IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 9, 2021, accepted March 13, 2021, date of publication March 18, 2021, date of current version April 7, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3067043

Short-Term Load Forecasting Based on PSO-KFCM Daily Load Curve Clustering and CNN-LSTM Model

CHUAN SHANG, JUNWEI GAO, HUABO LIU, (Member, IEEE), AND FUZHENG LIU

College of Automation, Qingdao University, Qingdao 266071, China
Shandong Key Laboratory of Industrial Control Technology, Qingdao 266071, China

Corresponding author: Junwei Gao (qdgao163@163.com)

This work was supported in part by the Shandong Provincial Natural Science Foundation under Grant ZR2019MF063 and Grant ZR2020MF064, in part by the National Key Research and Development Program of China under Grant 2019YFC010167, and in part by the Key Research and Development Plan of Shandong Province under Grant 2017GGX10115.

ABSTRACT Short-term load forecasting (STLF) with excellent precision and prominent efficiency plays a significant role in the stable operation of power grid and the improvement of economic benefits. In this paper, a novel model based on data mining and deep learning is proposed. Firstly, the preprocessing of data includes normalization of historical load, and fuzzification of influencing factors (meteorological factors, date types and economy) based on Pearson correlation coefficient (PCC). Secondly, kernel fuzzy c-means (KFCM) modified by particle swarm optimization (PSO-KFCM) algorithm clusters the daily load curve. In the clustering experiments, the within-cluster sum of squared error (SSE) index is presented to determine the number of clusters and the clustering validity has a 31.9% enhancement compared with the traditional FCM algorithm. Thirdly, the cosine similarity establishes the resemblance between the prediction date and each cluster, and the similar cluster is determined according to the principle of maximum similarity. Finally, a multivariate and multi-step hybrid model MMCNN-LSTM based on convolution neural network (CNN) and long short-term memory (LSTM) neural network is proposed to forecast the load in following 24 hours, in which similar cluster data is applied to training set. To demonstrate the effectiveness of proposed integrated technique, the accuracy has been verified in three predictive experiments. The fruitful results indicated that the average mean absolute percent error (MAPE) in the entire test set was only 1.34%, a 3.02% reduction compared to a single LSTM.

INDEX TERMS Short-term load forecasting, Pearson correlation coefficient, PSO-KFCM, cosine similarity, CNN, LSTM.

I. INTRODUCTION

Power load forecasting is to forecast the future load data with historical data as the key component [1]. The level of power load forecasting has become a remarkable sign to measure whether the management of an electric power enterprise is going towards modernization. Accurate power load forecasting plays a significant role in realizing the modernization and scientific management of power grid [2]. Power load forecasting can be divided into long-term load forecasting (LTLF), medium-term load forecasting (MTLF), short-term load forecasting (STLF) and very short-term load forecasting (VSTLF) according to the forecast duration. Among them, STLF refers to the prediction of the future daily load or weekly load, which is mainly worked for power system operation dispatching, guaranteeing the safety of power grid process and improving the operational efficiency. Therefore, it is a substantial task of power system from the perspective of security, economy and development [3].

The traditional forecasting model takes historical load data as the whole basis. For instance, the trend extension method usually deduces its future trend and state according to the gradient law. Similarly, methods of regression analysis is to adjust the parameters and extrapolate the prediction [4]. Time series method is one of the most customary forecasting methods, whose core is to establish a mathematical model by analyzing the metabolic law between historical data and time information. The ordinally adopted time series models are:

The associate editor coordinating the review of this manuscript and approving it for publication was Seyedali Mirjalili.

autoregressive moving average (ARMA)model [5], autoregressive integrated moving average (ARIMA) model [6], seasonal autoregressive integrated moving average (SARIMA) model [7], auto regressive integrated moving average models with external input (ARIMAX) model [8]. However, the conventional prediction methods require a minor transformation in development trend, and the relationship between historical data and forecast data is relatively simplistic. Obviously, the traditional method with inferior accuracy caused by the tremendous alterations in tendency has a fatal shortcoming.

