IoT Enabled Smart Parking System for Improvising the Prediction Availability of the Parking Space

Anchal Dahiya, Pooja Mittal
MDU, Department of Computer Science, Rohtak, India

***Abstract*—Smart cities are a result of persistent technological advancements aimed at improving the quality of life for their residents. IoT-enabled smart parking is one of the foundations of smart transportation which seeks to be versatile, long-lasting, and integrated into a Smart City. One of the studies shows that the drivers who are searching for free parking space can cause congestion problems up to 30%. There is a possibility to reduce air pollution and fluidity noise traffic by combining Internet of Things (IoT) sensors positioned in different parking areas with a mobile application and help the drivers to search for free places in different areas of the city and also provide guidance toward the parking space. In this paper, we show and explain a unique Data Mining-based Ensemble technique for anticipating parking lot occupancy to reduce parking search time and improve car flow in congested locations, with a favorable overall impact on traffic in urban centers. In this paper multi scanning, IoT Enabled smart parking model is proposed along with ensemble classifier that improvises the predictive availability of the free parking space. The predictors' parameters were additionally optimized using a Bootstrap and bagging algorithm. The proposed method was tested an IoT dataset containing a number of sensor recordings. The tests conducted on the data set resulted in an average mean absolute error of 0.07% using the Bagging Regression method (BRM).**

Keywords—IoT; Data mining; sensors; ensemble; decision tree; bagging technique; boosting technique

I. Introduction

Data Mining plays a significant role in the digitalization of the technologies fusion with the Internet of things, virtual reality, 5G connectivity, and many more. This technology's influence is not only restricted to information systems [1]. It also affect other sectors too such as design, transportation, healthcare, shipping, and business procedures. The integration and amalgamation of IoT, Data Mining, and pervasive interconnection can accumulate enormous data inside a city, distribute the data to the central database, and use it for processing. For example, to make city parking and operations more efficient, as well as to support the household waste and hospitality sectors [2]. It is important to recognize that a "Smart City" is not only a city that is interconnected with IoT technology or simply "uses" the data connection. Future Smart City is an interconnected system, a "brain," in which innovation, management, infrastructure, and residents all communicate with one another [3]. IoT and Data mining allows us to track the patterns in a city's everyday existence, then modify them according to the requirements to create them more effective and real. Tragedies, accidents for example, is discovered by direct reporting, permitting for quicker action and actual traffic reporting to suggest alternate paths to escape congestion bottlenecks, simplifying movement and lowering consumption and CO2 emissions. In addition, to the persistent parking issue in large cities, Data Mining can assist both in terms of detecting and reporting to residents the available parking spots in the neighborhood, as well as in terms of emphasizing to governments where new ones are needed.

According to Juniper Research's "Smart Cities: Leading Platforms, Segment Analysis & Forecasts 2019–2023," innovative solutions for smart city traffic, which are used to alleviate traffic jams in towns, would produce $4.4 billion in revenue in 2023, up from $2 billion in 2019. Smart parking methods, which are a result of technological advancement, are causing a revolution in the parking industry. Nowadays, finding a parking space in a city is a time-consuming and difficult process that can quickly turn into a nightmare. Due to the Internet of Things and Data Mining, it become possible to examine, and discover parking utilizing technology instruments like sensors, cameras, linked, and integrated things. As a result, smart parking becomes an important component of smart cities. IoT enabled Smart parking is an approach that integrates technology and human to use fewer resources (time, energy, and area) for attaining quicker, and optimum parking of the vehicles during the time when the cars are not in use[4]. To put it another way, IoT-enabled smart parking systems determine which parking areas are engaged and which are vacant, as well as build a real-time parking map. Real-time maps are useful in different areas.

- Assist the drivers to search the vacant space rapidly by using the mobile application.
- Ability to make actual information accessible, which allows the agents to spot any problems.
- Assist individuals in deciding on another mode of transport if the parking area is unavailable.

To conclude, IoT enabled smart parking system is made up of sensors, collection of actual records and analytics, and a smart payment methods which permit users to search for parking space in the preferred location and make the payment in advance if required. Traffic is severely limiting mobility in cities around the world, resulting in enormous expenses not only for travelers but for society itself. According to the Texas Transportation Institute, the average U.S. passenger spent 34 hours in traffic in 2010, up from 14 hours in 1982, and is likely to spend more than 40 hours in 2020. In areas where people work and reside, traffic contributes to pollution and noise. IoT-enabled smart parking provides a solution to the problem.

As the above scenario is described, to designing a predictive analytics framework by using the predicting algorithm and IoT-enabled sensor-generated data is a big challenge. It can also be used in a variety of real-world situations, such as air pollution forecasts, novel healthcare services, weather forecasts, etc. [5, 6, 7]. We propose a predictive technique in the Smart Mobility realm, with an emphasis on parking space availability. To solve the cities' parking problem, we are providing a consistent immediate estimation of parking space availability. In this regard, the goal is to develop a reliable structure of day-to-day 1-hour and 4-hour range estimations based on data provided by IoT sensors. Data mining techniques are used to produce a 1/2-hours-horizon prediction of parking space availability, improving the prediction of the well-known methodologies.

A. Organization of the Article is as Follow

The paper is organized as follows. Literature survey is presented in Section 2. In Section 3 methodology is described that contains a block diagram of multi-scanning model for IoT Enabled smart parking system. In this section, an overview of different techniques are also described that will help in the prediction of parking space. In Section 4 the performance of these techniques is represented using the performance measure R2 and Mean absolute error (MAE). And finally, conclusion is presented in Section 5.

II. Related Work

The section is related to the review of the literature and seeks to highlight the key terms, perceptions, and flows of thinking in the area of parking through the readings. The important concern is the Parking; the following section emphasizes the features relevant to the precise objectives of the study.

There are two subsystems in the parking management system. A vehicular detection method (VDM) and a vehicular control method (VCM) are both available. The VDM monitors the availability of free spaces for parking and delivers the data to the VCM module for distribution of the availability of the space to drivers [8]. An intelligent parking system is also proposed by another author Kumar et al.[9] .The authors compared different types of sensors used in smart parking systems and data generated by different sensors are sent to the central database in a predefined amount of time. Another author proposed a parking system with intelligence that is created on the visual processing, the technique decides the accessibility of the space using deep learning. The proposed model is equated using the methods that are exit in the PKLot and CNRPark_EXT[10,11,12].Other authors implemented a prototype that uses, RFID, sensor, and IoT to identify vehicle particulars and also uses IR sensor to locate the vehicle's existence, allowing all details to be accessible via IoT devices[13].Pawowicz et al. [14] used an RFID-based technique to better handle traffic control in an urban areas, although the issue of predicting continues with this technique.

