ISSN 2563-7568

RESEARCH ARTICLE

Car Parking Availability Prediction: A Comparative Study of LSTM and Random Forest Regression Approaches

Kishwara Sadia[1], Rehnuma Reza[1], Albina Alam[1], Muhammad Arifur Rahman[2*]

[1]*Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh*

[2]*Department of Physics, Jahangirnagar University, Dhaka, Bangladesh*

Abstract

Drivers spend an enormous amount of time searching for parking spots every year. Waste of time, emission of carbon and air pollution have been issues in hunting for parking spots without proper prediction. In this paper, we have proposed to build a framework based on Recurrent Neural Network (RNN) using Long Short Term Memory (LSTM) and Random Forest Regression model to provide prediction of parking availability and compared results afterwards. A real-world case of parking spots availability consisting of 5,500 parking spots in Kuala Lumpur City Centre (KLCC), Malaysia, has been used for regression implementation in this comparative analysis. The results showed that random forest outperformed LSTM approach based on performance metrics.

Key Words: *Parking spot prediction; Recurrent neural network; Long short term memory; Random forest regression*

***Corresponding Author**: *Muhammad Arifur Rahman, Associate Professor, Department of Physics, Jahangirnagar University, Dhaka, Bangladesh, Tel: +88 017 2642 8888; E-mail: arif@juniv.edu*

***Received Date**: January 06, 2021, **Accepted Date**: March 20, 2021, **Published Date**: March 31, 2021*

***Citation**: Sadia K, Reza R Alam A, et al. Car Parking Availability Prediction: A Comparative Study of LSTM and Random Forest Regression Approaches. Int J Auto AI Mach Learn. 2021;2(1):16-29.*

1. Introduction

Due to excessive growth of population all over the world, the growth of private and public vehicles has increased immeasurably. The reason behind the increment in the number of vehicles is because people are tending for ease of life and sustainability. Having your own vehicle lessens the time to reach the destination without waiting for the mass transports. As a result, the number of vehicles has been increasing more abruptly in the past 10 years. It has been recorded by WHO that there was a 16% increase in the number of registered vehicles in the world in between 2010 and 2013 [1]. It is estimated that by 2050, the total number of registered vehicles will increase twice a time to 2.5 billion [2]. As a result of this, the traffic congestion is increasing in a random and uncontrollable fashion. Traffic congestion leads to air pollution, greenhouse gas emission and energy consumption in Metropolitan areas as well as Central Business District (CBD) areas [2,3]. More than one-third of congestion is caused by parking space searching tasks.

To solve the parking space issues, the prediction for parking availability is necessary. With the advent of technology, this problem can be facilitated by data driven models introduced by machine learning. Machine learning approaches can train itself from a large historical dataset and outcome a most efficient prediction in solving the issue [4-7]. This data-driven and robust solution for parking availability prediction can guide the drivers, thereby reducing traffic congestion and the time cost.

Considering the influence of time intervals on parking space availability issues, it is always required to contemplate the whole problem in a time series scenario to get a better perspective.

Therefore, in this paper, we have discussed two new trending models to solve the parking space availability issue. Firstly, we have trained the dataset with a deep learning approach, Long Short-Term Memory (LSTM), a specialization of Recurrent Neural Network. Further we have used a regression model approach, namely Random Forest Regression. We have chosen these two models as they have a great impact in time series analysis and prediction. We first trained and analyzed the prediction result with these two models and in the end, we have compared between them in the accuracy in prediction using different regression metrics.

The rest of the paper is organized in the following manner- in section 2 we have described the literature review. In sections 3 and 4, we have presented the dataset used in this experiment and the methodology respectively. In section 5, we have discussed the result and analysis and finally conclusion and future work have been discussed in section 6.

2. Literature Review

Many researchers have proposed different machine learning approaches to solve the parking occupancy prediction problem such as Neural Network approaches, regression models and so on.

N. Feng et al. in their paper [8] analyzed the parking behavior under the impact of different weather conditions such as temperature, humidity, rainfall and wind speed. To categorize features and test the correlation, they used Anova Test. They used Linear Regression, Ridge Regression,

Decision Tree, Lasso Regression and Random Forest to predict the parking behavior. Finally, they found out that prediction using Random Forest achieved higher accuracy.

Jesper C. Provoost et al. used both real time data and historical data to predict the occupancy rates. They developed two models, feed-forward neural networks (FFNN) and Random forest. Finally, the models are compared by MSE, MAE and MASE where FFNN outperforms the random forests [9].

A. Camero et.al. discussed an approach to building a car parking availability model using recurrent neural network (RNN) analyzing real world case study consisting of the occupancy values of 29 car parks in Birmingham, UK. and also compared the proposed model with the different machine learning approach done so far. Further, they also focused on the fact of RNN model, i.e. high computational cost and applied two optimization approaches such as GA (Genetic Algorithm) and ES (Evolutionary Strategy) for the establishment of an optimal RNN model. Their model predictions were made based on already predicted data, this can be improved if a real-time approach is used. Further, the attributes based on which the prediction took place are not clear [10].

