IEEE *Access*
Multidisciplinary : Rapid Review : Open Access Journal

Prediction of EV Charging Behavior using Machine Learning

Sakib Shahriar[1], A. R. Al-Ali[1], (Life Senior Member, IEEE), Ahmed H. Osman[2], (Senior Member, IEEE), Salam Dhou[1], (Member, IEEE), and Mais Nijim[3], (Member, IEEE)

[1] Computer Science and engineering, Department American University of Sharjah, Sharjah, UAE
[2] Electrical Engineering Department, Department American University of Sharjah, Sharjah, UAE
[3] Electrical Engineering and Computer Science Department, Texas A&M University – Kingsville, Kingsville, TX 78363 USA

Corresponding author: A. R. Al-Ali (e-mail: aali@aus.edu).

ABSTRACT As a key pillar of smart transportation in smart city applications, electric vehicles (EVs) are becoming increasingly popular for their contribution in reducing greenhouse gas emissions. One of the key challenges, however, is the strain on power grid infrastructure that comes with large-scale EV deployment. The solution to this lies in utilization of smart scheduling algorithms to manage the growing public charging demand. Using data-driven tools and machine learning algorithms to learn the EV charging behavior can improve scheduling algorithms. Researchers have focused on using historical charging data for predictions of behavior such as departure time and energy needs. However, variables such as weather, traffic, and nearby events, which have been neglected to a large extent, can perhaps add meaningful representations, and provide better predictions. Therefore, in this paper we propose the usage of historical charging data in conjunction with weather, traffic, and events data to predict EV session duration and energy consumption using popular machine learning algorithms including random forest, SVM, XGBoost and deep neural networks. The best predictive performance is achieved by an ensemble learning model, with SMAPE scores of 9.9% and 11.6% for session duration and energy consumptions, respectively, which improves upon the existing works in the literature. In both predictions, we demonstrate a significant improvement compared to previous work on the same dataset and we highlight the importance of traffic and weather information for charging behavior predictions.

INDEX TERMS Electric vehicles (EVs), charging behavior, machine learning, smart city, smart transportation

I. INTRODUCTION

Climate change has become a growing concern in recent years with thirty-three countries jointly declaring a climate emergency as of January 2021 [1]. Global energy consumption is a major contributor to the climate crisis, and in particular, the transportation sector accounts for over a quarter of the global energy consumption [2]. The United Nations (UN) projects that two thirds of the world's population will reside in urban areas by 2050 [3]. This would increase the demand for urban mobility, leading to further energy consumption and emissions of greenhouse gases. Studies have shown that electric vehicles (EVs) have the potential to reduce carbon emissions by 45% compared to conventional internal combustion engine (ICE) vehicles [4]. EVs were initially limited by factors such as reliability and battery range, which have significantly improved in recent years and led to an increase in EV popularity [5]. As a result, the trust in EV

reliability has grown and satisfaction among EV owners are higher [6]. The driver flexibility has also increased with the addition of charging stations in many parts of the world, often lead by various government initiatives encouraging further adoption of EVs. These factors have placed EVs to be in a pole position with regards to providing a clean source of transportation.

There still remains a few challenges, most notably the charging time and public charging needs, despite the promising potential. Although EV charging time has significantly decreased over the years, it is still on average much higher than the refueling time for ICE vehicles. Emerging charging technologies such as extreme fast charging [7] and wireless charging [8] are promising but are still overcoming various challenges and will require years before being adopted. The constraints from charging

infrastructure means that most EV owners rely on public charging stations, which poses a strain on power distribution grid due to the high-power requirements of the EVs [9]. To avoid power grid degradation and failures, un-coordinated charging behavior must be avoided. The optimal solution is to better manage the scheduling of charging stations. The research on smart scheduling using data driven approaches are plentiful and include optimization [10] and metaheuristic [11] approaches. Furthermore, psychological factors influencing charging behavior [12] as well as transactions data and interviews with EV drivers [13] have been used for charging behavior analysis. A comprehensive review of charging behavior analysis using machine learning and data-driven approaches is presented in [14], which concludes that machine learning based approaches are more suitable to scheduling approaches with the ability to provide quantification and more realistic representation.

A. RELATED WORKS

Although predictions of EV charging behavior can have various categories, the focus of this work will be on session duration and energy consumption. Examples of other charging behavior include the prediction of whether the EVs will be charged the next day [15], identification of the use of fast charging [16], prediction of the time to next plug [17], charge profile prediction [18], charging speed prediction [19] and prediction of charging capacity and the daily charging times [20]. These behaviors provide valuable insights, but the prediction of session duration and energy time is more valuable for scheduling purposes.

As will be defined in the following sections, session duration is directly related to the departure time. It is the arrival time, which is a known variable, minus the departure time. Therefore, one can assume the prediction of either the session duration or the departure time to have the same application. Lee *et al*. [21] introduced a novel dataset for non-residential EV charging consisting of over 30000 charging sessions. They used gaussian mixture models (GMM) to predict session duration and energy needs by considering the distribution of the known arrival times. The testing dataset included the month of December 2018 and the reported symmetric mean absolute percentage errors (SMAPEs) were 14.4% and 15.9% for the session duration and energy consumption, respectively. In this work, only historical charging data was considered for obtaining the predictions. In [22], the authors used support vector machines (SVM) for the prediction of arrival and departure time for EV commuters in a university campus. Using historical arrival and departure times and temporal features i.e., week, day, and hour, the reported mean absolute percentage error (MAPE) was 2.9% and 3.7% for arrival and departure times, respectively. For comparison, a simple persistence model was used as reference and SVM hyperparameter tuning was not addressed in the work. Frendo *et al*. [23] predicted the departure time of EVs using

regression models. Historical charging data was utilized, and eight features were used including, car ID, car type, weekday, charging point, car park location, parking floor and arrival time. For prediction, three regression models were trained namely, linear regression, XGBoost and artificial neural network (ANN). XGBoost achieved the best results with mean absolute error (MAE) of 82 minutes. In [24], ensemble machine learning using SVM, random forest (RF) and diffusion-based kernel density estimator (DKDE) was used for session length and energy consumption predictions. For training, historical charging records from two separate datasets were used, with one of them being public and the other being residential charging. The ensemble model performed better than the individual models in both predictions and the reported SMAPEs were 10.4% for duration and 7.5% for the consumption.

