Research Article

Predicting Short-Term Electricity Demand by Combining the Advantages of ARMA and XGBoost in Fog Computing Environment

Chuanbin Li , Xiaosen Zheng , Zikun Yang , and Li Kuang

School of Software, Central South University, Changsha 410075, China

Correspondence should be addressed to Li Kuang; kuangli@csu.edu.cn

Received 26 January 2018; Accepted 25 March 2018; Published 6 May 2018

Academic Editor: Xuyun Zhang

Copyright © 2018 Chuanbin Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of IoT, the disadvantages of Cloud framework have been exposed, such as high latency, network congestion, and low reliability. Therefore, the Fog Computing framework has emerged, with an extended Fog Layer between the Cloud and terminals. In order to address the real-time prediction on electricity demand, we propose an approach based on XGBoost and ARMA in Fog Computing environment. By taking the advantages of Fog Computing framework, we first propose a prototype-based clustering algorithm to divide enterprise users into several categories based on their total electricity consumption; we then propose a model selection approach by analyzing users' historical records of electricity consumption and identifying the most important features. Generally speaking, if the historical records pass the test of stationarity and white noise, ARMA is used to model the user's electricity consumption in time sequence; otherwise, if the historical records do not pass the test, and some discrete features are the most important, such as weather and whether it is weekend, XGBoost will be used. The experiment results show that our proposed approach by combining the advantage of ARMA and XGBoost is more accurate than the classical models.

1. Introduction

In recent years, with the rise of Cloud Computing [1, 2], more and more computing and storage processing are taking place in Cloud, and the vast employment of Cloud inevitably leads to high latency, network congestion, and low reliability. At the same time, with the wide adoption of IoT services, a variety of household appliances and sensors will be connected to the Internet and produce a large amount of data [3–5]. It has been estimated that the number of devices connected by IoT will reach 50 billion to 100 billion by 2020, which means there will be more and more data without the control of existing techniques on data processing and analysis, privacy leaks may be caused, and the quality of service will be decreased [6, 7]. In this regard, the rapid development of IoT has deepened the dilemma of Cloud Computing. The emergence of Fog Computing makes up for these shortcomings but also brings new opportunities and challenges to the transformation and upgrading of traditional industries. Electricity system, which

aims at providing enterprises with safe, reliable, and high-quality electric power, has become an indispensable part in the construction of national economy and people's life, so it is affected at first. Under the current technical conditions, it is still not possible to achieve large-scale storage of electric energy; therefore, it is required to generate electricity according to the system load at any time, or else the quality of electricity supply and usage may be affected, and even the safety and stability of the system may be endangered. It has become an urgent and important research issue to improve the accuracy of electricity demand prediction in Fog Computing framework.

In the field of electricity demand prediction, scholars have carried out extensive research. In the early stage, scholars basically followed the technology in the field of economic prediction, focusing on the rule of the load sequence in the form of time series itself. The prediction model is established by analyzing the qualitative relationship between the historical load and related factors, and the parameters are estimated

according to historical data. However, time series based approach requires historical data with high accuracy, insensitive to the factors such as weather and holidays. Actually, it is very difficult to express the nonlinear relation between the input and output by using a clear mathematical equation since the electricity data are nonlinear, time-varying, and uncertain. In order to further improve the accuracy of electricity demand prediction, artificial intelligence methods have been applied since the 1990s, such as neural network, expert system, and wavelet analysis. However, the existing methods usually can just be applied to a limited scenario and only effective for simple electricity systems with a small quantity of factors.

The main deficiencies of existing work include the following: (1) electricity data samples have been proven to have anomalies but few of the existing solutions detect and deal with the abnormal data. (2) When classifying users and their data, the number of clusters needs to be specified in advance, and the density distribution on electricity consumption is usually ignored in clustering users. (3) In the process of modeling, the temporal features of data are not fully excavated, and the interaction among the features is not fully considered. (4) Most of the solutions just adopt a sole model, which cannot give full play to the advantage of each model.

In order to overcome the shortcomings of existing solutions, in this paper, we propose a short-term electrical-demand prediction approach by combining the advantages of XGBoost and ARMA in Fog Computing framework. Firstly, sensors collect the real-time data of electricity consumption, and then the Fog Nodes would classify enterprise users into different groups according to the amount of electricity consumption and perform anomaly and outlier data detection and procession for each group. Secondly, a model selection process would be performed, that is, based on a series of tests including stationarity, white noise and Pearson correlation coefficient of data, as well as the observed electricity consumption rule; we will decide whether to use time series model or decision tree based model for the modeling of each enterprise group. Finally, the prediction values of each enterprise user are combined to obtain the final result. The accuracy of the proposed approach is verified to be 20% higher than classical models by experiments.

The rest of the paper is organized as follows: Section 2 reviews the related work on predicting short-term electricity demand. Section 3 describes the framework of the proposed solution, as well as the details of the involved key techniques. Section 4 introduces the experiment and analyzes the results. Section 5 summarizes our work and provides the future research plan.

2. Related Work

Because of the nonlinear, time-varying, and uncertainty characteristics of electricity data, it is difficult to accurately grasp the related factors and the rules of electricity consumption change. How to effectively improve the accuracy of electricity demand prediction has become a major challenge to researchers [8]. At present, the methods used for short-term electricity demand prediction mainly include time series [9–11], Regression Analysis [12, 13], Support Vector Regression

[14–16], Neural Network [17–20], Bayes [21], Fuzzy Theory [20, 22], and Wavelet Echo State Network [23]. Each kind of method has its own applicable scenario, and no model can achieve desired satisfying result alone.

In order to improve the accuracy of prediction, the current research works mainly focus on three directions. The first one is to explore the optimization of single model. Zhu et al. [24] propose to predict daily load roughly by ARMA first, then obtain the difference sequences that are noncyclical and strongly influenced by the weather, and finally propose an improved ARIMA prediction model with strong adaptability to weather. Factors which influence the electricity consumption are recognized and the mapping relation between key factors and electricity consumption are mined [25, 26]. Ghelardoni et al. [27] use the empirical mode decomposition method to divide the time series into two parts, describing the trend and the local oscillation of energy consumption values, respectively, and then use them to train the support vector regression model. Che et al. [28] use the human knowledge to construct fuzzy membership functions for each similar subgroup and then build an adaptive fuzzy comprehensive model based on self-organizing mapping, support vector machine, and fuzzy reasoning for prediction. An electricity load model is established based on improved particle swarm optimization algorithm and genetic algorithm [29, 30].

The second direction is to improve the accuracy of prediction by integrating different models. Haque et al. propose a hybrid intelligent algorithm based on wavelet transform and fuzzy adaptive resonance theory [31]. In [32–34], the wavelet decomposition is used to project the load sequence decomposition onto different scales, different models are used to predict the different components, and finally the final result is obtained by reconstructing the components. Pindoriya et al. [35] propose an adaptive wavelet neural network (AWNN) for short-term price prediction in the electricity market. Pany and Ghoshal [36] propose a local linear wavelet neural network (LLWNN) model instead of wavelet neural network for the electricity price prediction. Che and Wang [37] propose a hybrid model that combines the unique advantages of SVR and ARIMA models in both nonlinear and linear modeling.