J. Arjona et al in [11], introduced a new model of RNN which is Gated Recurrent Unit (GRU) for the car parking availability purpose developed according to city Rhyad. They divided the input variables in two classes: exogenous data (weather and calendar information obtained from some available APIs) and endogenous data (data acquired from sensor networks buried in the parking slot). Their model resulted in different MSE according to the variations in combinations of variables. And concluded that, the model best fits in predicting parking availability using the parking (endogenous) and the calendar information (days, week etc.). They didn't use the graphical attributes in their research.

In [12] Yang et al. used graphical convolutional neural network (GCNN) to extract the relation of traffic in large scale network and the model of recurrent neural network (RNN) using the Long-short term memory (LSTM) to capture the time series features in predicting the car parking occupancy. They also utilized the traffic speed, weather conditions which added to the prediction accuracy to some extent.

S. S. Ghosal et al. in [13], proposed a deep learning model for block-level parking occupancy prediction which is centered on the concept of heterogeneous clustering and regression learning. The model incorporates CNN, stacked LSTM auto-encoder and regression model using FNN. A dissimilarity measure is proposed based on the weights of each feature of input data in the regression model to form the clusters. During each iteration of learning and clustering, a classifier is used to predict the current cluster labels, and the cluster belonging probabilities are used to control the subsequent re-estimation of cluster centers. They found that incorporating information about spots available, temperature and weather that potentially influence parking behavior can significantly improve the performance of parking occupancy prediction.

T. Anagnostopoulos et al. proposed a model with a multiagent system (MAS)using long-term memory (LSTM) neural network in [14]. LSTM was used for stochastic prediction based on periodic data provided by parking sensors. Stochastic prediction of available SP spots has attracted

much interest since it is an efficient tool used in B2B and B2C marketplace. They used prediction accuracy to evaluate the efficiency of the system. It achieves higher prediction accuracy per daily basis due to stochastic prediction design and input to the proposed MAS and LSTM model.

Shao et al. [15] proposed a framework using LSTM to predict the occupancy rate of on-street parking spots. They used the K-means clustering method to cluster parking spots for a specific area and finally analyzed the performance of the LSTM model using performance metrics.

Awan et al. [16] discussed a comparative analysis of various machine learning and deep learning techniques such as Multilayer Perceptron, K-Nearest Neighbors, Decision Tree, Random Forest, and Voting Classifier for the prediction of parking space availability. They used Santander's parking data set for the experiment. Their experiment led to the conclusion that less complex algorithms like Decision Tree, Random Forest, and KNN outperform complex algorithms like Multilayer Perceptron based on performance metrics.

Jiachang Li et al. has proposed a deep learning-based parking prediction system architecture in cloud in [17]. The LSTM network is used to predict the parking availability. To improve the prediction accuracy, they take into account more factors such as time of day, weather condition and holiday. Their proposed economical workflow works based on elastic computing service. The model training and updating processes need not be in a running state all the time, and it can significantly reduce the computation cost. They used Ali Cloud Platform to deploy it.

In [18], Jamie Arjona et al. developed two RNN based architecture (LSTM, GRU) for accurately forecasting parking availability in urban areas. The developed forecasting models predict the occupancy/hour for a sector. This is computed by aggregating the occupancy/hour for all the sensors. The GRU architecture achieves better results in nearly all cases compared to the LSTM version.

3. Data Set

The dataset for this project has been collected from Kaggle platform. This is a dataset consisting of parking data of Kuala Lumpur City Centre (KLCC), Malaysia [19].

This parking occupancy data has been collected every 15 minutes using parking sensors based on date and time. It consists of 47,605 parking information entries from 2016 to 2017. The attributes to be used in the data set are summarized in Table 1.

Table 1: *Data Set Attributes.*

Feature name	Description
Area Name	The region where the parking slot belongs
Date	Date of the parking spot availability information
Time	Exact time of the parking spot availability information
Parking Spot availability	Number of spots available

From 2016 to 2017, KLCC had 5,500 parking spots. So apart from numerical value indicating available parking spots, the value "FULL" means no parking available at that period and "OPEN" means there were problems in the sensor for reading data.

4. Methodology

4.1. Preprocessing

Before diving into the process of training and testing our models, we have analyzed the characteristics of our dataset. After analyzing the dataset, we came to a decision that our dataset is not purely normally distributed (i.e. not gaussian). Therefore, we have calculated the skewness and Kurtosis of normal distribution to see if the distribution departs from normal distribution. Kurtosis measures the heaviness and lightness of tails of probability density function. On the other hand, skewness measures if the dataset is asymmetrical to the normal distribution.

The values of skewness and Kurtosis have resulted in as such -

Kurtosis of normal distribution: -1.0125348127380422
Skewness of normal distribution: -0.3547449843324632

From the results, since the kurtosis is less than 0, and skewness has a value between -0.5 to 0.5, we can conclude that our dataset distribution is light tailed and fairly symmetrical respectively. Figure 1 shows the probability density function of our dataset.