The authors of Giufr et al. [15] suggested an Intelligent Parking Assistant. Their intelligent parking management system is based mostly on the utilization of sensor networks. However, this research ignores the technique of machine learning and the benefits of the Internet of Things. Another author uses Haar-AdaBoosting and CNN algorithms, Xiang et al. [16] suggested a method for the identification of real-time parking utilization at gas stations. Another deep learning case study [17] developed an automatic valet parking that is based on robotic valets which uses hybridization in smart parking that helps to maximize the use of parking area that uses the well-known technique that is Deep Q-Learning, a reinforcement learning. The authors of Camero et al.[18] (2018) introduced a new technique for processing parking occupancy rate predictions that uses deep learning by combining recurrent neural networks. The paper [19] provided a framework for designing an efficient parking control system that uses an innovative video processing technology that will help to assign vehicles to the vacant open parking slot at the entry point. This method is used for real-time forecasting of events. Stolfi et al. [20] use a study using parking occupancy data to examine a variety of forecasting methodologies that includes time series, Fourier series, k-means grouping, and polynomial alteration. The developed model is still critical and does not include rising technologies such as IoT and can be improved. Bachani et al.[21,32] provided a thorough examination of one of the most important components to develop an intelligent parking system, which includes the selection of appropriate sensors and optimal placement for precise detection.

In this paper, we propose an IoT-enabled smart parking model that will help in predicting the available parking space that fuses two technologies that is data mining and IoT technology.

III. Methodology

In the literature, many works are presented as smart parking system but fails to meet the major challenges. The issues which are to be at the limelight are user privacy, selection of parking slots in quick time, presenting an efficient and real-time considerable system, etc. In the current work major of the issues presented are considered for a solution. The current work presents a predictive system, which uses data mining techniques for fast and easy retrieval. For complete prediction proves Ensemble technique is considered. The entire system is designed considering two different types of agents as detection agent and the smart parking agent, which reacts as per the learning and retrieval results.

A. System Model of IoT Enabled Smart Parking System

There is the number of models available but the shortcoming of the available models is that a large amount of data is generated by the sensors to handle that data there is no specific model available that will handle a large amount of data. That is the reason we proposed multi-scanning model because sensors generate a large amount of data every second and we need to accumulate only that data which is important for parking-related information. The main idea behind making the smart parking system is to utilize the available parking space efficiently.

"The block diagram of the multi-scanning proposed methodology is shown in Fig. 1".

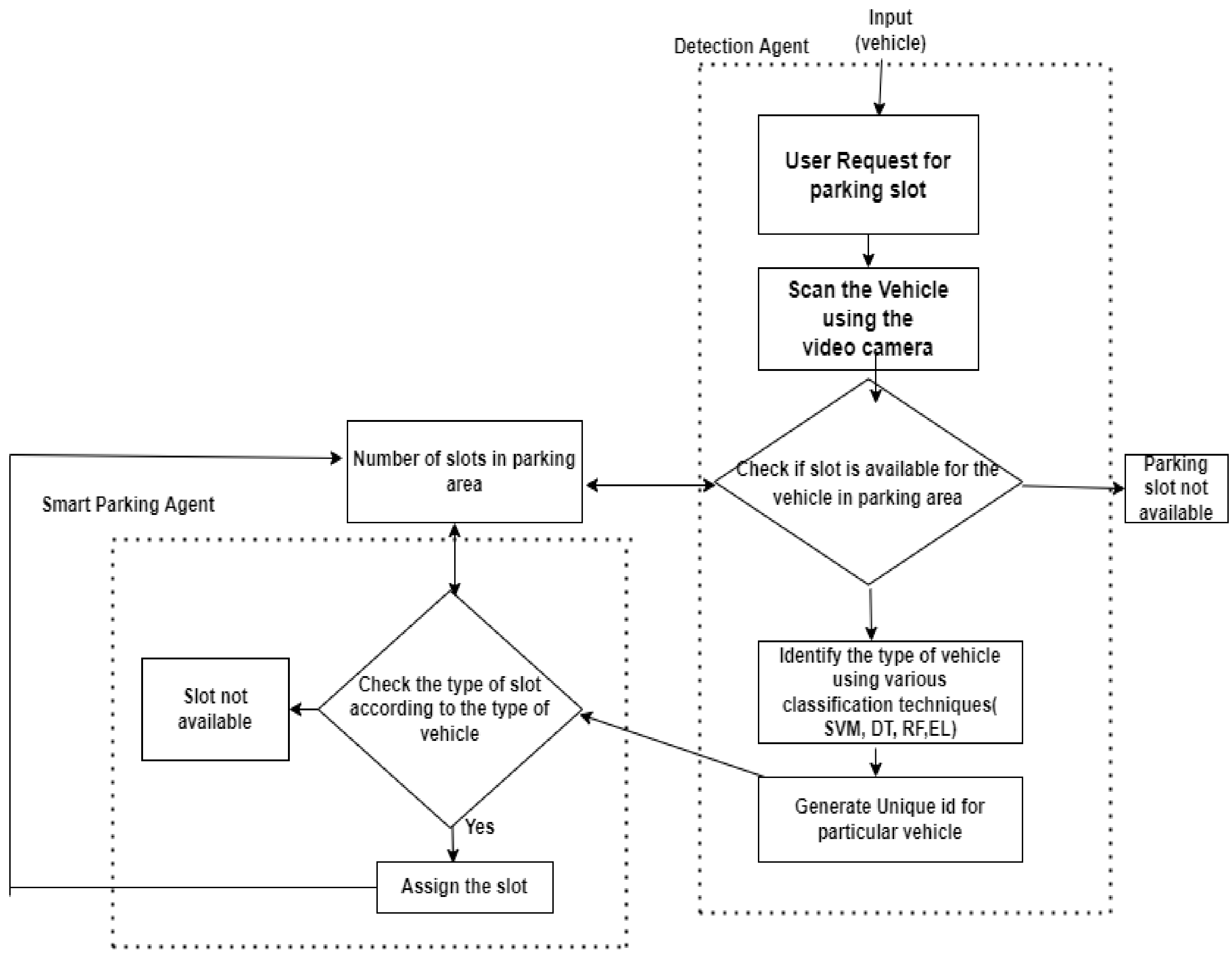

Fig. 1. Multi Scanning Model for IoT Enabled Smart.