The third direction is to explore composite models for prediction. The weighted average of all the results by various algorithm is usually used, and there are two kinds of ways to determine the weights. The first kind is to improve the fitting accuracy of historical electricity consumption by minimizing the fitting error. The main methods include monotone iterative algorithm [38], evolutionary programming [39], and quadratic programming [40]. Wang et al. [41] propose using adaptive practical swarm optimization algorithm to optimize the weight of the integrated model. The second kind is to determine the weights by evaluating the algorithm's score. Elliott and Timmermann [42] introduce the concept of the loss function, quantify the negative impact caused by different prediction errors, then take the minimum loss expectation as the goal, and perform optimization to get the weights. Yao et al. [43, 44] employ analytic hierarchy process (AHP) in multiobjective decision analysis to get the relative merits of each algorithm in fitting accuracy, model adaptability,

TABLE 1: Examples of the enterprise electricity consumption dataset.

record_date	user_id	power_consumption
2015/1/1	1	1135
2015/1/2	1	570
2015/1/3	1	3418
2015/1/4	1	3968

and result reliability, the judgment matrices are obtained, and then the weights are combined by calculating the main eigenvectors of each matrix. Petridis et al. study the use of probability models to determine the weight of each model and combine the values of each algorithm to obtain the final result [45]. Without enough quantitative theoretical basis, the weights of such models only reflect the advantages and disadvantages of the algorithms.

In summary, there have been many research works on the prediction of short-term electricity demand, and exploring composite models for prediction is the main trend. However, existing methods still have limitations. In the context of the rapid development of smart electricity grid, in this paper, we propose a short-term electricity demand prediction method based on XGBoost and ARMA.

3. Predicting Short-Term Electricity Demand

3.1. Problem Definition. Given a dataset $R = \{r_1, r_2, r_3 \dots r_n\}$, r_i is the i th electricity consumption record for a certain enterprise user, and r_i can be expressed as a 3-tuple, that is, $r_i = \langle \text{record_date}, \text{user_id}, \text{power_consumption} \rangle$, where record_date represents the date time, user_id is the ID of the enterprise user, and power_consumption represents the electricity consumption amount of the enterprise user on that day.

We use the dataset from Tianchi, which contains the historical records of 1454 enterprises in Yangzhong High-Tech Industrial Development Zone of Jiangsu Province from 2015-01-01 to 2016-11-30, for the following illustration and experiments. Examples of the dataset are shown in Table 1.

Given the dataset R and a month time in future, we aim to predict the total amount of the electricity demand on the desired month in this region on the basis of historical records.

3.2. Framework of the Proposed Approach

3.2.1. Smart Electricity System in Fog Computing. Fog Computing framework has the advantages of low latency, saving core bandwidth, and high reliability [46, 47]. Fog Nodes are located lower in the network topology and thus they have less network latency and more reactivity. As an intermediate between Cloud and terminals, Fog Layer can filter and aggregate enterprise messages and only send the necessary messages to Cloud, thus reducing the pressure on the core network. In order to serve enterprises in different regions, the same services will be deployed on the Fog Nodes in each region. Once the services in a certain area are abnormal, the requests can be quickly forwarded to other same services nearby, which makes the framework highly reliable.

As an extension of Cloud Computing, the framework pre-processes enterprise data and makes real-time decision and provides temporary storage to enhance the users' experience. In the current electricity systems, the number of hops from enterprise terminal to Cloud is generally 3 to 4 or even more, so the system will have to face the network delay when making real-time decisions. Figure 1 shows the framework of a smart electricity system in Fog Computing. Electricity meters collect data as sensors, and, for some enterprises with major changes of data and high real-time requirement, we can make real-time decisions in Fog Nodes to meet the need of real-time electrical-demand prediction; otherwise, the data can be buffered at Fog Nodes, compressed to save network bandwidth, and then transmitted to the Cloud.

3.2.2. Process of Electricity Demand Prediction Approach.

Figure 2 shows the process of electricity demand prediction approach based on multimodel fusion algorithm, which includes five main steps.

(1) *Data Preprocessing.* After data are collected, the missing values will be filled and their form will be unified.

(2) *Enterprise Users Clustering.* The size of electricity demand for each enterprise user is measured by counting the total amount of its historical electricity consumption. Then enterprises users will be divided into different groups by clustering them according to their sizes of electricity demand.

(3) *Model Selection.* We then aim to determine an appropriate training model for each group of enterprise users. First, the rules of electricity consumption for different groups of enterprise users are analyzed for prejudgement of model selection. If the electricity consumption changes with time, showing a periodical change or an obvious rising/falling trend, we consider to model the users' data by XGBoost. If the electricity consumption shows an irregular change over time or fluctuates around a certain constant and the fluctuation range is limited, we consider to model the users' data by time series model.

Second, a series of tests would be performed to verify the prejudgement and finally determine the selected model for each group of users. Feature correlation analysis and feature importance scores would be performed before selecting XGBoost modeling. Stationary and white noise test would be performed before selecting ARMA modeling. If neither XGBoost nor ARMA is appropriate, mean model will be used.

(4) *Model Building.* After model is selected for a given group of users, data cleaning is performed first, including the detection and processing of anomalies and outliers. For XGBoost modeling, anomalies and outliers will be deleted first, the influence of time factor and temperature on the prediction will be considered emphatically, Pearson correlation coefficient will be used to identify redundant features, and appropriate features will be selected to build the model by combining the feature importance outputted by pretraining. For ARMA modeling, anomalies and outliers will be modified by average value, and parameters p and q will be

FIGURE 1: Smart electricity system in Fog Computing.

FIGURE 2: Process of electricity demand prediction approach.

optimized based on the minimum amount of information principle of BIC.

(5) *Predicting Electricity Demand*. After modeling for different enterprise groups, the prediction values of each group will be summed up to get the final daily electricity demand in the desired month of this region.

3.3. Key Techniques. In this section, we will elaborate on the detailed key techniques in the five steps of the electricity demand prediction approach.

3.3.1. Data Preprocessing

(1) *Collecting External Weather Data*. Taking into account the impact of temperature on electricity consumption, we first

collect external weather data from <http://lishi.tianqi.com>. Samples are shown in Table 2.

(2) *Processing Missing Value*. We find that there are some missing values in Tianchi dataset, so we then fill the missing values with the mean value of the three days before and after the date with missing value. The detailed calculation is shown as follows:

$$c_{ij} = \frac{(c_{i,j-3} + c_{i,j-2} + c_{i,j-1} + c_{i,j} + 1, c_{i,j+2}, c_{i,j+3})}{6}, \quad (1)$$

in which c_{ij} represents user i 's missing record on the j th day of the year.

(3) *Unifying Data Form*. The electricity consumption records are reorganized to facilitate the follow processing. Each

TABLE 2: Samples of weather information.