TABLE I
SUMMARY OF RELATED WORKS

Source	Prediction	Model	Features	Results
[21]	Session length, energy consumption	GMM	Historical charging data	SMAPE: 14.4% duration, 15.9% consumption
[22]	Arrival time, departure time	SVM	Historical charging data	MAPE: 2.9% arrival, 3.7% departure
[23]	Departure time	XGBoost	Historical charging data, vehicle type, charging location	MAE: 82 minutes
[24]	Session length, energy consumption	Ensemble model of SVM, RF & DKDE	Historical charging data	SMAPE: 10.4% duration, 7.5% consumption
[25]	Start time, session length, energy consumption	Linear regression	Historical charging data	-
[26]	Energy requirements	XGBoost	Historical charging data, season, weekday, location type, charging fees	R^2: 0.52, MAE: 4.6 kWh
[27]	Energy consumption	k-NN	Last few days energy consumption	SMAPE: 15.3%
[28]	Energy consumption	PSF	Last few days energy consumption	SMAPE: 14.1%

Xiong *et al*. [25] predicted the start time and session duration using mean estimation. Session duration was then used to obtain energy consumption predictions using linear regression. The charging behavior predictions were integrated

to flatten the charging load profile and stabilize the power grid. However, the prediction performances were not evaluated quantitatively. In [26], several regression models were used to predict the energy requirements from public charging stations data for the US state of Nebraska. Besides historical charging data, parameters such as season, weekday, location type and charging fees were used as input features. On the test set, XGBoost model outperformed linear regression, RF and SVM obtaining a R^2 score of 0.52 and MAE of 4.6 kWh. The authors in [27] used k-nearest neighbor (k-NN) to predict the energy consumption at a charging outlet using data from a university campus. The problem was formulated as time-series forecast whereby energy consumption prediction for the next day (next 24 hours) was made using energy consumption of previous days. The highest SMAPE was 15.3% using k value of 1 (1-NN) and a time-weighted dot product dissimilarity measure. Similarly, Majidpour *et al*. [28] also predicted the next day energy needs of a charging station based on previous days energy consumption using various algorithms including SVM and RF. They also experimented with pattern sequence-based forecasting (PSF) [29], where clustering is first applied to classify the days and predictions are made for that day. The PSF-based approach provided the most accurate results with average SMAPE value of 14.1%. Table 1 provides a summary of the related works in the literature.

B. OBJECTIVES

Although the above works from the literature have successfully applied machine learning for the prediction of session duration and energy consumption, they have mainly focused on utilizing historical charging data. In some cases, additional derived features such as vehicle information, charging location information and seasonal information were used. This has motivated us in this work to investigate the use of additional input features including weather, traffic and local events and observe its impact on the accuracy of charging behavior predictions. The key contributions of this work are the following:

1) We propose a novel approach in EV charging behavior prediction that utilizes weather, traffic, and local events data along with historical charging records.
2) We use several machine learning algorithms including RF, SVM, XGBoost and ANN for predictions of session duration and energy consumption on the adaptive charging network (ACN) dataset.
3) We empirically show that the use of additional data has a positive impact on the accuracy of predictions and significantly improves upon the previous work on the same dataset that used only historical charging information.

The rest of the paper is organized as follows. Background information including key concepts in machine learning is provided in Section II. This is followed by a detailed explanation of the methodology, including dataset description, and experimental setup in Section III. Section IV presents and discusses the results of this work. Future research directions are provided in Section V, and Section VI concludes the paper.

II. BACKGROUND

This section summarizes the background information including the algorithms used in this work and the evaluation metrics for predictions.

A. SUPERVISED MACHINE LEARNING

The main objective in machine learning (ML) is to develop a learning framework that can learn from experience, i.e., the training dataset, without explicit programming. Primarily, ML algorithms are classified as either supervised learning or unsupervised learning. In unsupervised ML, the training data is not labeled, and the goal of the algorithm is to group similar data points. Conversely, in supervised learning, the models are trained from labeled dataset that contains the specified output or target variable, i.e., the variable to be predicted. The representation between the input and target variable is learned iteratively by optimizing a specific objective function. In this work, the target variables, i.e., the session duration and the energy consumption are both labeled, and thus supervised learning will be used. Furthermore, since both target variables are continuous values, we are going to use regression models as opposed to classification models which deals with categorical target values. The four regression models used in this work are RF, SVM, XGBoost and deep ANN. The following paragraphs describe each of them briefly.

A decision tree (DT) can be used to separate complex decisions into a combination of simpler decisions using split points from the input features. Leaf nodes are the points where no further split is made whereas a decision node is the point where decisions take place. Predictions are made by taking the average value of all the items in the leaf node in regression. Although simple to implement, a single DT is prone to overfitting. To overcome this problem, multiple DTs can be aggregated, and this is the essence of a random forest (RF) algorithm. Bagging method is used in this case where the trees are created from various bootstrap sample which is sample with replacement. The average value of the predictions across all the trees are taken as the final prediction for regression problems [30].

Similar to a RF, a gradient boosting algorithm [31] makes use of multiple DTs. However, in this algorithm each tree is built sequentially and as a result the errors made by previous trees are taken into consideration which often leads to superior performance. XGBoost [32] is a more recent variation of the gradient boosting algorithm. XGBoost has gained popularity over the last few years for its success in machine learning competitions mainly due to it being effective in dealing with the bias-variance tradeoff [33]. This means that the algorithm is able to avoid overfitting on the training data while at the same time maintaining enough complexity to obtain meaningful representations.

A support vector machine (SVM) [34] is used for both classification and regression problems. It is sometimes referred to as support vector regression [35] when exclusively applied to regression problems. SVM separates the classes with the best hyperplane that can maximize the margin between the respective classes. The key objective is to map the inputs to high dimensional feature spaces where they are linearly separable. This is achieved using kernels such as linear, polynomial, and radial basis function (RBF). SVM is not suitable for larger datasets due to its long training time.

Deep learning-based models contain a large amount of composition of learned functions. Using layered hierarchy of concepts, complex concepts are defined in terms of simpler concepts and more abstract representations are gathered using less abstract ones [36]. Variations of deep learning algorithms include convolutional and recurrent neural networks, which have been successful in image and audio classification tasks. In this work, we consider artificial neural networks (ANN), often referred to as a multilayer perceptron (MLP). MLPs utilize non-linear approximation given a set of input features and can be used for both regression and classification. An MLP consists of input layer which is fed with a given set of input features, the hidden layers which learns the representations and the output layer which makes the final predictions. When the number of hidden layers is two or more, the model is referred to as deep ANN.

In ensemble learning, set of individually trained classifiers are combined and then used to predict new instances, often providing more accurate predictive performance than the individual classifiers [37]. Figure 1 illustrates the concept of ensemble learning. Both RF and XGBoost are examples of ensemble learning, where individual models (in these cases DTs) are first evaluated and then integrated into a single model. The motivation behind such approach is similar to asking multiple experts about an opinion, and then taking their votes to make the final decision [38].
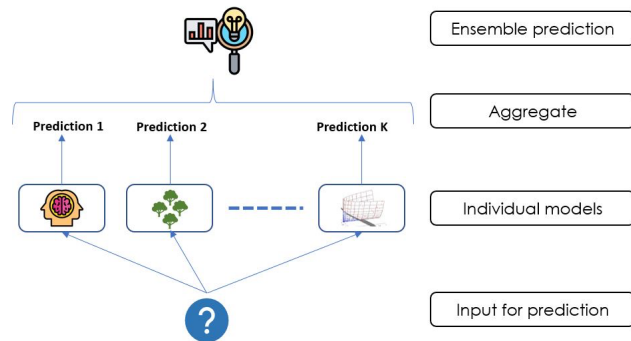