In recent decades, with the rapid progression of machine learning, experts and scholars all over the world have conducted in-depth research on STLF and put forward numerous effective forecasting models. Despite the dilemma in identifying the optimal parameters, kernel function parameter σ and penalty factor c, Hua *et al*. introduced a supervised learning model called support vector machine (SVM) into STLF [9]. The addition of the optimization-seeking algorithm to SVM was proposed to alleviate the underlying problem, such as SVM optimized by particle swarm optimization (PSO-SVM) [10], genetic algorithm (GA-SVM) [11], fruit fly algorithm (FF-SVM) [12], dragonfly algorithm (DA-SVM) [13]. Because inequality constraints are changed into equality constraints, the least squares support vector machine (LS-SVM) model applied by Yang *et al*. simplified the algorithm and enhanced solving speed [14]. In 1991, Artificial neural network (ANN) was first proposed by Park DC [15] for load forecasting of power system, and back propagation (BP) algorithm was adopted. Afterwards, due to the optimization by intelligence algorithms, the ANN got improved accuracy, but it relied heavily on the quality of training data [16]. Nowadays it is well known that deep neural network (DNN) has dominated load prediction in recent years. Shi *et al*. imported the novel recurrent neural network (RNN) into household STLF domain, which was available for time characteristics [17]. In addition, long short-term memory (LSTM) network is a variant of RNN, which overcomes the gradient disappearance and gradient explosion of RNN, therefore, the LSTM adopted by Liu *et al*. performed more prominently on long sequences [18]. Other DNN baseline models were trained with abundant samples, such as gate recurrent unit (GRU) [19] and bidirectional recurrent neural network (Bi-RNN) [20], however their results implied that it is not a promising realisation due to the ease of overfitting. In the process of STLF development, single machine learning models had difficulty meeting load accuracy requirements and a few hybrid preprocessing methods were mixed into them. These usual preprocessing methods include grey theory [21], wavelet packet analysis [22], empirical mode decomposition (EMD) [23], random forest [24] and so on.

To sum up, although the STLF based on modern prediction method has achieved great performances in theory and application, the reasoning process is quite complex and tough to meet the demands of practical problems. This paper proposes a novel model based on data mining and deep learning, which not only takes historical data into account, but also meteorology, date type, economy and others. The normalization of historical load prevents the gradient from falling sluggishly and the fuzzification avoids that the influencing factors cannot be fed directly into prediction model due to different weights. The proposed PSO-KFCM algorithm is applied to daily load curve clustering, which cracks the obstacle that the initial clustering center is easily limited to local optimization. After that, cosine similarity of the influencing factor is exerted to establish relationship between the prediction day and all clusters. Finally, a hybrid deep learning model CNN-LSTM whose input mode is multivariate and multi-step (MMCNN-LSTM) is proposed to forecast the load data in the next 24 hours. The above comprehensive technique which combines the PSO-KFCM algorithm and CNN-LSTM model so far have not been applied in the field of STLF. The effectiveness of the above comprehensive technique is verified by three predictive results, which will provide a reference for future load forecasting.

The main contributions in this paper are summarized as follows:

1) **Advanced preprocessing methods:** From the perspective of practical problems, historical load data and influencing factors (meteorology, date type, economy and others) should be amply taken into account, where historical load data are normalized and influencing factors are fuzzy mapped according to the PCC.
2) **Enhanced algorithms:** KFCM algorithm maps the points form the original space to the high-dimensional space through the kernel function, which increases operational efficiency. After the PSO optimization, PSO-KFCM algorithm has a better global searching ability which overcomes the problem of sensitivity to the initial clustering center. The CNN-LSTM hybrid model not only has the ability to extract features, but also is skillful at processing time series.
3) **Multitudinous comparative experiments:** In this research, one clustering experiment and three predictive experiments were executed. Firstly, the validity of clustering method is confirmed by the clustering experiment. In addition, unclustered models, three other input methods and five DNN baseline models are compared in accuracy with the data mining-based novelty MMCNN-LSTM model.
4) **Fruitful evaluation indicators:** On the one hand, the clustering validity indicators refer to silhouette coefficient (SC), Davies-Bouldin index (DB), Calinski-Harabasz index (CH), Krzanowski-Lai index (KL). On the other hand, root mean square error (RMSE), mean absolute error (MAE) and MAPE represent prediction evaluation indicators. The excellent results are validated by the optimal evaluation indicators as well.

The remainder of this paper is organized as follows. Part II introduces the theory of algorithms applied in this study. Part III presents the source of the dataset and modeling.

