Household Power Consumption Prediction Method Based on Selective Ensemble Learning

KUN LIANG*, FEI LIU*, AND YIYING ZHANG*
College of Artificial Intelligence, Tianjin University of Science and Technology, Tianjin 300457, China
Corresponding author: Fei Liu (1006124568@qq.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61807024.

ABSTRACT In the context of power big data, the household power consumption data on the user side has the characteristics of large quantity, wide distribution and many types. Ensemble learning is very excellent in the analysis and mining of power data with large amount of data, strong timeliness and many influencing factors. Based on the data of household power consumption, this paper analyzes and predicts the power consumption of some users in a city by using selective ensemble learning technology and combining with meteorological factors. In this paper, K-means algorithm is used to cluster the household power consumption, and then the clustering results are combined with the meteorological information. In the stage of power consumption prediction, a Filter Iterative Optimization Ensemble Strategy (FIOES) is proposed to selectively ensemble the basic learners and get the final prediction model. Experimental results show that the FIOES algorithm has better performance in time cost and prediction accuracy than the traditional ensemble learning algorithm.

INDEX TERMS Power big data, ensemble learning, power consumption, iterative optimization.

I. INTRODUCTION
In the context of smart grid, the power big data, especially residential power consumption data, has potentially valuable information. How to use these massive data is of great significance to promote production, improve services, and ensure grid security. The power information collection equipment on demand side collects a large amount of data, which constitutes the marketing side big data in the smart grid. The information structure in power big data is very complicated. There are not only structured data, but also many semi-structured and unstructured data. Therefore, the processing of power big data is currently a difficult problem.

The power system consists of five parts: generation, transmission, transformation, distribution and consumption [1]. Power load prediction [2] is to analyze the historical data of power load and study the influence of related external factors, so as to establish a mathematical model and estimate the demand of the power system. It is mainly divided into short-term load prediction, medium-term load prediction and long-term load prediction [3]. User power consumption prediction is part of load prediction, and its essence is to analyze and model the user’s historical power consumption information and predict the power consumption in the next stage. There are many influencing factors of power load prediction, mainly including time-space factors [4], meteorological factors and so on. Due to the wide geographical distribution of household users, this time-space factor can also be considered in consumption prediction, so as to be different from the centralized prediction of some pilot measurement nodes and better adapt to the needs of customized load analysis. Because the data used in this paper is used by users in a certain province and city for power data, the influence of time and space factors is not taken into account when prediction power consumption. In this paper, the temperature, precipitation and sunshine time of the area are taken into account when predicting the power consumption.

In this paper, the prediction on stage is mainly medium-term load prediction. We collected the power consumption information of home users in a certain city for one year as a data source. Considering the important factors that affect the residents’ power consumption, such as average temperature, monthly precipitation, and sunshine statistics, users are clustered based on their historical power consumption. Using neural network and selective integrated learning method, the user power consumption is predicted. This paper proposes a method for home users’ power consumption prediction based on neural networks and selective ensemble
learning [5], [6]. This method provides necessary reference for power grid planning and transformation.

II. RELATED WORK

Power big data mining is mainly carried out from the prediction of power data [7], [8]. Machine learning is widely used in classification and prediction problems. By training a large number of data, we can find the rules contained in the training data, and then use these learner models to classify and predict the new data. Generalization ability has always been one of the important criteria to evaluate a machine learning algorithm model. It means that a machine learning system should be able to process new data well according to the model trained by existing data.

In the power big data processing, because of the complexity of data and the particularity of semi-structured and unstructured, the effect of a single learner is not very ideal. The research of scholars shows that ensemble learning can effectively improve the generalization ability of machine learning system. Ensemble learning is to get several classifiers (called basic learners) in the process of training samples, and then combine these basic learners in a certain way, so that multiple basic learners work together to solve a learning task. For the combination of basic learners, the traditional methods are generally as follows: majority voting method [9], weight voting method [10], hierarchical combination method [11], etc. These integrated learning algorithms can effectively solve all kinds of classification and prediction problems, but they still have shortcomings. For example, although they have good generalization performance, they are still unsatisfactory in some aspects of performance and efficiency. In reference [12], when the amount of data is too large, although it is relatively simple in the construction of various learners, but in the process of ensemble, the hardware memory and computing capacity of the server system are required to be high. In the earliest ensemble learning, all the basic learners are used to combine the learners. This method maybe ensemble the learners with poor performance, which will affect the ensemble effect of the model. Selective ensemble learning can solve this problem well. During ensemble stage, only the learners with better performance are selected to avoid the impact of the basic learners with poor performance on the ensemble model. The process of selective ensemble learning is shown in Fig. 1.

The main idea of selective ensemble learning [13] is to select only a part of the learners with better performance among many base learners, so as to obtain the better effect than that of ensemble all base learners [14]. In the search process of the basic learner, the method used is very important, which has a great impact on the ensemble results. At present, there are four methods of selective ensemble: iterative optimization, ranking, clustering and pattern mining. In reference [15], the author summarizes and analyzes the existing selective integration learning methods, and points out that scholars first adopt the method of selective integration according to the accuracy. With the deepening of research, GASSEN and the improved selective integrated learning algorithm are proposed by researchers. This algorithm adopts the combination of voting method and genetic algorithm. Firstly, voting method is used to select the part of the learners first, and then genetic algorithm is used to search the expected basic learners.