Figure 1. Illustration of ensemble learning

B. EVALUATION OF REGRESSION MODELS
To assess the performance of predictions made by regression models, numerous metrics are used as discussed in [39]. In this work, we will define and use four measures that were commonly used in related works. Equations (1)-(4) defines the metrics that will be used in this work:

Root mean square error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}{n}} \tag{1}$$

Mean absolute error (MAE):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \bar{y}_i| \tag{2}$$

Coefficient of determination or R^2:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}{\sum_{i=1}^{n}(y_i - \mu)^2} \tag{3}$$

Symmetric mean absolute percentage error (SMAPE):

$$SMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \bar{y}_i|}{(|y_i| + |\bar{y}_i|)/2} * 100\% \tag{4}$$

where y represents the actual value, $\bar{y}$ is the predicted value, μ is the average of the actual values and n represents the groups of values in the dataset. Generally, lower scores of RMSE, MAE and SMAPE indicate accurate predictions, and this occurs when the predicted value, $\bar{y}$ is very close to the actual value y. The R^2 value is a measure of goodness of fit for regression and is usually a score between 0 and 1. A score of 1 indicates perfect predictions and generally a higher value represents better performance. We do not consider mean absolute percentage error because it is inconvenient when the actual value y is close to 0, therefore creating a bias. Rather we consider SMAPE which is more suitable for EV charging prediction because both the original and the predicted values are in the denominator [24].

III. METHODOLOGY
In this section, we define the approach used for the prediction of charging behavior. We formulate the problem, describe the dataset, highlight the preprocessing steps, and discuss the methods for training the learning models.

A. EV CHARGING BEHAVIOR
Assuming t_{con} represents the connection time when the car first plugs in, t_{discon} represents the disconnection time when the car plugs out and leaves the station and e represents the energy delivered to the car during the session, we consider the session charging behavior $B_{session}$ as following:

$$B_{session} \triangleq (t_{con}, t_{discon}, e) \tag{5}$$

Based on the above, we can define the length of charging session or the session duration, S_{dur}, as follows:

$$S_{dur} = t_{discon} - t_{con} \qquad (6)$$

In this work, we predict both the session duration and the session energy consumption of an individual charging record and assume that the connection time is known.

B. DATASET DESCRIPTION

Besides the charging dataset, we also make use of weather, traffic, and local events data in order to predict the charging behavior. We will briefly describe the datasets used and highlight their attributes.

Scheduling of EV charging is more significant in public charging structures due to the unpredictable nature of the charging behavior, especially in places like shopping malls. The ACN [21] dataset is among the few publicly available datasets for non-residential EV charging and will be utilized in this work. The dataset contains charging records from two stations in the university campus, namely JPL and Caltech. Unlike the Caltech station, which is open to public, the JPL station is only accessible to employees and therefore will not be considered in this work. Registered users can manually enter additional details, such as their estimated departure time and requested energy, by scanning a QR code through their mobile applications. The dataset can be accessed from [40] by either a web portal or python application programming interface (API).

Although there is a small weather station located at the Caltech campus [41], we did not consider it for this work due to missing values and irregular interval recordings for the wind variable. Additionally, this station did not record variables such as rainfall and snowfall which could potentially impact charging behavior. We therefore used the weather data from NASA's Modern-Era Retrospective analysis for Research and Applications, Version 2 (MERRA-2) [42] which provides data for the precise location of the charging station. The accuracy of satellite weather data in comparison to ground stations has been compared in [43]. Although it has been shown that given a specific location some weather parameters may be more accurately detected using ground stations, for the purpose of this work we do not require a high level of accuracy but rather a more general perception of the impact of weather on charging behaviors. For example, we are interested in observing how the charging behavior is impacted during heavy rainfall as opposed to drier conditions.

Obtaining historical traffic data for specific roads and regions is challenging. Conventional traffic collection methods include intrusive approaches such as road tubes and piezoelectric sensors and non-intrusive approaches including microwave radar and video image detection [44]. With most of these approaches, scalability is an issue, and in most cases, specific roads are not covered. For instance, the city of

Pasadena (where the charging data originates from) provides an open data site [45] for the traffic count around the city. However, for most roads in the city it contains traffic count for some period of time and therefore is not usable in our case where we require regular interval data. Additionally, not all roads and streets are covered. As a result, we decided to use traffic data from google maps, which has also been used in previous machine learning applications [46]. The data is collected by recording the location data from the commuter's mobile devices provided they use the application and have agreed to share their location. The data collected from individuals is anonymized and aggregated to address any privacy concerns [47]. The google maps distance matrix API can be used to retrieve the data. Given a source and destination coordinates, the travel distance and the time taken is returned for a given departure time. We retrieved historical trip time for 9 of the closest roads and streets which one must take to access the charging station.

Since the charging station is located in the Caltech university campus, we decided to include campus events and find out if the number of events have an impact on the charging behavior. The number of events in an hour were obtained from the Caltech university website calendar [48]. For simplification, we decided to round the minutes to the nearest hour, therefore if an event started at 10.20 am, it was counted as an event starting at 10 am.

C. DATA PREPROCESSING

Cleaning and preprocessing the dataset is vital to ensuring the quality of the predictive models. These include removing faulty records and outliers.

The presence of outliers can negatively impact the model performance. A common technique of graphically detecting outliers is boxplots [49]. The boxplots for both target variables contained outliers, as shown in Figure 2. We notice that the outliers for both variables are not consistent, i.e., we have far too many outlier points for energy consumption than the session duration. It is possible that certain vehicles consume far greater amount of energy even if the session duration is not too long.
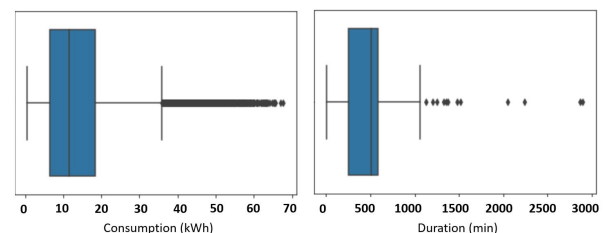
Figure 2. Boxplots of energy consumption (left), session duration (right)