The methodology is divided into two different parts:

- Detection agent.
- Smart parking agent.

The detection agent performs the identification of the type of vehicle. First, the user gives a request for an available parking slot. After getting the user's request, the system starts to detect the type of vehicle the sedan, coupe, SUV, bus, auto, etc., and assign a particular vehicle id to the specific vehicle. In the next part, the smart parking agent gives a particular location to the vehicle if the slot is available in the parking space. The slots of the parking areas are classified in advance by the smart parking agent according to the type of vehicle. Based on classified slots the smart parking agent will assign the available parking slot to the vehicle. The classifier that will help the smart parking agent is the ensemble technique.

The IoT-enabled smart parking model is a smart structure that uses a detection method to determine the parking space availability to assist the drivers with the accessibility of the parking space. A smart parking system needs the following elements:

- Sensors.
- Smart mobile application.
- Database.
- Gateway Hardware.

B. *Ensemble Predictive Technique*

This classifier combines several basic techniques such as decision tree or random forest. Every basic model provides an alternative approach to predicting the datasets and the final forecasting is the result of the combination of the basic models that will enhance the accuracy of the results. Because the uncertainty is significantly smaller than that provided by one of the individual base models that make up the overall model, the prediction technique of combining the predictions of a collection of individual base models often provides for more consistent and accurate output prediction. Therefore, the final ensemble-based model rectifies the individual errors generated by the fundamental models, resulting in a significant reduction in total error. The basic models should be required to meet two characteristics to be effective: they must be independent and weak models.

The original plan was to split the training data D into n basic data sets to train n models (m1;m2,..., mn). When n becomes high, however, this strategy is eventually outperformed because it promotes under-fitting. To deal with this kind of limitation various approaches are used that will re-sample the training data into n independent and greater data sub-samples to develop weak models. The number of techniques has been employed to accomplish it, the most well-known of which include bagging and boosting. As a result, the suitable algorithms for such kind of situations are volatile algorithms that include decision trees and neural networks that are modified during data set to produce a different model. A flow chart of a prediction system based on a set-based model is shown in Fig. 2.

The next sections covers our basic model which is decision trees and also contain two sampling strategies bagging and boosting and three-set models that contain the Random Forest regressor method (RFRM), Gradient boosting method (GBM), and Adaboost regressors(ARM) methods.

C. Decision Tree

The decision tree is used to depict the repeated division of the original entire space in the prediction tree technique. The terminal node or the leaves nodes represents a partition cell and is linked with a basic model that will apply only to that particular cell. For better understanding, think about a regress equation with two independent variables X1 and X2, and a continuous output variable Y. The concept is to divide the space into two sections and model the Y response (mean of Y) in every area separately. We again divide each section into two more sections and repeat the procedure till reaches a halt rule [22]. Each region's answer is frequently treated as a Cm constant.

$$\text{Mininize}(SSE) = \sum_{Ek-1}^{Em} \sum_{i=1}^{n} (yi - ck)^2 \quad (1)$$

$$\text{Minimize SSE}+\alpha|T| \quad (2)$$

The best Cm is simply the average of Yi in the Rm region because the optimization objective is to minimize the sum of squares [23]. Applied a cost complexity parameter ($\propto$) to identify the optimal region, which disciplines independent function in Equation (1) for leaf nodes of the tree T as given in Equation (2). The greedy algorithm is used to find the best division variable and split point. The optimal partition point for each splitting variable can be found by scanning all important parameters; it should be done fast. It is feasible to determine the finest pair of division parameters and division points by examining all of the input variables.

D. KNN for Regression

KNN uses non-parametric methods to resolve any regression problem, it is also known as lazy learner [24].If the incident values are categorical that it will assign the regression value by aggregating the nearest k neighbor values for quantitative occurrences or implementing the majority vote to the k neighbors. By aggregating the k neighbor outputs (constant weights) the values can be forecasted [25].

E. Ensemble Technique for Regression

The summary of the general idea is shown in Fig. 3 it shows that the model is based on three parts that is bootstrapping, intermediate model, and aggregation. Bootstrap split all the data D into n data D1 ,D2…Dn. Now we create an intermediate regressor Rj for each data set Dj and an aggregate of the successive regressors Ri will be the final regressor. The bootstrapping used in the Random Forest algorithm and the Boosting used by Gradient Boosting and Adaptive Boosting are two of the most powerful approaches derived from this fundamental principle.

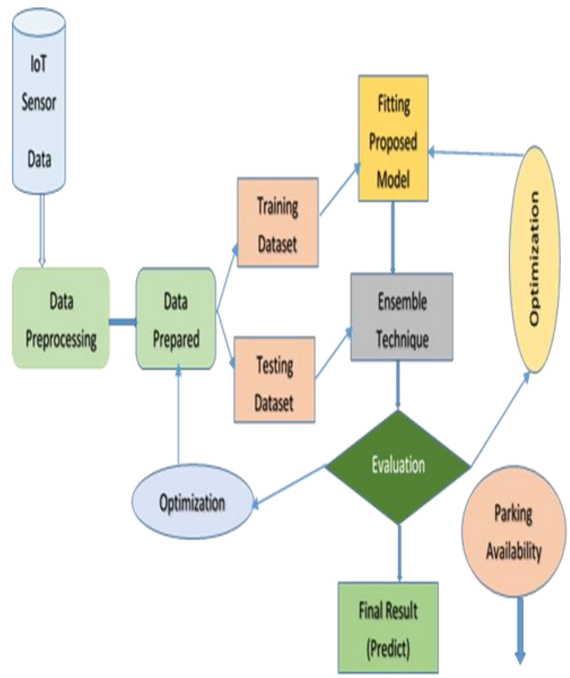

Fig. 2. Ensemble Technique for Prediction of Real Time Parking Space.