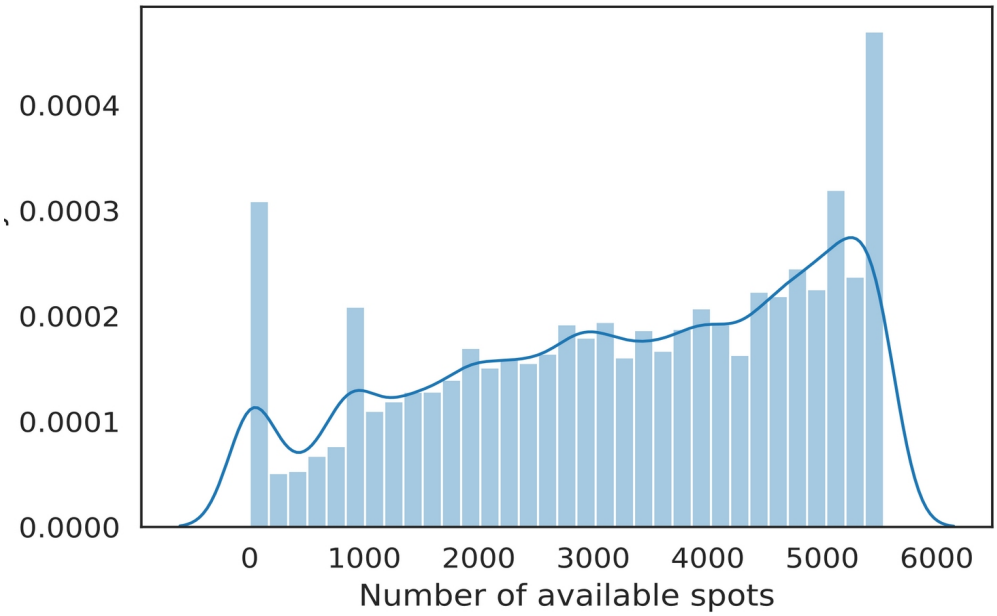

Figure 1: *A statistical representation of the parking dataset gathered from Kuala Lumpur City Center from the date ranges from 2016 to 2017. The statistics shows the probability density of the dataset over the specified period of time and reflects its light tailed and fairly symmetrical nature.*

After analyzing the dataset distribution characteristics, we have revised our dataset as such that there is to be only numerical values in all the columns of the dataset. There are a number of non-numerical values in "available parking spots" columns (e.g. "FULL" and "OPEN"). Since non-numerical values in certain columns could lead to an undesirable estimation in prediction, we have replaced the value "FULL" with value '0', as there is no available spot for parking. Similarly, we have omitted the "OPEN" values of that column, as the sensors faced problems while reading the data and returned errors.

After processing this dataset, 34,933 entries got ready for the experiment. We have trained 27,946 of our dataset (approximately 80% of the entire dataset) following a specialized deep learning approach, LSTM and machine learning approach, Random Forest and retained rest of 6,986 entries (consumes 20% of the dataset) aside for testing purposes.

4.2. Estimators

In this paper, we have used the LSTM model and later run the Random Forest Regression model on the same dataset as estimators for regression. In the end we have compared both models based on the evaluation metrics and graphs constituting the training and testing results.

4.2.1. Long Short Term Memory (LSTM)

Long Short Term Memory (LSTM) is one of the specialization classes of Recurrent Neural Network (RNN). In time series analysis, RNN has marked its significance over other neural network models, building on the concept of recurrence on features that are fed into it [20]. Although there is an issue in predicting time series data using RNN when the dataset is significantly large. Addition of more layers with the activation function [20] with a large dataset may cause the gradient of the loss function go zero, which is called vanishing gradient problem. This phenomenon makes the time series training more infeasible.

LSTM has facilitated RNN by solving the vanishing gradient [21] problem. LSTM introduces a number of gates which makes the model decide upon input features based on which one to retain from the previous cell and which one to forget. The working model of LSTM is shown in Figure 2.