As a result, we opted to perform multivariate outlier detection using the isolation forest algorithm which constructs an ensemble of iTrees for a given data set. The outliers are those instances which have short average path lengths on the

iTrees [50]. By randomly selecting a variable and a split value between the minimum and maximum of the selected variable, the observations are 'isolated'. Partitioning of observations are repeated recursively until all of them have been isolated. After the partitioning, observations that have shorter path lengths for some particular points are likely to be the outliers. Figure 3 illustrates the process in detecting the outlier of the target variables. A total of 697 outliers were detected which accounts for 4% of the total observations.
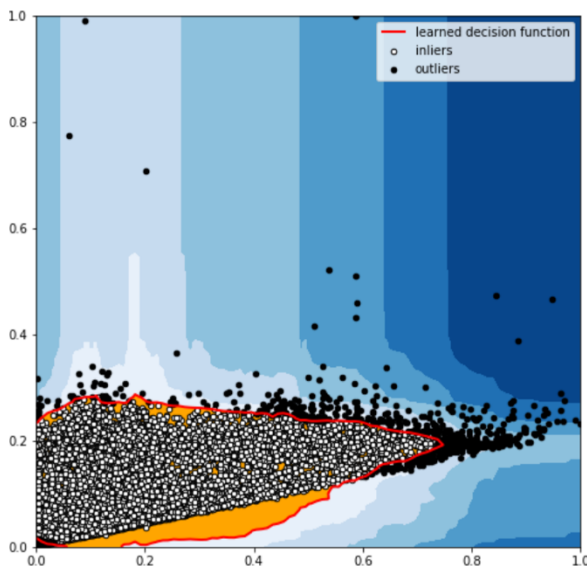

Figure 3. Outlier detection using isolation forest

For the charging data, we only considered charging records that were registered, i.e., contained user IDs, and this accounted for 97% of the records. For the weather data, the time of recording was in universal time and we used the *pytz* [51] library in python to convert the time zone to be the same as that of the charging records. We also converted the temperature units from kelvin to degrees Celsius. Then for each given hour, we also computed average of the previous 7 hours of weather and the average of the next 10 hours. This would allow us to understand how the previous weather and the weather after charging impacts charging decision. For instance, heavy snowfall in the previous hours may account for shorter charging duration and so on. We also had to convert the time zone from coordinated universal time for the traffic data. We then aggregated the traffic for each hour across the nine selected roads and streets. It must be noted that we considered the average trip time as well as the maximum trip time as estimated by google maps. Finally, we aggregated the total events in the campus for each hour.

To merge the various data, the time-series fields were converted to date-time objects using *pandas* [52] library. Then to obtain weather, traffic, and events for a particular charging record, we first obtained the nearest hour that the connection time belongs to. For example, the connection time of 22:11 belongs to 10 pm. This allows us to easily extract the other

information. Instead of simply selecting the traffic level for a given time, we selected the total traffic after arrival until the end of the day. If a vehicle arrived at 2 pm, for instance, we accumulated the traffic from 2 pm until the end of that day. This would allow the model to learn how the traffic level impact the charging behavior. Similarly, we considered the total events after arrival until the end of the day.

D. FEATURE ENGINEERING
Feature engineering refers to the transformation of data into meaningful representation using human knowledge. This process is labor intensive but important nonetheless as this is a weakness of the learning algorithms. Feature engineering relies on human ingenuity and prior knowledge to compensate for the inability of the algorithms to extract and organize the discriminative information from the data [53]. We discuss the future engineering steps next.

Firstly, we convert the time fields that will be used by the models into numeric format by simply dividing the minute by 60 and adding to the hour. Then, for each charging record, we find out their average departure time, session duration and energy consumption. This is done by finding out the user ID of the charging record and aggregating his previous records. We use the arrival time as a numeric feature. However, the arrival time also has other components such as the date information. Using this, we extract the hour of the day, day of the month, month of the year, day of the week, whether the day is a weekend and whether the day falls in a US federal holiday. However, temporal information such as day, hour, and month are cyclic ordinal features. This is because the hour value of 23 corresponding to 11 pm, for example, is actually close to the hour value of 0 which corresponds to 12 am. To represent the proximity of these values, trigonometric transformation is performed as following:

$$f_x = \sin\left(2\pi f / max(f)\right) \qquad (7)$$

$$f_y = \cos\left(2\pi f / max(f)\right) \qquad (8)$$

where f represents the cyclic feature to be transformed, f_x and f_y represents the first and second components of the cyclic feature, respectively. To transform other categorical variables, one-hot encoding was used, where a single variable with n points and k distinct classes is transformed into k binary variables with n points each. For numeric variables, feature scaling is a common transformation where the goal is to normalize the range of the numeric features. There are various scaling techniques, including scaling by domain where all the features are scaled to a specific range such as [0, 1] and scaling to minmax where the features are scaled to the range [0, R], in which case the minimum of the maximum value of feature in all directions is assigned as the radius of the sphere R [54]. However, in this work we have used standardization which

ensures the values of each feature to have zero mean and unit variance. The transformations were performed using the *preprocessing* package of the *Scikit-learn* [55] library. Table 2 lists the features used for training.

TABLE II
LIST OF FEATURES AND THEIR DESCRIPTIONS

Feature	Description
session_length	Length of charging duration, target variable
kWh_delivered	Session energy consumption, target variable
time_con	Numerical representation of the connection time (arrival time)
day_of_week	Day of the week, one-hot encoded
is_weekend	Binary variable indicating whether the session took place in a weekend
holiday	Binary variable indicating whether the session took place on a US federal holiday
hr_x, hr_y	Sine and Cosine components of the hour
day_x, day_y	Sine and Cosine components of the day
mnth_x, mnth_y	Sine component of the month
mean_d_time	Historical average departure time
mean_con	Historical average consumption
mean_dur	Historical average session length
traffic_aft_arvl	average traffic level after arrival
max_traffic_aft_arvl	maximum traffic level after arrival
events_after_arrival	total campus events after arrival
avg_temp_prv	average temperature of last 7 hours
avg_temp_nxt	average temperature of next 10 hours
avg_hum_prv	average humidity of last 7 hours
avg_hum_nxt	average humidity of next 10 hours
avg_win_prv	average wind speed of last 7 hours
avg_win_nxt	average wind speed of next 10 hours
avg_rain_prv	average rainfall of last 7 hours
avg_rain_nxt	average rainfall of next 10 hours
avg_snwfall_prv	average snowfall of last 7 hours
avg_snwfall_nxt	average snowfall of next 10 hours
avg_snwdpth_prv	average snow depth of last 7 hours
avg_snwdpth_nxt	average snow depth of next 10 hours
avg_irradiation_prv	average irradiation of last 7 hours
avg_irradiation_nxt	average irradiation of next 10 hours

E. MODEL SELECTION AND EXPERIMENTAL SETUP

We selected all charging sessions from the ACN dataset that belonged to the 2019 calendar year, which ensures we take the seasonal factors into consideration during training. The dataset was split such that 80% of the records were used for model training and 20% for evaluation. During the training phase, we performed K-fold cross validation where the algorithms are repeatedly trained K times with a fraction 1/K training examples left out for testing [56]. In this case, we selected the common K value of 10. To determine model hyperparameters, we utilized grid search method which determines the optimal set of parameters from a given list by trying out all possible values of the specified parameters [57]. We performed the grid search across K-folds, selected to be 5 in this case to speed up the grid search. We then evaluated all the models using the aforementioned regression metrics. Inspired by the success of ensemble learning methods in previous works, we also decided to experiment with ensemble learning. We used two variants of ensemble stacking, namely voting regressor and stacking regressor, using the *ensemble* package of the *Scikit-learn* library. In a voting regressor, several base regressors are trained on the entire training set, and the average predictions made by the base models are treated as the final prediction. Stacking regressor is based on the concept of stacked generalization where predictions made by the base models are used as inputs to a final estimator, which is trained using cross-validation, to generate predictions [58]. Figure 4 provides a graphical representation of the framework.

