



## Article

# EV Charging Behavior Analysis Using Hybrid Intelligence for 5G Smart Grid

Yi Shen <sup>1,\*</sup> , Wei Fang <sup>2</sup>, Feng Ye <sup>3</sup>  and Michel Kadoch <sup>4</sup><sup>1</sup> Rutgers Business School, Rutgers University, Newark, NJ 07102, USA<sup>2</sup> Rutgers Law School, Rutgers University, Newark, NJ 07012, USA; wfang@rutgers.edu<sup>3</sup> School of Engineering, University of Dayton, Dayton, OH 45469, USA; fye001@udayton.edu<sup>4</sup> Department of Electrical Engineering, École de Technologie Supérieure, Montreal, QC H3C 1K3, Canada; michel.kadoch@etsmtl.ca

\* Correspondence: yi.shen@rutgers.edu

Received: 21 November 2019; Accepted: 20 December 2019; Published: 1 January 2020



**Abstract:** With the development of the Internet of Things (IoT) and the widespread use of electric vehicles (EV), vehicle-to-grid (V2G) has sparked considerable discussion as an energy-management technology. Due to the inherently high maneuverability of EVs, V2G systems must provide on-demand service for EVs. Therefore, in this work, we propose a hybrid computing architecture based on fog and cloud with applications in 5G-based V2G networks. This architecture allows the bi-directional flow of power and information between schedulable EVs and smart grids (SGs) to improve the quality of service and cost-effectiveness of energy service providers. However, it is very important to select an EV suitable for scheduling. In order to improve the efficiency of scheduling, we first need to determine define categories of target EV users. We found that grouping on the basis of EV charging behavior is one effective method to identify target EVs. Therefore, we propose a hybrid artificial intelligence classification method based on the charging behavior profile of EVs. Through this classification method, target EVs can be accurately identified. The results of cross-validation experiments and performance evaluations suggest that this method is effective.

**Keywords:** smart grid; 5G; V2G; cloud computing; fog computing; machine learning; hybrid intelligence

## 1. Introduction

Modern power grids need to be smarter to provide a sustainable and affordable power supply in the context of ever-increasing consumer demand. The smart grid (SG) is a modern version of the traditional power network that is emerging as monitoring technology advances and power use continues to grow [1]. This network functionally includes control centers, transmission networks, and substations.

The SG is a cost-effective distributed power line condition monitoring system that allows for the two-way flow of power and information between the customers and providers of an energy service, which improves the efficiency and stability of power delivery [2]. By using this two-way communication, the advanced metering infrastructure (AMI) makes the energy service not only reliable and secure, but also cost-effective throughout the system [3]. Technically, the smart grid is a modern solution that adopts a unique digital platform to provide innovative sensing [4], signal processing, communication, and computing capabilities [5]. Together with advanced metering and the automation of distribution, the creation of the technical foundation for a smart grid that combines power communication and computer control can lead to a significant improvement in reliability and grid capacity. Furthermore, smart grids track every data point in the system of the energy distribution

cycle, which makes it possible to monitor and control the energy service in real time, including pricing, production, and consumption. Consequently, the energy service providers are able to achieve efficient load balancing and a reliable transfer of energy. As a result, smart grids provide better autonomous control measures and improved efficiency through innovative product type solutions.

Along with the development of smart grids, the proportion of electric vehicles (EVs) is expected to increase due to climate concerns, rising crude oil prices, technological advances, and advances in the automatic control of EVs [6]. International Energy has a plan to have more than 20 million EVs by 2020. As EVs consistently need to be recharged, they require a plan for effective and intentional use of energy otherwise, a large number of EVs can cause an overload [7]. EVs and their peripheral products constitute a large disperse network of data-collecting elements, promoting the development of the Internet of Things (IoT) [8]. EVs can also promote urban development through the integration of renewable energy and comparatively low emissions. There are two modes of EV in the power grid: Consumers and producers. In this context, consumer means grid-to-vehicle (G2V) and the producer means vehicle-to-grid (V2G) [9]. In the consumer mode (G2V), the electric vehicle can be charged.

In recent years, various electric vehicle charging scheduling schemes and pricing strategies have been proposed to reduce charging operation costs and improve grid reliability. A centralized hybrid game strategy for charging and discharging has been proposed in a V2G network [10], so as to eliminate the need for a centralized control system, developing a distributed charging strategy for plug-in hybrid vehicles. In addition to charging scheduling control [11], more and more research has been conducted on designing appropriate demand response mechanisms to improve overall system efficiency [12]. Dynamic pricing decision mechanisms based on charging stations and electric vehicles have been considered [13]. However, most existing studies assume a certain understanding of the arrival of future electric vehicles and electricity prices [14]. Some researchers have proposed that the Markov Decision Process (MDP) is based on the premise that the distribution of future events is known [15]. However, the assumed distribution and actual time distribution often do not match, resulting in inefficient pricing and scheduling strategies in actual scenarios. In recent years, machine learning has become one of the most active technologies in many research fields. In natural language processing, image recognition has made remarkable achievements. Therefore, a learning-based approach driven by real data is ideal for solving this problem. For example, some papers have proposed a real-time EV charging strategy based on reinforcement learning [16]. When EVs participate in demand response plans with real-time price signals, making full use of these price signals can greatly reduce charging costs. Another paper proposed a ML-based demand response strategy to control electric vehicle charging in response to real-time pricing, thereby minimizing the overall energy cost of electric vehicles. However, these algorithms all arrange users' charging strategies based on electricity prices to minimize user charging costs. In order to maximize the profit of the electricity supplier, it is necessary to predict the actual charging behavior of the user so as to formulate an optimal electricity price to maximize profit.

Based on the above reasons, we propose a 5G smart grid electric vehicle charging behavior prediction method based on hybrid intelligence. This method consists of two parts, one is a smart grid network architecture based on cloud computing and fog computing, and the other is a user behavior analysis method based on human in the loop (HITL)-based K-Means clustering and k-nearest neighbors (k-NN) algorithm running on this structure. The smart grid network architecture based on cloud computing and fog computing is the hardware foundation for implementing intelligent algorithms and consists of three layers. The user layer is used to collect user information, the service layer is responsible for data analysis, request response, and provides cloud computing, fog computing, data driven charge and discharge scheduling determined by artificial intelligence technology, and finally multiple communication technologies are integrated at the network layer to ensure the interaction between the user layer and service layer. This structure can give full play to the big data computing capability of cloud computing and the real-time response capability of fog computing, ensuring the efficient real-time operation of the proposed algorithm. The HITL-based K-Means clustering and k-NN algorithm is the core of software predicting the charging behavior of EV. It marks the

data set through an automatic process, making it unnecessary for new users to join the charging network clustering. This method can dispatch electric vehicles in the smart grid more efficiently. Cross-validation experiments and performance evaluation measures show that this technology can classify the charging behavior of users and has good prediction ability. The classification results can be used as an important basis for scheduling charge and discharge.