1) RANKING METHOD

Ranking is the most common selective ensemble algorithm [16]. Its main idea is to sort the basic learners according to their performance. Generally, the accuracy rate is used. After ranking, only the top parts of the basic learners are selected for ensemble to improve the performance of the overall model. The main advantage of ranking method is that its selection speed is relatively fast, the cost in the ensemble stage is relatively small, and it is suitable for small-scale ensemble learning. The common ranking based selective ensemble learning algorithms are: direction-based ranking, boundary-based ranking, kappa coefficient-based ranking [17], etc., which all show good performance in the ensemble stage.

However, most of the ranking algorithms only consider the performance of the prediction results when selecting the base learners, which is one-sided. If the performance of multiple base learners is good, but the difference is large, the integration model has good prediction performance, but its generalization ability has not been significantly improved. In the selection of basic learners, the relevance and difference between each basic learner are not considered, so the ranking method has some defects.

2) ITERATIVE OPTIMIZATION METHOD

The main idea of iterative optimization method [18] is to use a specific optimization method when finding the optimal solution, and then optimize the target value according to the result of each iteration. The advantage of iterative optimization is that it avoids the problem of too large data combination. By introducing the optimization method, the optimization is carried out step by step until the optimal combination of the basic learners is obtained, so that the optimal solution is obtained when the basic learners are selected.

At present, the commonly used iterative optimization methods include hill climbing method, GASSEN algorithm, EPRL algorithm and SDP algorithm [19]–[22]. Hill climbing
is the most widely used algorithm in iterative optimization. It is a process of gradually finding the optimal solution for the selection of each basic learner, which is similar to hill climbing. In the search of the optimal solution, the results will gradually decrease and the search speed will gradually increase. In the process of searching for the optimal solution, it will keep approaching to the optimal solution to obtain the optimal combination. Hill climbing is the most widely used algorithm in iterative optimization. It is a process of gradually finding the optimal solution for the selection of each basic learner, which is similar to hill climbing. In the search of the optimal solution, the results will gradually decrease and the search speed will gradually increase. In the process of searching for the optimal solution, it will keep approaching to the optimal solution to obtain the optimal combination. Because of its fast speed and good performance, it is one of the most commonly used algorithms. GASEN algorithm mainly uses genetic algorithm to optimize the combination of base learners. In the process of optimization, the weight of each basic learner is divided, and the final integrated model has better prediction performance. EPRL algorithm uses reinforcement learning method in iterative optimization. At the beginning of iteration, a decision function is introduced, and then each basic learner is optimized according to the function, so as to get the optimal ensemble model. SDP algorithm is to transform the selection problem of each basic learner into a quadratic integer problem. In order to get the optimal solution, the integer programming method is used to select the basic learner.

III. ANALYSIS OF POWER USER CHARACTERISTICS

A. POWER DATA AND PREPROCESSING

The power data used in this paper is part of the power consumption information from residents of a certain city provided by the State Grid Corporation of China (SGCC). The source data mainly includes field information such as Electricity Meter Box ID (EM_BoxID), User ID (UserID), Collection Time (ColTime), Power Consumption (PowCsp), Latitude and Longitude (Lat_Lon), and Weather (Weather). The power management method for residents in this city is: the meter box is a first-level unit, which is connected to multiple users, and each user has its own unique User ID. In general, the collection time is once a month, the Power Consumption is the user’s current power consumption for this month. The original data includes 9,374 meter boxes, more than 100,000 connected users, and a total of 2.73 million records of data power consumption information collected for 14 months from January 2018 to February 2019.

1) MISSING VALUE PROCESSING

The main research object of this article is the monthly power consumption information of home users, and the problem of missing values in the “Power Consumption” field needs to be considered. During the clustering process, adding and deleting a large amount of data has a great impact on the clustering results, so we analyze and process the data from the following three cases:

1. If the user’s Monthly average power consumption is less than 30 kW·h, the household is likely to be a vacant house. This kind of data has little meaning to the clustering results, so it can be eliminated.

2. If the user’s power consumption information is missing for 1-3 months, fill it with the average value.

3. If the month with missing values is greater than 4 months, it is eliminated.

We use the tool software Tableau [23] for data processing. Through the field information, we find the user ID of the power user with the missing value. According to the analysis, there are 3541 users and 2532 related meter boxes whose power consumption is 0 every month. There are 492 users with a missing value of 1-3 months and 277 related meter boxes. There are 49 users with a missing value of more than 4 months and 36 related meter boxes. The above detailed data information is shown in Table 1.

| Case | Calculated field | Number of meter boxes | Number of users | processing method |
|------|------------------|-----------------------|----------------|------------------|
| (1)  | Avg(PowCsp)<30   | 2532                  | 3541           | Eliminated      |
| (2)  | Count(PowCsp =NULL)<4 | 277                    | 492            | Filled average  |
| (3)  | Count(PowCsp =NULL)>4 | 36                     | 49             | Eliminated      |

2) OUTLIER PROCESSING

Statistical discrimination is a method of dealing with outliers. This paper mainly uses the Box-plot method [24] to deal with outliers in the original data. The Box-plot method is shown in Fig 2. Its main processing is as follows:

1. Calculate the upper quartile, the median, and the lower quartile;

2. Calculate the difference between the upper and lower quartile, that is, the quartile difference;

3. Draw the upper and lower ranges of the boxplot. The upper limit is the upper quartile and the lower limit is the

![Box plot schematic diagram](image-url)
lower quartile. Draw a horizontal line at the median position inside the box;

(4) Values greater than 1.5 times the quartile difference in the upper quartile, or values less than 1.5 times the quartile difference in the lower quartile are classified as outliers;

(5) In addition to the outliers, draw the horizontal line at the two values closest to the upper and lower edges as the whisker of the boxplot.