IV. RESULTS AND DISCUSSION

We begin the experiment with RF algorithm which can be used to visualize the variable importance [30]. This is a method for feature selection where certain variables that are not important and can often hinder performance are removed. In this case, the inclusion of the least important variables had a very insignificant performance increase and hence we decided to include them in model training. Additionally, variables can be ranked in terms of their relative importance. This is determined by each feature's contribution in determining the most effective splits. In Figures 5 and 6, we plot the top 10 important variables for session duration and energy consumption, respectively. The two most important predictors of session duration are the maximum traffic after arrival and the time of connection. This indicates the

Figure 4. Graphical representation of the proposed framework

usefulness of including traffic information for the prediction of session duration. However, for energy consumption, the historical average consumption is by far the most significant. This is because a specific vehicle will consume similar energy if the session duration is consistent.
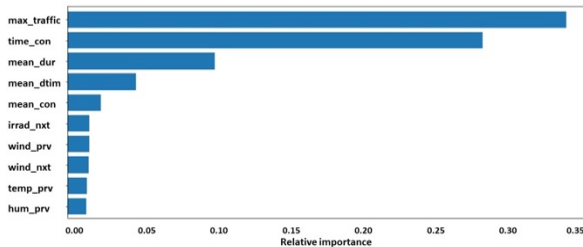

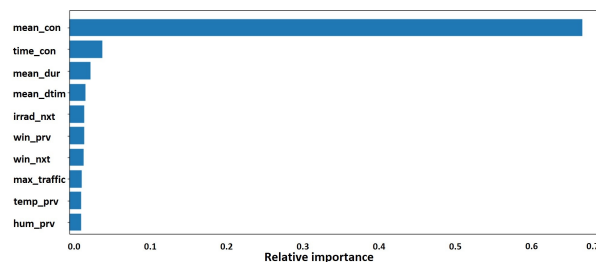**Figure 5.** Top ten features for session duration

Figure 6. Top ten features for energy consumption

A. SESSION DURATION PREDICTIONS

As mentioned earlier, the hyperparameters for the models were determined using the grid search approach. For the deep ANN training, we experimentally determined an architecture with 3 hidden layers of 64, 32 and 16 nodes respectively to be the most suitable. Rectified linear units (Relu) [59] was used as the activation function for all hidden layers and the output layer contained a linear activation as we are expecting the prediction to be a numeric value. The learning rate value was set to 0.001 and we used the Adam [60] algorithm for model optimization. The training batch size was 32 and the number of iterations were 15 epochs. Appendix 1 displays the training loss curve and Table 3 summarizes the 10-fold cross validation scores on the training set.

TABLE III
TRAINING SCORES FOR SESSION DURATION

Metrics/Model	RMSE (mins)	MAE (mins)	R^2	SMAPE (%)
RF	100	68.7	0.72	10.1
SVM	102	68.2	0.71	10.2
XGBoost	100	69.3	0.72	10.3
Deep ANN	103	73.1	0.71	10.7
Voting Ensemble	99.5	67.4	0.72	10.0
Stacking Ensemble	99.5	68.0	0.72	10.1

The training scores are very similar for RF, SVM and XGBoost whereas deep ANN performs slightly worse.

Therefore, we aggregated the 3 best performing models in the training phase into 2 ensemble models, which resulted in improved cross validation scores. Next, we present the results on the test set. For reference, we also selected the user estimates of their departures as prediction. This value was collected through a smart phone app where users were asked to enter their estimates of their departure time and consumption upon arrival. We summarize the results on the test set in Table 4.

TABLE IV
TEST SCORES FOR SESSION DURATION

Metrics/Model	RMSE (mins)	MAE (mins)	R^2	SMAPE (%)
RF	98.7	68.0	0.63	10.1
SVM	101	67.4	0.64	10.1
XGBoost	97.9	68.0	0.63	10.1
Deep ANN	101	73.7	0.57	10.9
Voting Ensemble	97.7	**66.5**	**0.73**	**9.92**
Stacking Ensemble	**97.5**	67.1	**0.73**	9.95
User predictions	430	394	-4.20	69.9

As highlighted, the best results are obtained using the ensemble learning approach, which is consistent with previous works [15], [24]. Voting regressor performs best on 2 metrics and stacking regressor performs the best in terms of RMSE, whereas they both achieve the same R^2 score. The results are consistent with the training performance with RF, SVM and XGBoost resulting in similar performance and deep ANN performs the worst of the four base models. Predictions made by user about their own session length is also far off the actual session length. This indicates that perhaps relying on users to provide an estimate of their own departure time is perhaps not suitable.

B. ENERGY CONSUMPTION PREDICTIONS

Similar approach to the session duration prediction was also used here. The only exception was the deep ANN architecture which in this case contained 2 hidden layers with 64 and 16 nodes, respectively. The training batch size was 64 and the number of epochs was set to 20. Appendix 2 presents the loss curve from the training phase. Table 5 summarizes the 10-fold cross validation scores on the training set.

TABLE V
TRAINING SCORES FOR ENERGY CONSUMPTION

Metrics/Model	RMSE (kWh)	MAE (kWh)	R^2	SMAPE (%)
RF	5.49	3.40	0.69	11.9
SVM	5.65	3.53	0.67	12.6
XGBoost	5.56	3.49	0.68	12.4
Deep ANN	5.61	3.60	0.67	12.9
Voting Ensemble	5.50	3.42	0.69	12.0
Stacking Ensemble	5.48	3.40	0.69	11.9

RF has the best cross validation scores whereas the other 3 models have similar scores. We selected the top 3 models, i.e., RF, SVM and XGBoost to form the 2 ensemble models. In this case, the ensemble models did not improve upon the

best performing RF model but rather achieved similar results on training. The results from the test set are presented in Table 6. We also compare the results with user predictions about their consumptions.

The best results as highlighted were obtained using the stacking ensemble model. The improvement using ensemble learning for energy consumption prediction was perhaps not as significant when we compare with the session duration. The user predictions about their consumptions are not accurate in this case as well.