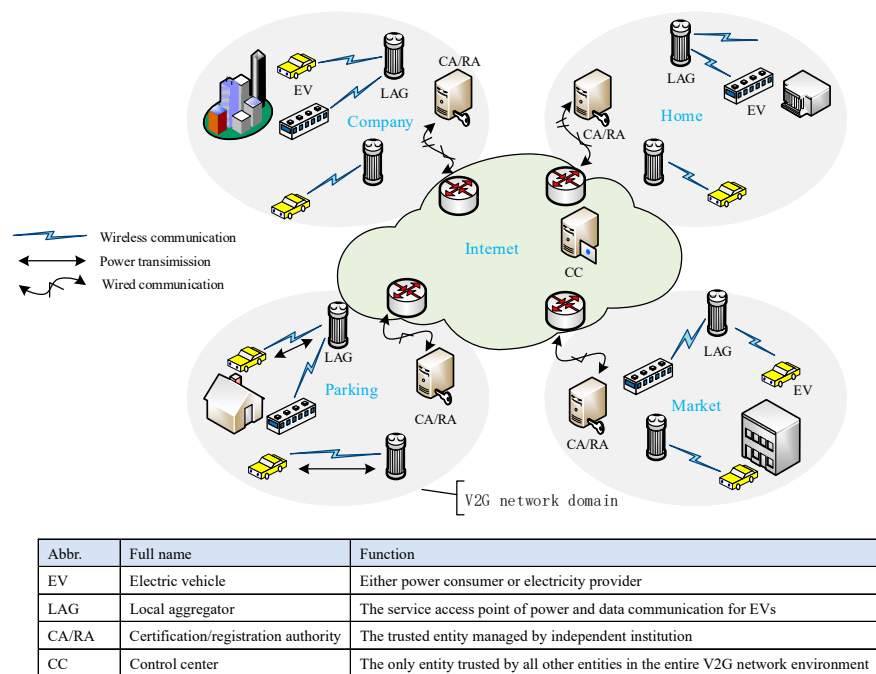
In this article, we first present the three-tier architecture of a smart grid. Then we introduce the application of hybrid artificial intelligence in V2G systems. Finally, we conclude the article.

## 2. Smart Grid Network Architecture with EV Access

### 2.1. Network Architecture Design

In this section, we introduce the smart grid network architecture based on cloud computing and fog computing, which is the hardware foundation for implementing intelligent algorithms. The smart grid network architecture consists of three layers, namely the user layer, service layer, and network layer. The user layer is used to collect user information, the service layer is responsible for data analysis, requesting responses, and providing data-driven charging and discharging scheduling determined by cloud computing, fog computing, and AI technology, and the network layer integrates multiple communication technologies to ensure the interactions of the user layer and service layer. The smart grid network architecture based on cloud computing and fog computing is the hardware foundation, which can realize various intelligent algorithms.

As shown in Figure 1, according to the cross-layer design, the V2G auxiliary service system framework of this model can be divided into the service layer [17], network layer, and user layer. This cross-layer design takes into account the information integration and service interaction requirements of the service network and security requirements of all levels [18].



**Figure 1.** An illustration of vehicle-to-grid (V2G) network environment.

The V2G auxiliary service system adopts service-oriented technology, optimized control, and coordinated control theory to guide the distributed control system. The specific application of service-oriented technology is based on the actual charging demand of EV users and the real-time operating state of the power grid. Information interaction and hierarchical control can be achieved through visualization technology and user interface technology.

Figure 2 is the implementation diagram of EV grid integration, which shows the three layers of the V2G auxiliary service system mentioned above. The mesh communication between EV and EV supply equipment can be achieved through power line communication (PLC) and 5G networks. The data from electric vehicles and any other devices is stored by cloud. It could be processed for a variety of application purposes, such as optimized charging.

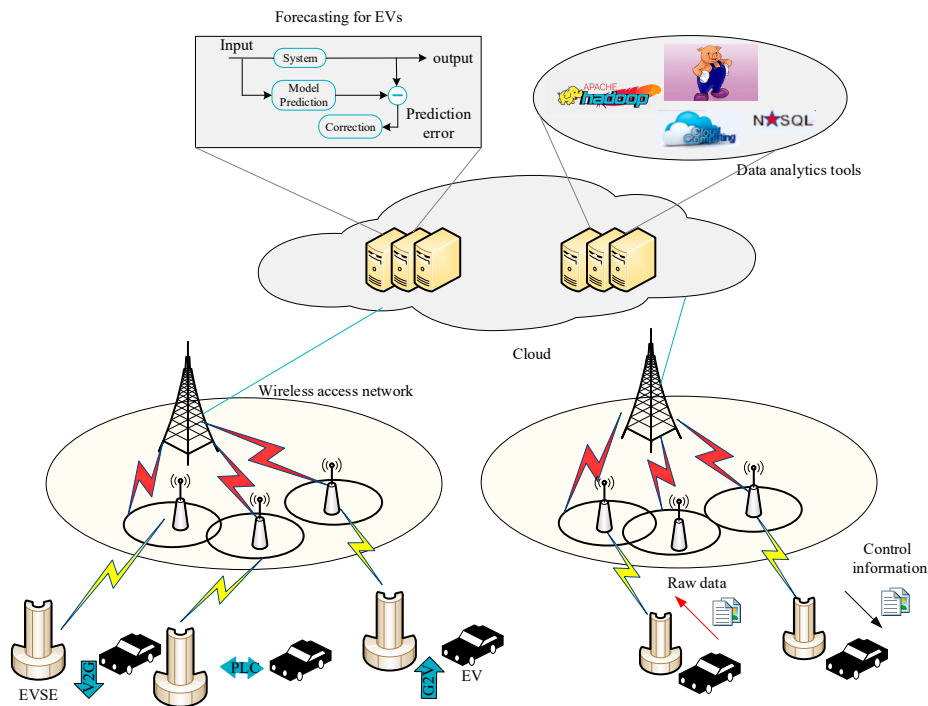


Figure 2. Implementation diagram of electric vehicles (EV) grid integration.

As shown, the user layer could provide an interface for the user to interact with the service and collect the information of each entity in real time and upload it to the network layer. The network layer is the connection between the user layer and service layer. The service layer is responsible for dealing with user requests, making intelligent decisions, and providing macro control over the entire grid.

## 2.2. The User Layer

The user layer is the lowest level and it is composed of various types of end-users. The end-users are comprised of all vehicle-to-grid system entities and electric vehicle users that need to use the services in service layer. The end-users can initiate a service request to the service layer through the service access point (SAP) using the communication device. Accordingly, they will also receive responses from the upper layers through a similar SAP.

The data collected by sensors on EVs are diverse and plentiful. It includes the user's driving behavior as uploaded by smart devices and wearable devices, battery security data collected by the battery sensors, and EV charging information uploaded by the charging station. This makes it easy to collect EV user behavior information, including the distance of the trip, the connection and disconnection times of the charger, and the battery state of charge (SOC). These data could be used to analyze the behavior of EV users through artificial intelligence algorithms.

## 2.3. The Network Layer

The network layer consists of a variety of communication devices and technologies that enable interaction between data in the user layer and computing resources in the service layer. In the network layer, 5G communication is the most widely used. The network layer also integrates other common

communication technologies such as 3G/4G, satellite, GPS, etc., to form a heterogeneous converged network [19].

In smart grids, utilities may face bandwidth blocking caused by large amounts of data transmission, which may lead to communication barriers. This is because with the popularity of V2G services, the number of uploaded data has grown rapidly. However, in the context of 3G or 4G, the service layer could not fully meet the V2G service performance specifications. Fortunately, the 5G mobile network supports a variety of connectivity and transmission advantages and is expected to readily facilitate services provided by various clouds. 5G technology features high speed, high capacity, low power consumption, low delay, and high reliability [20]. It is the main development direction of mobile Internet and IoT, which can fully meet the bandwidth, delay, reliability, security, and mobility requirements of the V2G system [21].