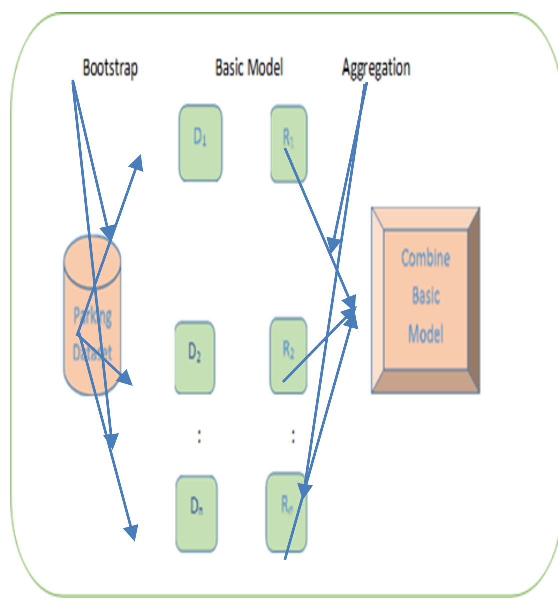

Fig. 3. Basic Working Approach of Ensemble Technique.

F. Bootstrap Aggregation / Bagging Technique

1) Bagging Regression Method (BRM): Bagging term is derived from "Bootstrap Aggregating." The algorithms are shown in the context of regression and they can be easily extended as a supervised classes. Random vector describes the learning data is denoted by the values that is (X, Y), where X is in Rq and Y is in R. Dn = (X1; Y1); (Xn; Yn) is a sample that is independent and evenly distributed. The regression function (X, Y) and m˜(x) = E [Y|X=x] are both defined by the same law. The mean square error of an estimated m and its bias-variance representation for x ∈ Rp is as follows:

$$(\widehat{\mathrm{m}}(\mathrm{x}) - \mathrm{m}(\mathrm{x}))^2 = (\mathrm{E}\widehat{\mathrm{m}}(\mathrm{x}) - \mathrm{m}(\mathrm{x}))^2 + \mathrm{V}(\widehat{\mathrm{m}}(\mathrm{x})) \quad (3)$$

These comprise of integrating a number Z of models m1; m2,..., mz in such a way that:

$$\widehat{\mathrm{m}}(\mathrm{x}) = \frac{1}{\mathrm{Z}}\sum_{\mathrm{k}=1}^{\mathrm{Z}}(\mathrm{m}_{\mathrm{k}}(\mathrm{x})) \quad (4)$$

And

E(m)(x)=Em_1 (x) (5)

And

V(m(x))=1/Z V(m_1 (x)) (6)

As a result, the composite model's bias is the same as the m ˆ k, but the variable reduces. By generating aggregated models on bootstrap samples, the bagging method attempts to reduce the correlation between them.

2) Random Forest Regression Method (RFRM): It is a specific bagging approach that involves a collection of trees that is based on random factors. The classification and regression tree algorithm, whose concept is to iteratively split the space formed by the independent variable reciprocally, is most commonly used to create trees. For each phase of the partitioning, a section of the space is divided into two sub-parts based on a variable Xj[23].

G. Boosting Method

The basic idea behind this method is to create a group of models that are combined using a weighted average of predictions. In contrast to the previous bootstrap aggregation, the standard increase in subgroup formation is not random. The main impression behind boosting algorithms is to train forecasters in order, with everyone attempting to accurate the one before it [26,27]. Since every new subdivision is repeated on the former comprised features that could have been mislabeled by prior models, so the performance depends on the performance of preceding models. Most general boosting methods are AdaBoost and Gradient Boosting.

1) AdaBoost Method (ABM): It is also known as the adaptive boosting method. It is centered on the assumption that a new forecaster pays a bit more recognition to the training sample under which the antecedent has adjusted to rectify the fault of its precedent. The outcome is the new predictor whose main focus is more on difficult cases. Let's take an example, Assume a predictor which is nothing more than a decision tree for creating an AdaBoost classifier. Predictions on the set of patterns are made using a fundamental tree structure. After that, the weight assigned to the misclassified training instances is increased. The altered weights are then used to create a second classifier [28]. The second classifier predicts the outcome of the training game once more. After that, the weights are adjusted and the process will continue until all the predictors are in order, the set makes predictions in the same way as bagging do. The primary difference is that the weights of the generated predictors are dependent on their total accuracy calculated over the weighted training set [29].

2) Gradient Boosting Method (GBM): Gradient Boosting is similar to AdaBoost in that it gradually adds predictors to a set, with each one attempting to rectify the faults of the one before it. Instead of modifying the class weights at each repetition as AdaBoost does, this method seeks to match the new classifier to the prior one's residual errors [29].

IV. Implementation Results

The given section, analyzes and discusses the performance of the proposed methodology using a data mining classifier that is Ensemble technique. Section A, discuss about the dataset.

A. Dataset

Fig. 4 shows the real-time prediction of parking space using the ensemble technique. Sensors are installed in different parking areas in smart cities. These sensors are connected to the database that collects the IoT data with the help of sensors installed in different places. In this paper, data was collected using Birmingham in (U.K) dataset as a CSV file format. Parking sensors [30] collect over 6 months of data. The process of selecting relevant data involves removing inappropriate and duplicate information [30].The Parking dataset consists following features.

- Code Number: It is described by an alphanumeric value that will identify the parked vehicle.
- Last Modification: It provides the time and date of the most recent updation for every parking block's availability data. Between 9:00 a.m. and 5:00 p.m., schedules are logged.
- Size: It contains the capability of each parking space.
- Occupancy: It includes every vehicle park's activities that are modified every 30 minutes.
- Status: It represents the status of the parking space i.e. free or available.

We created a particular feature called the accessibility rate (AR) from these characteristics, which is the ratio of the availability excluding the occupancy on the parking areas space at time t of the dated d. In our scenario, we use the following formula to calculate it:

$$\mathrm{AR}_{\mathrm{q}}(\mathrm{t}) = \frac{\mathrm{Capicity}_{\mathrm{q}} - \mathrm{Occupancy}_{\mathrm{q}}(\mathrm{t})}{\mathrm{Capacity}_{\mathrm{q}}} \quad (7)$$

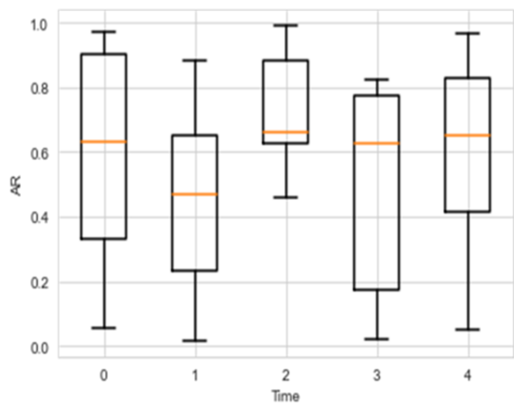

Fig. 4. Scattering of Availability of Parking Area as a Function of Time.