TABLE VI
TEST SCORES FOR ENERGY CONSUMPTION

Metrics/Model	RMSE (kWh)	MAE (kWh)	R^2	SMAPE (%)
RF	5.50	3.39	0.54	11.7
SVM	5.69	3.54	0.51	12.4
XGBoost	5.61	3.48	0.51	12.1
Deep ANN	5.65	3.55	0.55	12.5
Voting Ensemble	5.54	3.41	0.69	11.8
Stacking Ensemble	**5.50**	**3.38**	**0.70**	**11.6**
User predictions	20.6	11.8	0.04	55.0

C. COMPARISON AND DISCUSSION

When we compare across both predictions, looking at the overall R^2 and the SMAPE, it appears that the prediction of energy consumption is perhaps more difficult. This is consistent with the previous work on the ACN data [21]. However, in another case the opposite was observed [24], i.e., the prediction of energy consumption was easier. Moreover, in both scenarios, it was also noticed that the user predictions about their own behavior is very different to their actual behavior, which further emphasizes the need for predictive analytics. The users' predictions in terms of their energy consumption are slightly more accurate when compared to their predictions of session duration as indicated by better R^2 and SMAPE values. This could be due to the users' lack of interest in entering their estimates every time they decide to charge their vehicles. We also noticed that the performance using deep ANN was the least accurate in both cases. Although deep learning models are proven superior in dealing with images and audio data where feature extraction is not performed, in applications such as this where we perform feature extraction, traditional ML models usually perform better. Furthermore, predictions made by ensemble learning outperformed predictions made by individual ML models in both scenarios, although the impact was more significant for session duration prediction. This is most likely because in the first scenario, the top 3 performing models had similar training performance and combining their predictions resulted in an improvement. However, in the latter scenario, one model clearly outperformed the rest in training and hence the improvement using ensemble learning was not significant.

Looking at the previous works in the literature, the results in this work outperformed all the previous works that reported similar evaluation metrics ([21], [23], [26], [27],

[28]). We summarize the results from the previous works in comparison to the one achieved in this work in Table 6. In comparison to [24], the results obtained in this work for session duration is more accurate although we do not improve upon their results for energy consumption. This is most likely because the authors in [24] utilized both residential and non-residential data for their predictions, and residential charging behavior in most cases are more consistent. However, it must be noted that all previous works except [21] used a different dataset to this work and therefore a comparison is perhaps not suitable. Therefore, keeping the comparison across the same dataset, we can conclude that the utilization of the additional weather, traffic and events data resulted in an improvement in the EV charging behavior predictions.

TABLE VI
PERFORMANCE COMPARISON WITH PREVIOUS WORKS

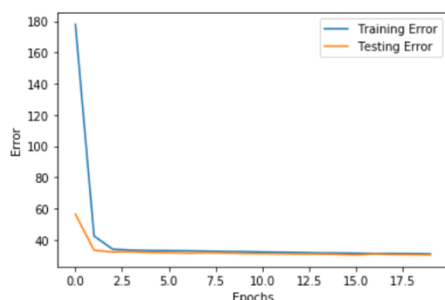| Source	Session Duration	Energy Consumption	Dataset Used
[21]	SMAPE: 14.4%	SMAPE: 15.9%	ACN (historical charging)
[23]	MAE: 82 minutes	Not considered	German charging data (historical charging, vehicle & location info)
[24]	SMAPE: 10.4%	SMAPE: 7.5%	UCLA campus (historical charging) and Residential charging data from UK
[26]	Not considered	R^2: 0.52	Nebraska public charging (historical charging, temporal & location)
[27]	Not considered	SMAPE: 15.3%	UCLA campus (historical energy)
[28]	Not considered	SMAPE: 14.1%	UCLA campus (historical energy)
Our work	**SMAPE: 9.92%, MAE: 66.5 minutes**	**SMAPE: 11.6%, R^2: 0.7**	**ACN, weather, traffic, and events data**

V. RECOMMENDATIONS AND FUTURE WORK

We have quantitatively shown in the previous section that the traffic and weather data were important predictors in EV charging behavior, particularly in the case of session duration. Although the use of local events data (campus events in this case) had insignificant impact in terms of performance gain, it cannot be ruled out for future work. In this work, we obtained all campus events from the university calendar. However, perhaps only the significant events that draw more crowd should be taken into consideration. It is possible that events data may not impact predictions in a university campus setting. However, for other public spaces such as shopping malls for example, events like end of the year sale could be important predictors. Therefore, similar experiments on other public charging spaces should be carried out to find the impact

of local events. Social media can also be explored as a means to obtain information about local events as well as driver behavior. For instance, social media has been shown to be a good tool for estimating human behavior [61] and also is a significant predictor of truck drivers' travel time [62]. It is also likely that the use of vehicle information such as the vehicle model and vehicle type can improve predictions, especially in terms of energy consumption. Some of the previous works have utilized vehicle information [23] but not in conjunction with weather, traffic, and events. Finally, to better understand the charging behavior during the COVID-19 pandemic, a case study should be conducted using the proposed approach to validate the predictive performance in uncertain situations.

VI. CONCLUSION

In this work, we presented a framework for the prediction of two of the most important EV charging behaviors with regards to scheduling, namely EV session duration and energy consumption. Unlike previous work, we utilized weather, traffic, and events data along with the historical charging data. We trained four popular ML models along with two ensemble learning algorithms for the prediction of charging behavior. The results obtained in terms of prediction performance is superior to the results in the previous works. We have also provided a significant improvement of charging behavior prediction on the ACN dataset and demonstrated the potential of utilizing traffic and weather information in charging behavior prediction.

APPENDIX

Appendix 1. Validation loss curve for session duration
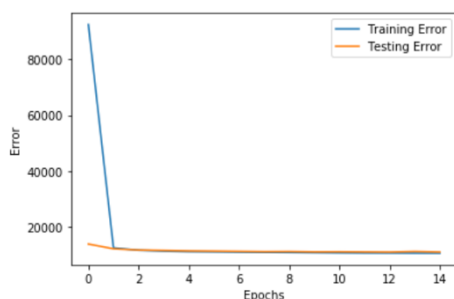
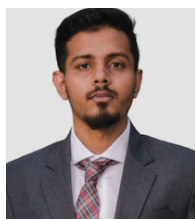**Appendix 2.** Validation loss curve for energy consumption

ACKNOWLEDGMENT

REFERENCES

[1] "Climate Emergency Declaration and Mobilisation In Action," *CEDAMIA*. https://www.cedamia.org/global/ (accessed Jan. 17, 2021).

[2] "Key World Energy Statistics 2018 – Analysis," *IEA*. https://www.iea.org/reports/key-world-energy-statistics-2019 (accessed Jun. 01, 2020).

[3] "68% of the world population projected to live in urban areas by 2050, says UN," *UN DESA | United Nations Department of Economic and Social Affairs*, May 16, 2018. https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html (accessed Jun. 01, 2020).

[4] X. Zhang, F. Gao, X. Gong, Z. Wang, and Y. Liu, "Comparison of Climate Change Impact Between Power System of Electric Vehicles and Internal Combustion Engine Vehicles," in *Advances in Energy and Environmental Materials*, Singapore, 2018, pp. 739–747.

[5] "Global EV Outlook 2019 – Analysis," *IEA*. https://www.iea.org/reports/global-ev-outlook-2019 (accessed Jun. 01, 2020).

[6] Y. Kwon, S. Son, and K. Jang, "User satisfaction with battery electric vehicles in South Korea," *Transp. Res. Part Transp. Environ.*, vol. 82, p. 102306, 2020, doi: https://doi.org/10.1016/j.trd.2020.102306.