Record_date	Highest temperature	Lowest temperature	Conditions	Wind direction	Wind force
2016/10/12	21	16	Overcast	West	breezy
2016/10/13	20	16	Partly Cloudy	Calm	light breeze
2016/10/14	19	17	Mostly Cloudy	West	light breeze
2016/10/15	20	16	Showers	East	breezy
2016/10/16	23	17	Overcast	NorthWest	light breeze

column represents records of an enterprise user, and there are totally 1454 users. Each row represents the records on a certain day, and the dates are sorted in ascending order. Each grid represents the electricity consumption of a specific user on a certain day, which can be expressed as $r_{ij} = \{\text{Record_date}_j, \text{Power}_i, c_{ij}\}$. The unified data sheet is shown as Table 3.

3.3.2. Clustering Users by Prototype-Based K-Means Algorithm. Clustering analysis can find locally strongly related object groups. Outlier detection can detect objects that are significantly different from most objects by detecting the discreteness of the data. Based on the characteristics of the two techniques, in this paper, we use a prototype-based clustering method to detect the degree of data discreteness, so as to learn the distribution of the data and then determine the range of k first, and next use K-means algorithm to cluster enterprise users in order to achieve enterprise groups division.

The principle of the prototype-based clustering method is to cluster all the objects first and then evaluate the degree that the objects belong to the clusters according to the distance. In traditional method, the distance between the object and the cluster center is used to measure the degree that the object belongs to the cluster. In this paper, we consider the density of data distribution and adopts the relative distance. Based on this idea, we design the clustering algorithm for enterprise groups division as shown in Algorithm 1.

We will take the Tianchi dataset as an example to illustrate the process of prototype-based K-means clustering algorithm:

First, we calculate the historical electricity consumption of each enterprise from 01/01/2015 to 10/31/2016, and cluster all the samples by using prototype-based algorithm. We take the relative distance to measure the discrete degree of samples and choose the threshold λ as 20, so the sample with relative distance greater than 20 is deemed as an outlier. Figure 3 shows the discrete points with relative distances.

Each of the points in Figure 3 is marked with a pair, which indicates the enterprise ID and its relative distance to the cluster center. There are 7 points with higher dispersion that are 1416, 175, 174, 90, 129, 1262, 1307, and 1310 from high to low. According to Figure 3, the data are generally distributing in four distance segments. User 1416 is located at the top of the figure, whose relative distance is greater than 400, much higher than the other points, and it should be classified to the first category. At the middle of the figure, 175 and 174, whose relative distances are between about 150 and 200, should be

classified to the second category. The users with ID 90, 129, 1262, 1307, and 1310, whose relative distances are between 100 to 20, should be the third category. And the remaining 1447 enterprise users, whose relative distances are below 20, should be the fourth category. So we set $k = 4$ when using K-means algorithm. And the final result of users clustering is shown as Table 4.

3.3.3. Model Selection. Model selection is the basis of next step and it is especially important. Figure 4 shows the process and rules of model selection, which mainly includes four parts.

(1) *Data Preparing.* The data now mainly include the enterprise user ID, its group ID, the history electricity consumption, and the weather data from 2015/1/1 to 2016/11/30.

(2) *Model Prejudgement Based on Periodicity/Trend and Nonlinear/Weak Stationary Analyzing*

(2a) *Periodicity/Trend Analyzing.* Periodicity analyzing is to find out whether electricity consumption of a user will change periodically with time. If a user's electricity consumption shows regular fluctuations, the increasing speed is the similar at both sides of the wave peak, and the appearance of peak is strongly related to time period, that is, $\text{Power}(n) = \text{Power}(n + T)$, where T is the time span; then we can make a prejudgement that XGBoost model would be appropriate with this character.

Trend analyzing is to find out whether change pattern of electricity consumption will show some trend. If a user's electricity consumption shows a tendency to rise or fall, or the appearance of peak changes with time and shows a relation of positive and negative correlation, then we can also consider using XGBoost model for this character.

(2b) *Nonlinear/Weak Stationary Analyzing.* If the user's electricity consumption presents irregular change or mutation or no obvious change, it shows that the sequence of electricity consumption is nonlinear. If the user's electricity consumption shows a fluctuation around a certain constant and the fluctuation is approximately in the same range, it can be seen that the mean and variance of the data are all constant, which indicates that the data has no obvious trend. If the mean value has nothing to do with the change of time, and the influence among the sequence variables is almost the same after delaying k periods, it reflects the weak stationary of data. So we can make a prejudgement that time series modeling like

TABLE 3: Unified data sheet.

Record_date	Power 1	Power 2	Power 3	Power 4	Power 5	Power i	Power 1454
2015/1/1	1135	24	385	206	156	...	976
2015/1/2	570	22	475	5134	368	...	520
2015/1/3	3418	18	526	6784	359	...	536
2015/1/4	3968	119	1	6475	467	...	488
2015/1/5	3986	108	535	6592	433	...	406
2015/1/6	4082	109	663	6742	452	...	554
2015/1/7	4172	86	606	6963	467	...	803
2015/1/8	4022	106	735	6301	495	...	975
2015/1/9	4025	103	698	6693	542	...	796

Input: Historical electricity consumption records R

Output: the group division of enterprise users $\text{ClusterSet}(\text{Power}_i)$

- (1) /* Calculate the historical total electricity consumption of each enterprise */
INPUT: $R = \{\text{Record_date}_j, \text{user}_i, c_{ij}\}$
FOR $i = 1$ To $\text{length}(\text{user}_i)$
DO
SumPower $_i = 0$
FOR $j = 1$ To $\text{length}(\text{Record_date}_j)$
DO
SumPower $_i = \text{SumPower}_i + c_{ij}$;
END
END
OUTPUT: $\{\text{SumPower}_i\}$
- (2) /* Select the clustering algorithm (K -means) to cluster all the objects and then choose $k = 1$ to cluster all the samples into a cluster and find the centroid of the cluster */
INPUT: $\{\text{SumPower}_i\}$
DO
Select $K = 1$
ClusterCenter = $K\text{means}(\text{SumPower}_i)$
END
OUTPUT: ClusterCenter
- (3) /* Calculate the relative distance */
INPUT: ClusterCenter, $\{\text{SumPower}_i\}$
FOR $i = 1$ To $\text{length}(\text{user}_i)$
DO
AbsoluteDistance $_i = \text{SumPower}_i - \text{ClusterCenter}$
MedianDistance $_i = \text{median}(\text{AbsoluteDistance}_i)$
RelativeDistance $_i = \text{AbsoluteDistance}_i / \text{MedianDistance}_i$
END
OUTPUT: $\{\text{RelativeDistance}_i\}$
- (4) Make discrete point relative distance error figure and roughly determine the range of k accord to the relative distance
- (5) /* Use K -means algorithm to do clustering for enterprises, and get the groups of enterprises according to the result of clustering */
INPUT: $\{\{\text{user}_i, \text{SumPower}_i\}\}, k$
DO
Select $K = k$
ClusterSet(user_i) = $K\text{means}(\text{user}_i)$
END
OUTPUT: ClusterSet(user_i)

ALGORITHM 1: Prototype-based K -means clustering algorithm

TABLE 4: Clustering result.