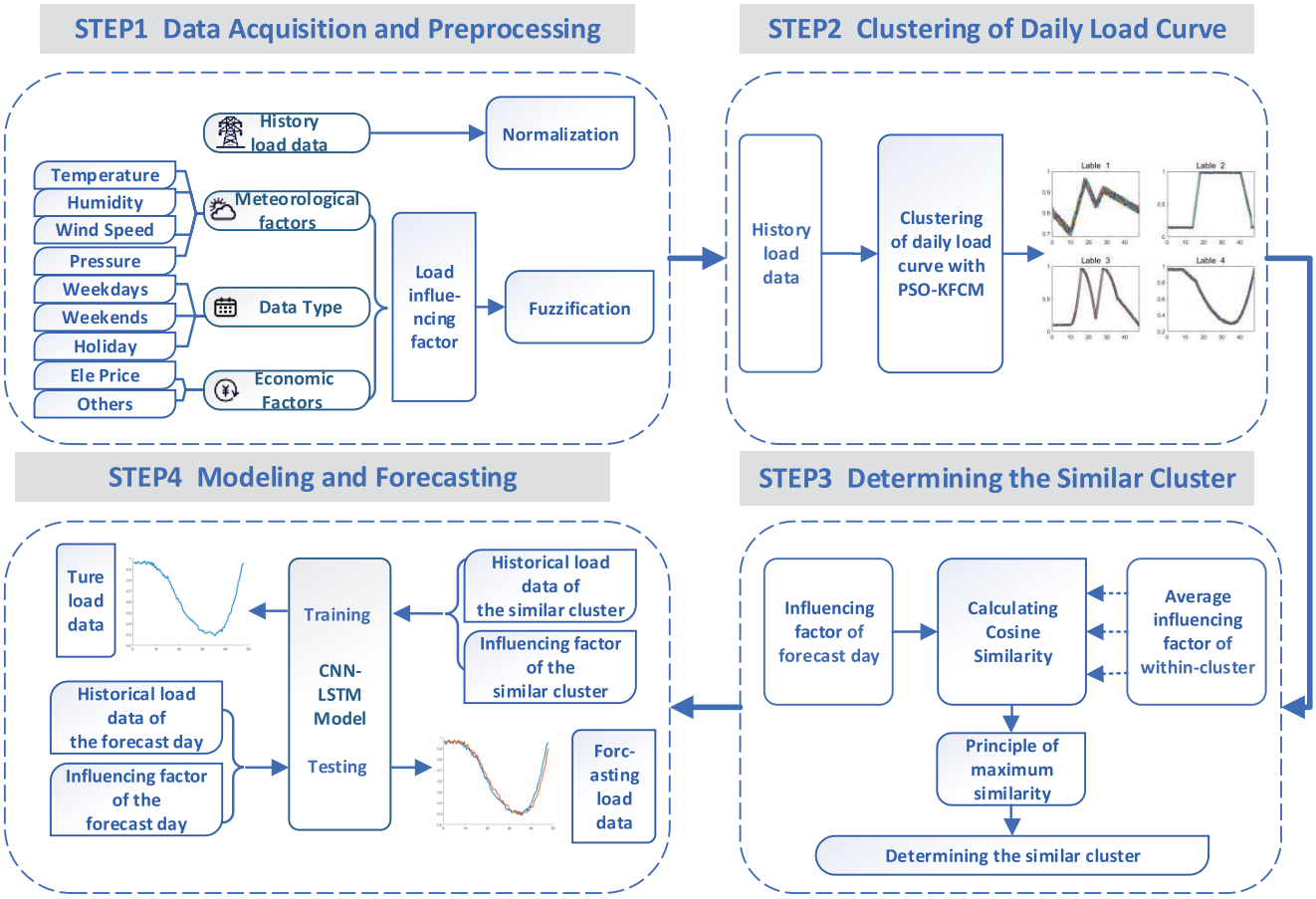


FIGURE 1. The proposed overall design for short-term load forecasting.

In Part IV, exploratory experiments are carry out and experimental results are shown. Finally, Part V summarizes this study.

