Parking lot occupancy prediction using long short-term memory and statistical methods

Ercan Avşar[1*], Yusuf Can Anar[2], Abdurrahman Özgür Polat[3]

[1] Dokuz Eylül University, Department of Computer Engineering, İzmir, Turkey, ercan.avsar@deu.edu.tr, ORCID: 0000-0002-1356-2753
[2] Çukurova University, Department of Electrical and Electronics Engineering, Adana, Turkey, yusufcan2617@gmail.com, ORCID: 0000-0002-2227-768X
[3] Karamanoğlu Mehmet Bey University, Department of Electrical and Electronics Engineering, Karaman, Turkey, ozgrpolat@kmu.edu.tr, ORCID: 0000-0002-4922-6567

ABSTRACT

In crowded city centers, drivers looking for available parking space generate extra traffic and in addition, the resulting excessive exhaust gases cause air pollution. Therefore, directing the drivers to a parking spot in an intelligent way is an important task for smart city applications. This task requires the prediction of occupancy states of parking lots which involves appropriate processing of the historical parking data. In this work, Long-Short Term Memory (LSTM) and Autoregressive Integrated Moving Average (ARIMA) methods were applied to parking data collected from curbside parking spots of Adana, Turkey for predicting the parking lot occupancy rates of future values. The experiments were performed for making predictions with different prediction horizons that are 1 minute, 5 minutes, and 15 minutes. The performances of the methods were compared by calculating root mean squared error (RMSE) and mean absolute error (MAE) values. The experiments were performed on data from five different days. According to the results, when the prediction horizon is set to 1 minute, LSTM achieved RMSE and MAE values of 0.98 and 0.72, respectively. For the same prediction horizon, ARIMA achieved RMSE and MAE values of 0.62 and 0.35, respectively. On the other hand, LSTM achieved smaller error values for larger prediction horizons. In conclusion, it was shown that LSTM is more suitable for larger prediction horizons, however, ARIMA is better at predicting near-future values.

1 Introduction

As a result of increment in world population, city centers are getting more crowded every day. Finding an available parking spot in such city centers is a challenging and frustrating task. Besides, extra time and fuel are consumed while looking for a parking spot. In addition, searching action of drivers causes slowing down the traffic movements and this recursive action causes more traffic problems. It is reported that parking spot searching drivers cause more than 1/3 of the traffic congestions [1]. Another source for this problem has emerged due to the Covid-19 situation. Today, more people started to prefer personal vehicles instead of urban transportation to decrement the risk of being infected. Hence, more vehicles may be present in traffic when compared to pre-pandemic times. Consequently, finding a parking spot is becoming critical for the economy, health, and air pollution.

To address this parking problem, various technologies such as wireless communication, sensors, and machine learning are used in smart city applications. Within the scope of Internet of Things (IoT) systems, a network of low-cost and low-power sensors may provide a useful infrastructure for a smart parking system. Such a system should involve more features than showing the instantaneous information for available parking spots to the users. Because of the dynamic properties of the parking problem, a smart parking system needs to be capable of organizing the drivers in an intelligent way. One key feature may be the processing of historical parking data using machine learning methods to make statistical inferences and future predictions.

There are various cities in the world that are being equipped with various smart city features with high investment costs. In general, sensors are one type of the important tools for these

systems. For smart parking applications, sensors placed at the curbside parking spots make it possible to collect both spatial and temporal data. Some of the cities in which sensor-based parking lot occupancy data collection scheme is deployed are San Francisco, USA [2], Melbourne, Australia [3], and Westminster, England [4].

Recent advancements in various technologies enable the widespread usage of the sensors in smart city applications. One example of these technologies is the wireless communication protocols. Relatively recent protocol "long range wide area network" (LoRaWAN) makes wireless data transmission possible over distances up to 15 kilometers while consuming very small amount of power. Investments on other enabling technologies like cloud computing, big data analytics, and embedded systems triggers deployment of more smart parking systems in various cities. As a result, the amount of collected parking data increases and it becomes necessary to analyze the data using advanced methods. Analysis of such data involves performing statistical analysis on the previous data as well as predicting the parking space availability for a particular area.