Cluster ID	Enterprise user ID	Semantic meaning
1	1416	400 < relative distance, large enterprise
2	174, 175	150 < relative distance < 200, Second large enterprise
3	90, 129, 1262, 1307, 1310	20 < relative distance < 100, medium enterprise
4	Other enterprises except 1416, 174, 175, 90, 129, 1262, 1307, 1310	0 < relative distance < 20, small enterprise

FIGURE 3: Discrete points with relative distances.

ARMA would be appropriate for such kind of data with weak stationarity.

(3) *Model Verification for XGBoost.* If XGBoost is prejudged as an appropriate choice for a user, a series of tests including Pearson correlation coefficient and feature importance analyzing would be performed before finally deciding to use XGBoost.

(3a) *Feature Generating.* Feature is the information extracted from the data that is useful in prediction. Feature extraction is mainly based on the existing background knowledge, so that the feature can play a better role in the machine learning algorithm. Based on thorough analysis of the features, we mainly extract the time and weather features as the input features.

(3b) *Correlation Analyzing.* We then use Pearson correlation coefficient test to analyze the correlation between the features and the prediction value. We use the results of feature generation and electricity consumption data for testing. If the correlation between the prediction target and some discrete features, such as weather, holidays, workday/nonworkday, is strong, it is better to use XGBoost for modeling.

(3c) *Feature Importance Analyzing.* Feature importance analysis refers to analyzing the importance relationship between

each feature and the target value and studying the influence of each feature on the change of the target. By outputting the importance of all features through the pretraining of XGBoost model, if the factors which cannot be characterized in time series, such as weather, vacation, and workday/nonworkday, have high scores, it can be determined to use XGBoost model.

If both tests have low scores, it indicates that the user's data are not suitable for XGBoost model, and we can turn to test whether they are suitable for the time series (ARMA) model.

(4) *Model Verification for ARMA.* If ARMA is prejudged as an appropriate choice for a user, or if XGBoost is not suitable, we then perform a series of tests including Pearson correlation coefficient, stationarity testing, and white noise testing to verify whether ARMA is suitable.

(4a) *Correlation Analyzing.* Same as in (3b), we test the correlation between features and the target value by using Pearson's correlation coefficient. If the correlation between the target and discrete features is low, we can consider to use the time series (ARMA) model preliminarily.

(4b) *Stationarity Testing.* Stationarity is the prerequisite for time series modeling. If the data is stationary, the fitting curve obtained through the sample time series can still inertially continue for a period of time in the future. If the data is not stationary, it indicates that the shape of the sample fitting curve does not have the characteristic of "inertia" continuation; that is, the curve fitting based on the sample time series to be obtained in the future will be different from the current sample fitting curve. To test the stationarity of data, we use the unit root for inspection. If the electricity consumption sequence data has unit root, the data is stationary.

(4c) *White Noise Testing.* White noise sequence is a stationary random sequence without any information. If the sequence is white noise, it indicates that there is no relationship between the values of sequence, and it is a purely random sequence. The autocorrelation coefficient is equal to zero, that is, $\lambda(k) = 0, k \neq 0$. If the white noise test is passed, it shows that the sequence is a non-white noise sequence.

If both stationarity and white noise testing are passed, we can determine that ARMA model is suitable for modeling the user's data.

FIGURE 4: Framework of model selection.

(5) *Mean Value Model*. If neither XGBoost nor ARMA is suitable for modeling, we then use the mean model as a final choice. The mean model takes the mean of the historical electricity consumption data as the prediction value.

We then show the process of model selection by two examples: one is a small enterprise and another one is the enterprise with ID 1307. Figure 5 shows the daily electricity consumption amount of the 1307th enterprise from 1/1/2015 to 10/31/2016, in which the x -axis is represented as the x th day in the period.

From Figure 5, we can find that the curve of the 1307th enterprise fluctuates on the mean value line from 2015 to 2016 and the fluctuation amplitudes are almost the same, which is consistent with characteristics of weak stationarity

and nonlinear. So it can be prejudged to employ ARMA for modeling.

Figure 6 shows the data of a small enterprise.

From Figure 6, it can be seen that the curve presents a cyclical fluctuation with the change of time. Through analysis, we can find that curve presents a “W” shape, it is symmetric on the 1/1/2016, and every small peak fluctuates in the unit of week. The data wave within each week is like a convex line, showing a high middle and low sides. It can be judged that the electricity consumption of small enterprises is greatly influenced by the year and week. So it can be prejudged to use XGBoost for modeling.

From the analysis above, we can find out that most enterprises are highly correlated with time features, so we can

FIGURE 5: Daily electricity consumption amount of the 1307th enterprise.

FIGURE 6: Daily electricity consumption amount of a small enterprise.

extract temporal features [48] as attributes for feature construction. Sahay et al. [49] introduced the influence of temperature to electricity consumption, so we also consider the effect of temperature. Then we use Pearson correlation coefficient to test the correlation between electricity consumption and features for enterprises from each group.

The main factors which may cause changes in electricity consumption are specifically shown in Table 5.

Figure 7 is the Pearson correlation coefficient test result of 1307th enterprise.

The correlation coefficient matrix is a symmetric matrix. The correlation between the feature and the target can be regarded as the importance of the feature, which is more important if it is closer to 1 or -1 . From Figure 7, we can find that the scores of time features for the 1307th enterprise are relatively low, so it can be inferred that its electricity consumption characteristics are irrelevant to the time features. Next we perform stationarity test and white noise test on the data of the 1307th enterprise.

Table 6 shows the test results of the 1307th enterprise.

From Table 6, after smooth processing, we can find that the p value of the unit root test statistic (0.0089) of the series is significantly less than 0.01, so the original hypothesis is strictly rejected, which judges that the series is a stationary sequence. The p value of white noise test is significantly less than 0.01, so we strictly reject the original hypothesis, which judges that the logarithmic processed series is a stationary non-white noise sequence. Combined with the prejudgement and the result of the three tests, we can determine to choose ARMA for modeling the data of the 1307th enterprise.

From Figure 8, we can find that the scores of time features for the small enterprise are relatively high, so it can be inferred that the electricity consumption characteristics of small enterprises are highly correlated with the time features. In particular, the feature “holidays” has the strongest correlation, while time series model cannot make full use of temperature and holiday features. So time series modeling is not suitable for small enterprises.

We then calculate feature importance scores for the small enterprise by pretraining in XGBoost, and the result is shown in Figure 9.

After training, the XGBoost sorts the importance of features from high to low and the result is doy, daydis, dow, maxt, mint, dom, holiday, and woy (f2, f7, f0, f9, f10, f1, f6, and f3). We can find out that small enterprises are influenced by doy, dow, daydis, maxt, and mint greatly, which is consistent with previous analysis.

Combining the prejudgement with the test of Pearson correlation coefficient and feature importance scores, we can determine that XGBoost modeling is more suitable for small enterprises.

3.3.4. Abnormal Data Processing. Data quality is crucial for the performance of models. A large number of abnormal data in the original data may lead to the deviation of the result, so it is necessary to clean the data. Missing value processing has been done in the data preprocessing, and then, in this part, we mainly perform the detection and processing of abnormal data.