In the V2G communication of the auxiliary service system, we propose to apply 5G technology. Specifically, as V2G uses multicast to aggregate data about the remaining energy in EVs and release the energy requirements of the smart grid. It has shortcomings such as bandwidth, delay, reliability, safety, and mobility, which can be solved by 5G technology. First, we analyzed the communication requirements and communication technologies in the V2G system. Next, we designed an overall system architecture and information transmission process based on 5G technology. This process includes a V2G service connection process initiated by the EV, a V2G service connection process initiated by the V2G server, and an aggregator registration update process initiated by the EV. Our proposed 5G-based V2G system can meet the requirements of wireless communication networks, allowing it to improve the integration of renewable energy with the power grid. Together, this improves the economic and environmental efficiency of the power grid.

EVs send V2G service data messages to the aggregator through the Internet interface, and then the aggregator updates the corresponding Internet ID in the main table of the EV information module and sets the network label of the data exchange. Therefore, the aggregator can receive IP packets from the EV [22]. When the electric vehicle uses a two-way charger for charging, the PLC and the aggregator are used for data interaction, and the aggregated data is transmitted to the cloud computing center via 5G communication. When the electric vehicle is traveling, the sensor real-time data and user requests are sent to the cloud computing center through the car-connected wireless network. When the power demand on the consumer side of the grid changes drastically, the charging point (CP) accesses these sensors through multicast technology to summarize the battery status of the EV, issue the power demand, and assign the state of charge (SOC) transition instructions in V2G [23].

#### 2.4. The Service Layer

The service layer consists of two parts: A traditional cloud layer and a temporary fog layer. The service layer can provide various V2G services to end-users of the user layer through SAP.

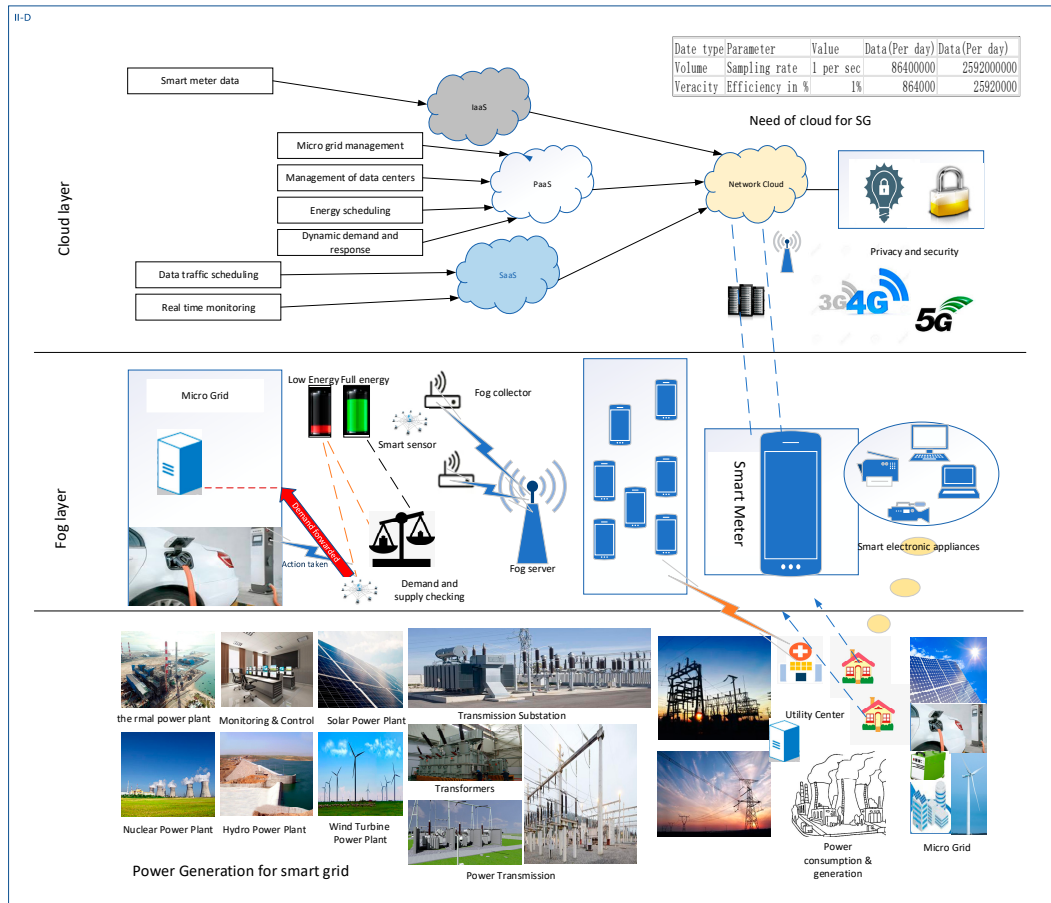
Cloud computing (CC) is a kind of distributed computing technology. It splits a huge computational processing program into a myriad of smaller subroutines through the network and passes the processing results back to the user after calculation and analysis. We have integrated cloud computing and V2G systems to enable V2G system entities to handle more complex service applications.

However, providing a reliable and low latency response V2G service is a new challenge. In detail, in V2G system, smart meters can collect large amounts of data from EVs, drivers, charging stations, and infrastructure, which requires the execution of data analysis tools on the cloud. Since this data is different in volume, velocity, and variety, storing, processing, and analyzing the data is a challenging task. Although SG can transfer all data to the cloud-based infrastructure for further processing, it is difficult for Cloud Computing (CC) to store and process these collected data in a short time. Therefore, we apply fog computing (FC) to solve these problems.

Fog computing, as shown in Figure 3, is an integrated version of mobile cloud computing (MCC) and mobile edge computing (MEC) that demonstrates a better infrastructure that includes resources for network edge services to provide applications for future Internet and services [24]. Generally, it



acts as a bridge between the terminal device and the cloud. FC collects data from SG entities and then stores and calculates them locally. The data is transferred to the cloud for storage after processing. It reduces the need for network bandwidth for overly large data and can help protect sensitive data by providing local analytics.