Machine learning methods are suitable for such time-series prediction problems; hence, various machine learning methods are utilized for prediction of parking spot states. Among the traditional machine learning methods, Artificial Neural Networks (ANN) [5], Support Vector Machines (SVM) [6] and Regression Trees (RT) [6, 7] are widely used for this purpose. On the other hand, Long Short-Term Memory (LSTM), an architecture based on Recursive Neural Networks (RNN) for time-series prediction, is a commonly used deep learning method [8].

In this work, Autoregressive Integrated Moving Average (ARIMA) and LSTM methods are applied to parking data collected from curbside parking spots of Adana, Turkey. ARIMA is a statistical method where the data is regressed on its own prior values. LSTM is a deep learning method that can handle the long-term dependencies in the data with the help of memory cells in its structure. Both of these methods are developed for time-series analysis problems, hence this study compares the performances of these methods on the parking data and underlines the advantages of one method over other.

The rest of the paper is organized as follows. In section 2, related literature review is given. The background information about the LSTM and ARIMA models are provided in section 3. Section 4 contains the details about the dataset and the experiments. The results are given in section 5 and the paper is concluded with section 6.

2. Literature review

The importance of parking lot occupancy prediction has been mentioned in the literature frequently with an increased rate in

the recent years. The reason for this situation may be the emergence of IoT concept and its widespread applications. Within the scope of smart cities, the prediction of occupancy rates contributes to various aspects such as reduction in traffic congestion and environmental pollution. One other useful application can be given as efficient pricing for the smart parking systems. A machine learning based pricing system was proposed by Saharan et al., where on-street occupancy rates are predicted using four well-known methods, namely, linear regression, decision tree (DT), neural networks (NN), and random forest (RF) [9].

Typically, machine learning methods are utilized for solving this prediction problem. In a recent study by Awan et al., prediction performances of multilayer perceptron, k-nearest neighbors (KNN), DT, RF, and an ensemble of these methods were compared and it was shown that the voting-based ensemble method outperforms the others [10]. The ensemble methods attracted the attention of different researchers. For example, Sampathkumar et al. proposed a majority voting ensemble model that combines outputs of seven different prediction methods for predicting parking availability in a smart city [11]. On the other hand, further analysis of different ensemble methods for parking space availability was performed by Tekouabou et al. using four regression methods that are bagging, RF, adaptive boosting, and gradient boosting [12]. Instead of considering only the occupancy rate, Provoost et al, defined state of a parking area using occupancy rate as well as in- and out-flux of the vehicles [13]. They showed that this definition is less sensitive to the prediction horizon. Furthermore, there is a large amount of publications for this problem utilizing different variants of machine learning methods. These studies include road segment clustering and kalman filter based prediction [14], neural networks considering dynamic distribution characteristics [15], and wavelet neural networks [16].

Strategies utilizing deep learning methods for parking availability prediction is also very common [17]. For example, a hybrid approach was proposed by Qiu et al., where the parameters of an RNN model were optimized using genetic algorithm [18]. In the work proposed by Yang et al., several traffic data sources such as parking meter transactions, traffic speed, and weather conditions were used as input parameters to a Graph-Convolutional Neural Network (GCNN) models [19]. It was shown that GCNN has better prediction performance than the baseline LSTM and lasso models. Usage of ensemble learners together with deep learning models is also available in the literature. Lv et al. proposed an ensemble learning algorithm with an RNN model [20]. The optimized values for the parameters of the algorithm were determined using particle swarm optimization. At this point it is worth to mention that the number studies using time series analysis is limited when compared to machine learning and deep learning based methods [21, 22].