In order to detect abnormal data, outlier detection is usually used to find out the values which obviously deviate from most of the samples. For some enterprises, we use pretraining modeling to mark the sudden points which deviate too much from the fitted curve as the outlier. Prototype-based clustering outlier detection method is used to detect the outliers which are deviating from the centroid obviously. This outlier detection algorithm also adopts the relative distance which is similar to Algorithm 1. And there is a little difference; that is, the outlier detection algorithm based on prototype clustering filters the outliers by using the appropriate threshold value a in the 4th step and outputs the detected outliers.

Figure 10 shows the original daily electricity consumption amount of the 175th enterprise.

It can be seen from Figure 10 that the electricity consumption of the 175th enterprise has a large difference between the first year and the second year, so segment detection will be used. We take year as a unit to carry out segment detection. Figure 11 shows the results of outlier detection based on prototype-based clustering.

FIGURE 7: Pearson correlation coefficient test result of the 1307th enterprise.

FIGURE 8: Pearson correlation coefficient test result of a small enterprise.

TABLE 5: Features.

Feature number	Feature name	Feature Description
F0	dow	Day of the week, integer, between 0 and 6
F1	dom	Day of the month, integer, between 1 and 31
F2	doy	Day of the year, integer, between 0 and 365
F3	woy	Week of the year, integer, between 1 and 52
F4	moy	Month of the year, integer, between 1 and 12
F5	weekend	Whether the day is a weekend, 0 for weekdays, and 1 for weekends
F6	holiday	Whether the day is a holiday, 0 is a holiday, 1 is not a holiday
F7	daydis	The distance between this day and the first day
F8	mondis	The distance between this month and the first month
F9	maxt	The highest temperature
F10	mint	The lowest temperature
F11	season	Season of the year, integer, between 1 and 4

TABLE 6: The test results of the 1307th enterprise.

The ADF test result	0.0089
The white noise test result	$1.3282e - 142$

FIGURE 9: The importance score of each feature for the small enterprise user.

In order to facilitate data detection, data is numbered according to the date, which is the distance from 1/1/2015 to that day. For the first year, we use threshold $a_1 = 5$; that is, the points with relative distance larger than 5 are deemed as outliers. For the second year, we use threshold $a_2 = 2.6$; that is, the points with relative distance larger than 2.6 are detected as outliers. From Figure 11, we can see that the red dots are significantly deviated from the data centroid, which are abnormal data, and green spots are normal data.

For abnormal data, we adopt different strategies for different models. If XGBoost is used, the abnormal value will be deleted directly; otherwise, if ARMA or mean value model is used, the abnormal value will be modified by the average value of three days before and after the abnormal value day.

For the 175th enterprise, since it is determined to use XGBoost model, we delete the abnormal values directly. Figure 12 shows the daily electricity consumption of the 175th

FIGURE 10: Electricity consumption amount of the 175th enterprise.

enterprise after data cleaning. From Figure 12, we can see that the curve becomes smooth after data cleaning. It is obvious that the electricity consumption pattern changes softly in the first year. And in the second year, there is an obvious small peak, showing the increase of electricity consumption.

3.3.5. Building ARMA Model. The basic idea of ARMA is, according to a stationary time series, which may be differential or logarithmic, processed to be a stationary series if necessary, a model is built to describe the stochastic process, and then the best prediction value of future time would be obtained by the built model and observed time series values.

The modeling process of ARMA is shown as Figure 13. It mainly consist of four steps.

(1) The square root test (ADF) is used to test the stationarity of the series. If the series is stationary, the white noise test will be performed. Otherwise, differential or logarithmic operation will be used to make it as a stationary series.

(2) The white noise test is performed. If the series is a stationary random series that has no information to extract, we quit the process. If the series passes the white noise test, which shows the series is a stationary non-white noise series, it can be modeled by the ARMA.

FIGURE 11: Outlier detection result (threshold $a_1 = 5$, $a_2 = 2.6$).

FIGURE 12: Daily electricity consumption of the 175th enterprise after data cleaning.

(3) Using parameter optimization, we determine p, q based on the minimum amount of information principle of BIC.

(4) Predict the electricity demand using the built AMRA model.

As we mentioned before, data of the 1307th enterprise is suitable for ARMA model. In data processing, logarithmic processing is done. The logarithmic processing can make the data smooth and make the data more stationary without changing the trend of the data. According to the result of ADF test, we judge that the series of 1307th enterprise is a stationary non-white noise series, and then we use parameter optimization to determine p, q based on the minimum amount of information principle of BIC.

The fitting result of the 1307th enterprise by ARMA is shown in Figure 14. The blue curve represents its actual electricity consumption. The red one represents the fitting line of ARMA (2, 0). From Figure 14, we can see that the prediction values of ARMA model are basically consistent, and the fitting performance is good.

FIGURE 13: Modeling process of ARMA.

3.3.6. Building XGBoost Model. An integrated process of building XGBoost model is shown as Figure 15. It mainly consists of four steps.

(1) Feature correlation testing: the correlation test is a statistical test on whether the variables are related and the degree

FIGURE 14: The fitting curve of 1307th enterprise.

FIGURE 15: XGBoost algorithm modeling process.

of correlation. We use the Pearson correlation coefficient to measure the correlation between the features. If the correlation between two features is relative high, it indicates that the linear correlation between them exists, and there must be feature redundancy.

(2) Feature importance testing: features are important to the model, but too many features can cause redundancy and overfitting. Therefore, we need to filter features. According to

the scores of feature importance, the higher the score is, the more important the features are, and the features with lower scores can be discarded.

(3) Modeling training: after processing the features, we can build the model. Choose XGBoost to train the model and use the 5-fold cross-validation method to verify the model during the training process.

(4) Predict the electricity demand using the built XGBoost model.

Take the data of a small business enterprise as an example to illustrate the process of XGBoost modeling. The data have already been cleaned. For the features listed in Table 5, we should filter features first.

Figure 16 shows the results of Pearson correlation coefficient test of features for the small enterprise. The score of correlation coefficient matrix can be regarded as the similarity between features, which are better if lower. If the correlation of two features is very high, it means that one of them is redundant. From Figure 16, we can see that *doy*, *woy*, and *moy* (*f2*, *f3* and *f4*); *daydis* and *mondis* (*f7* and *f8*); *maxt* and *mint* (*f9* and *f10*); *doy*, *woy*, and *moy* (*f2*, *f3*, *f4*); and *season* (*f11*) are highly correlated, which means there is feature redundancy. At the same time, combined with the scores of feature importance which is pretraining output of XGBoost model in Figure 9, the features retained in the end are *dow*, *dom*, *doy*, *woy*, *holiday*, *daydis*, *maxt*, and *mint*.

Figure 17 shows the fitting curve of the small enterprise after 5-fold cross-validation, in which the blue line represents the actual values and the red represents the fitting curve of the XGBoost model. It can be seen that the fitting curve of the XGBoost model is basically consistent with the actual curve.