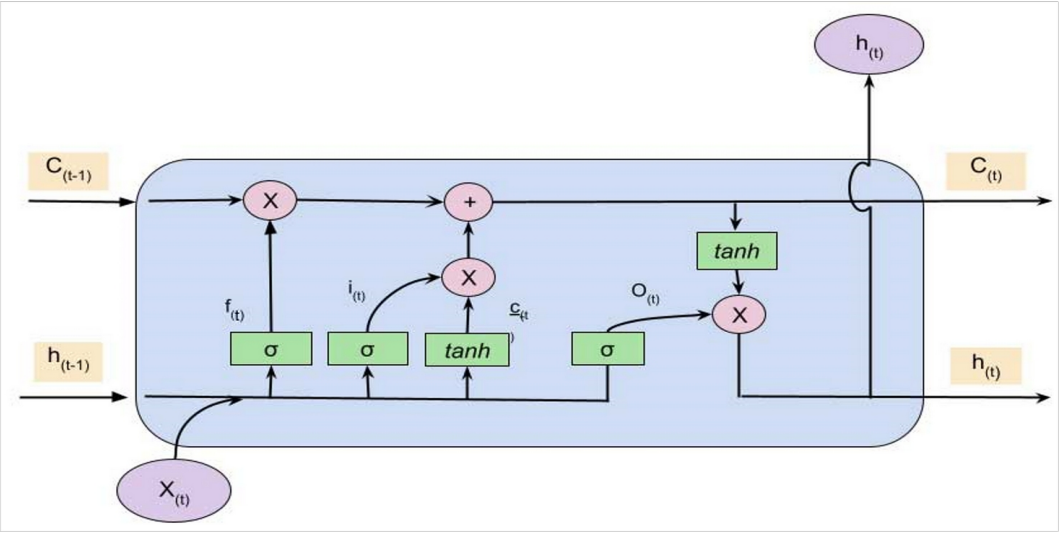

Figure 2: *A LSTM model cell [21]; representing the workflow inside one particular cell. Input data* $x_{(t)}$ *are fed into the cell which undergoes several activation gates* σ *and hyperbolic tangent function* $(\tanh)$ *and thus generates forgate gate* $f_{(t)}$*, input gate* $i_{(t)}$*, update cell* $c_{(t)}$ *and output gate* $O_{(t)}$ *for the purpose of estimating the candidate memory gate* $C_{(t)}$*, hidden layer* $h_{(t)}$ *which in turn fed into the next cell unit.*

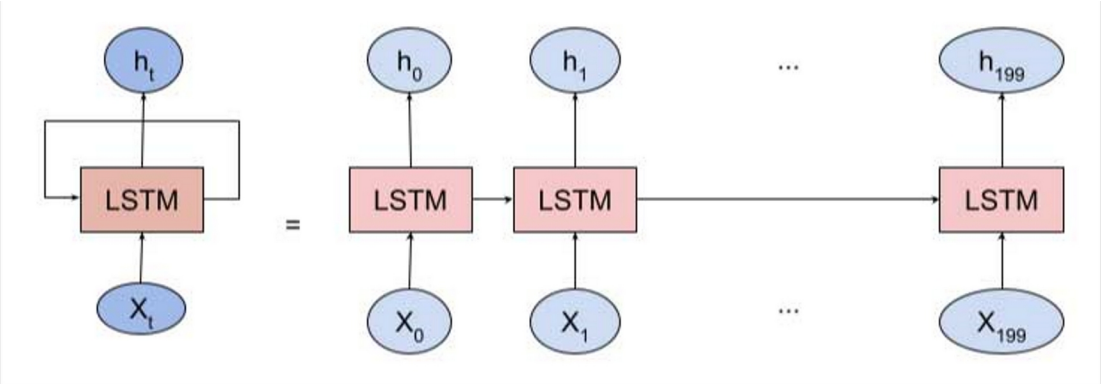

Figure 3: *A simple representation of one of the LSTM layers used for building the learning model comprising 200 neurons. Every neuron of the layer is fed with unit of input data* $X_{(t)}$ *to predict the hidden output* $h_{(t)}$ *which is also fed into the next neuron to predict* $h_{(t+1)}$ *along with* $X_{(t+1)}$.

LSTM comprises four distinct gates unlike RNN. The gates perform distinct operations on the input data. They are namely forget gate, input gate, candidate memory gate and output gate. The presence of Forget gate and Candidate Memory gate make the LSTM model unique to RNN. The equations for the aforementioned gates are shown below.

Input Gate: Controls whether the memory cell is updated.

$$i_{(t)}=\sigma\left(W_i\left[h_{(t-1)},x_{(t)}\right]+b_i\right)$$; Where W_i is the recurrent weight.

Forget Gate: Controls if the memory cell is reset to 0.

$$f_{(t)}=\sigma\left(W_f\left[h_{(t-1)},x_{(t)}\right]+b_f\right)$$

Output gate: Controls the visibility of information of the current cell.

$$o_{(t)}=\sigma\left(W_o\left[h_{(t-1)},x_{(t)}\right]+b_o\right)$$

Candidate Memory Gate: All these three gates have sigmoid activation function which constitutes smooth curves in the range of 0 and 1. Vector $\underline{c}$ modifies the cell state.

$$\underline{c}_{(t)}=\tanh\left(W_c\left[h_{(t-1)},x_{(t)}\right]+b_c\right)$$

To drop the old subject's features and add the new subject information:

$$c_{(t)}=f_{(t)}c_{(t-1)}+i_{(t)}\underline{c}_{(t)}$$

And to calculate the hidden layer output:

$$h_{(t)} = \tanh\left(c_{(t)}\right) * o_{(t)}$$

Where, W_f, W_i, W_c, W_o are the weighted parameters and b_f, b_i, b_c, b_o are the biases for the forget, input, candidate and output gates respectively.

4.2.2. Random Forest Algorithm

Random forest (RF) is a machine learning approach where numerous decision trees are built and integrated together so that the prediction attains more accuracy.

The main idea behind random forest is that multiple models or trees working in integrated form outperforms any individual single model or tree.

It always looks out for the best feature from an arbitrary subset of features in case of splitting any node. This ensures finding better models by creating diversity.

The working model of RF is shown in Figure 4.