II. METHODOLOGY

A. THE PROPOSED METHOD

The proposed short-term load forecasting based on PSO-KFCM algorithm and CNN-LSTM model is shown in Fig.1. Firstly, historical power load data can be obtained from the supervisory control and data acquisition (SCADA) system. Load forecasting should not take the historical load data as whole evidences, influencing factors are ought to obtain from the local administrative section such as meteorological factors, date types, economic factors, etc. The load features composed of historical data and influencing factors guarantee the accuracy of the prediction and enhance the anti-interference ability. Then, the historical load data and the influencing factor are normalized and fuzzified respectively, which overcomes the odd of slow gradient descent caused by vast values and and not being able to input directly into the prediction model due to non-uniform units or no units. In the second step, PSO-KFCM algorithm clusters the normalized historical load data to excavate typical power consumption characteristics. Third, cosine similarity establishes the resemblance between the prediction day and each cluster, where the input is the influencing factor of the predicted day and the average influencing factor of within-cluster both after fuzzification. The similar cluster of the prediction day is identified according to the principle of maximum similarity. All three steps mentioned above apply to the preprocessing method for extracting hidden features and internal laws. Finally, the multivariable and multi-step CNN-LSTM model is committed to forecasting 24h ahead load data in the end.

B. THE PRINCIPLE OF THE PSO-KFCM ALGORITHM

1) KFCM

Fuzzy c-means (FCM) algorithm is an unsupervised fuzzy clustering method based on objective function proposed by Bezdek *et al.* [25]. FCM belongs to soft clustering, which is different from traditional hard c-means (HCM) clustering in that it allows the same object to belong to the same cluster. This fuzzy partition enables each data point to determine the degree of its relevance with other groups through the membership grade between [0,1]. Set the dataset as $X = \{x_1, x_2, \cdots, x_n\} \subset R^p$, each data x has p characteristics and n

is the number of sample data in dataset X. To divide a dataset into k classes, the objective function of FCM algorithm is as follows:

$$J_m(U,V)=\sum_{i=1}^{k}\sum_{j=1}^{n}u_{ij}^{m}\left\|x_j-c_i\right\|_2^2 \tag{1}$$

where m represents the fuzzy weighting coefficient, u_{ij} represents the membership grade of sample data j to cluster i, c_i represents the clustering center of i cluster. U is a $k*n$ matrix, representing the membership matrix, V is a $k*p$ matrix, representing the clustering center matrix. Obviously, this 2-norm $\left\|x_j-c_i\right\|_2$ is the Euclidean distance from each data point to the clustering center. The constraint condition is $\sum_{i=1}^{k}u_{ij}=1,\forall j=1,2,\cdots n$, that is, the sum of membership grades of each sample data to all clusters is equal to 1 [26]. More seriously, the membership grade is inversely proportional to Euclidean distance, which makes FCM sensitive to noise and outliers. In the case of data with strong interference, this fatal shortcoming lead to poor clustering quality.

For the purpose of settling the problem of poor clustering quality caused by the constraint condition, the kernel function is introduced into KFCM algorithm, which maps the points of the original space to the high-dimensional feature space. Contrasted with FCM algorithm, KFCM algorithm has been greatly improved in performance and classification effect, because it enlarges the feature differences among various samples through nonlinear mapping [27]. The objective function of KFCM algorithm is as follows:

$$J_m(U,V)=\sum_{i=1}^{k}\sum_{j=1}^{n}u_{ij}^{m}\left\|\Phi(x_j)-\Phi(c_i)\right\|_2^2 \tag{2}$$

where Φ represents a nonlinear mapping, the Euclidean distance $\left\|x_j-c_i\right\|_2$ in the traditional FCM algorithm is rewritten as $\left\|\Phi(x_j)-\Phi(c_i)\right\|_2$, $\Phi(x_j)$ and $\Phi(c_i)$ are the images of sample data and clustering center mapped from the original space to the high-dimensional feature space respectively.

The common kernels are radial basis function (RBF) kernel, rational quadratic (RQ) kernel, exponential kernel, sigmoid kernel and so on. In this study RBF kernel function is dedicated to nonlinear mapping, which has the characteristics of rotational symmetry and separability [28]. RBF kernel function can be decomposed into the following forms:

$$K(x,\hat{x})=<\Phi(x),\Phi(\hat{x})>=\exp\left(-\frac{\left\|x-\hat{x}\right\|^2}{2\sigma^2}\right) \tag{3}$$