The boxplot method can be used to clearly identify outliers in the data set, which plays a role in data cleaning. In addition, it can also observe the data distribution. Firstly, users whose monthly power consumption is less than 30 kW·h are searched with tableau tool, and these data are eliminated. And then, we used Tableau to make a boxplot of power consumption, setting the upper and lower boundaries to 377 and 31. After analysis, it was found that there were 56339 power consumption records exceeding this limit, the number of related meter boxes was 1,682, and the number of users was 8,943. Among them, 743 records have increased power consumption, setting the upper and lower boundaries to 377 and 31. After analysis, it was found that there were 56339 power consumption records exceeding this limit, the number of related meter boxes was 1,682, and the number of users was 8,943. Among them, 743 records have increased power consumption by more than 60% compared with the neighboring months. If there are errors in statistics, they are filled with the average value of neighboring months. Another 55596 records with higher power consumption are not outliers.

### B. POWER CONSUMPTION CLUSTERING

#### RESULTS AND ANALYSIS

1) CLUSTERING PROCESS

In this paper, K-means algorithm [25] is used to cluster the data set. When clustering, we should first determine the initial cluster centroid and select \( k = 3 \). The setting of initial cluster center value is shown in Table 2.

Power consumption (PowCsp) represents the power used by the user for a month.

The change of power consumption (PowCsp_Chg) refers to the difference in power consumption between two consecutive months. The calculation method is as shown in formula (1):

\[
\text{PowCsp}_{\text{Chg}} = \text{PowCsp}_i - \text{PowCsp}_{i-1}
\]

Power Consumption Change Rate (PowCsp_ChgRt) is the change of power consumption divided by power consumption last month. The calculation method is as shown in formula (2):

\[
\text{PowCsp}_{\text{ChgRt}} = \frac{\text{PowCsp}_i - \text{PowCsp}_{i-1}}{\text{PowCsp}_i}
\]

In this paper, the clustering is the user’s power consumption data, that is, numeric data. For this data, the K-means algorithm is used to perform clustering analysis on the data source. The clustering center is first determined when clustering, according to the square error criterion:

\[
\min E = \min \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|
\]

In formula (3), \( E \) is the integrated square error of all samples in the data source, \( p \) represents the monthly power consumption, and \( m_i \) is the average value of the cluster \( C_i \). This criterion ensures that the generated clustering results are as compact and independent as possible.

In the K-means algorithm, the type of the sample point is divided by the distance from the current sample point to the center of the class, and the distance is calculated using the Euclidean distance, as shown in formula (4):

\[
\text{EUCLID} (x, y) = \sqrt{\sum (x_i - y_i)^2}
\]

In equation (4): \( x_i \) represents the value of the i-th variable in the sample, and \( y_i \) represents the value of the i-th variable in the clustering center. After subtracting the two, the squares are accumulated, and then the square can be obtained to obtain its Euclidean distance.

The specific process of K-means clustering algorithm is:

1) Select the number of cluster centers required for the experiment;
2) Calculate the Euclidean distance from the data sample to the cluster center according to formula (4), and divide it according to its distance;
3) Then recalculate each cluster center according to formula (3);
4) Repeat steps (2) and (3) until the position of each cluster center no longer changes;
5) Output results.

According to the above steps, the experiment first selected three clustering centers, and their initial settings are shown in Table 2.

According to the steps of K-means algorithm, the clustering centroid is calculated and the users are clustered. After 474 iterations, the position of cluster centroid no longer changes, and the iteration stops. The final clustering centroid iteration results are shown in Table 3.

The clustering results are as follows: Class I users have the largest power consumption, with a total of 16,784 households, and their monthly average power consumption is 563 kW·h. Followed by Class II users, a total of 53,167 households, and their average monthly power consumption is 277 kW·h. Class III users is the smallest, with a total of 26459 households, and their monthly average power consumption is 103 kW·h.

#### TABLE 2. Initial cluster centroid.

| Unit (kW·h) | Cluster category |
|-------------|-----------------|
|             | Cl I | Cl II | Cl III |
| PowCsp      | 800  | 350   | 31     |
| PowCsp_Chg  | 200  | -50   | -1     |
| PowCsp_ChgRt| 0.3278 | -0.2000 | -0.2000 |

#### TABLE 3. Cluster centroid iteration results.

| Unit (kW·h) | Cluster category |
|-------------|-----------------|
|             | Cl I | Cl II | Cl III |
| PowCsp      | 563  | 277   | 103    |
| PowCsp_Chg  | 37   | 3     | 2      |
| PowCsp_ChgRt| 0.0914 | -0.0373 | -0.0736 |
2) CLUSTERING CATEGORY ANALYSIS

We use the boxplot to analyze the above clustering results. Fig 3 is a boxplot of power consumption, which clearly shows the number of users and power consumption in three categories.

Class I users: the upper boundary is 714 kW·h, and the lower boundary is 492 kW·h, with 328 users out of range.