For reference, car park "BHMBCCMKT02" has a capacity of 455 parking spaces and an occupancy of 51 on 05/11/2020 at 06:49:32, thus we may infer our occupancy rate at this moment is 51/455, or 10.53 percent. The accessibility rate in the parking lot at the present time of day is 88.42%, indicating that the user can park here without causing traffic congestion. Fig. 4 illustrates the rate of parking availability as a function of time. Having the availability rates in real-time we are now applying data mining algorithms to forecast the future availability rates without knowing occupancy.

B. Performance Measure

We evaluated the performance of the model using two primary parameters to develop an ideal approach: the r-square (R2) and the mean absolute error (MAE). In various aspects, the two terms can be used to assess the gap between actual and predicted parking availability rates. The following formula is used [31].

$$R^2 = 1 - \frac{\sum_{j=1}^{N}(AR_{j,p} - AR_j)}{\sum_{j=1}^{N}(AR_{j,p} - AR_J)} \quad (8)$$

$$MAE = \frac{\sum_{j=1}^{N}|AR_{j,p} - AR_i|}{N} \quad (9)$$

Here N stands for a complete number of cases, ARj,p, it is the predicted accessibility rate of the case j & ARj is the actual available rate in the particular event. The use of a single metric may not always allow the models to be distinguished. The absolute error may reflect the effect of precision in the prediction of the waiting time, and R2 would also demonstrate to us the percentage of the exact waiting time that has been successfully predicted, whereas the MAE will show the error characterized by the deviation and mean between the predicted and the real by choosing the effects of high variations. The consistency of these two measures will lead to the best model.

C. Optimization of Hyper Parameters

Here we will compare the performance of different mining techniques discussed in the section for the prediction of available space in the parking lot. We separated our dataset as discussed in the section into two sets: a training set (80%) for building the models and a testing set (20%) for evaluating the model and comparing their performance. To begin, we searched for factors that would allow us to construct reliable and efficient models. It is important to determine the number of iterations, the learning rate, and the optimal weak learner for the boosting procedure. The number of repetitions, the out-of-bag error, and the best weak learner is the most critical factors in bagging algorithms.

The tests that were conducted to choose these variables using a basic decision tree as a weak learner are shown in the figure.

The appropriate parameters for the boosting method are shown in Fig. 5. We can see from Fig. 5 that, how our model works on optimized parameters. The default value learning rate is 0.1, and the effective rate for Gradient Boosting is about 0.4. For AdaBoost, the same frequency might be used. The appropriate number of predictions for Gradient Boosting is at least 200, although it is not as necessary for Adaboost (about 30 predictor variables). Beginning with the estimation, a persistent optimum classifier based on bagging is created. The out-of-bag error is nearly zero. We can observe from the graph that the distribution of BRM followed by RFRM is more suitable for greater performance than GBM and ABM, which are both less efficient.

1) Result analysis of total dataset of parking area: In this, we evaluate the result on the total dataset of all the parking areas. To evaluate overall parking capacity, we used the attribute "Code Number" that is used for giving the identity numbers ranging from 0 to 25 for the 24 parking slots. The evaluations involved assessing and making comparisons of four different approaches. In the bagging algorithm, we use an optimization approach that is (BRM and RFRM) and boosting algorithm uses (GBM and ABM) approach. Table I shows the results, from which we created in Fig. 6 and Fig 7, which depict the comparison results. In comparison to the boosting method, the bagging methods (BRM and RFRM) produce the best results for all two metrics.

MAE produces 0.000771 & BRM produces 0.999951 here it is clear that BRM gave an optimal performance. In respect of R2, both BRM and RFRM achieved a near-perfect score of 0.999951. These results are somewhat better than RFRM, with an improvement of less than 0.00006 for these two metrics.

MAE and R2 received expert evaluations of 0.024523, and 0.982871 respectively, from GBM. These results were superior to ABM's 0.087872 and 0.834494 for these metrics, respectively. The findings of the two boosting methods evaluated (GBM and ABM) were substantially different from those of the very similar bagging methods, as demonstrated by the trend in Fig. 6 and 7. For the tests conducted, boosting approaches were slightly less effective than bagging approaches.

TABLE I. COMPARISON OF PERFORMANCE

		MAE	R^2
Bagging	BRM	0.000771	0.999951
	RFRM	0.000783	0.999949
Boosting	ABM	0.087872	0.834494
	GBM	0.024523	0.982871

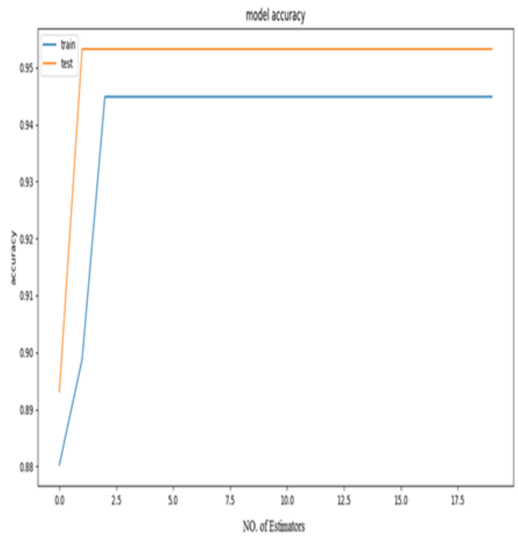

Fig. 5. Optimized Parameters of Boosted Method.