3. Background

3.1. Long short-term memory

LSTM is a type of RNN architecture and has been applied in various problems such as language modeling, speech-to-text transcription, machine translation and time-series prediction. LSTM networks are able to learn long-term dependencies in the data with the repeating memory cells in its structure. There are three main building blocks of the memory cells which are input, output and forget gates. These gates process the data from the previous hidden state (h_{t-1}), previous cell state (C_{t-1}) and the current input (x_t). The previous hidden state and the current input data are processed together by concatenating them to form current information (I_t). Specifically, forget gate decides whether to keep or forget the previous state of the cell by passing the current information through a sigmoid activation function. In the input gate, the important information within the current information is determined to allow the less important ones be removed from the cell. The outputs of these two gates are summed to update the cell state. This updated cell state (C_t) is passed to a tanh function and then multiplied with the sigmoid output of the current information. This final step is called the output gate because it outputs the updated hidden state (h_t) for the cell. A memory cell and corresponding operations are illustrated in Figure 1.

Repeating structure of such cells can be achieved by providing the output states of a cell as inputs to the next one, eventually yielding an LSTM network. The weights associated with the gates can be learned appropriately when a sequence data is used for training the network. As a result, the dependencies between the states of the sequence are modelled using these networks. An LSTM network having a visible layer with one input and a hidden layer with four neurons is used in this work.
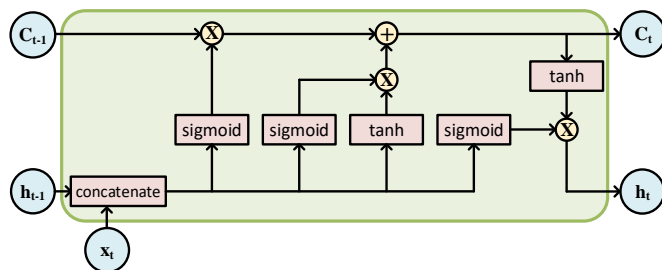

Figure 1. *Memory cell of an LSTM network*

3.2. Auto regressive integrated moving average

ARIMA is a statistical method particularly for time series analysis by which future values are predicted. It involves three main parts that are auto regression (AR), moving average (MA) and integration (I) of these two.

For a time series data, y_t, AR can be described as a model that depends on linear combination of its past values, y_{t-n}. Thus, it is a regression model based on its own lagged version. $AR(p)$ notation is used to denote an AR model of order p and is defined as

$$y_t = c + \sum_{i=1}^{p} \varphi_i y_{t-i} + n_t \qquad (1)$$

where c is a constant number, φ_i are the model parameters, and n_t is the white noise.

On the other hand, rather than past values, MA method uses linear combination of previous prediction errors and current error to model the current value of a variable. MA model of order q can be defined as

$$y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} \qquad (2)$$

where μ is the mean of the time series, ε_t are the white noise error terms, θ_i are the model parameters.

The combination of AR and MA methods cannot capture the non-stationary properties in the data. To overcome this situation, "integrated" part (I) of the ARIMA model is introduced. In this step the difference between the data points and their previous values are calculated. The data points are replaced with these difference values. This differencing process may be repeated more than once. Generally, the order of the integrated step is denoted with d and represents the number of consecutive differencing operations. As a result, an ARIMA (p, d, q) model is defined as

$$y'_t = c + \sum_{i=1}^{p} \varphi_i y'_{t-i} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots \\ + \theta_q \varepsilon_{t-q} \qquad (3)$$

where y'_t is the differenced version of the time series y_t. Obviously, for such a model, p is the order of AR part; d is the number of differencing operations, and q is the order of MA part.

In this work and ARIMA$(1,1,0)$ model is used to make predictions on the parking data.

4. Materials and methods

4.1. The dataset

The dataset used in this work is taken from Adana Metropolitan Municipality. It contains data for a curbside parking lot in the city center for five working days. The raw data was recorded in a vehicle-based format where each record