4. Experiments

We use the electricity consumption data of 1454 enterprises in Yangzhong High-Tech Industrial Development Zone of Jiangsu Province from 2015-01-01 to 2016-11-30 for experiments. The data between 2015-01-01 and 2016-10-31 are used as training set, and the data between 2016-11-01 and 2016-11-30 are used as test set to verify the model. The experiments mainly include two parts: the parameter optimization, which can be referred to in Section 4.3, and effectiveness verification of the proposed model, which can be referred to in Sections 4.4–4.6. Before the detailed result analysis, we will introduce evaluation indicators and classical models for comparison in Sections 4.1 and 4.2, respectively.

4.1. Evaluation Indicators

(1) *MAE*. We use *MAE* for one of the indicators. *MAE* refers to the mean absolute error between the predicted values and real ones. The formula is shown as follows:

$$MAE = \frac{\sum_{i=1}^n |p_i - q_i|}{n}, \quad (2)$$

where $\{p_1, p_2, \dots, p_n\}$ refers to the prediction values and $\{q_1, q_2, \dots, q_n\}$ refers to the real ones. The smaller *MAE* value is, the more accurate the model is.

FIGURE 16: The results of the Pearson correlation coefficient test.

FIGURE 17: The fitting curve of small enterprises.

(2) *Score*. In order to measure the average deviation between the prediction values and real one, we use Score as the second indicator, and the detailed computation is shown as the following:

$$\text{Score} = \frac{1}{n} \sum_{k=1}^n e^{-3(|p_i - q_i|/q_i)}. \quad (3)$$

Score is a function to calculate relative error. The bigger the Score value is, the more accurate the model is.

4.2. Models for Comparison. We choose the following four classical algorithms for comparison.

(1) *ARMA*. This algorithm regards the data sequence which is the electricity consumption with time changes as a random sequence and uses a specific mathematics model to describe the sequence.

(2) *GBDT Model*. Features are first extracted from original data and then selected by Pearson correlation coefficient and feature importance scores. The scores are obtained by pretraining of GBDT. Finally, the prediction value is obtained by training and modeling with GBDT.

(3) *Random Forest Model*. Features are first extracted from original data and then selected by Pearson correlation coefficient and feature importance scores. The scores are obtained by pretraining of Random Forest. Finally, the prediction value is obtained by training and modeling with Random Forest.

(4) *XGBoost Model*. Features are first extracted from original data and then selected by Pearson correlation coefficient and feature importance scores. The scores are obtained by pretraining of XGBoost. Finally, the prediction value is obtained by training and modeling with XGBoost.

4.3. Parameter Optimization

Depth of Tree in XGBoost Model. The depth is the primary parameter for XGBoost model, so we first work on optimizing the depth in XGBoost model.

FIGURE 18: MAE scores with tree depth of XGBoost model changes.

Figure 18 shows the change of MAE based on XGBoost model when depth has different values. The horizontal coordinate refers to the value of depth, the vertical coordinate refers to MAE value, and curves with different colors represent different enterprises. In general, MAE becomes smaller when depth increases. But when depth is large enough, MAE will not change any more. There are problems that will cause overfitting when depth is too large, and an overly fine classification would enlarge calculation. According to Figure 18, for small enterprise, when depth is 3, the MAE is the smallest; that is, the performance is the best. And similarly, the best depths for the enterprises whose ID are 174, 175, 90, 129, and 1262 are 3, 3, 1, 2, and 2, respectively.

(p, q) in ARMA Model. For ARMA model, the most important parameters are p and q . Table 7 is the BIC information for the 1307th enterprise when p, q in ARMA (p, q) has different values.

According to the smallest amount of information principle, the best ARMA parameter is found within all the pairs of p, q . From Table 7, the pair (1, 0) has the smallest amount of information for the 1307th enterprise, so the parameter pair (1, 0) suits better for the 1307th enterprise.

4.4. Verifying the Rationality of User Clustering. In our proposed approach, we propose to cluster users first. Therefore, in this section, we aim to verify the rationality of the step. We compare the MAE and Score values for the two cases, with and without user clustering, under five models, which are ARMA, XGBoost, GBDT, Random Forest, and our proposed XGB-ARMA. The result is shown in Table 8.

From Table 8, by comparing MAE and Score values vertically, the performances of the five models except XGBoost have improved when clustering enterprise users. It proves the rationality of the step. By comparing MAE and Score values

FIGURE 19: MAE values of five models in each month.

horizontally, the performance of XGB-ARMA is the best since it has the smallest MAE and the highest Score.

4.5. Comparison of Different Models. In this section, we aim to make a detailed comparison between our proposed model and 4 classical models. Figure 19 shows how the MAE value changes with the month when we use ARMA, GBDT, Random Forest, XGBoost, or XGB-ARMA separately from January 2015 to October 2016. In Figure 19, the x -axis represents the month; y -axis represents the MAE value. Curves with different colors represent different models.

It can be seen from Figure 19 that the MAE values of XGB-ARMA are the lowest in most of the 22 months, and the MAE values of XGB-ARMA gradually decrease with time, indicating that the model is more and more stationary with the increase of time. On the other hand, it can be seen that the MAE values of various models around February (near the Spring Festival) are relatively high, indicating that the models are disturbed by the Spring Festival. Overall, XGB-ARMA outperforms other models, further demonstrating the effectiveness of the model.

4.6. Results on Test Set. In this section, we use the prediction results on test set to verify the reliability of our proposed model. Figure 20 shows the fitting curve based on XGB-ARMA Model in Nov. 2016.

In Figure 20, x -axis represents each day in Nov. 2016, while y -axis represents the power amount. Red curve represents the fitting result; blue curve represents real values. From Figure 20, we can see that fitting curve is smooth and has good generalization. It has similar tendency with the real values. The blue curve has sudden drops on 26th, 27th, and 28th, since the 1416th enterprise that takes up the 1/4 electricity consumption stopped working on the three days.

According to statistical analysis, the MAE of prediction based on XGB-ARMA in Nov. 2016 is 171641.423967, and the Score is 92.61. It proves that the model has good fitting performance. From the results, we can conclude that different models have different strengths and weaknesses when

TABLE 7: BIC information for the 1307th enterprise.

p	q					
	0	1	2	3	4	5
0	16004.1	15557.81	15391.2	15289.85	15253.54	15230.46
1	15161.09	15164.31	15170.7	15176.47	15182.02	15188.47
2	15164.25	15170.76	15177.1	15182.34	15187.98	15194.47
3	15170.76	15176.62	15182.36	15188.78	15189.81	15196.37
4	15176.3	15182.33	15188.81	15194.94	NaN	15200.08
5	15182.11	15187.76	NaN	15195.52	15206.96	15209.5

NaN: an infinite number, that is, not a number.

TABLE 8: MAE and Score values of models with or without user clustering.