Obviously, $K(x,x)=1$.Therefore, the objective function of KFCM can be simplified as follows:

$$J_m(U,V)=2\sum_{i=1}^{k}\sum_{j=1}^{n}u_{ij}^{m}(1-K(x_j,c_i)) \tag{4}$$

The perfect clustering result is the smallest similarity within a cluster and the largest similarity between the clusters, which means calculating the minimum objective function. The Lagrange multiplier method is applied to this extreme value problem with constraint condition. In the end, calculation formula of the clustering center c_i and membership grade u_{ij} are shown in Eq.(5) and Eq.(6) respectively:

$$c_i^{(l+1)}=\frac{\sum_{j=1}^{n}(u_{ij}^{(l)})^m K(x_j,c_i^{(l)})x_j}{\sum_{j=1}^{n}(u_{ij}^{(l)})^m K(x_j,c_i^{(l)})} \tag{5}$$

$$u_{ij}^{(l+1)}=\frac{(1-K(x_j,c_i{}^{(l+1)}))^{\frac{-1}{m-1}}}{\sum_{i=1}^{k}(1-K(x_j,c_i{}^{(l+1)}))^{\frac{-1}{m-1}}} \tag{6}$$

where l represents the iteration step at present. The concrete steps of KFCM algorithm are as follows:

1) Initialize the maximum number of iteration steps M, number of clusters k, fuzzy weighting coefficient m, RBF kernel parameters σ, termination threshold of the objective function δ and iteration step $l=0$.
2) Initialize the membership matrix $U^{(0)}$ through the random numbers between [0,1] in case of satisfying the constraint condition.
3) According to the Eq.(5) and Eq.(6), The clustering center matrix $V^{(l+1)}$ and membership matrix $U^{(l+1)}$ are constantly updated respectively. Then, compute the value of the objective function $J_m^{(l+1)}$.
4) If $\left|J_m^{(l+1)}-J_m^{(l)}\right|<\delta$ or reaching the maximum number of iteration steps, the calculation is terminated, otherwise let iteration step $l=l+1$ and skip back to step (3).

The KFCM mentioned above has overcome the problem of poor quality caused by outliers. However, extra attention should be paid to step (3), KFCM itself is an iterative descent algorithm, which makes it sensitive to the initial clustering center and tough to converge to global optimality.

2) PSO-KFCM

In view of the shortcoming of KFCM clustering algorithm which is sensitive to initial value and easy to drop into local optimum, the kernel fuzzy c-means optimized by particle swarm optimization (PSO-KFCM) is figure out to escape poor robustness.

PSO is a new global optimization algorithm with winged convergence speed and few parameters proposed by Eberhart and Kennedy [29]. It simulates the predatory behavior of birds searching for food randomly through mass free particles. Particles have two important properties: velocity and position. Let the particle population size be N, where the position of the i-th particle in the D-dimensional space can be expressed as $x_i=(x_{i1},x_{i2},\cdots,x_{id},\cdots,x_{iD})$. The velocity of i-th particle is defined as the moving distance in each iteration, expressed by $v_i=(v_{i1},v_{i2},\cdots,v_{id},\cdots,v_{iD})$. The optimal position of the i-th particle at present is called individual extremum, which is denoted as $p_{best}=(p_{i1},p_{i2},\cdots,p_{id},\cdots,p_{iD})$. The optimal position of the

whole population at present is called global extremum, denoted as $g_{best} = (p_{g1}, p_{g2}, \cdots, p_{gd}, \cdots, p_{gD})$. The formula for the i-th particle to update its velocity and position in d-dimensional space is as follows respectively:

$$v_{id}(t+1) = wv_{id} + c_1 r_1 (p_{id} - x_{id}(t)) + c_2 r_2 \left(p_{gd} - x_{id}(t)\right) \tag{7}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \tag{8}$$

where w is the inertia weight, c_1 and c_2 are the acceleration constant, r_1 and r_2 the random number between [0,1], p_{id} is the individual optimal position at present in d-dimensional space, and p_{gd} is the optimal position of the whole population at present in d-dimensional space. The specific process of PSO algorithm is as follows:

1) Initialize the particle swarm, including the population size N, the speed v_i and position x_i of each particle.
2) Calculate the fitness value of each particle.
3) According to the fitness value, search for the individual extremum p_{best} and the global optimal solution g_{best}.
4) Update the speed and position of each particle according to Eq.(7) and Eq.(8).
5) If the error is little enough or reaches the maximum iteration steps, the optimal result is output, otherwise skip back to step (2) and proceed with calculation.