contained enter and exit time of a vehicle. A preprocessing step was performed to convert this data into a time-series format in which each sample denotes the total number of vehicles for one minute of a day. A sample parking data after preprocessing is shown in Figure 2. Since the curbside parking system is active only during the daytime, the data belonging to intervals of hours [00:00,09:00] and [17:00,23:59] are all zero. When performing the experiments, these hours are simply ignored and only the data for active hours is used and this corresponds to nearly 500 samples of the time series data.
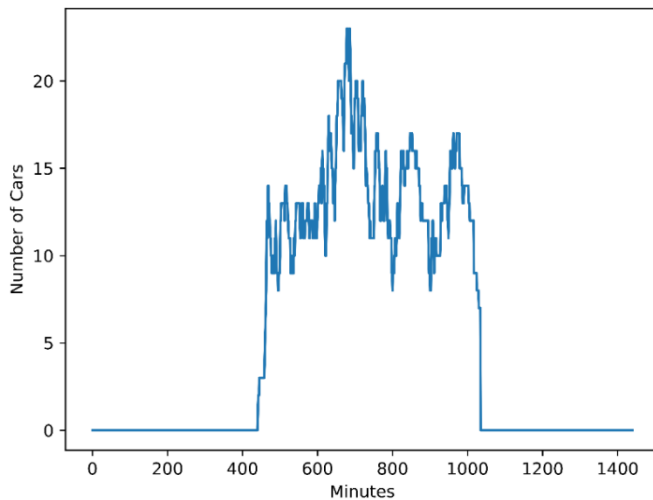

Figure 2. *Parking lot occupancy data for one day*

4.2. Experiments

For both ARIMA and LSTM methods, three different experiments were performed to predict the parking lot occupancy of 1, 5, and 15 minutes ahead. These experiments were applied separately to the days of data. In other words, the data belonging to each day is processed individually.

The sliding window approach is employed for making the predictions for one-step ahead. If x_t denotes the tth sample in the time-series data, x_{t+1} is predicted by training the model on all the samples from x_1 to x_t. Next, the samples from x_1 to x_{t+1} are used to predict x_{t+2}. This procedure is continued until the end of the time-series data. For this purpose, 66% of the one-day data is taken as the training set and the next single record in the dataset is used as test data. The remaining 34% of the one-day data is predicted in this manner and each prediction result is stored for performance analysis.

When training the LSTM model, five most recent data (i.e. x_{t-i} where $i \in \mathbb{Z}$ and $0 \le i \le 4$) are used as lookback features and the training is performed for 30 epochs. These values are determined by increasing the lookback features and the training epochs gradually. It was observed that larger values for these hyperparameters have no considerable improvement in the performance of the model. The learning curve showing the training loss for different epochs is given

in Figure 3. These parameters are not usable in the ARIMA model because it makes new predictions according to total updated history.
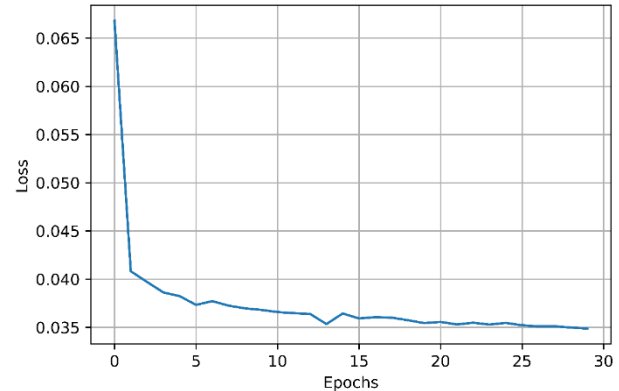

Figure 3. *Change of the training loss in LSTM*

The performances of the models are evaluated by computing root mean squared error (RMSE) and mean absolute error (MAE) which are given by the following equations:

$$RMSE = \sqrt{\frac{\sum_{k=1}^{n}(p_k - a_k)^2}{n}} \qquad (4)$$

$$MAE = \frac{1}{n}\sum_{k=1}^{n}|a_k - p_k| \qquad (5)$$

where p_k, a_k, and n denote the predicted values, actual values, and the total number of samples, respectively.

5. Results and discussion

As indicated earlier, the dataset contains data for five different days and future predictions for three different times (1, 5 and 15 minutes ahead) are performed. The prediction results for LSTM and ARIMA methods are given in Tables 1 and 2, respectively.

As can be seen from the tables, the minimum error for both methods are achieved when predicting one minute ahead. This is an expected situation as the problem of predicting the future values gets more difficult as the prediction horizon increases. This is consistent with the obtained results because the amount of error increases when the prediction horizon is larger. The increment is observable in all of the experimented days with different prediction horizons for both LSTM and ARIMA methods as well as both performance metrics. Sample predictions of both methods for different times are provided in Figures 4 and 5 where there is an apparent shift in case of 5-minute ahead predictions. In addition, the shift in the predicted values, there is a slight distortion and hence, the calculated error values increase eventually.