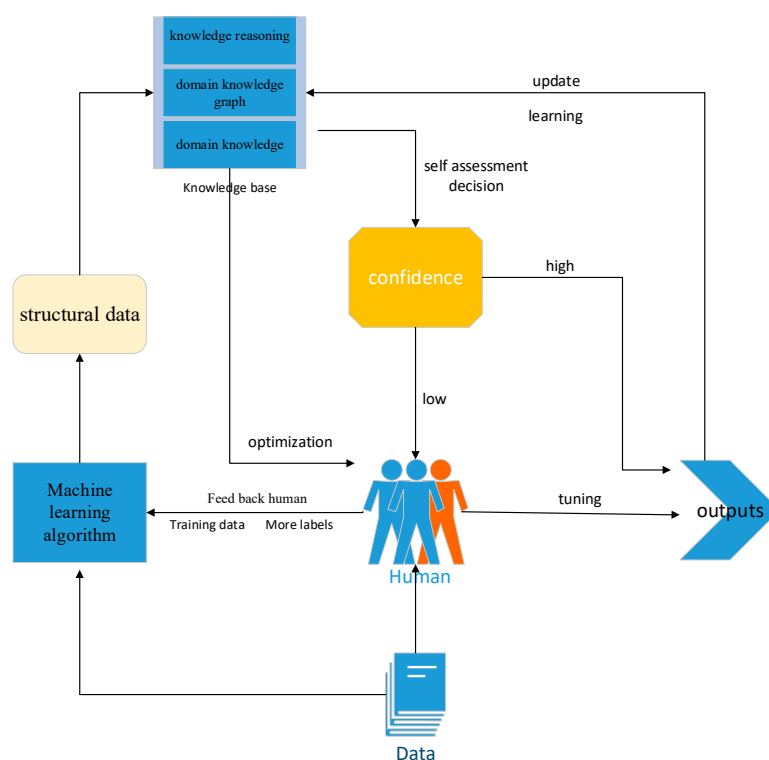
**Figure 3.** Three-tier architecture of smart grid with cloud computing (CC) and fog computing (FC).

### 3. Hybrid Artificial Intelligence

#### 3.1. The Introduction of Hybrid Artificial Intelligence

AI is an area of computer science that addresses common cognitive problems related to human intelligence, such as learning, problem solving, and pattern recognition. Despite the great progress made in the artificial intelligence approach, most effective artificial intelligence systems require a high degree of manual supervision to prevent errors [25]. This is largely because artificial intelligence can only move toward a specific goal, while human intelligence can flexibly adjust the target according to the current situation. The successful application of AI depends not only on big data and powerful algorithms but also on people-oriented design. Specifically, AI should reflect the realities of user needs. To achieve this, we introduced the concept of Human-in-the-loop (HITL) hybrid intelligence.

HITL is the combination of supervised machine learning (SML) and active learning (AL) [26]. This concept uses human and machine intelligence to create machine learning models. In this process, humans directly participate in algorithm training, parameter adjustment, and result judgment, as shown in Figure 4. User feedback can help overcome the deficiencies of the AI system, leading to a virtuous cycle of tasks. In this way, the decision-making ability of the AI system can be greatly enhanced and errors would largely be decreased. In conclusion, our goal is to use human training to reduce errors in AI predictions and improve the accuracy of the AI algorithm.



**Figure 4.** Base framework of human-in-the-loop (HITL) hybrid intelligence.

We could apply HITL in situations where: (1) There is little data available at present. Because of the scarcity of data in the early stages of a project, it is difficult to classify the data. At this stage, people can make initial judgments, from which the machine can learn and eventually take over the tasks. (2) The cost of errors is high. If errors will cause great interference to the AI prediction results, manual supervision is required. (3) There are class imbalances. Machines are prone to errors when dealing with imbalanced data, but humans can solve these problems by modifying the AI algorithm.

### 3.2. The Impact of EV User Charging Behavior on the Grid

Unregulated charging of electric vehicles will likely pose great challenges to a city's power supply system. In contrast, the controlled charging behavior of an EV can actually enhance the stability of the smart grid and achieve peak shaving and valley filling effects [27]. The user's charging behavior mainly affects the stability of the grid by the following factors: (1) The time to start charging and time to end charging: If all users choose to start charging their electric car after work and end charging in the morning, this activity would accentuate the peaks and valleys of the grid. (2) The amount of charging required: Users that need more charge will absorb more power from the grid. (3) Charging scene of the electric car: If an electric car is charged during travel, it has a higher charging priority and is not very predictable. However, if a user is charging at home at night, the charging priority is lower, and the charging behavior of the user can be reasonably scheduled to improve the stability of the smart grid.

### 3.3. Use HITL to Analyze User's Charging Behavior

Power companies should develop appropriate rules to manage the charging behavior of electric vehicles. By analyzing the user behaviors of EV users, we can formulate rules to schedule the charging behavior of EVs. By using artificial intelligence, we can cluster and classify user behaviors and predict the user's charging behavior.

In details, in order to increase the ability of the smart grid to handle charging loads, we collected user information and applied artificial intelligence technology to interpret the data. By utilizing user's charging data in the smart grid, we can categorize EV users and select users who are more suitable for

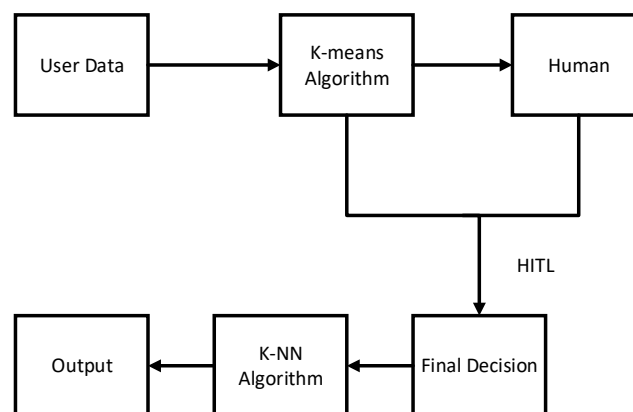
load balancing. This can help efficiently manage the charging or discharging behavior of EVs in the SG and achieve efficient load balancing and reliable energy transfer.

However, for a small number of users with special charging behaviors, using artificial intelligence methods alone cannot achieve good results, and can even increase the error of the entire system. Therefore, we propose the use of the HITL method to analyze the user's charging behavior.

#### 4. Simulation

We used real charging records of electric vehicles, released by Adaptive Charging Network (ACN)-Data as a data set to analyze user charging behavior. The ACN-Data is a dynamic dataset of workplace EV charging which currently includes over 30,000 sessions with more added daily. We compiled 26,000 charge records for 318 electric car users, including the user's charge connection time, charge completion time, charge disconnect time, and charging amount [28].

The data suggest that, for the vast majority electric car users, their charging behavior is predictable, e.g., starting to charge after work every day and disconnecting the electric car and charging pile before work the next morning. However, the charging behavior of a small number of users is not regular. Even if only those users with regular charging behavior can be scheduled, the stability of the power grid will be greatly improved. As a result, we use the HITL-based K-Means clustering and k-NN algorithm to predict the charging behavior of EV. The AI algorithm flow chart based on hybrid artificial intelligence is shown in Figure 5. In this simulation we used MATLAB R2016a as the simulation software. The computer's CPU is Intel core i5 5200U and the memory is 4 GHz.



**Figure 5.** Artificial intelligence (AI) algorithm flow chart based on hybrid artificial intelligence.

Clustering analysis is a crucial analytical algorithm in data mining [29]. Clustering is to divide similar objects into different groups or more subsets by static classification so that the member objects in the same subset have similar properties. The K-means algorithm has been widely used for its simple algorithmic design, faster clustering speed, and good clustering effect. Hence, in this article, we used K-means to analyze the uncertainty of EV user charging behavior.

In order to analyze the users' charging behavior, we first clustered the users' behavior. When selecting variables as the basis for user behavior clustering, we mainly considered the following ideas: (1) The user mainly affects the grid in their choice of charging start time, charging end time, charging amount, and charging scene. (2) The amount of charge is roughly positively related to the charging time. (3) Some users have regular charging behavior and some users have no regular charging behavior. Combining the above points, in order to achieve better clustering results, we chose the average charging time, the standard deviation of the charging time, and the connection time as the basis for clustering user behavior.