Class II users: the upper boundary is 453 kW·h, and the lower boundary is 237 kW·h, with 453 users out of range.

Class III users: the upper boundary is 213 kW·h, and the lower boundary is 31 kW·h, with 33 users out of range.

Taken together, the out-of-bounds users account for 1.95% of total users in Class I. The out-of-range users accounted for 0.85% of total users in Class II. The above two categories of clustering results are better. The out-of-range users accounted for 0.12% of the total class III users, and the clustering effect is the best.

3) CLUSTERING RESULTS ANALYSIS

The relationship between power consumption and the change of power consumption can be obtained by clustering results, as shown in Fig 4. Class I users have the largest range and fluctuation in the change of power consumption, and even extreme changes of −54 kW·h to 31 kW·h have occurred. The range of change for Class II users is relatively modest, and it is basically maintained within ±19 kW·h, with most positive values, and this category of users is relatively stable. Class III users have the smallest change in power consumption, which is in a relatively low range and relatively stable.

The relationship between power consumption and the change rate of power consumption can be obtained by clustering results, as shown in Fig 5. Class I users have the largest power consumption and the smallest range of change rates. It ranges from −9% to 5% and is the most stable class. The power consumption of category II users is second, and the range of change rate is also relatively small, ranging from −5% to 14%, which is relatively stable; The power consumption of Class III users is the least, but its range of change rate is the largest, between −17% and 22%.

Through clustering analysis, we summarize the power consumption characteristics of the three categories, as shown in Table 4.

Fig 6 shows the monthly average power consumption of the three categories between January 2018 and February 2019. Figure 5 shows that:

| CAT | PowCs | PowCap_C | PowCap_C | PowCap_C | Range | Out of | Ratio |
|-----|-------|----------|----------|----------|-------|-------|-------|
| I   | large | large    | small    | large    | 492-714 | 1.95% |
| II  | medium| medium   | medium   | medium   | 237-415 | 0.85% |
| III | small | small    | large    | small    | 31-213  | 0.12% |
IV. POWER CONSUMPTION PREDICTION BASED ON SELECTIVE ENSEMBLE LEARNING

Based on the clustering of power consumption in the previous section, this paper adopted the theory of selective ensemble learning to establish a selective ensemble prediction model to predict the power consumption of home power users. We propose a Filtering Iterative Optimization Ensemble Strategy (FIOES), which uses the advantages of the Ranking method to optimize the traditional Iterative Optimization method, trains multiple base learners, and each base learner makes predictions through the Neural Network [26]–[28]. Experiments have proved that the model has a good performance for home users’ power consumption prediction.

A. ANALYSIS OF METEOROLOGICAL FACTORS

Meteorological factor is one of the important factors that affect the household power consumption, which has a very important significance for the accurate prediction of power consumption. In the meteorological data, the influence of temperature, precipitation and sunshine time on power consumption is very obvious. Therefore, the above three aspects are the main weather factors to be considered in the prediction of household power consumption. The temperature has the most obvious effect on the power consumption. According to the data in the previous section, residents use the most power in summer every year. The main reason is that residents use high-power electrical appliances such as air conditioning for a long time in summer, resulting in a sharp increase in power consumption, reaching the peak of the year. Because this is a northern city, during the heating period in winter, the power demand of residents is relatively less. The impact of precipitation on power consumption varies with the month. Some studies show that the power consumption of some months increases with the increase of precipitation, and decreases with the decrease of precipitation [29]. The influence of sunshine time on household power consumption is also related to the month. The sunshine intensity in different months affects the household power consumption differently.

The meteorological data in this paper are from China Weather Network. The collected data are the monthly average temperature, monthly precipitation and monthly sunshine time of the province. The specific data after processing is shown in Table 5.

| Month | Monthly average temperature (°C) | Precipitation (mm) | Sunshine time (h) |
|-------|----------------------------------|--------------------|------------------|
| 1     | 3.3                              | 3.3                | 89.4             |
| 2     | -0.7                             | 4.0                | 107.6            |
| 3     | 5.8                              | 7.7                | 118.2            |
| 4     | 11.2                             | 20.9               | 125.8            |
| 5     | 21.2                             | 37.6               | 183.3            |
| 6     | 24.4                             | 41.1               | 217.4            |
| 7     | 26.2                             | 170.6              | 220.1            |
| 8     | 24.9                             | 145.7              | 214.5            |
| 9     | 20.0                             | 46.1               | 147.3            |
| 10    | 13.4                             | 22.7               | 97.8             |
| 11    | 4.6                              | 10.4               | 64.3             |
| 12    | -1.5                             | 4.1                | 53.6             |

Table 5 shows the average temperature, precipitation and sunshine time of the city in 2018. The average temperature of the whole year is 13.9 °C, in which the average temperature in January is the lowest, and the average temperature in August is the highest. The total precipitation of the whole year is 474.2mm, the month with the most precipitation is July, and the month with the least precipitation is January. Import the sorted data into tableau, merge with the original household power consumption data and clustering results data, and lay the data foundation for the subsequent power consumption prediction.