Table 1. *Prediction Results for LSTM*

| | 1 minute | | 5 minutes | | 15 minutes | |
	RMSE	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**
Day 1	**0.98**	**0.72**	1.13	0.89	1.88	1.57
Day 2	1.13	0.83	1.50	1.23	2.39	1.94
Day 3	1.19	0.90	1.56	1.23	2.28	1.84
Day 4	1.04	0.77	1.05	0.81	1.29	1.04
Day 5	1.14	0.85	1.29	1.00	1.91	1.51
Mean	1.10	0.81	1.31	1.03	1.95	1.58

Table 2. Prediction Results for ARIMA

| | 1 minute | | 5 minutes | | 15 minutes | |
	RMSE	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**
Day 1	**0.62**	**0.35**	1.47	1.07	2.82	2.47
Day 2	0.75	0.48	1.96	1.55	3.05	2.47
Day 3	0.73	0.47	1.63	1.21	2.71	2.17
Day 4	0.66	0.43	1.45	1.10	2.21	1.83
Day 5	0.78	0.48	1.71	1.27	2.26	1.83
Mean	0.71	0.44	1.64	1.24	2.61	2.15

According to the results presented in Tables 1 and 2, for all the experiments with a prediction horizon of 1 minute, ARIMA has lower error rate, while the error rate achieved by LSTM is generally lower when the prediction horizon is 5 minutes or 15 minutes. Thus, it is possible to compare the overall performances of the methods. On average, ARIMA is able to predict the future values when the prediction horizon is smaller. However, for larger values of prediction horizon the relative error for ARIMA gets higher than those for LSTM. It means that LSTM may be a better choice than ARIMA in case of a large prediction horizon.

This information can be handy when developing a hands-on application for a smart city parking system. If the driver is looking for a parking lot a nearby location, then ARIMA will be more suitable method for generation suggestions to the user. On the other hand, in case of a user looking for a parking lot a distant location (in other words, it will take some time for the driver to reach to the target location), usage of LSTM will allow the user to find a parking lot with a higher probability.
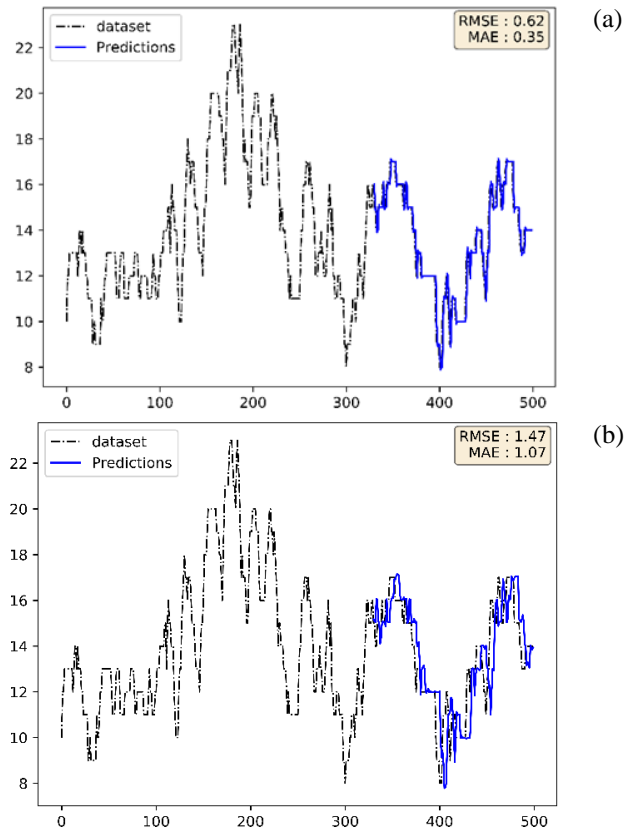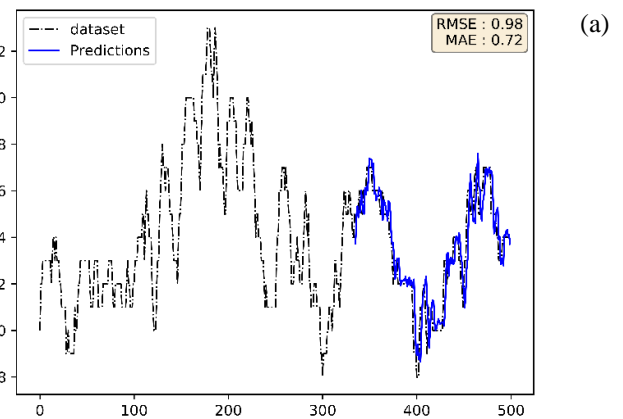