Before clustering the data, we preprocessed the data. We manually screened and removed low-quality data from the raw data set. For example, we removed users with fewer than three records,



which were random and not useful to determining patterns of behavior. Then, the user data was normalized using the linear normalization method, with specific rules:

$$X^* = \frac{X - X_{\min}}{X_{\max} - X_{\min}}. \quad (1)$$

We used the K-means algorithm to classify EV user charging behavior, however, for data at cluster boundaries, the clustering effect may not be ideal [30]. Drawing on the idea of HITL, after clustering the data with the K-means algorithm, we used manual methods to check and re-mark data at the boundary.

Finally, for the new user data, we used the k-NN algorithm to for classification. This allowed us to filter out users with unstable behavior for further processing.

#### 4.1. HITL Based K-Means Clustering for EV Driver Behavior

K-means is an unsupervised learning algorithm, which means the data to be processed has no labels [31]. It divides data points into  $k$  clusters so that each point belongs to the cluster corresponding to his nearest mean [32]. The 'k' represents the number of clusters and the 'means' represents the mean of the data objects in the cluster. The algorithm makes each point belong to the cluster corresponding to his nearest mean (this is the cluster center) and uses it as the standard of clustering. There are many ways to calculate distance. The most popular way in this algorithm uses Euclidean distance to measure distance. The formula is given below:

$$,dict(x_i, y_i) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2)$$

In this formula,  $x_i$  and  $y_i$  are two points in a  $n$ -dimensional Euclidean space.

In the beginning of K-means algorithm,  $m$  objects are randomly selected from  $n$  data objects as the initial cluster center, and then the distance between each object and the cluster center is calculated and the corresponding object is re-divided according to the minimum distance. Finally, the mean of each cluster as a new cluster center is recalculated. The  $k$ th iteration cluster center is updated as follows:

$$Center_k = \frac{1}{|C_k|} \sum_{x_i \in C_k} x_i, \quad (3)$$

where  $|C_k|$  represents the number of data points in  $k$ th cluster, center  $k$  is a vector containing  $D$  attributes as shown as below:

$$Center_k = (Center_{k,1}, Center_{k,2}, \dots, Center_{k,D}). \quad (4)$$

Moreover, the K-means algorithm requires constant iteration to re-divide the clusters and update the cluster center. In this article, we use the sum of squared errors (SSE) to determine when to terminate the iteration. The SSE is as follows:

$$SSE = \sum_{k=1}^K \sum_{x_i \in C_k} dist(x_i, Center_k), \quad (5)$$

where  $K$  represents the number of clusters. When the difference of SSE between two iterations is less than a certain threshold  $\delta$ , the iteration is terminated.

The main disadvantage of this algorithm is that the random selection of the initial  $K$  point may lead to local convergence. Therefore, we used hybrid intelligence to refine the results. In this work, the use of HITL helped refine the classification and clustering results. After clustering the data with the K-means algorithm, we used manual methods to check and re-mark the data at the boundary [33], so that we could obtain a better result. The detailed description of the HITL-based K-means algorithm is shown in Figure 6.

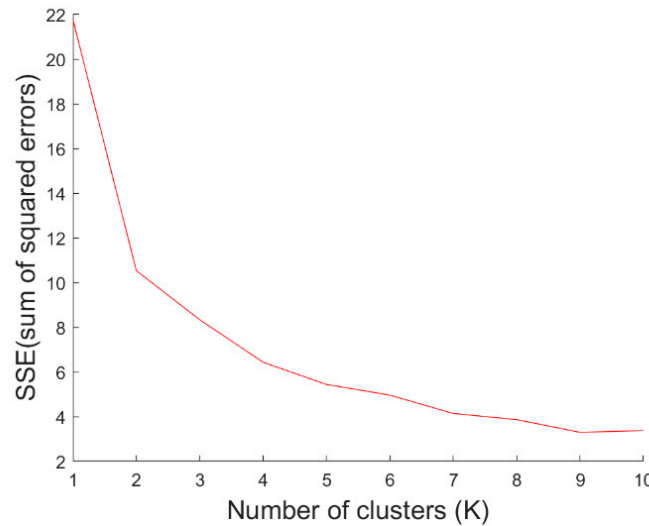
**1: Generate  $k$  random numbers and corresponding  $k$  cluster centroids;**  
**2: For each instance**  
**3:   Assign the instance to cluster with closest centroid;**  
**4: Repeat**  
**5:   Update the centroid;**  
**6:   Reassign the instances to clusters;**  
**7: Until no update;**  
**8: Manual methods to check and re-mark the data at the boundary**

**Figure 6.** The detailed description of HITL based K-means algorithm.

Before classification, we first normalized the data. As mentioned before, the user data were processed into the same tuple structure is as follows:

$$\mu := (t_{charge}, \sigma_{charge}, \sigma_{connect}). \quad (6)$$

To find the best number of clusters, we used the elbow algorithm to get the K-SSE curve, as shown in Figure 7.

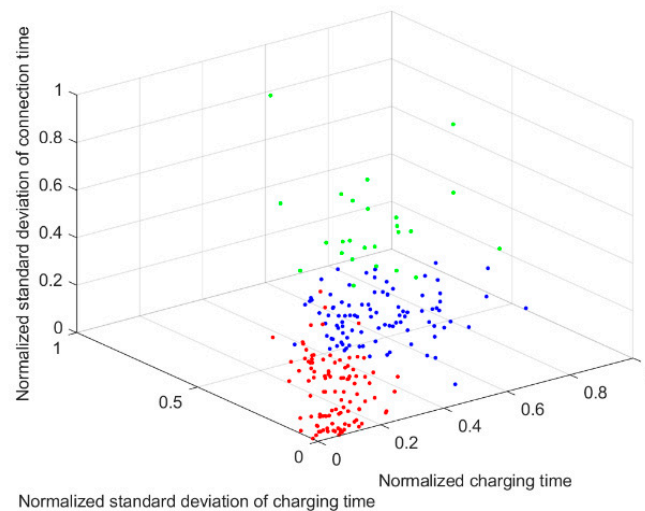


**Figure 7.** The curve obtained by the elbow algorithm.

When the number of clusters is approaching the ideal cluster number, SSE shows a rapid decline. However, when the number of clusters exceeds the ideal cluster number, SSE will continue to decline. At the meantime, the rate of decline will slow down. By plotting the K-SSE curve and finding the inflection point on the way down, the  $K$  value can be better determined. Therefore, our goal is to determine a small value of  $k$  that still has a low SSE, and the elbow usually represents where increasing  $k$  provides diminishing returns. By observing the curve, the clustering effect is better when the number of classifications  $k = 3$ . Next, on the basis of clustering, we used manual discriminant methods to check and re-mark the data at the boundary, with final results shown in Figure 8.

After clustering, human intelligence is used to assign labels to user behavior. Based on the simulation results, we define red and blue users as behaviorally stable EV users because their standard deviation of charging behavior is low, which means that their charging behavior is regular and predictable, and such users can participate in the centralized scheduling of the grid. In contrast, green users are unstable EV users because their standard deviation of charging behavior is large, their

charging behavior is unpredictable, and it is not useful for such users to participate in the centralized scheduling of the grid.