The theory of PSO has been mentioned before, the following is the combination of PSO and KFCM. The prime problem to be solved is the encoding of particles. Let each particle represent the solution of the clustering center, so the velocity and position of each particle are $k * p$ matrix. Fitness function is another trouble to be resolved, which evaluates the position of each particle. As we all know, the smaller the KFCM objective function, the higher the clustering quality, that is, the larger the fitness value. Therefore, the objective function of KFCM is inversely proportional to the fitness value. The fitness function is defined as:

$$Fitness(x) = \frac{K_0}{K_0 + J_m(U, V)} \tag{9}$$

where K_0 is an arbitrary minor positive number and 1 is a decent choice to avoid the denominator of fitness function being 0 and ignoring the subject. Then the overall flow of PSO-KFCM algorithm is shown in Fig.2.

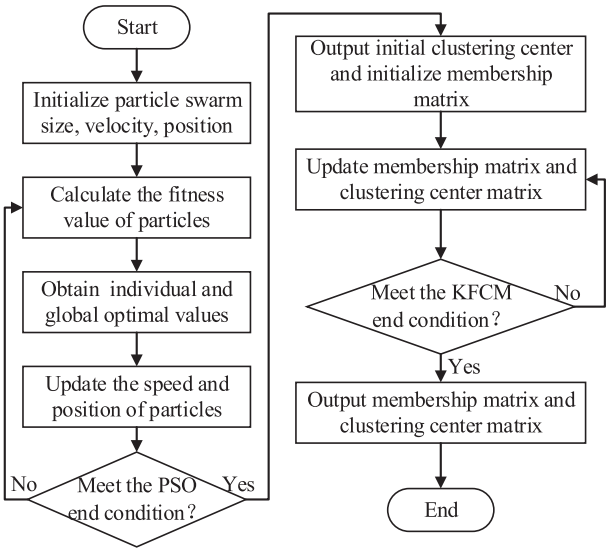


FIGURE 2. The overall flow chart of the PSO-KFCM algorithm.

C. THE PRINCIPLE OF THE COSINE SIMILARITY

Cosine similarity is an effective method to calculate the similarity between two unknown datasets. As we all know, Euclidean distance measures the absolute distance, which is directly related to the location coordinates of each point. Furthermore, it equates the differences between different attributes of samples, which can not occasionally meet the actual requirements [30], [31]. Contrary to Euclidean distance, cosine similarity measures the angle of space vector, which is more reflected in difference of direction rather than position. The closer the cosine value is to 1, the closer the angle is to 0 degree, that is, the more similar the two vectors are. In extreme cases, two vectors are completely coincident. The cosine similarity analysis between vector a and b is as follows:

$$\cos(\theta) = \frac{a \cdot b}{\|a\|_2 \cdot \|b\|_2} \tag{10}$$

Customarily, cosine similarity is applied in multidimensional positive space. In the field of load forecasting, load features are generally multidimensional positive numbers. After preprocessing, the data are in the first quadrant, the angle is 0-90 degree, thus the cosine similarity value is between [0,1]. Obviously, cosine similarity is absolutely suitable for the determination of similar cluster. The similar cluster is established by the maximum similarity intensity between the prediction date and each cluster, which is equivalent to performing a classification problem.

D. THE PROMCIPLE OF THE CNN-LSTM MODEL

1) CNN MODEL

Convolutional neural network (CNN) is a feedforward neural network (FNN) with feature extraction ability, which was proposed by Fukushima [32]. It can be divided into 1DCNN,2DCNN,3DCNN, among which 1DCNN is the most suitable for the prediction of time series. The integrated 1DCNN network consists of input layer, convolution layer, pooling layer, flattening layer and dense layer [33].