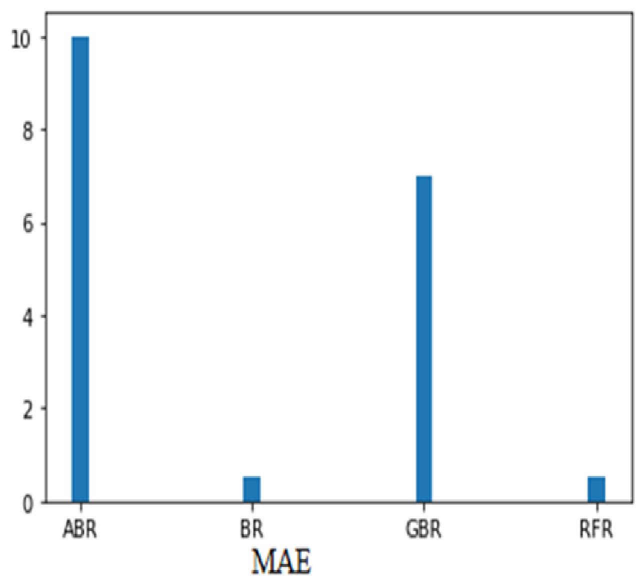

Fig. 6. Comparison of Performance using Mean Absolute Error (MAE).

D. Result Analysis

To validate the algorithms, we performed two tests: the first on the total dataset of all parking areas, and another on every parking area data.

1) Result analysis of every parking area: We put our approaches to the test to see if we could estimate parking availability in each area separately. Comparison to earlier work that simply employed one loss function. In Table II we compare the results using two loss functions that are Mean absolute error and r- square (MAE, and R2). We also compare the minimum, maximum and average performance (i.e. R2, MAE) values in the same table.

2) Comparison of MAE of predicted value with each parking space with previous work: MAE values of the various approaches, the RFRM technique finds the smallest mean value of 0.00055. For R2 we observe that BRM gave the optimal mean value i.e. 0.99980 compared to other approaches. As a result, we reached the conclusion that optimizing using the bagging method, particularly BRM, was the most effective way of estimating the available space in each parking space.

The fundamental goal of prediction is to forecast values that are close to accurate as feasible, we compared the results of our method to those achieved in earlier studies (Table III). In the previous work, only one measure is used i.e. MAE. The comparison table depicts that our best method (BRM) reduced mean absolute error by 7.6% on average when compared to RNN [18] and by more than 6.8 percent when compared to [20], which used and compared to time series(TS), Fourier series (F), k-means clustering (KM), shift and phase modifications (SP), polynomial(P), polynomial fitted by k-mean centroid (KP). Furthermore, the standard deviation is also low using BRM (0.00023) when compared to earlier work, which had a minimum of at least 0.026. Even improved, when compared to earlier algorithms, BRM proves to be faster.

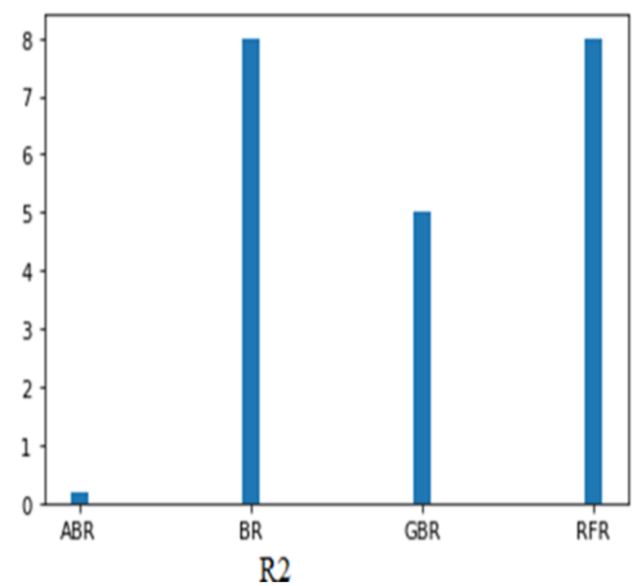

Fig. 7. Comparison of Performance using r-square (R^2).

TABLE II. MAE AND R2 PREDICTION AVAILABILITY IN EACH PARKING

Vehicle ID	**R^2**				**MAE**			
	BRM	**ABM**	**GBM**	**RFRM**	**BRM**	**ABM**	**GBM**	**RFRM**
BHMBCCMKT02	0.9996	0.9951	0.9996	0.9996	0.0004	0.0090	0.0013	0.0005
BHMBCCLKT04	0.9996	0.9945	0.9996	0.9996	0.0005	0.0013	0.0012	0.0007
BHMBCCMKT05	0.9996	0.9972	0.9996	0.9996	0.0005	0.0092	0.0013	0.0005
BHMBCCMKT08	0.9996	0.9972	0.9996	0.9996	0.0004	0.0110	0.0011	0.0006
BHMBCCMKT10	0.9996	0.9971	0.9997	0.9996	0.0006	0.0142	0.0013	0.0004
BHMBCCMKT12	0.9997	0.9973	0.9996	0.9997	0.0006	0.0106	0.0015	0.0004
BHMBNCMKT15	0.9998	0.9962	0.9997	0.9996	0.0004	0.0060	0.0009	0.0007
BHMBCCMKT20	0.9972	0.9907	0.9962	0.9960	0.0012	0.0020	0.0012	0.0007
BHMBPKMKT23	0.9996	0.9961	0.9996	0.9996	0.0004	0.0145	0.0009	0.0007
BHMBCCMKT26	0.9997	0.9962	0.9997	0.9997	0.0003	0.0154	0.0012	0.0006
BHMBCCMKT01	0.9998	0.9900	0.9998	0.9996	0.0003	0.0150	0.0012	0.0005
BHMBCCMKT07	0.9989	0.9971	0.9985	0.9963	0.0006	0.0090	0.0014	0.0006
BHMBCCMKT09	0.9996	0.9956	0.9995	0.9965	0.0012	0.0080	0.0012	0.0005
BHMBCCMKT34	0.9971	0.9992	0.9970	0.9996	0.0004	0.0100	0.0009	0.0007
BHMBCCMKT27	0.9989	0.9941	0.9985	0.9981	0.0006	0.0061	0.0008	0.0006
NIA Park C26	0.9996	0.9951	0.9996	0.9996	0.0012	0.0101	0.0012	0.0005
NIA South	0.9996	0.9981	0.9996	0.9996	0.0005	0.0153	0.0012	0.0005
NIA North	0.9996	0.9955	0.9996	0.9996	0.0004	0.0170	0.0009	0.0005
BHMBDDMYT29	0.9996	0.9919	0.9994	0.9991	0.0012	0.0170	0.0010	0.0006
BHTBFFDTYK37	0.9996	0.9957	0.9994	0.9991	0.0005	0.0080	0.0005	0.0004
BHTBFFDTYK40	0.9996	0.9932	0.9996	0.9996	0.0004	0.0082	0.0008	0.0005
BHTBFFDTYK45	0.9996	0.9972	0.9996	0.9996	0.0003	0.0045	0.0012	0.0006
BHTBFFDTYK48	0.9996	0.9956	0.9996	0.9996	0.0005	0.0076	0.0009	0.0005
BHTBFFDTYK42	0.9996	0.9919	0.9996	0.9996	0.0004	0.0003	0.0058	0.0004
mean Value	**0.99980**	0.99471	0.99978	0.99972	0.00056	0.00976	0.00111	**0.00055**
max Value	0.99996	0.99819	0.99995	0.99996	0.00143	**0.01812**	0.00162	0.00152
min Value	0.99718	**0.98563**	0.99749	0.99642	0.00029	0.00208	0.00055	**0.00028**
standard deviation	0.00051	0.00298	**0.00041**	0.00050	**0.00023**	0.00428	0.00025	0.00024