[7] D. Ronanki, A. Kelkar, and S. S. Williamson, "Extreme Fast Charging Technology—Prospects to Enhance Sustainable Electric Transportation," *Energies*, vol. 12, no. 19, 2019, doi: 10.3390/en12193721.

[8] S. A. Q. Mohammed and J.-W. Jung, "A Comprehensive State-of-the-Art Review of Wired/Wireless Charging Technologies for Battery Electric Vehicles: Classification/Common Topologies/Future Research Issues," *IEEE Access*, vol. 9, pp. 19572–19585, 2021, doi: 10.1109/ACCESS.2021.3055027.

[9] A. Ramanujam, P. Sankaranarayanan, A. Vasan, R. Jayaprakash, V. Sarangan, and A. Sivasubramaniam, "Qantifying The Impact of Electric Vehicles On The Electric Grid: A Simulation Based Case-Study," in *Proceedings of the Eighth International Conference on Future Energy Systems*, New York, NY, USA, 2017, pp. 228–233. doi: 10.1145/3077839.3077854.

[10] M. Ş. Kuran, A. Carneiro Viana, L. Iannone, D. Kofman, G. Mermoud, and J. P. Vasseur, "A Smart Parking Lot Management System for Scheduling the Recharging of Electric Vehicles," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2942–2953, 2015, doi: 10.1109/TSG.2015.2403287.

[11] J. García-Álvarez, M. A. González, and C. R. Vela, "Metaheuristics for solving a real-world electric vehicle charging scheduling problem," *Appl. Soft Comput.*, vol. 65, pp. 292–306, 2018, doi: https://doi.org/10.1016/j.asoc.2018.01.010.

[12] L. Hu, J. Dong, and Z. Lin, "Modeling charging behavior of battery electric vehicle drivers: A cumulative prospect theory based approach," *Transp. Res. Part C Emerg. Technol.*, vol. 102, pp. 474–489, 2019, doi: https://doi.org/10.1016/j.trc.2019.03.027.

[13] J. Spoelstra, "Charging behaviour of Dutch EV drivers," Master's Thesis, 2014.

[14] S. Shahriar, A. R. Al-Ali, A. H. Osman, S. Dhou, and M. Nijim, "Machine Learning Approaches for EV Charging Behavior: A Review," *IEEE Access*, vol. 8, pp. 168980–168993, 2020, doi: 10.1109/ACCESS.2020.3023388.

[15] S. Ai, A. Chakravorty, and C. Rong, "Household EV Charging Demand Prediction Using Machine and Ensemble Learning," in *2018 IEEE International Conference on Energy Internet (ICEI)*, 2018, pp. 163–168.

[16] Y. Yang, Z. Tan, and Y. Ren, "Research on Factors That Influence the Fast Charging Behavior of Private Battery Electric Vehicles," *Sustainability*, vol. 12, no. 8, 2020, doi: 10.3390/su12083439.

[17] S. Venticinque and S. Nacchia, "Learning and Prediction of E-Car Charging Requirements for Flexible Loads Shifting," in *Internet and Distributed Computing Systems*, Cham, 2019, pp. 284–293.

[18] O. Frendo, J. Graf, N. Gaertner, and H. Stuckenschmidt, "Data-driven smart charging for heterogeneous electric vehicle fleets," *Energy AI*, vol. 1, p. 100007, 2020, doi: 10.1016/j.egyai.2020.100007.

[19]   J. J. Mies, J. R. Helmus, and R. Van den Hoed, "Estimating the Charging Profile of Individual Charge Sessions of Electric Vehicles in The Netherlands," *World Electr. Veh. J.*, vol. 9, no. 2, 2018, doi: 10.3390/wevj9020017.

[20]   Y. Lu *et al.*, "The Application of Improved Random Forest Algorithm on the Prediction of Electric Vehicle Charging Load," *Energies*, vol. 11, no. 11, 2018, doi: 10.3390/en11113207.

[21]   Z. J. Lee, T. Li, and S. H. Low, "ACN-Data: Analysis and Applications of an Open EV Charging Dataset," in *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, New York, NY, USA, 2019, pp. 139–149. doi: 10.1145/3307772.3328313.

[22]   Z. Xu, "Forecasting Electric Vehicle Arrival & Departure Time On UCSD Campus using Support Vector Machines," PhD Thesis, UC San Diego, 2017.

[23]   O. Frendo, N. Gaertner, and H. Stuckenschmidt, "Improving Smart Charging Prioritization by Predicting Electric Vehicle Departure Time," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–8, 2020.

[24]   Y.-W. Chung, B. Khaki, T. Li, C. Chu, and R. Gadh, "Ensemble machine learning-based algorithm for electric vehicle user behavior prediction," *Appl. Energy*, vol. 254, p. 113732, 2019, doi: https://doi.org/10.1016/j.apenergy.2019.113732.

[25]   Y. Xiong, C. Chu, R. Gadh, and B. Wang, "Distributed optimal vehicle grid integration strategy with user behavior prediction," in *2017 IEEE Power Energy Society General Meeting*, 2017, pp. 1–5.

[26]   A. Almaghrebi, F. Aljuheshi, M. Rafaie, K. James, and M. Alahmad, "Data-Driven Charging Demand Prediction at Public Charging Stations Using Supervised Machine Learning Regression Methods," *Energies*, vol. 13, no. 16, p. 4231, 2020.

[27]   M. Majidpour, C. Qiu, P. Chu, R. Gadh, and H. R. Pota, "Fast Prediction for Sparse Time Series: Demand Forecast of EV Charging Stations for Cell Phone Applications," *IEEE Trans. Ind. Inform.*, vol. 11, no. 1, pp. 242–250, 2015.

[28]   M. Majidpour, C. Qiu, P. Chu, R. Gadh, and H. R. Pota, "A novel forecasting algorithm for electric vehicle charging stations," in *2014 International Conference on Connected Vehicles and Expo (ICCVE)*, 2014, pp. 1035–1040.

[29]   N. Bokde, M. W. Beck, F. [Martínez Álvarez, and K. Kulat, "A novel imputation methodology for time series based on pattern sequence forecasting," *Pattern Recognit. Lett.*, vol. 116, pp. 88–96, 2018, doi: https://doi.org/10.1016/j.patrec.2018.09.020.

[30]   A. Liaw, M. Wiener, and others, "Classification and regression by randomForest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[31]   J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, 2001.

[32]   T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[33]   D. Nielsen, "Tree boosting with xgboost-why does xgboost win" every" machine learning competition?," Master's Thesis, NTNU, 2016.

[34]   C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Min. Knowl. Discov.*, vol. 2, no. 2, pp. 121–167, Jun. 1998, doi: 10.1023/A:1009715923555.

[35]   M. Awad and R. Khanna, "Support Vector Regression," in *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*, M. Awad and R. Khanna, Eds. Berkeley, CA: Apress, 2015, pp. 67–80. doi: 10.1007/978-1-4302-5990-9_4.

[36]   I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.