Model without user clustering	ARMA	XGBoost	GBDT	Random forest	XGB-ARMA
MAE	188274.29	191378.12	257591.77	277195.20	194278.54
Score	0.8652	0.8678	0.8258	0.8158	0.8616
Model with user clustering	ARMA	XGBoost	GBDT	Random forest	XGB-ARMA
MAE	145605.80	245134.24	176533.17	187297.21	124609.62
Score	0.8954	0.8318	0.8755	0.8674	0.9085

FIGURE 20: Fitting curve based on XGB-ARMA Model in Nov. 2016.

explaining data from different angles. Some works utilize single model for prediction and therefore abandon better chance, because for some enterprises there may have better models. Different enterprises have different electricity usage patterns. It is better to choose different models based on their own characteristic rather than adopt single model. XGB-ARMA model combines the advantages of ARMA model and XGBoost model, so it can capture the changing rules of electricity consumption for different enterprises more comprehensively blessed with strengths of different models.

5. Conclusion

In this paper, we propose a XGB-ARMA model to predict short-term electricity demand by combining the advantages of XGBoost and ARMA in Fog Computing framework. It can fully utilize the storage and computing ability of Fog Nodes and achieve the mass flow and low latency requirements of smart electricity system by data preprocessing, local

computing, and real-time decision. The main contributions of this paper mainly include the following.

(1) We propose clustering enterprise users based on prototype-based K -means algorithm first, and the clustering result shows the density distribution on electricity consumption and clear semantic meaning. It is consistent with the Pareto Principle; that is, 20% of enterprise users consumes 80% of electricity energy.

(2) We propose choosing different models for different users according to the characteristic of their historical electricity consumption. A rigid model selection process is proposed, which includes model prejudgement and model determination. The prejudgement is achieved by analyzing the periodicity/trend and nonlinear/weak stationary of the historical curve, while the model determination is achieved by a series of tests, including correlation test, feature importance scores, stationary, and white noise test.

(3) Before the model building, we propose a processing strategy of abnormal data for different models. In addition, we construct a rich feature set by extending the single column of date time, such as weather, weekend, and holidays.

Future work includes the following: first, we aim to introduce local economic and population flow data to explore the influence of other factors on electricity consumption; second, we would like to explore a new method of enterprise users clustering which can classify users according to data distribution and different premodeling results; third, we would like to employ visualization techniques [50] in the presentation of our solution.

Data Availability

The data used to support the findings of this study are provided by Tianchi under license and so cannot be made freely

available. Access to these data will be considered by the author upon request, with permission of Tianchi.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

The research is supported by National Natural Science Foundation of China (no. 61772560), Natural Science Foundation of Hunan Province (no. 2016JJ3154), Scientific Research Project for Professors in Central South University, China (no. 904010001), and Innovation Project for Graduate Students in Central South University (no. 1053320181628).

References

- [1] L. Qi, X. Zhang, W. Dou, and Q. Ni, "A distributed locality-sensitive hashing-based approach for cloud service recommendation from multi-source data," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2616–2624, 2017.
- [2] L. Qi, X. Xu, X. Zhang et al., "Structural balance theory-based e-commerce recommendation over big rating data," *IEEE Transactions on Big Data*, 2016.
- [3] I. Stojmenovic and S. Wen, "The fog computing paradigm: scenarios and security issues," in *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS '14)*, pp. 1–8, IEEE, Warsaw, Poland, September 2014.
- [4] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, "Fog computing: A platform for internet of things and analytics," *Studies in Computational Intelligence*, vol. 546, pp. 169–186, 2014.
- [5] R. Mahmud, R. Kotagiri, and R. Buyya, "Fog Computing: A Taxonomy, Survey and Future Directions," in *Internet of Everything, Internet of Things*, pp. 103–130, Springer Singapore, Singapore, 2018.
- [6] L. Kuang, Y. Wang, P. Ma et al., "An Improved Privacy-Preserving Framework for Location-Based Services Based on Double Cloaking Regions with Supplementary Information Constraints," *Security and Communication Networks*, vol. 2017, pp. 1–15, 2017.
- [7] L. Kuang, Z. Liao, W. Feng, H. He, and B. Zhang, "Multimedia services quality prediction based on the association mining between context and QoS properties," *Signal Processing*, vol. 120, pp. 767–776, 2016.
- [8] Y. J. Wang, X. Z. Ji, and J. M. Shi, "Scenario analysis and application research on big data in smart power distribution and consumption systems," *Proceedings of the CSEE*, vol. 35, no. 8, pp. 1829–1836, 2015.
- [9] A. E. Clements, A. S. Hurn, and Z. Li, "Forecasting day-ahead electricity load using a multiple equation time series approach," *European Journal of Operational Research*, vol. 251, no. 6, pp. 522–530, 2016.
- [10] M. Bessec, J. Fouquau, and S. Meritet, "Forecasting electricity spot prices using time-series models with a double temporal segmentation," *Applied Economics*, vol. 48, no. 5, pp. 361–378, 2016.
- [11] F. Yasmeen and M. Sharif, "Functional Time series (FTS) Forecasting of Electricity Consumption in Pakistan," *International Journal of Computer Applications*, vol. 124, no. 7, pp. 15–19, 2015.
- [12] V. Bianco, O. Manca, and S. Nardini, "Electricity consumption forecasting in Italy using linear regression models," *Energy*, vol. 34, no. 9, pp. 1413–1421, 2009.
- [13] F. Zhang, H. Wang, T. Han et al., "Short-term load forecasting based on partial least-squares regression," *Power System Technology*, vol. 3, pp. 36–40, 2003.
- [14] W.-C. Hong, Y. Dong, W. Y. Zhang, L.-Y. Chen, and B. K. Panigrahi, "Cyclic electric load forecasting by seasonal SVR with chaotic genetic algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 44, no. 1, pp. 604–614, 2013.
- [15] K. Kavaklioglu, "Modeling and prediction of Turkey's electricity consumption using support vector regression," *Applied Energy*, vol. 88, no. 1, pp. 368–375, 2011.
- [16] C.-C. Hsu, C.-H. Wu, S.-C. Chen, and K.-L. Peng, "Dynamically optimizing parameters in support vector regression: An application of electricity load forecasting," in *Proceedings of the 39th Annual Hawaii International Conference on System Sciences, HICSS'06*, p. 30, USA, January 2006.
- [17] H. S. Hippert, E. C. Pedreira, and C. R. Souza, "Neural networks for short-term load forecasting: a review and evaluation," *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, 2001.
- [18] J. P. S. Catalão, H. M. I. Pousinho, and V. M. F. Mendes, "Hybrid wavelet-PSO-ANFIS approach for short-term electricity prices forecasting," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 137–144, 2011.
- [19] Y. T. Chae, R. Horesh, Y. Hwang, and Y. M. Lee, "Artificial neural network model for forecasting sub-hourly electricity usage in commercial buildings," *Energy and Buildings*, vol. 111, pp. 184–194, 2016.
- [20] A. G. Bakirtzis, J. B. Theoharis, S. J. Kiartzis, and K. J. Satsios, "Short term load forecasting using fuzzy neural networks," *IEEE Transactions on Power Systems*, vol. 10, no. 3, pp. 1518–1524, 1995.
- [21] B. W. Tao, Z. L. Zhang, H. Pan et al., "Spatial electric load forecasting based on double-level Bayesian classification," in *Proceedings of the Chinese Society of Electrical Engineering*, vol. 27, pp. 13–17, 2007.
- [22] T. Niimura and T. Nakashima, "Deregulated electricity market data representation by fuzzy regression models," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 31, no. 3, pp. 320–326, 2001.
- [23] A. Deihimi, O. Orang, and H. Showkati, "Short-term electric load and temperature forecasting using wavelet echo state networks with neural reconstruction," *Energy*, vol. 57, pp. 382–401, 2013.
- [24] T.-Y. Zhu, Y.-Q. Li, Y. Zhang, X.-Z. Zhang, and C.-Y. He, "New algorithm of advancing weather adaptability based on ARIMA model for day-ahead power load forecasting," *Proceedings of the Chinese Society of Electrical Engineering*, vol. 26, no. 23, pp. 14–19, 2006.
- [25] A. Bogomolov, B. Lepri, R. Larcher, F. Antonelli, F. Pianesi, and A. Pentland, "Energy consumption prediction using people dynamics derived from cellular network data," *EPJ Data Science*, vol. 5, no. 1, article no. 13, 2016.
- [26] Y. Han, X. Sha, E. Grover-Silva, and P. Michiardi, "On the impact of socio-economic factors on power load forecasting," in *Proceedings of the 2nd IEEE International Conference on Big Data, IEEE Big Data 2014*, pp. 742–747, USA, October 2014.
- [27] L. Ghelardoni, A. Ghio, and D. Anguita, "Energy load forecasting using empirical mode decomposition and support vector regression," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 549–556, 2013.