TABLE III. COMPARISON WITH PREVIOUS WORK

Vehicle ID	**SP**	**KPF**	**KM**	**F**	**P**	**TS**	**RNN**	**BRM**	**ABM**	**GBM**	**RFRM**
BHMBCCMKT02	0.032	0.051	0.086	0.085	0.058	0.066	0.062	0.0004	0.0090	0.0013	0.0005
BHMBCCLKT04	0.067	0.071	0.147	0.148	0.082	0.112	0.136	0.0005	0.0013	0.0012	0.0007
BHMBCCMKT05	0.123	0.142	0.18	0.147	0.138	0.068	0.116	0.0005	0.0092	0.0013	0.0005
BHMBCCMKT08	0.123	0.143	0.136	0.132	0.122	0.087	0.102	0.0004	0.0110	0.0011	0.0006
BHMBCCMKT10	0.133	0.149	0.147	0.148	0.132	0.094	0.132	0.0006	0.0142	0.0013	0.0004
BHMBCCMKT12	0.078	0.115	0.123	0.121	0.096	0.087	0.110	0.0006	0.0106	0.0015	0.0004
BHMBNCMKT15	0.048	0.087	0.112	0.110	0.073	0.058	0.075	0.0004	0.0060	0.0009	0.0007
BHMBCCMKT20	0.055	0.056	0.086	0.084	0.035	0.041	0.076	0.0012	0.0020	0.0012	0.0007
BHMBPKMKT23	0.067	0.034	0.065	0.062	0.066	0.067	0.061	0.0004	0.0156	0.0009	0.0007
BHMBCCMKT26	0.076	0.084	0.073	0.071	0.083	0.128	0.085	0.0003	0.0154	0.0012	0.0006
BHMBCCMKT01	0.087	0.083	0.15	0.126	0.072	0.033	0.049	0.0003	0.0150	0.0012	0.0005
BHMBCCMKT07	0.086	0.057	0.086	0.083	0.035	0.071	0.071	0.0006	0.0090	0.0014	0.0006

BHMBCCMKT09	0.048	0.119	0.080	0.077	0.067	0.081	0.101	0.0012	0.0080	0.0012	0.0005
BHMBCCMKT34	0.039	0.079	0.14	0.153	0.119	0.073	0.123	0.0004	0.0101	0.0009	0.0007
BHMBCCMKT27	0.029	0.108	0.065	0.065	0.089	0.057	0.075	0.0006	0.0061	0.0008	0.0006
NIA Park C26	0.090	0.024	0.143	0.14	0.084	0.055	0.089	0.0012	0.0101	0.0012	0.0005
NIA South	0.033	0.050	0.173	0.071	0.041	0.034	0.100	0.0005	0.0153	0.0012	0.0005
NIA North	0.067	0.090	0.113	0.112	0.101	0.074	0.033	0.0004	0.0170	0.0009	0.0005
BHMBDDMYT29	0.032	0.055	0.048	0.049	0.028	0.054	0.079	0.0012	0.0170	0.0010	0.0006
BHTBFFDTYK37	0.092	0.089	0.102	0.101	0.073	0.067	0.053	0.0005	0.0080	0.0005	0.0004
BHTBFFDTYK40	0.057	0.075	0.064	0.064	0.031	0.078	0.091	0.0004	0.0082	0.0008	0.0005
BHTBFFDTYK45	0.083	0.119	0.119	0.121	0.05	0.095	0.049	0.0003	0.0045	0.0012	0.0006
BHTBFFDTYK48	0.016	0.084	0.081	0.92	0.089	0.061	0.033	0.0005	0.0076	0.0009	0.0005
BHTBFFDTYK42	0.035	0.054	0.065	0.066	0.032	0.032	0.037	0.0004	0.0003	0.0058	0.0004
Mean value	0.068	0.078	0.102	0.101	0.073	0.067	0.079	0.00056	0.00976	0.00111	**0.00055**
Max value	0.122	0.147	0.177	0.179	0.139	0.129	0.137	0.00143	**0.01812**	0.00162	0.00152
Min value	0.015	0.024	0.148	0.049	0.025	0.023	0.033	0.00029	0.00208	0.00055	**0.00028**
standard deviation	0.030	0.035	0.035	0.035	0.033	0.026	0.028	**0.00023**	0.00428	0.00025	0.00024

V. Conclusion

Among the important features of smart cities that directly help residents on day to day, basis is urban transportation. One of the common problems in urban areas is congestion which is intensified by the search for free parking spaces by at least 35%. The capability to predict the available space for parking the vehicle in the urban cities is a big challenge and the smart solution will significantly lower traffic jams and also reduces metropolitan pollution. Some writers presented methodologies and models for the prediction of available space for parking the vehicle have several limitations. The proposed model helps the users to predict the parking space in advance using smart devices. As we know lots of data is generated every second it filters the data according to the need of the user and provides valuable information in less time using limited memory space. IoT enabled smart parking predictive model should allow us to take advantage of all of the interconnected devices in smart parking lots for the collection of data, evaluate it, and communicate the results with the users. Data mining technique that is Ensemble predictive analytic, enhanced the prediction of free space available in smart parking areas.

References

[1] C. Stracener, Q. Samelson, J. MacKie, M. Ihaza, P. A. Laplante, and B. Amaba. 2019. The internet of things grows artificial intelligence and data sciences. IT Profess. 21, 3 (2019), 55–62. https://doi.org/10.1109/MITP.2019.2912729.

