. Jain, "Short-Term Traffic Flow Prediction: Using LSTM," Proc. - 2020 Int. Conf. Emerg. Trends Commun. Control Comput. ICONC3 2020, 2020, doi: 10.1109/ICONC345789.2020.9117329.
- [11] T. E. N et al., "Long Short-Term Memory in Recurrent Neural Networks," vol. 2366, 2366.
- [12] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," Neural Comput., vol. 12, no. 10, pp. 2451–2471, 2000, doi: 10.1162/089976600300015015.
- [13] J. Wang et al., "Spatiotemporal modelling and prediction in cellular networks: A big data-enabled deep learning approach," Proc. - IEEE INFOCOM, 2017, doi: 10.1109/INFOCOM.2017.8057090.
- [14] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," Transp. Res. Part C Emerg. Technol., vol. 54, pp. 187–197, 2015, doi: 10.1016/j.trc.2015.03.014.
- [15] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU Neural Network Methods for Traffic Flow Prediction," no. November 2016, 2018, doi: 10.1109/YAC.2016.7804912.
- [16] Y. Kamarianakis and P. Prastacos, "Space-time modelling of traffic flow," Comput. Geosci., vol. 31, no. 2, pp. 119–133, 2005, doi: 10.1016/j.cageo.2004.05.012.
- [17] M. Lippi, M. Bertini, and P. Frasconi, "Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning," IEEE Trans. Intell. Transp. Syst., vol. 14, no. 2, pp. 871–882, 2013, doi: 10.1109/TITS.2013.2247040.
- [18] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Y. Wang, "Traffic Flow Prediction with Big Data: A Deep Learning Approach," IEEE Trans. Intell. Transp. Syst., vol. 16, no. 2, pp. 865–873, 2015, doi: 10.1109/TITS.2014.2345663.
- [19] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," pp. 1–9, 2014, [Online]. Available: <http://arxiv.org/abs/1412.3555>.