B. CONSTRUCTION OF BASIC LEARNERS

In the construction of the basic learner, the weather data and power data are combined, and the Multilayer Perceptron (MLP) is used to predict the power consumption. MLP is an Artificial Neural Network (ANN) with multiple hidden layers. Due to different data sets, the number of hidden layers can be modified. The structure of neural network is shown in Figure 7, which contains two hidden layers. The first layer is the input layer, which is generally an n-dimensional vector, such as: $x_1, x_2, \ldots, x_7$. They are respectively expressed as: clustering result, power consumption change, monthly average temperature, distance to the clustering centroid, power consumption change rate, precipitation and sunshine time. The second and third layers are hidden layers, the second layer is 8 neurons, the third layer
is 6 neurons, and the output of the second layer in Fig 7 is:

$$y_2^{(z)} = f\left(\sum_{i=1}^{n} \omega_2^{(z)} x_i + b_2^{(z)}\right) = f\left(\sum_{i=1}^{n} \omega_2^{(z)} x_i + b_2^{(z)}\right)$$

where $z = \{1, 2, \ldots, 8\}$.

After transformation, we can get:

$$y_2 = f\left(\begin{bmatrix} \omega_2^{(1)} & \omega_2^{(2)} & \cdots & \omega_2^{(8)} \\ \omega_2^{(1)} & \omega_2^{(2)} & \cdots & \omega_2^{(8)} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_2^{(1)} & \omega_2^{(2)} & \cdots & \omega_2^{(8)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_8 \end{bmatrix} + \begin{bmatrix} b_2^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_2^{(8)} \end{bmatrix}\right) = f(W_2x + b_2)$$

(6)

By analogy, the calculation process of this propagation can be extended to any layer, and the rules are as follows:

$$y_1^{(j)} = f\left(\begin{bmatrix} u_1^{(j)} \\ u_1^{(j)} \\ \vdots \\ u_1^{(j)} \end{bmatrix}\right) = f\left(\begin{bmatrix} \omega_1^{(j)} & \omega_1^{(j)} & \cdots & \omega_1^{(j)} \end{bmatrix} \begin{bmatrix} y_{l-1} \\ y_{l-1} \\ \vdots \\ y_{l-1} \end{bmatrix} + b_1^{(j)}\right)$$

$$y_l = f\left(W_ly_{l-1} + b_l\right)$$

(7)

where $l$ represents the number of neurons in the current layer, $y_l$ represents its output. $u_l^{(j)}$ represents the input of node $j$, $y_l^{(j)}$ represents the output of node $j$, $W_l$ is the weight matrix between two layers, $\omega_l^{(j)}$ is the weight from the $i$-th node in the previous layer to the $j$-th node in the current layer, $b_1$ represents the offset of the node, $f$ is the activation function.

C. FILTERING ITERATIVE OPTIMIZATION ENSEMBLE STRATEGY

1) SELECTION OF BASIC LEARNER

FIOES uses a selective ensemble method that combines the ranking method [30], [31] with the iterative optimization method. The main steps are as follows:

Step 1: During the iterative optimization, the base learner is selected by the Ranking method, and some base learners with poor performance are filtered in advance.

Step 2: The unfiltered base learner is integrated using an iterative optimization method until the iteration is within a set threshold.

Step 3: The remaining base learners after the iteration are selected by the ranking method for the second filtering and integrated.

When filtering all the basic learners, the ranking method is used to sort them. Here, kappa coefficient method is adopted, and kappa coefficient is used to preliminarily filter each basic learner. The filtering process is shown in formula (8):

$$k_i = \frac{p_0 - p_i}{1 - p_i}$$

(8)

where, $p_0$ is the average prediction accuracy of all base learners, that is, the overall prediction accuracy, $p_i$ is the prediction accuracy of this learner.

The results of the Kappa coefficient are usually between 0 and 1. The base learners are sorted according to the calculated Kappa coefficients, and the base learners whose accuracy is above 0.8 are selected for ensemble, and others are filtered out.

The main advantage of the Ranking method is that it is simple and fast. There are also advantages in generalization performance and the calculation method of the difference between learners. A common approach is to divide a validation set $M_{val}$ from the overall data set to calculate the prediction accuracy $R$ of each base learner $C_i$:

$$R_i = \text{Accuracy}(M_{val}) \text{ of } C_i$$

(9)

In general, the verification set $M_{val}$ is used to calculate the differences among the basic learners. However, this method may have two shortcomings: (a) during the training process, it may lead to the loss of part of the training data, especially when the training data is small, it may easily cause the model to underfit. (b) The optimization target is set as a part of the data set in advance. This method may cause a deviation of the generated model in some algorithms, which is not conducive to the generation of the best model.

2) ITERATIVE OPTIMIZATION ENSEMBLE ALGORITHM

The FIOES algorithm flow is shown in Table 6.

D. EXPERIMENT AND RESULT ANALYSIS

1) EXPERIMENTAL ENVIRONMENT AND DATA SET

In this paper, the experimental environment is the Ubuntu system, the experimental programming language is Python 3.0, the research and development tool is PyCharm, and the deep learning TensorFlow framework is used.

The data from January to December 2018 is used as a training set to predict the power consumption in January and February 2019.