- [28] J. Che, J. Wang, and G. Wang, "An adaptive fuzzy combination model based on self-organizing map and support vector regression for electric load forecasting," *Energy*, vol. 37, no. 1, pp. 657–664, 2012.
- [29] P. Regulski, D. S. Vilchis-Rodriguez, S. Djurović, and V. Terzija, "Estimation of Composite Load Model Parameters Using an Improved Particle Swarm Optimization Method," *IEEE Transactions on Power Delivery*, vol. 30, no. 2, pp. 553–560, 2015.
- [30] I. F. Visconti, D. A. Lima, J. M. C. De Sousa Costa, and N. R. D. B. C. Sobrinho, "Measurement-based load modeling using transfer functions for dynamic simulations," *IEEE Transactions on Power Systems*, vol. 29, no. 1, pp. 111–120, 2014.
- [31] A. U. Haque, P. Mandal, J. Meng, A. K. Srivastava, T.-L. Tseng, and T. Senjyu, "A novel hybrid approach based on wavelet transform and fuzzy artmap networks for predicting wind farm power production," *IEEE Transactions on Industry Applications*, vol. 49, no. 5, pp. 2253–2261, 2013.
- [32] N. Sovann, P. Nallagownden, and Z. Baharudin, "Electricity load forecasting using hybrid wavelet neural network based on parallel prediction method," in *Proceedings of the 6th International Conference on Intelligent and Advanced Systems, ICIAS 2016*, Malaysia, August 2016.
- [33] G.-X. Li, "Electricity consumption forecast based on wavelet neural network," in *Proceedings of the 2016 International Conference on Information System and Artificial Intelligence, ISAI 2016*, pp. 361–364, China, June 2016.
- [34] T. Nengling, J. Stenzel, and W. Hongxiao, "Techniques of applying wavelet transform into combined model for short-term load forecasting," *Electric Power Systems Research*, vol. 76, no. 6–7, pp. 525–533, 2006.
- [35] N. M. Pindoriya, S. N. Singh, and S. K. Singh, "An adaptive wavelet neural network-based energy price forecasting in electricity markets," *IEEE Transactions on Power Systems*, vol. 23, no. 3, pp. 1423–1432, 2008.
- [36] P. K. Pany and S. P. Ghoshal, "Dynamic electricity price forecasting using local linear wavelet neural network," *Neural Computing and Applications*, vol. 26, no. 8, pp. 2039–2047, 2015.
- [37] J. Che and J. Wang, "Short-term electricity prices forecasting based on support vector regression and Auto-regressive integrated moving average modeling," *Energy Conversion and Management*, vol. 51, no. 10, pp. 1911–1917, 2010.
- [38] X. Cheng, C. Kang, Q. Xia, and Y. Shen, "Integrated model of short term load forecasting," *Automation of Electric Power Systems*, vol. 24, no. 9, pp. 42–44, 2000.
- [39] G. Chen, K. Li, T. Chung, H. Sun, and G. Tang, "Application of an innovative combined forecasting method in power system load forecasting," *Electric Power Systems Research*, vol. 59, no. 2, pp. 131–137, 2001.
- [40] C. Kang, X. Cheng, Q. Xia, Y. Huang, and F. Gao, "Novel approach considering load-relative factors in short-term load forecasting," *Electric Power Systems Research*, vol. 70, no. 2, pp. 99–107, 2004.
- [41] J. Wang, S. Zhu, W. Zhang, and H. Lu, "Combined modeling for electric load forecasting with adaptive particle swarm optimization," *Energy*, vol. 35, no. 4, pp. 1671–1678, 2010.
- [42] G. Elliott and A. Timmermann, "Optimal forecast combinations under general loss functions and forecast error distributions," *Journal of Econometrics*, vol. 122, no. 1, pp. 47–79, 2004.
- [43] Y. Yao, Z. Lian, S. Liu, and Z. Hou, "Hourly cooling load prediction by a combined forecasting model based on analytic hierarchy process," *International Journal of Thermal Sciences*, vol. 43, no. 11, pp. 1107–1118, 2004.
- [44] R. K. Rietz and S. Suryanarayanan, "A review of the application of analytic hierarchy process to the planning and operation of electric power microgrids," in *Proceedings of the 40th North American Power Symposium, NAPS2008*, Canada, September 2008.
- [45] V. Petridis, A. Kehagias, L. Petrou et al., "A Bayesian multiple models combination method for time series prediction," *Journal of Intelligent & Robotic Systems*, vol. 31, no. 1–3, pp. 69–89, 2001.
- [46] M. Peng, S. Yan, K. Zhang, and C. Wang, "Fog-computing-based radio access networks: Issues and challenges," *IEEE Network*, vol. 30, no. 4, pp. 46–53, 2016.
- [47] A. V. Dastjerdi and R. Buyya, "Fog Computing: Helping the Internet of Things Realize Its Potential," *The Computer Journal*, vol. 49, no. 8, Article ID 7543455, pp. 112–116, 2016.
- [48] G. Barta, G. B. G. Nagy, S. Kazi, and T. Henk, "GEFCOM 2014—probabilistic electricity price forecasting," *Smart Innovation, Systems and Technologies*, vol. 39, pp. 67–76, 2015.
- [49] K. B. Sahay, N. Kumar, and M. M. Tripathi, "Short-term load forecasting of Ontario Electricity Market by considering the effect of temperature," in *Proceedings of the 6th IEEE Power India International Conference, PIICON 2014*, ind, December 2014.
- [50] Z. Liao, L. Kong, X. Wang et al., "A visual analytics approach for detecting and understanding anomalous resident behaviors in smart healthcare," *Applied Sciences (Switzerland)*, vol. 7, no. 3, article no. 254, 2017.