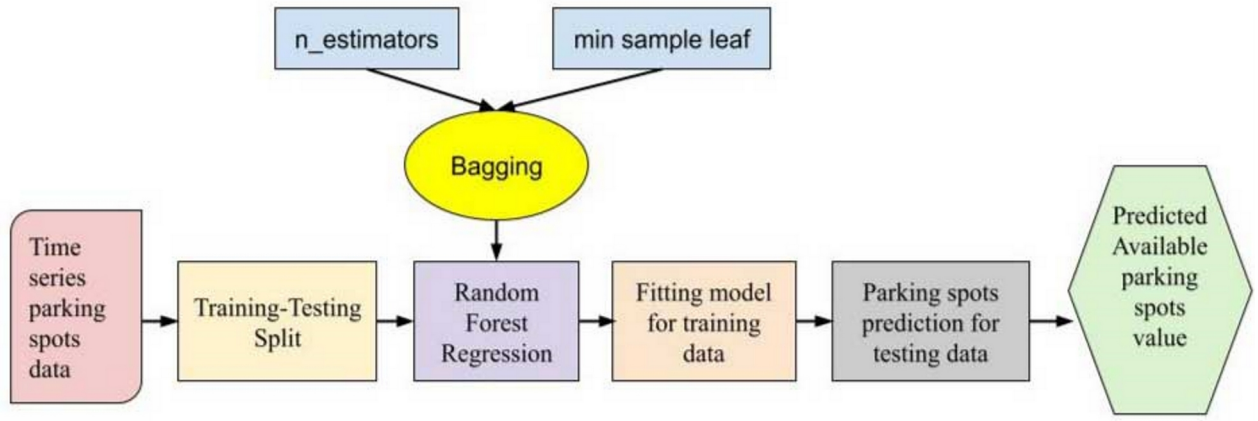

Figure 4: *Workflow of Random Forest model. The time series data of parking spots availability has been split between training and testing data and used the training data to fit. The random forest regression uses the ensemble bagging method where the n_estimators (=250) and minimum sample leaf value (=40) used as parameters here. Definite number of decision trees gets to be integrated for final predictions.*

In case of time series data, it is a challenge to track the frequent changes of real time data. Being an ensemble technique, random forest handles this situation quite well. To train the model properly, there are several parameters for tuning in random forest. Multiple trees are built for accurate prediction by parameter n estimators indicating the number of trees. The default value for n estimators is 100. For indicating the minimum number of samples for each leaf node, min samples leaf parameter is used.

4.2.3. Model Development

We have run our entire experiment in a step by step process. At first, we fine-tuned our data so that all the attributes contain numeric data. After the processing, we have split our dataset maintaining a standard ratio (e.g. 80:20) for training and testing purposes. After having separate splits, we have run our models separately and acquire the evaluation metrics for the respective models. Finally, we have analyzed and compared the model's performance based on the metrics we have obtained. The entire workflow of our experiment is depicted in Figure 5.

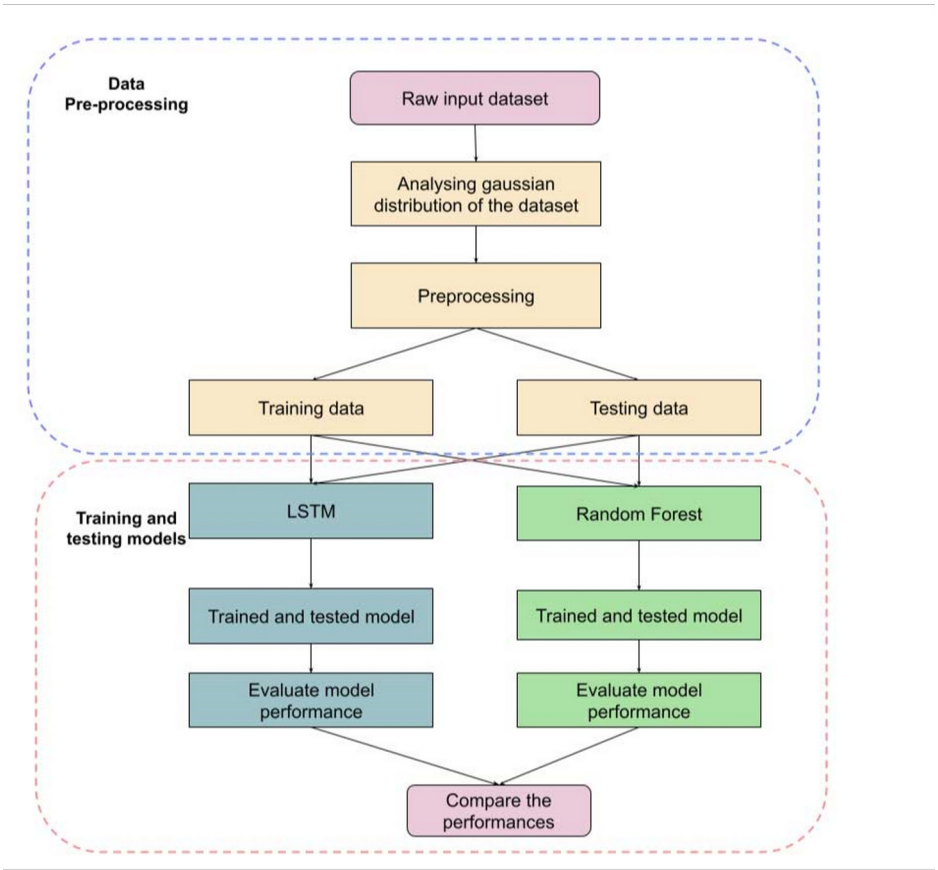

Figure 5: *Workflow of the proposed model. The entire process can be divided into 2 sections: data pre-processing and training and testing the model for evaluation. After presenting, the training and testing splits were fed into LSTM and Random Forest models separately and in the end, we evaluated the performances of different models based on the evaluation metrics.*

5. Result and Analysis

Before training our dataset with any model, it was required to scale the data within the range 0 and 1, so that all the feature values range from 0 to 1. After the processing we have trained and tested our dataset with LSTM and Random Forest Regression model.