In the extraction of training data, the Out of Bag method in the Bagging algorithm is used for random sampling [32], [33]. The main process is: First, the data of 100,000 users in the entire data set were processed for missing values and outliers, and 3590 households were eliminated, leaving...
TABLE 6. FIOES algorithm flow.

| INPUT | OUTPUT |
|-------|--------|
| data set: D, base learner algorithm: L, number of initial base learners: T, initial filtering ratio: k_i, final remaining ratio: k_f, minimum difference increment: ΔE | For each new sample, the prediction results of the integrated system are output. |

1) For t = 1,2,⋯,T do
2) h_t = L(D, D_boot)
3) END FOR
4) Calculate the accuracy of each learner on out-of-bag samples ACC_{obg}
5) According to the accuracy rate, the learner is ranked to get the set: {h_1, h_2, ⋯, h_T}(ACC_{obg} < ACC_{max})
6) Filter out the learners in front of the learner set according to the initial filtering ratio k_i, and the remaining set S(m_1, m_2, ⋯, m_{T−i})(ACC_{obg} < ACC_{max}), the remaining quantity is Num
7) WHILE Numbers > k_f * Num
8) Calculate the difference of the current learner group: E_{avg}
9) Calculate the learner group difference E_{avg} after filtering the i_{th} learner
10) IF E_{avg} > ΔE; DELETE m_i from S; i++;
14) STOP //When the deletion process cannot be continued or the number of deletions reaches the threshold
15) Average the results of the residual learners.

TABLE 7. The importance of learner predictor variables.

| Fields               | Importance |
|----------------------|------------|
| Clustering results   | 0.40       |
| PowCsp_Chp           | 0.25       |
| Monthly average temp | 0.15       |
| Distance to centroid | 0.05       |
| PowCsp_Chp_Rt        | 0.05       |
| Precipitation        | 0.05       |
| Sunshine intensity   | 0.05       |

96410 households. Then it randomly samples 96410 times, so the probability that a piece of data is collected each time is 1/96410, and the probability that it is not collected is 1-1/96410. If the probability of not being collected in 96410 acquisitions is (1-1/96410)^{96410} ≈ 0.3679, and each extraction is performed randomly, it is guaranteed that the data at each sampling is not exactly the same. We use this method to train each base learner.

In the construction of the base learning, in order to ensure that all data sets can be selected to participate in the training of the base learner, a total of 15 random samplings were performed during the construction phase, that is, the number of base learners is 15. Each random sampling not only guarantees the differences between the base learners, but also ensures that almost all data sets participate in the training of the base learners. This method can ensure that the final integration model has good generalization performance.

2) EXPERIMENTAL DESIGN

This paper uses the FIOES model to classify and prediction the power consumption information of household power users. The importance of each base learner to the predictor variables is shown in Table 7.

This importance will be used as a weight when the neural network is trained. When analyzing the prediction result, the error between the actual value and the prediction value is used to evaluate the quality of the prediction result. There are two groups of comparative experiments in this paper. The specific methods are as follows:

(1) Compare the prediction results of the selective ensemble framework model with a single base learner model to verify whether the selective ensemble framework model is better than the traditional model.

(2) Set different values of Kappa coefficient k_i, compare the classification effect under different k_i values, and prove the impact of Filtering Iterative Optimization Ensemble Strategy on the results.

3) COMPARISON AND ANALYSIS OF PREDICTION RESULTS

Table 8 shows the prediction results of some users in January and February 2019 by using the FIOES model, which mainly includes the actual value, predicted value, relative error, and absolute error. After statistical analysis, the error range of the FIOES model is between −18 and 23, and the average error is 7.35.

Table 9 shows the prediction results of a single base learner. Here we use the traditional neural network model. The error range of this model is between −31 and 38, and the average error is 12.33. Compared with the FIOES model, the prediction accuracy of the single learner model is worse.

Based on the above data, Fig 8 shows the comparison results of prediction with FIOES model and single base learner. Fig 8 and Fig 9 are the prediction results of January and February 2019 respectively. Through comparison, it can be seen that the prediction results with FIOES model are closer to the actual values.

In the selective integration stage of the FIOES model, different Kappa coefficients k_i are used to make the prediction results significantly different. Table 10 shows the deviation range and average deviation between the predicted result and the actual value when the Kappa coefficient k_i is adjusted from 0.6 to 0.9. When k_i changes from 0.6 to 0.8, the deviation range gradually decreases, indicating that the effect of the model increases with increasing k_i. When k_i changes from
TABLE 8. Prediction results by FIOES model.

| UserID       | Actual value (Jan) | predicted value (Jan) | absolute error | relative error (%) | true value (Feb) | predicted value (Feb) | absolute error | relative error (%) |
|--------------|--------------------|-----------------------|----------------|--------------------|------------------|-----------------------|----------------|--------------------|
| 6020912533   | 203                | 207                   | 4              | 1.97               | 197              | 200                   | 3              | 1.52               |
| 7011829901   | 286                | 288                   | 2              | 0.70               | 279              | 283                   | 4              | 1.45               |
| 3280811518   | 255                | 248                   | -8             | -3.14              | 248              | 260                   | 12             | 4.84               |
| 6449918768   | 314                | 319                   | 5              | 1.59               | 307              | 309                   | 2              | 0.65               |
| 5478241759   | 277                | 270                   | -7             | -2.52              | 284              | 279                   | -5             | 1.76               |
| 1771214636   | 269                | 273                   | 4              | 1.49               | 275              | 271                   | -4             | 1.45               |
| 8553050114   | 447                | 452                   | 5              | 1.06               | 442              | 450                   | 8              | 1.80               |
| 3145746770   | 183                | 180                   | -3             | -1.64              | 205              | 209                   | 4              | 1.95               |
| 7645243142   | 259                | 259                   | 0              | 0                  | 264              | 261                   | -3             | -1.15              |
| 2396659493   | 331                | 335                   | 4              | 1.21               | 330              | 331                   | 1              | 0.30               |