Figure 4. *Prediction results of ARIMA belonging to "Day 1" for (a) 1 minute and (b) 5 minutes ahead*
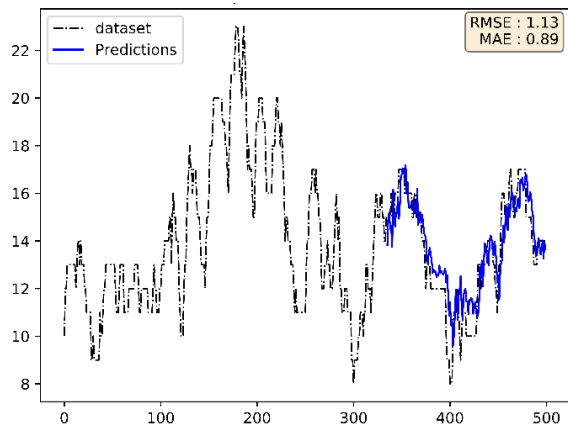
Figure 5. *Prediction results of LSTM belonging to "Day 1" for (a) 1 minute and (b) 5 minutes ahead*

6. Conclusion

As the city centers get more crowded, it becomes a necessity to organize and manage the facilities of the cities in an intelligent way. Car parking problem is one of the consequences of high population, which brings about several other issues such as generating extra traffic and increased carbon emission to the atmosphere. The machine learning and data analysis methods available today may allow for predicting the occupancy status of a specific parking spot, thereby it becomes possible to direct the drivers to the available place. In this work, ARIMA and LSTM methods are used to predict parking lot occupancy rate using the parking data collected from Adana, Turkey. For both of the methods, the experiments were performed with three different prediction horizons that are 1 minute, 5 minutes and 15 minutes. According to the results, it was observed that ARIMA outperforms LSTM when the prediction horizon is smaller. On the other hand, for the increased prediction horizons, the performance of LSTM is higher. This situation may be due to the presence of memory cells in the structure of LSTM that enables considering the long-term dependencies when making predictions. Since the predictions with horizons more than one minute are more feasible in practical applications, it may be concluded that LSTM method is more suitable than ARIMA for such problems. Increasing the amount of data for training the model as well as inclusion of traditional machine learning methods in the experiments are planned as the future works.

References

[1]. Shao W., Zhang Y., Guo B., Qin K., Chan J., Salim F.D., "Parking availability prediction with long short term memory model," in International Conference on Green, Pervasive, and Cloud Computing, Springer, (2018), 124-137.

[2]. https://www.sfmta.com/demand-responsive-parking-pricing, Accessed: 24 August 2021.

[3]. https://www.melbourne.vic.gov.au/about-council/governance-transparency/open-data/Pages/on-street-parking-data.aspx, Accessed: 24 August 2021.

[4]. https://www.smartparking.com/latest/case-studies/city-of-westminster, Accessed: 24 August 2021.

[5]. Vlahogianni E.I., Kepaptsoglou K., Tsetsos V., Karlaftis M.G., "A Real-Time Parking Prediction System for Smart Cities," Journal of Intelligent Transportation Systems, (2015), 20, 2, 192-204, doi: 10.1080/15472450.2015.1037955.

[6]. .Yanxu Zheng S.R, Christopher Leckie, "Parking Availability Prediction for Sensor-Enabled Car Parks in Smart Cities," 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP) Singapore, 2015.