[37]   D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," *J. Artif. Intell. Res.*, vol. 11, pp. 169–198, 1999.

[38]   R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits Syst. Mag.*, vol. 6, no. 3, pp. 21–45, 2006.

[39]   A. Botchkarev, "A new typology design of performance metrics to measure errors in machine learning regression algorithms," *Interdiscip. J. Inf. Knowl. Manag.*, vol. 14, pp. 45–76, Jan. 2019.

[40]   "ACN-Data -- A Public EV Charging Dataset." https://ev.caltech.edu/dataset (accessed Jul. 02, 2020).

[41]   "TCCON Weather." http://tccon-weather.caltech.edu/ (accessed Jul. 02, 2020).

[42]   R. Gelaro *et al.*, "The Modern-Era Retrospective Analysis for Research and Applications, Version 2 (MERRA-2)," *J. Clim.*, vol. 30, no. 14, pp. 5419–5454, Jul. 2017, doi: 10.1175/JCLI-D-16-0758.1.

[43]   R. Mendelsohn, P. Kurukulasuriya, A. Basist, F. Kogan, and C. Williams, "Climate analysis with satellite versus weather station data," *Clim. Change*, vol. 81, no. 1, pp. 71–83, Mar. 2007, doi: 10.1007/s10584-006-9139-x.

[44]   G. Leduc and others, "Road traffic data: Collection methods and applications," *Work. Pap. Energy Transp. Clim. Change*, vol. 1, no. 55, pp. 1–55, 2008.

[45]   "Pasadena Traffic Count Website." https://data.cityofpasadena.net/datasets/eaaffc1269994f0e8966e2024647cc56 (accessed Jan. 21, 2021).

[46]   F. Goudarzi, "Travel Time Prediction: Comparison of Machine Learning Algorithms in a Case Study," in *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, 2018, pp. 1404–1407. doi: 10.1109/HPCC/SmartCity/DSS.2018.00232.

[47]   D. Barth, "The bright side of sitting in traffic: Crowdsourcing road congestion data," *Google Off. Blog*, 2009.

[48]   "Calendar | www.caltech.edu." https://www.caltech.edu/campus-life-events/master-calendar (accessed Jan. 23, 2021).

[49]   J. W. Tukey, *Exploratory data analysis*, vol. 2. Reading, Mass., 1977.

[50]   F. T. Liu, K. M. Ting, and Z. Zhou, "Isolation Forest," in *2008 Eighth IEEE International Conference on Data Mining*, 2008, pp. 413–422. doi: 10.1109/ICDM.2008.17.

[51]   S. Bishop, *Pytz–world timezone definitions for Python*. Online][Cited: May 25, 2012.] http://pytz. sourceforge. net, 2016.

[52]   W. McKinney and others, "pandas: a foundational Python library for data analysis and statistics," *Python High Perform. Sci. Comput.*, vol. 14, no. 9, pp. 1–9, 2011.

[53]   Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, 2013, doi: 10.1109/TPAMI.2013.50.

[54]   P. Juszczak, D. Tax, and R. P. Duin, "Feature scaling in support vector data description," in *Proc. asci*, 2002, pp. 95–102.

[55]   F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[56]   Y. Bengio and Y. Grandvalet, "No unbiased estimator of the variance of k-fold cross-validation," *J. Mach. Learn. Res.*, vol. 5, no. Sep, pp. 1089–1105, 2004.

[57]   Qiujun Huang, Jingli Mao, and Yong Liu, "An improved grid search algorithm of SVR parameters optimization," in *2012 IEEE 14th International Conference on Communication Technology*, 2012, pp. 1022–1026. doi: 10.1109/ICCT.2012.6511415.

[58]   D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, 1992, doi: https://doi.org/10.1016/S0893-6080(05)80023-1.

[59]   V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," 2010.

[60]   D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *ArXiv Prepr. ArXiv14126980*, 2014.

[61]   M.-A. Abbasi, S.-K. Chai, H. Liu, and K. Sagoo, "Real-world behavior analysis through a social media lens," in *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*, 2012, pp. 18–26.

[62]   D. Yuniar, L. Djakfar, A. Wicaksono, and A. Efendi, "Truck driver behavior and travel time effectiveness using smart GPS," *Civ. Eng. J.*, vol. 6, no. 4, pp. 724–732, 2020.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3103119, IEEE Access

IEEE Access

Author Name: Preparation of Papers for IEEE Access (February 2017)

Sakib Shahriar received his B.S. (2018) in Computer Engineering from American University of Sharjah, U.A.E. He is currently pursuing his M.S. degree in Computer Engineering at the American University of Sharjah, U.A.E. He is also a research assistant at the same university. His research interests include Machine Learning, Big Data Analytics and Deep Learning.

A. R. Al-Ali received the Ph.D. in electrical engineering and a minor in computer science from Vanderbilt University, Nashville, TN, USA, 1990; MS from Polytechnic Institute of New York, USA, 1986 and B. Sc.EE from Aleppo University, Syria, 1979. From 1991-2000, he was a faculty with the EE Dept. at KFUPM, Saudi Arabia. Since 2000–present, he is working as professor of computer engineering with American University of Sharjah, UAE. His research, teaching interests include embedded systems, cyber physical systems, IoT and IIoT applications in smart cities.

Ahmed H. Osman received the B.Sc. and M.Sc. degrees in electrical engineering from Helwan University, Cairo, Egypt, in 1991 and 1996, respectively, and the Ph.D. degree in electrical engineering from the University of Calgary, Calgary, AB, Canada, in 2003. From 2004 to 2008, he was an Assistant Professor with the Department of Electrical and Computer Engineering, University of Calgary. He is currently a Professor at the Department of Electrical Engineering, American University of Sharjah, Sharjah, UAE. His research interests and activities include power system analysis and power system protection.

Salam Dhou (Member, IEEE) received the B.Sc. and M.Sc. degrees in computer science from the Jordan University of Science and Technology, Irbid, Jordan, in 2004 and 2007, respectively, and the Ph.D. degree in electrical and computer engineering from Virginia Commonwealth University, Richmond, VA, USA in 2013. She worked as a Postdoctoral Research Fellow in the Department of Radiation Oncology at Harvard Medical School, Boston, Massachusetts, USA, between 2013 and 2016. She is currently an Assistant Professor in the Department of Computer Science and Engineering, and the Biomedical Engineering Graduate Program with the American University of Sharjah, Sharjah, United Arab Emirates. Her research interests include machine learning and data mining, computer vision, and medical imaging and informatics.

Mais Nijim received her B.S in Computer Science from Princess Sumaya University in Amman, Jordan. She received her M.S. degree in Computer Science from New Mexico State University in 2004, and her Ph.D. in Computer Science from New Mexico Institute of Mining and Technology in 2007. She has been an Associate Professor with the Department of Electrical Engineering and Computer Science, Texas A&M University-Kingsville. Her research interests include Machine Learning, Cyber Physical System, Cybersecurity, and Wireless Sensor Network. She is the editor of the International Journal of Sensor Network.