TABLE 9. Prediction results of a single base learner.

| UserID       | Actual value (Jan) | Predicted value (Jan) | Absolute error | Relative error (%) | True value (Feb) | Predicted value (Feb) | Absolute error | Relative error (%) |
|--------------|--------------------|-----------------------|----------------|--------------------|------------------|-----------------------|----------------|--------------------|
| 6020912533   | 203                | 193                   | -10            | -4.93              | 197              | 218                   | 21            | 10.66              |
| 7011829901   | 286                | 297                   | 11             | 3.85               | 279              | 288                   | 9             | 3.23               |
| 3280811518   | 255                | 267                   | 12             | 4.71               | 248              | 231                   | -17           | -6.85              |
| 6449918768   | 314                | 303                   | -11            | 3.50               | 307              | 319                   | 12            | 3.91               |
| 5478241759   | 277                | 267                   | -10            | -3.61              | 284              | 302                   | 18            | 6.34               |
| 1771214636   | 269                | 277                   | 6              | 2.23               | 275              | 262                   | -13           | -4.73              |
| 8553050114   | 447                | 456                   | 9              | 2.01               | 442              | 448                   | 6             | 1.36               |
| 3145746770   | 183                | 195                   | 12             | 6.56               | 205              | 196                   | -9            | -4.39              |
| 7645243142   | 259                | 267                   | 8              | 3.09               | 264              | 254                   | -10           | -3.79              |
| 2396659493   | 331                | 335                   | 4              | 1.21               | 330              | 331                   | 1             | 0.30               |

FIGURE 9. The comparison results of prediction with FIOES model and single base learner-The prediction results of February 2019.

TABLE 10. Prediction deviation for different $k_i$ values.

| $k_i$ values | Deviation range | Average deviation |
|--------------|-----------------|------------------|
| $k_1 = 0.6$  | -27 to -39      | 9.74             |
| $k_2 = 0.7$  | -24 to -32      | 8.15             |
| $k_3 = 0.8$  | -18 to -23      | 7.35             |
| $k_4 = 0.9$  | -21 to -29      | 7.73             |

0.8 to 0.9, the deviation range continues to increase, indicating that the effect of the model will also decrease. Therefore, when $k_i$ is 0.8, the classification effect of this model is the best.

It can be clearly seen in Table 10 that the prediction result of FIOES model is the best when $k_i = 0.8$, and the minimum deviation and the maximum deviation are 18 and 23 respectively. Compared with other values, the prediction deviation is the lowest, so when $k_i = 0.8$, the integration effect of each basic learner is the best. The experimental results show that the ensemble strategy of each basic learner determines the classification effect of the model.

V. CONCLUSION

In this paper, we have proposed FIOES to improve the accuracy of power big data prediction. The proposed FIOES is effective for almost all power consumption prediction, which means that FIOES has good generalization ability. In order to make the prediction results more accurate, we first cluster the household power users, classify the users according to their power consumption characteristics, and bring data support to power management and prediction. To the best of our knowledge, this is the first work in the literature to adopt the combination of ranking method and iterative optimization method to selectively ensemble each basic learner. The experiments show that the prediction accuracy of the model can be improved by secondary filtering and iterative optimization. Moreover, different Kappa coefficient leads to different ensemble effects. Only by selecting the appropriate coefficient, the optimal prediction model can be obtained.

REFERENCES

[1] J. Zhang, M. Cui, and Y. He, “Robustness and adaptability analysis for equivalent model of doubly fed induction generator wind farm using measured data,” Appl. Energy, vol. 261, Mar. 2020, Art. no. 114362.

[2] A. Kaur, H. T. C. Pedro, and C. F. M. Coimbra, “Ensemble re-forecasting methods for enhanced power load prediction,” Energy Convers. Manage., vol. 80, pp. 582–590, Apr. 2014.
[3] N. Amjady and F. Keynia, “Mid-term load forecasting of power systems by a new prediction method,” Energy Convers. Manage., vol. 49, no. 10, pp. 2678–2687, Oct. 2008.

[4] M. Cui, J. Wang, and B. Chen, “Flexible machine learning-based cyber-attack detection using spatiotemporal patterns for distribution systems,” IEEE Trans. Smart Grid, vol. 11, no. 2, pp. 1805–1808, Mar. 2020.

[5] Z. H. Zhou, Y. Yu, and C. Qian, “Selective ensemble,” in Evolutionary Learning: Advances in Theories and Algorithms, 2019.

[6] W. Davy, S. Abeed, and K. Ari, “Deep neural networks ensemble for detecting medication mentions in tweets,” J. Amer. Med. Inform. Assoc., vol. 26, no. 12, pp. 1618–1626, 2019.

[7] L. Ismaiel and A. I. Ismaiel, “Prediction the data consumption for power demands by elman neural network,” Int. J. Electr. Comput. Eng., vol. 9, no. 5, p. 4003, Oct. 2019.

[8] J. von Kistowski, M. Deffner, and S. Kounev, “Run-time prediction of power consumption for component deployments,” in Proc. IEEE Int. Conf. Autonomic Comput. (ICAC), Sep. 2018, pp. 151–156.