**Figure 8.** The clustering results under HITL based K-means algorithm.

Then, we use the hybrid enhancement intelligence method to delete the users with unstable behavior by batch deletion and manual filtering. We regard the remaining users as users with stable behavior that can participate in the power supply scheduling of the grid.

#### 4.2. *k*-NN Classification for EV Driver Behavior

After clustering the data in the dataset by K-means and HITL methods, we also need to classify the new EV users' behavior to manage and optimize the charging behavior of new electric vehicle users. However, it is unrealistic to re-cluster the entire data set every time new data is added. So, we would like to use the K-Nearest Neighbors (*k*-NN) method to classify new EV driver behavior according to the results of the K-means clustering algorithm.

*k*-NN is a supervised learning method that classifies new data by measuring the distance between new data points and the remaining marked data points. In the *k*-NN algorithm, all the selected neighbors have been correctly classified. It is a distance-based algorithm where the category of new data can be determined based on the category of the nearest neighbors, e.g., using the Euclidean method (Equation (1)) to calculate the distance between data points. Then, we can select the training data of the first *k* distances closest to the new data and assign it the same data category. There is no fixed rule for the selection of *k* values. Generally, according to the distribution of samples, a smaller value is best, and a suitable *k* value can be determined by cross-validation. The detailed description of the *k*-NN algorithm is shown in Figure 9.

- 1: Calculating the distance between the labelled point and the current point;**
- 2: Sorting the distance in ascending order;**
- 3: Select *k* points with the smallest distance from the current point;**
- 4: Select the label with the most occurrences of the first *k* points as the prediction classification of the current point.**

**Figure 9.** The *k*-nearest neighbors algorithm.

We define the training error rate as the ratio of the K-nearest training sample marker to the input marker, and the error rate is expressed as:

$$\frac{1}{k \sum_{x_i \in N_k(x)} I(y_i \neq c_j)} = 1 - 1/K \sum_{x_i \in N_k(x)} I(y_i = c_j), \quad (7)$$

and as a result, the proper  $k$  can maximize the coefficient below in the training set.

$$1/K \sum_{x_i \in N_k(x)} I(y_i = c_j) \quad (8)$$

After that, we use cross-validation to evaluate the performance of the model, which can prevent over-fitting. As mentioned before, there were 223 data in total after screening, which we divided into six groups. As shown in Figure 10, the first five groups were used for cross-validation.

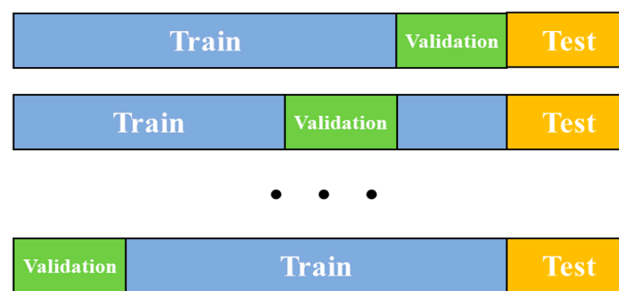


Figure 10. Cross-validated data grouping.

In cross-validation, four groups of data were used as a training set and 1 group of data remained as a test set. Each group of data had an opportunity to be test set. The result of cross-validation results showed that when  $k = 5$ , the training error rate was the smallest, and the correct rate for different groups acting as the test set is shown in Figure 11. Finally, the remaining 48 data were used as the final verification set, and the accuracy was found to be 97.91%.

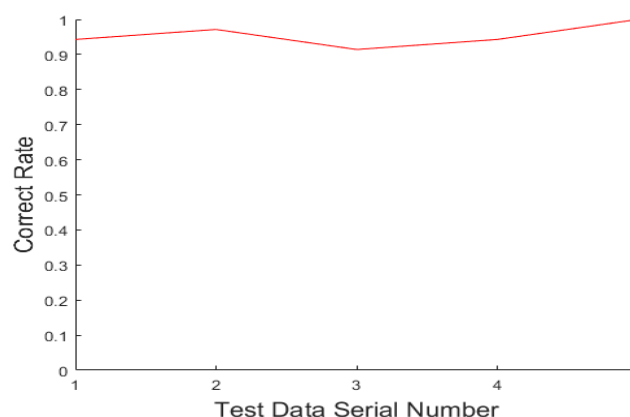


Figure 11. The correct rate of cross-validation.

The confusion matrix is used to visualize the performance of an algorithm [34]. In predictive analysis, the confusion matrix is a table made up of False Positives (FP), False Negatives (FN), True Positives (TP), and True Negatives (TN). By using a confusion matrix, it is easy to see if the learning machine will confuse two similar classes. Assuming that the False Positives, False Negatives, True Positives, and True Negatives represent the corresponding number of samples respectively:

$$TP + FP + TN + FN = \text{Total}. \quad (9)$$

The classification result's "confusion matrix" in this experiment is shown in Figure 12.

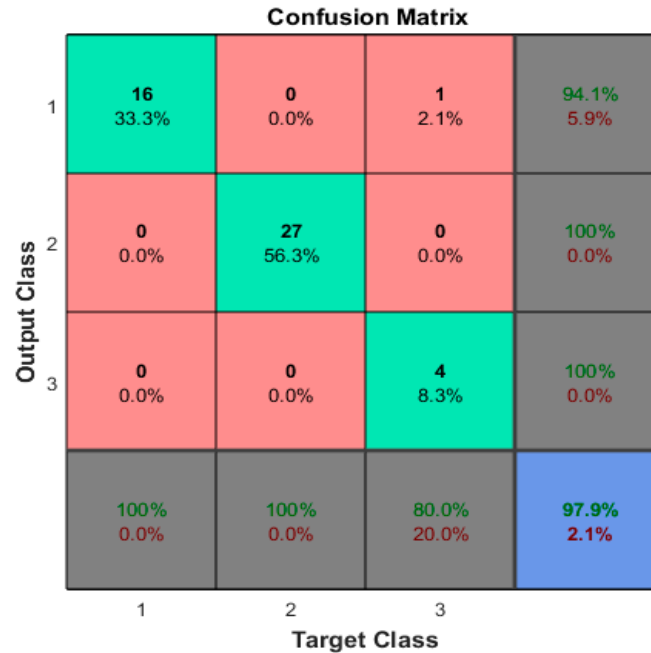


Figure 12. The confusion matrix of the classification result.

In order to comprehensively consider the generalizable performance of the model, the Receiver Operating Characteristics (ROC) curve can be drawn to see the quality of the model partition type [35]. In the case of uneven class distribution, the ROC curve is usually more efficient. In the ROC curve, the vertical axis is the True Positive Rate (TPR), and the horizontal axis is the False Positive Rate (FPR), which are defined as Equations (10) and (11).