For the evaluation of the accuracy of the models, we have chosen Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) matrices. The Mean Absolute Error (MAE) calculates the absolute value difference between the true values and the predictions. The Root Mean Squared Error (RMSE) measures the average squared root difference between the true values and the

predictions after squaring them. In case of having a high weight for large error calculation, RMSE performs well specially in rare errors.

During training with LSTM, we have added two LSTM layers, each consisting 100 neurons in the network for obtaining better prediction. We also added dropout layers so that the model can ignore over-fitting issues. But according to our experiment, increasing the number of epochs leads to overfitting of models. So, we have kept it 10 epochs to train our data in that neural network. Table 2 and table 3 show the parameters used here for the LSTM model and Random Forest Regression Model respectively.

Table 2: *LSTM model parameters.*

Parameter	Values
Number of layers	2
Number of neurons in each layer	100
Dropout	0.4
Epoch	10
batch size	40

Table 3: *Random Forest Regression model parameters.*

Parameter	Values
random state	10
n_estimators	250
min samples leaf	40

Figure 6 and 7 represent the comparative results of MAE and RMSE for LSTM and Random Forest Regression respectively.

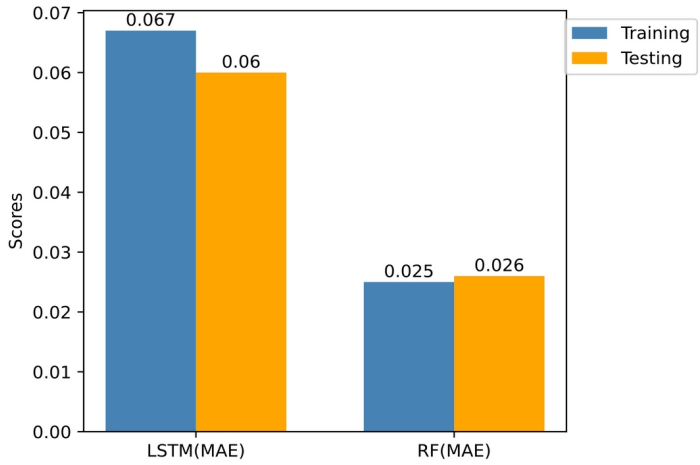

Figure 6: *Comparative result analysis of LSTM and RF models in terms of MAE; RF outperformed LSTM in this analysis.*

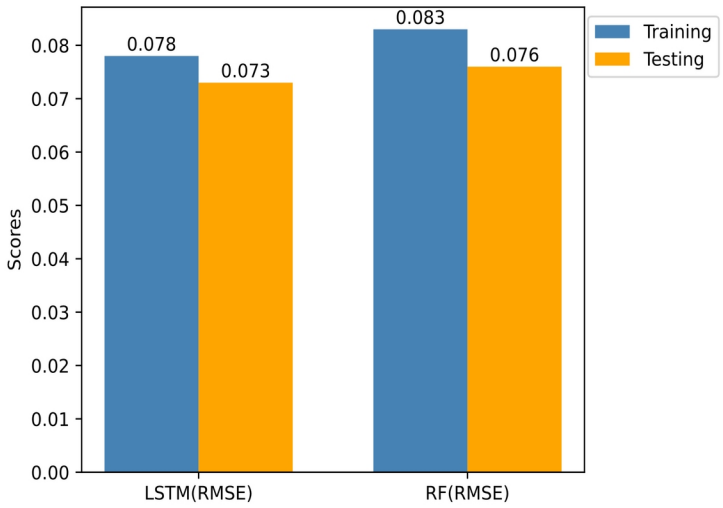

Figure 7: *Comparative result analysis of LSTM and RF models in terms of RMSE; LSTM performed better in this analysis having slight differences in RMSE.*

Comparing values from figure 6, we come to a result that Random Forest Regression, having less MAE value, performs better than LSTM model in this experiment.

We have analyzed the performance of our approach in comparison to previous approaches used in related works such as Shao et al.'s framework [15] and Feng et al. 's experiment [8].

Figure 8 shows the comparative result of performance of our approach and approach from related previous work.