[9] S. Site and S. K. Mishra, “Model for measuring accuracies of majority voting of ensemble classifier with COB and genetic algorithm,” in Proc. Int. Conf. Inf. Commun. Embedded Syst. (ICICES), Feb. 2013, pp. 99–103.

[10] A. Onan, S. Korukoçlu, and H. Bulut, “A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification,” Expert Syst. Appl., vol. 62, pp. 1–16, Nov. 2016.

[11] C. Deng and M. Perkowski, “A novel weighted hierarchical adaptive voting ensemble machine learning method for breast cancer detection,” in Proc. IEEE Int. Symp. Multiple-Valued Log., May 2015, pp. 115–120.

[12] B. Hu, J. Wang, Y. Zhu, and T. Yang, “Dynamic deep forest: An ensemble classification method for network intrusion detection,” Electronics, vol. 8, no. 9, p. 968, Aug. 2019.

[13] L.-J. Zhao, T.-Y. Chai, and D.-C. Yuan, “Selective ensemble extreme learning machine modeling of effluent quality in wastewater treatment plants,” Int. J. Autom. Comput., vol. 9, no. 6, pp. 627–633, Dec. 2012.

[14] M. Li and S. Wang, “Dynamic fault monitoring of pitch system in wind turbines using selective ensemble small-world neural networks,” Energies, vol. 12, no. 17, p. 3256, Aug. 2019.

[15] D. D. Margineantu and T. G. Dietterich, “Improved class probability estimates from decision tree models,” Nonlinear Estimation and Classification, 2003.

[16] R. Adhlakari, G. Verma, and I. Khandelwal, “A model ranking based selective ensemble approach for time series forecasting,” Procedia Comput. Sci., vol. 48, pp. 14–21, Jan. 2015.

[17] T. Wan, H. Jun, and Z. Hui, “Kappa coefficient: A common method to measure the consistency among evaluators,” Shanghai Psychiatry, to be published.

[18] J. S. Guo, J. C. Zeng, and J. X. Chen, “Selective ensemble learning with parallel optimization and hierarchical selection,” in Proc. Int. Conf. Mach. Learn. Cybern. (ICMLC), Jul. 2015, pp. 194–199.

[19] M. A. Al-Betar, I. Aljarah, and A. M. Awadallah, “Adaptive?—Hill climbing for optimization,” Soft Comput., to be published.

[20] K. Chen, “The study on a selective BagBoosting trees ensemble algorithm based on GASEN technology,” Stat. ThinkTank, vol. 12, p. 3, 2008.

[21] C. L. Wang and B. C. Xue, “Eprl-laf learning algorithm of RBF network,” Int. J. Autom. Comput., vol. 9, no. 9, p. 968, Aug. 2019.

[22] H. Tian, Y. Q. Ding, and S. Y. Yang, “Distributed node location algorithm based on MDS and SDP,” TV Technol., vol. 9, pp. 87–90, 2012.

[23] A. Loth, “Data preparation with tableau prep,” in Visual Analytics with Tableau. 2019.

[24] X. D. Sun, Y. J. Liu, and W. W. Chen, “Application of box plot method in inspection of abnormal values of animal health data,” China Animal Quarantine, vol. 7, pp. 72–74, 2010.

[25] C. Chinhtrungcueng and C. H. Sequin, “Optimal adaptive k-means algorithm with dynamic adjustment of learning rate,” IEEE Trans. Neural Netw., vol. 6, no. 1, pp. 157–169, Jan. 1995.

[26] I. Rebro, R. Arajandjelic, and J. Sivic, “Convolutional neural network architecture for geometric matching,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 11, pp. 2553–2567, Nov. 2019.

[27] X. Chen, S.-C. Wong, and C. K. Tse, “Adding randomness to modeling Internet TCP-RED systems with interactive gateways,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 57, no. 4, pp. 300–304, Apr. 2010.

[28] H. K. Huang, C. F. Chiu, and C. H. Kuo, “Mixture of deep Neural Network-based ensemble model for image retrieval,” Tech. Rep., 2016.

[29] J. Liu, C. Y. Zhou, and N. H. Cao, “Impact analysis of meteorological factor to electricity consumption and early warning study,” Meteorolog. Environ. Sci., vol. 33, pp. 61–64, 2010.

[30] C. Lin, W. Chen, C. Qiu, Y. Wu, S. Krishnan, and Q. Zou, “LibD3C: Ensemble classifiers with a clustering and dynamic selection strategy,” Neurocomputing, vol. 123, pp. 424–435, Jan. 2014.

[31] Y. Yu, Z. Li, X. Liu, K. Hirota, X. Chen, T. Fernando, and H. H. C. Lu, “A nested tensor product model transformation,” IEEE Trans. Fuzzy Syst., vol. 27, no. 1, pp. 1–15, Jan. 2019.

[32] X. Zhang, “Improvement on the vanishing component analysis by group- ing strategy,” EURASIP J. Wireless Commun. Netw., vol. 2018, no. 1, p. 111, Dec. 2018.

[33] H. Y. GAO, H. J. LU, and K. YAN, “Selective ensemble algorithm for gene expression data based on diversity and accuracy of weighted harmonic average measure,” J. Comput. Appl., vol. 5, p. 51, 2018.