[7]. Alajali W., Wen S., Zhou W., "On-Street Car Parking Prediction in Smart City: A Multi-source Data Analysis in Sensor-Cloud Environment," in Security, Privacy, and Anonymity in Computation, Communication, and Storage, (Lecture Notes in Computer Science, (2017), ch. Chapter 58, 641-652.

[8]. Li Xiangdong C.Y., CEN Gang, Xu Zengwei, "Prediction of short-term available parking space using LSTM model," The 14th International Conference on Computer Science & Education (ICCSE 2019).

[9]. Saharan S., Kumar N., Bawa S., "An efficient smart parking pricing system for smart city environment: A machine-learning based approach," Future Generation Computer Systems, vol. 106, pp. 622-640, 2020, doi: 10.1016/j.future.2020.01.031.

[10]. Awan F.M., Saleem Y., Minerva R., Crespi N., "A Comparative Analysis of Machine/Deep Learning Models for Parking Space Availability Prediction," Sensors (Basel), (2020), 20, 1, doi: 10.3390/s20010322.

[11]. Sampathkumar R.M.A., Pon Harshavardhanan, S. Murugan, P. Jayarajan, V. Sivasankaran, "Majority Voting based Hybrid Ensemble Classification Approach for Predicting Parking Availability in Smart City based on IoT," 11th ICCCNT 2020.

[12]. Koumetio Tekouabou S.C., Abdellaoui Alaoui E.A., Cherif W., Silkan H., "Improving parking availability prediction in smart cities with IoT and ensemble-based model," Journal of King Saud University - Computer and Information Sciences, (2020), doi: 10.1016/j.jksuci.2020.01.008.

[13]. Jesper L.J.J.W. C. Provoost, Sander J. van der Drift, Maurice van Keulen, Andreas Kamilaris, "Short Term Prediction of Parking Area states Using Real Time Data and Machine Learning Techniques," 2019.

[14]. Sergio Di Martino A.A., "Exploiting Recurrent Patterns to Improve Scalability of Parking Availability Prediction Systems," Electronics (2020), 9, 838.

[15]. Zhao Z., Zhang Y., Zhang Y., Ji K., Qi H., "Neural-Network-Based Dynamic Distribution Model of Parking Space Under Sharing and Non-Sharing Modes," Sustainability, (2020), 12, 12, doi: 10.3390/su12124864.

[16]. Liu J., Wu J., Sun L., "Control method of urban intelligent parking guidance system based on Internet of Things," Computer Communications, (2020), 153, 279-285, doi: 10.1016/j.comcom.2020.01.063.

[17]. Jamie Arjona M. P. L., Josep Casanovas-Garcia, Juan José Vázquez, "Improving Parking Availability Information Using Deep Learning Techniques," 22nd EURO Working Group on Transportation Meeting, EWGT 2019, 2020.

[18]. Qiu J., Tian J., Chen H., Lu X., "Prediction Method of Parking Space Based on Genetic Algorithm and RNN," in Advances in Multimedia Information Processing – PCM 2018, (Lecture Notes in Computer Science, (2018), ch. Chapter 79, 865-876.

[19]. Yang S., Ma W., Pi X., Qian S., "A deep learning approach to real-time parking occupancy prediction in transportation networks incorporating multiple spatio-temporal data sources," Transportation Research Part C: Emerging Technologies, (2019), 107, 248-265, doi: 10.1016/j.trc.2019.08.010.

[20]. Kedi Lv H.C., Yingda Lv, "Parking Space Predicting Algorithm Based on Recurrent Neural Network and Ensemble Learning Algorithm," Proceedings of CCIS2019.

[21]. Zhao Z., Zhang Y., Zhang Y., "A Comparative Study of Parking Occupancy Prediction Methods considering Parking Type and Parking Scale," Journal of Advanced Transportation, (2020), 1-12, doi: 10.1155/2020/5624586.

[22]. Stolfi D.H., Alba E., Yao X., "Can I Park in the City Center? Predicting Car Park Occupancy Rates in Smart Cities," Journal of Urban Technology, (2019), 1-15, doi: 10.1080/10630732.2019.1586223.