[2] Z. Allam and Z. A. Dhunny. 2019. On big data, artificial intelligence and smart cities. Cities 89 (2019), 80–91.

[3] M. Mohammadi and A. Al-Fuqaha. 2018. Enabling cognitive smart cities using big data and machine learning: Approaches and challenges. IEEE Commun. Mag. 56, 2 (2018), 94–101.

[4] Trista Lin, Hervé Rivano, and Frédéric Le Mouël. 2017. A survey of smart parking solutions. IEEE Trans. Intell. Transport. Syst. 18, 12 (2017), 3229–3253.

[5] Philippe Esling and Carlos Agon. 2012. Time-series data mining. ACM Comput. Surv. 45, 1 (2012), 12.

[6] Liangbo Qi, Hui Yu, and Peiyan Chen. 2014. Selective ensemble-mean technique for tropical cyclone track forecast by using ensemble prediction systems. Quart. J. Roy. Meteorol. Soc. 140, 680 (2014), 805–813.

[7] Huai-zhi Wang, Gang-qiang Li, Gui-bin Wang, Jian-chun Peng, Hui Jiang, and Yi-tao Liu. 2017. Deep learning based ensemble approach for probabilistic wind power forecasting. Appl. Energy 188 (2017), 56–70.

[8] Yoo, Seong-eun, Chong, Poh Kit, Kim, Taehong, et al., 2008. PGS: Parking GuidanceSystem based on wireless sensor network. In: 2008 3rd International Symposium on Wireless Pervasive Computing. IEEE, pp. 218–222.

[9] Kumar, Rakesh, Chilamkurti, Naveen K., Soh, Ben, 2007. A comparative study of different sensors for smart car park management. In: 2007 International Conference on Intelligent Pervasive Computing (IPC 2007). IEEE.

[10] Almeida, De, Paulo, R.L., et al., 2015. PKLot-A robust dataset for parking lot classification. Expert Syst. Appl. 42 (11), 4937–4949.

[11] Amato, Giuseppe et al., 2016. Car parking occupancy detection using smart camera networks and deep learning. In: 2016 IEEE Symposium on Computers and Communication (ISCC). IEEE.

[12] Amato, Giuseppe et al., 2017. Deep learning for decentralized parking lot occupancy detection. Expert Syst. Appl. 72, 327–334.

[13] Gandhi, B.K., Rao, M.K., 2016. A prototype for IoT based car parking management system for smart cities. Indian J. Sci. Technol. 9 (17), 1–6.

[14] Pawowicz, B., Salach, M., Trybus, B., 2019. Infrastructure of RFID-based smart city traffic control system. In: Conference on Automation. Springer, Cham, pp. 186–198.

[15] Giuffr, Tullio, Siniscalchi, Sabato Marco, Tesoriere, Giovanni, 2012. A novel architecture of parking management for smart cities. Procedia-Soc. Behav. Sci.53, 16–28.

[16] Xiang, Xuezhi et al., 2017. Real-time parking occupancy detection for gas stations based on Haar-AdaBoosting and CNN. IEEE Sens. J. 17 (19), 6360–6367.

[17] Shoeibi, N., Shoeibi, N., 2019. Future of smart parking: automated valet parking using deep Q-Learning. In: International Symposium on Distributed Computing and Artificial Intelligence. Springer, Cham, pp. 177–182.

[18] Camero, A., Toutouh, J., Stolfi, D.H., Alba, E., 2018. Evolutionary deep learning for car park occupancy prediction in smart cities. In: International Conference on Learning and Intelligent Optimization. Springer, Cham, pp. 386–401.

[19] Mago, N., Kumar, S., 2018. A machine learning technique for detecting outdoor parking. Int. J. Eng. Technol. 7 (2.30), 39–43.

[20] Stolfi, D.H., Alba, E., Yao, X., 2017. Predicting car park occupancy rates in smart cities. In: International Conference on Smart Cities. Springer, Cham, pp. 107–117.

[21] Bachani, Mamta, Qureshi, Umair Mujtaba, Shaikh, Faisal Karim, 2016. Performance analysis of proximity and light sensors for smart parking. Proc. Comput. Sci. 83, 385–392.

[22] Faris, H., Abukhurma, R., Almanaseer, W., Saadeh, M., Mora, A.M., Castillo, P.A., Aljarah, I., 2019. Improving financial bankruptcy prediction in a highly imbalanced class distribution using oversampling and ensemble learning: a case from the Spanish market. Progr. Artif. Intell., 1–23.

[23] Zhang, Y., Haghani, A., 2015. A gradient boosting method to improve travel time prediction. Transp. Res. Part C: Emerging Technol. 58, 308–324.

[24] Anchal , P. Mittal, 2021. Evaluation of Data Mining Techniques and Its Fusion with IoT Enabled Smart Technologies for Effective Prediction of Available Parking Space. Vol 12, Number 4, 187-197.

[25] Sinta, D., Wijayanto, H., Sartono, B., 2014. Ensemble k-nearest neighbor's method to predict rice price in Indonesia. Appl. Math. Sci. 8 (160), 7993–8005.

[26] Freund, Y., Schapire, R.E., 1996. Experiments with a new boosting algorithm. In icml vol. 96, pp. 148–156.

[27] Freund, Y., Schapire, R., Abe, N., 1999. A short introduction to boosting. J.-Jpn. Soc. Artif. Intell. 14 (771–780), 1612.

[28] Mishra, S., Mishra, D., Santra, G.H., 2017. Adaptive boosting of weak regressors for forecasting of crop production considering climatic variability: an empirical assessment. J. King Saud Univ.-Comput. Inf. Sci.

[29] Géron, A., 2017. Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media Inc..

[30] Camero, A., Toutouh, J., Stolfi, D.H., Alba, E., 2018. Evolutionary deep learning for car park occupancy prediction in smart cities. In: International Conference on Learning and Intelligent Optimization. Springer, Cham, pp. 386–401.

[31] Xu, H., Ying, J., 2017. Bus arrival time prediction with real-time and historic data. Cluster Comput. 20 (4), 3099–3106.

[32] Zheng, Yanxu, Rajasegarar, Sutharshan, Leckie, Christopher, 2015. Parking availability prediction for sensor-enabled car parks in smart cities. In: 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP). IEEE.