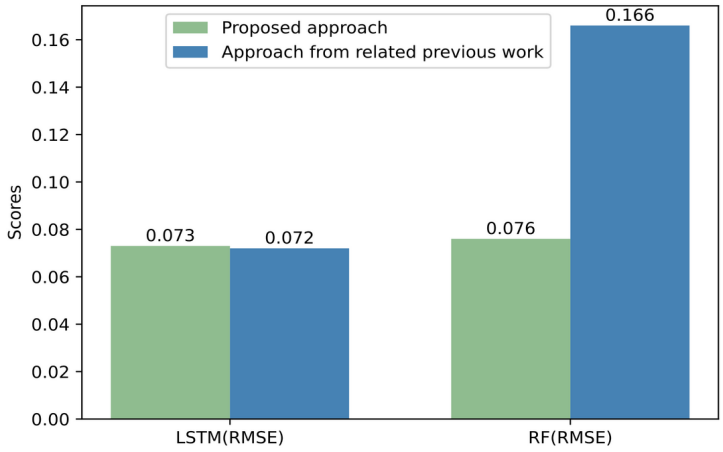

Figure 8: *Performance comparison of our approach and approach from related previous work in terms of RMSE value.*

Considering RMSE value, our proposed approach showed almost similar performance compared to Shao et al.'s parking data availability prediction with LSTM [15] with a narrow margin. On the other hand, our approach with Random Forest regression outperforms Feng's approach [8] of parking behavior.

Figure 9 shows the comparison between true values and the predictions for 5,500 parking slots availability in different times based on LSTM of our experiment.

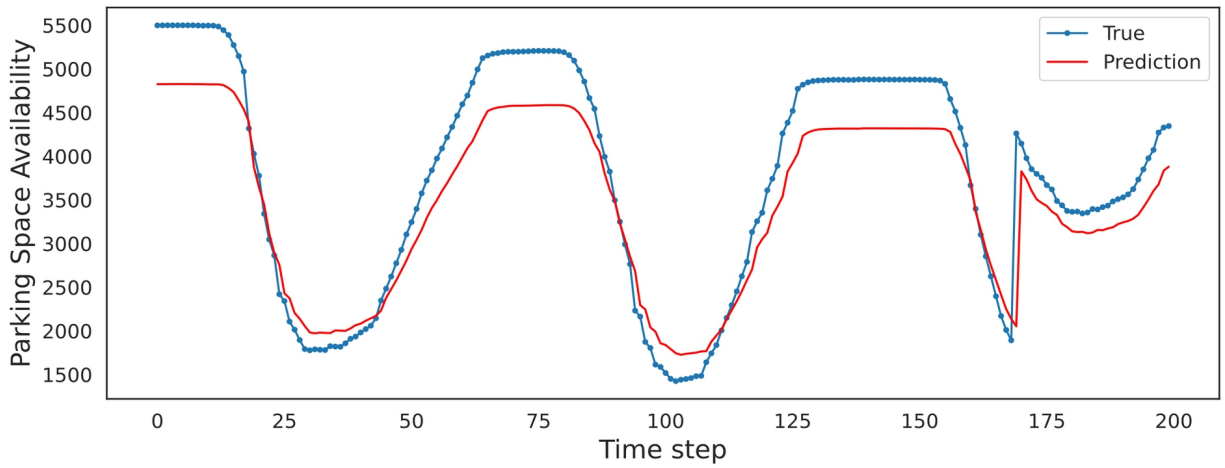

Figure 9: *Comparison between true values and predictions of KLCC parking dataset according to a time-series analysis for LSTM model.*

We have analyzed that our model faced difficulty predicting parking spots whenever it had an inflated number of spots available for a long-time step. It has performed well for other cases of time series prediction.

Again, for the random forest model, following its MAE measurement, it showed great accuracy in predicting the parking spots.

Figure 10 shows the comparison between true values and the predictions for 5,500 parking slots availability in different times based on random forest models.

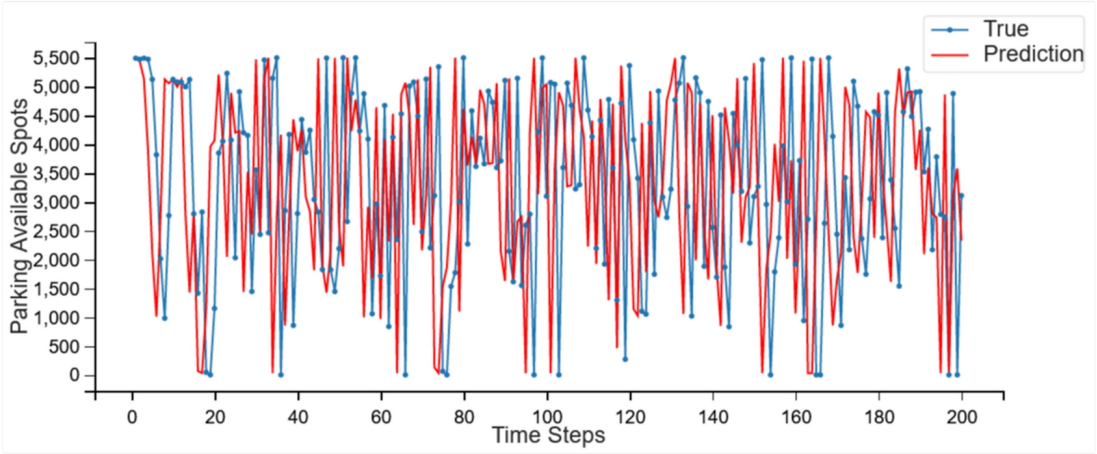

Figure 10: *Comparison between true values and predictions of KLCC parking dataset according to a time-series analysis for RF models.*

6. Conclusion and Future work

Analyzing the MAE values in this paper, we observed that RF performed competently in this experiment for the given dataset. The capability of combining the predictions of many decision trees into a single model by RF made it outperform LSTM. For a bigger dataset, LSTM, a neural network approach can be trained well and perform better. As our work focused on a specific region, it is not enough to discuss cases in real world issues consisting of multiple regions. So, data consisting of various regions with different distribution of occupancy rate cannot be analyzed following this model. To do that, in future, several groups can be created and clustered using various clustering techniques. We can consider a non-parametric framework like the Gaussian process [22] even for better performance.