$$TPR = \frac{TP}{TP + FN} \quad (10)$$

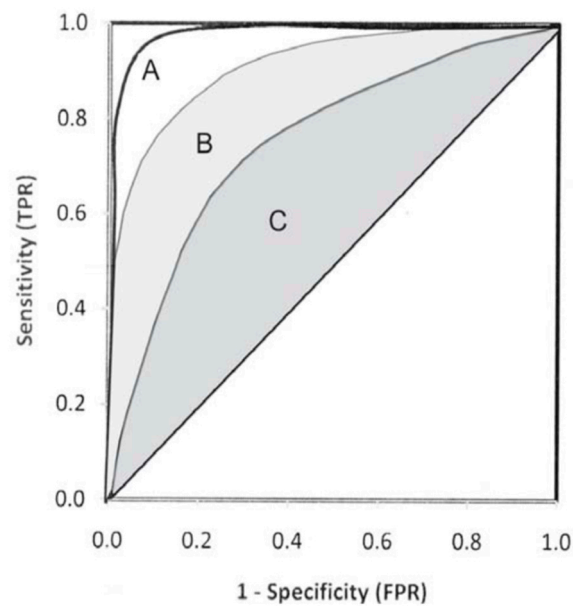
$$FPR = \frac{FP}{TN + FP} \quad (11)$$

Area Under ROC Curve (AUC) is the area under the ROC curve and is an indicator of the performance of the algorithm. The AUC can be obtained by calculating the total area under the ROC curve. For example, supposing that the ROC curve is formed by sequentially connecting a series of points of coordinates  $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots, (x_m, y_m)\}$ , the AUC can be estimated as:

$$AUC = \frac{1}{2} \sum_{i=1}^{m-1} (x_{i+1} - x_i) \cdot (y_{i+1} + y_i) \quad (12)$$

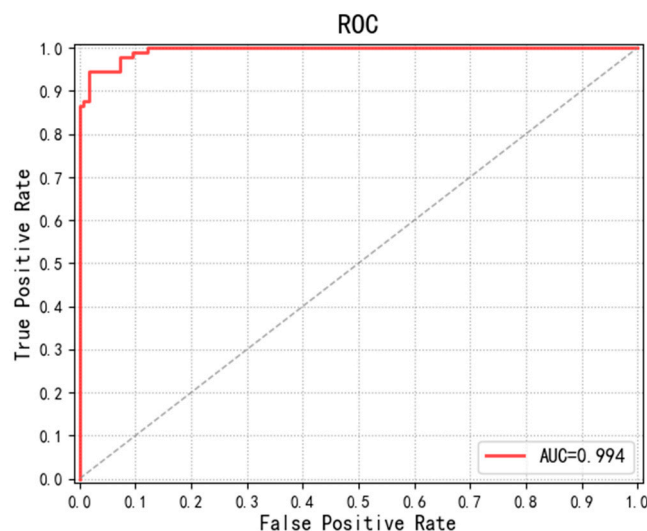
AUC-ROC curve is a performance measurement for a classification problem at various threshold settings. ROC is a probability curve and AUC represents degree or measure of separability. It tells how much a model is capable of distinguishing between classes. The higher the AUC, the better the model is at predicting 0 as 0 and 1 as 1. The higher the AUC, the better the model is at distinguishing between the drivers with regular behavior and irregular behavior.

As shown in Figure 13, the ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis. Area A, B and C are the AUC of three different classification results with different parameters. From Figure 9, it can be seen that the AUC-A is larger than AUC-B, and AUC-B is larger than AUC-C. After comparing these three curves, we could know that the AUC-ROC curve corresponding to AUC-A has the best performance.



**Figure 13.** The comparison between the different Area Under ROC Curve (AUC) curve.

The ROC curve of the classification model we built is shown in Figure 14. The grey dotted line is the set of points where FPR equals PTR. We found that the curve was close to the upper border with  $AUC = 0.994$ , indicating that the classification effect of our model was very good. Therefore, using this model, we could expect to classify the behavior of new users of electric vehicles to determine whether they could be efficiently uniformly scheduled. Power companies could potentially use these classification results to develop appropriate scheduling strategies to centrally coordinate the charging behavior of these electric vehicles in order to achieve peak-shaving and valley filling.



**Figure 14.** The ROC curve of the proposed model.

## 5. Conclusions

To effectively manage and dispatch the charging and discharging behavior of EVs in the SG, we proposed a smart grid communication network architecture based on cloud computing and fog computing in this paper. The whole architecture consisted of three layers: The user layer is used to collect user information, the service layer is responsible for data analysis, requesting responses, and providing data-driven charging and discharging scheduling determined by cloud computing,



fog computing and AI technology, and finally the network layer integrates multiple communication technologies to ensure the interactions of the user layer and the service layer. Utilizing the respective strengths of each of these technologies, this architecture could facilitate the rapid growth of G2V and V2G services. Thorough analysis of user behavior is the premise of G2V and V2G scheduling, we propose a user behavior analysis method combining HITL-based K-Means clustering and a k-NN algorithm as a potential solution. The proposed method could label the dataset through an automatic process and make it unnecessary to perform clustering each time when a new user joins the charging network. This method could schedule the electric vehicles for the smart grid more efficiently. Cross-validation experiments and performance evaluation measures suggest that the proposed technology can classify the user's charging behavior with excellent predictive capacity and the resulting classification could be an important basis for scheduling charging and discharging.

**Author Contributions:** Y.S. conceived and designed the experiments; Y.S. and W.F. performed the experiments; M.K. and F.Y. analyzed the data; Y.S. wrote the paper. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Acknowledgments:** Thanks to the editors and reviewers for their careful review, constructive suggestion, and reminding, which helped improve the quality of the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Zhou, L.; Chen, J.; Huang, J. Research on Smart Grid analysis based on informatization. In *2012 International Conference on Machine Learning and Cybernetics*; IEEE: Piscataway, NJ, USA, 2012; Volume 3, pp. 1025–1029.
2. Vineetha, C.P.; Babu, C.A. Smart grid challenges, issues and solutions. In *Proceedings of the 2014 International Conference on Intelligent Green Building and Smart Grid (IGBSG)*, Taipei, Taiwan, 23–25 April 2014; pp. 1–4.
3. Shin, J.; Song, B.; Lee, S.; Shin, S.; Kim, Y.; Jung, G.; Jeon, S. Contactless power transfer systems for On-Line Electric Vehicle (OLEV). In *Proceedings of the 2012 IEEE International Electric Vehicle Conference*, Greenville, SC, USA, 4–8 March 2012; pp. 1–4.
4. Wan, Y.; Cao, J.; Zhang, S.; Tu, G.; Lu, C.; Xu, X.; Li, K. An integrated cyber-physical simulation environment for smart grid applications. *Tsinghua Sci. Technol.* **2014**, *19*, 133–143.
5. Ferreira, J.C.A.; Monteiro, V.; Afonso, J.L. Vehicle-to-Anything Application (V2Anything App) for Electric Vehicles. *IEEE Trans. Ind. Informatics* **2013**, *10*, 1927–1937. [[CrossRef](#)]
6. Wu, Q.; Nielsen, A.H.; Ostergaard, J.; Cha, S.T.; Marra, F.; Chen, Y.; Træholt, C. Driving Pattern Analysis for Electric Vehicle (EV) Grid Integration Study. In *Proceedings of the 2010 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT Europe)*, Gothenberg, Sweden, 11–13 October 2010; pp. 1–6.
7. Ge, S.Y.; Wang, L.; Liu, H.; Feng, L. The impact of discharging electric vehicles on the distribution grid. In *Proceedings of the IEEE PES Innovative Smart Grid Technologies*, Tianjin, China, 21–24 May 2012; pp. 1–4.
8. Lopez, K.L.; Gagne, C.; Gardner, M.-A. Demand-Side Management Using Deep Learning for Smart Charging of Electric Vehicles. *IEEE Trans. Smart Grid* **2019**, *10*, 2683–2691. [[CrossRef](#)]
9. Masrur, M.A.; Skowronska, A.G.; Hancock, J.; Kolhoff, S.W.; McGrew, D.Z.; Vandiver, J.C.; Gatherer, J. Military-Based Vehicle-to-Grid and Vehicle-to-Vehicle Microgrid—System Architecture and Implementation. *IEEE Trans. Transp. Electrification* **2017**, *4*, 157–171. [[CrossRef](#)]
10. Yu, R.; Ding, J.; Zhong, W.; Liu, Y.; Xie, S. PHEV charging and dis-charging cooperation in V2G networks: A coalition game approach. *IEEE Internet Things J.* **2014**, *1*, 578–589. [[CrossRef](#)]
11. Fan, Z. A Distributed Demand Response Algorithm and Its Application to PHEV Charging in Smart Grids. *IEEE Trans. Smart Grid* **2012**, *3*, 1280–1290. [[CrossRef](#)]
12. Ghavami, A.; Kar, K. Non-linear pricing for social optimality of PEV charging under uncertain user preferences. In *Proceedings of the Information Sciences and Systems (CISS)*, Princeton, NJ, USA, 19–21 March 2014; pp. 1–6.
13. Chekired, D.A.; Khoukhi, L.; Mouftah, H.T. Decentralized Cloud-SDN Architecture in Smart Grid: A Dynamic Pricing Model. *IEEE Trans. Ind. Informatics* **2018**, *14*, 1220–1231. [[CrossRef](#)]