Conflict of interests: The authors declare that there is no conflict of interest.

References

1. https://www.who.int/gho/road safety/registered vehicles/number/en/
2. https://www.greencarreports.com/news/1093560_1-2-billion-vehicles-on-worlds-roads-now-2-billion-by-2035-report
3. https://www.brookings.edu/research/traffic-why-its-getting-worse-what-government-can-do/
4. Adiba FI, Islam T, Kaiser MS, et al. Effect of corpora on classification of fake news using naivebayes classifier. Int J Auto AI Mach Learn. 2020;1:80-92.
5. Ferdous H, Siraj T, Setu SJ, et al. Machine learning approach towards satellite image classification. In: Kaiser MS, Bandyopadhyay A, Mahmud M, Ray K (eds), Proceedings of International Conference on Trends in Computational and Cognitive Engineering. Advances in Intelligent Systems and Computing. Springer, Singapore. 2021;pp.627-37.
6. Mahmud M, Kaiser MS, Rahman MM, et al. A brain-inspired trust management model to assure security in a cloud based iot framework for neuroscience applications. Cogn Comput. 2018;10:864-73.
7. Das TR, Hasan S, Sarwar SM, et al. Facial spoof detection using support vector machine. In: Kaiser MS, Bandyopadhyay A, Mahmud M, Ray K (eds), Proceedings of International Conference on Trends in Computational and Cognitive Engineering. Advances in Intelligent Systems and Computing. Springer, Singapore. 2021;pp.615-25.
8. Feng, N, Zhang F, Lin J, et al. Statistical analysis and prediction of parking behavior. In: Tang X, Chen Q, Bose P, Zheng W, Gaudiot JL (eds), Network and Parallel Computing. NPC 2019. Lecture Notes in Computer Science. Springer, Cham. 2019;pp.93-104.
9. https://arxiv.org/abs/1911.13178
10. Camero A, Toutouh J, Stolfi DH, et al. Evolutionary deep learning for car park occupancy prediction in smart cities In: Battiti R, Brunato M, Kotsireas I, Pardalos P (eds), Learning and Intelligent Optimization. LION 12 2018. Lecture Notes in Computer Science. Springer, Cham. 2019;pp.386-401.
11. Arjona J, Linares MP, Casanovas J. A deep learning approach to real-time parking availability prediction for smart cities. DATA 19: Proceedings of the Second

International Conference on Data Science, E-Learning and Information Systems. 2019;1-7.
12. Yang S, Ma W, Pi X, et al. A deep learning approach to real-time parking occupancy prediction in transportation networks incorporating multiple spatio-temporal data sources. Transp Res Part C Emerg Technol. 2019;107:248-65.
13. Ghosal SS, Bani A, Amrouss A, et al. A deep learning approach to predict parking occupancy using cluster augmented learning method. 2019 International Conference on Data Mining Workshops (ICDMW), Beijing, China. 2019.
14. Anagnostopoulos T, Fedchenkov P, Tsotsolas N, et al. Distributed modeling of smart parking system using LSTM with stochastic periodic predictions. Neural Comput & Applic. 2020;32:10783-96.
15. Shao W, Zhang Y, Guo B, et al. Parking availability prediction with long short term memory model. In: Li S (eds) Green, Pervasive, and Cloud Computing. GPC 2018. Lecture Notes in Computer Science. Springer, Cham. 2019;pp.124-37.
16. Awan FM, Saleem Y, Minerva R, et al. A comparative analysis of machine/deep learning models for parking space availability prediction. Sensors. 2020;20:322.
17. Li J, Li J, Zhang H. Deep learning based parking prediction on cloud platform. 2018 4th International Conference on Big Data Computing and Communications (BIGCOM), Chicago, IL, USA. 2018.
18. Arjona J, Linares M, Casanovas-Garcia J, et al. Improving parking availability information using deep learning techniques. Transp Res Proc. 2020;47:385-92.
19. https://www.kaggle.com/mypapit/klccparking
20. Sadik R, Reza ML, Al Noman A, et al. COVID-19 pandemic: a comparative prediction using machine learning. Int J Auto AI Mach Learn. 2020;1:1-16.
21. https://medium.com/themlblog/time-series-analysis-using-recurrent-neural-networks-in-tensorflow-2a0478b00be7
22. Rahman MA. Gaussian Process in Computational Biology: Covariance Functions for Transcriptomics. PhD thesis. University of Sheffield. 2018.