14. Luo, C.; Huang, Y.F.; Gupta, V. Stochastic dynamic pricing for EV charging stations with renewables integration and energy storage. *IEEE Trans. on Smart Grid* **2017**, *9*, 1494–1505. [\[CrossRef\]](#)
15. Huang, Q.; Jia, Q.-S.; Guan, X. Robust Scheduling of EV Charging Load With Uncertain Wind Power Integration. *IEEE Trans. Smart Grid* **2016**, *9*, 1043–1054. [\[CrossRef\]](#)
16. Wan, Z.; Li, H.; He, H.; Prokhorov, D.; He, H.L.H. Model-Free Real-Time EV Charging Scheduling Based on Deep Reinforcement Learning. *IEEE Trans. Smart Grid* **2019**, *10*, 5246–5257. [\[CrossRef\]](#)
17. Chukwu, U.C.; Mahajan, S.M. Real-Time Management of Power Systems with V2G Facility for Smart-Grid Applications. *IEEE Trans. Sustain. Energy* **2013**, *5*, 558–566. [\[CrossRef\]](#)
18. Hong, H.-J. From Cloud Computing to Fog Computing: Unleash the Power of Edge and End Devices. In Proceedings of the 2017 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), Hong Kong, China, 11–14 December 2017; pp. 331–334.
19. Zhou, Z.; Lin, T.; Thulasiraman, K.; Xue, G.; Sahni, S. Cross-Layer Network Survivability Under Multiple Cross-Layer Metrics. *J. Opt. Commun. Netw.* **2015**, *7*, 540. [\[CrossRef\]](#)
20. Wang, J.; Jin, A.; Shi, D.; Wang, L.; Shen, H.; Wu, D.; Hu, L.; Gu, L.; Lu, L.; Chen, Y.; et al. Spectral Efficiency Improvement with 5G Technologies: Results from Field Tests. *IEEE J. Sel. Areas Commun.* **2017**, *35*, 1867–1875. [\[CrossRef\]](#)
21. Ge, C.; Wang, N.; Selinis, I.; Cahill, J.; Kavanagh, M.; Liolis, K.; Politis, C.; Nunes, J.; Evans, B.; Rahulan, Y.; et al. QoE-Assured Live Streaming via Satellite Backhaul in 5G Networks. *IEEE Trans. Broadcast.* **2019**, *65*, 381–391. [\[CrossRef\]](#)
22. Pinter, L.; Farkas, C.; Laszlo, P. Impacts of electric vehicle chargers on the power grid. In Proceedings of the 2015 5th International Youth Conference on Energy (IYCE), Pisa, Italy, 27–30 May 2015; pp. 1–6.
23. He, S.; Zhu, J.; Zhang, J.; Zheng, L. An optimal charging/discharging strategy for smart electrical car parks. *Chin. J. Electr. Eng.* **2018**, *4*, 28–35.
24. Abolfazli, S.; Sanaei, Z.; Alizadeh, M.; Gani, A.; Xia, F. An experimental analysis on cloud-based mobile augmentation in mobile cloud computing. *IEEE Trans. Consum. Electron.* **2014**, *60*, 146–154. [\[CrossRef\]](#)
25. Sharman, D.B.; Durrani, T.S. An overview of AI applied to signal processing: A perspective on coupled systems. In Proceedings of the IEE Colloquium on Application of Artificial Intelligence Techniques to Signal Processing, London, UK, 15 March 1989; pp. 1–4.
26. Liang, H.; Yang, L.; Cheng, H.; Tu, W.; Xu, M. Human-in-the-loop reinforcement learning. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 4511–4518.
27. Fluhr, J.; Ahlert, K.-H.; Weinhardt, C. A Stochastic Model for Simulating the Availability of Electric Vehicles for Services to the Power Grid. In Proceedings of the 2010 43rd Hawaii International Conference on System Sciences; Institute of Electrical and Electronics Engineers (IEEE), Honolulu, HI, USA, 5–8 January 2010; pp. 1–10.
28. Marcincin, O.; Medvec, Z. Active charging stations for electric cars. In Proceedings of the 2015 16th International Scientific Conference on Electric Power Engineering (EPE), Kouty nad Desnou, Czech Republic, 20–22 May 2015; pp. 444–447.
29. Salicrú, M.; Vives, S.; Zheng, T. Inferential Clustering Approach for Microarray Experiments with Replicated Measurements. *IEEE/ACM Trans. Comput. Boil. Bioinform.* **2008**, *6*, 594–604. [\[CrossRef\]](#) [\[PubMed\]](#)
30. Sun, S.; Huang, R. An adaptive k-nearest neighbor algorithm. In Proceedings of the 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery, Yantai, China, 10–12 August 2010; Volume 1, pp. 91–94.
31. Cohen-Addad, V.; Klein, P.N.; Mathieu, C. Local Search Yields Approximation Schemes for k-Means and k-Median in Euclidean and Minor-Free Metrics. In Proceedings of the 2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS), New Brunswick, NJ, USA, 9–11 October 2016; pp. 353–364.
32. Olukanmi, P.O.; Twala, B. K-means-sharp: Modified centroid update for outlier-robust k-means clustering. In Proceedings of the 2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech), Bloemfontein, South Africa, 30 November–1 December 2017; pp. 14–19.
33. Yang, F.; Yang, L.; Wang, D.; Qi, P.; Wang, H. Method of modulation recognition based on combination algorithm of K-means clustering and grading training SVM. *China Commun.* **2018**, *15*, 55–63.

34. Song, W.; Huang, C. Mining High Utility Itemsets Using Bio-Inspired Algorithms: A Diverse Optimal Value Framework. *IEEE Access* **2018**, *6*, 19568–19582. [[CrossRef](#)]
35. Slaby, A. ROC Analysis with Matlab. In Proceedings of the 2007 29th International Conference on Information Technology Interfaces, Cavtat, Croatia, 25–28 June 2007; pp. 191–196.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).