As can be seen from Fig.3, 1D does not mean that the input is 1-dimensional, but the direction of the convolution kernel motion is fixed, where the convolution kernel is a linear weighted function. The height H of input layer represents time steps and width W represents time features. In the convolution process, the input and the convolution kernel do point multiplication to extract features. If there are k convolution kernels of size f and the step size is s,then the formula for calculating the height of the convolution layer is illustrated

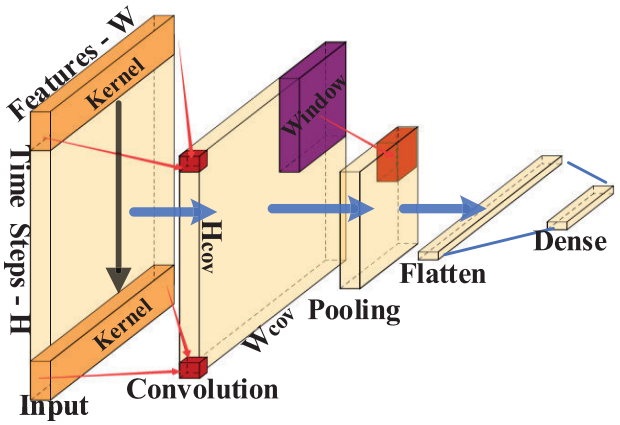


FIGURE 3. The network structure of CNN model.

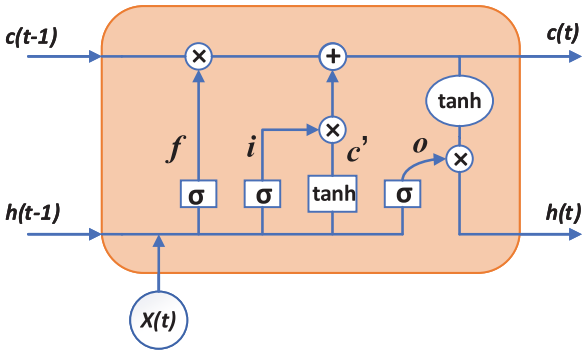


FIGURE 4. The internal structure of LSTM cell network.

below:

$$H_{cov} = \frac{(H - f)}{s} + 1 \tag{11}$$

The width of the convolution layer is determined by the number of convolution kernels, that is, $W_{cov} = k$.Then pooling layer concentrates data by narrowing the sampling window. The flatten layer stretches the data and connects it to the dense layer. Through this structure mentioned above, the characteristics of CNN can be summarized as follows:

1) Local receptive field: In contrast to the full connection, the convolution kernel is connected to the local part of input, which accelerates on operation.
2) Weight sharing: All the elements on the same feature map share the identical convolution kernel, that is to say, they assign a fixed weight, so that the parameter setting reduction.
3) Subsampled: In order to lessen redundancy and prevent overfitting, average pooling and maximum pooling are employed to concentrate data.

2) LSTM MODEL

In order to solve the gradient explosion and disappearance of traditional RNN, Sepp Hochreiter and Jürgen Schmidhuber proposed long short-term memory (LSTM) network in 1997 [34]. Compared with RNN, LSTM cell units are still calculated based on input and hidden layer output of the upper level, but the internal structure changes, while the external structure remains invariant. As shown in Fig.4, the internal structure of LSTM cell unit is composed of input gate i, forget gate f, output gate o and internal memory unit c. Forget gate f manages the forgetting degree of input $x(t)$ and output of the upper hidden layer $h(t-1)$. Input gate controls the update efficiency of input $x(t)$ and output of the upper hidden layer $h(t-1)$. The memory unit c determines which fresh information remains in the cell and renews the cell state. Finally, output gate o supervises how much information is output for cell [35]. The calculation of each component in LSTM is summarized as Eq.(12) - Eq.(17).

$$\text{Forget gate: } f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{12}$$
$$\text{Input gate: } i_t = \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right) \tag{13}$$
$$\text{New memory unit: } c'_t = \tanh\left(W_c x_t + U_c h_{t-1} + b_c\right) \tag{14}$$
$$\text{Final memory unit: } c_t = f_t * c_{t-1} + i_t * c'_t \tag{15}$$
$$\text{Output gate: } o_t = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right) \tag{16}$$
$$\text{Output: } h_t = o_t * \tanh\left(c_t\right) \tag{17}$$

where W_f, W_i, W_c and W_o are input weight vectors, U_f, U_i, U_c and U_o are upper output weight vectors, b_f, b_i, b_c and b_o are bias vectors. Sigmoid is generally selected as the excitation function for σ, which mainly plays a role of gating. Tanh function is an option to generate new memory unit c'_t due to faster convergence rate. Their expressions are manifested in Eq.(18) and Eq.(19) respectively.

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \tag{18}$$
$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{19}$$

3) THE HYBRID CNN-LSTM MODEL

In this paper, CNN can be regarded as "feature extractor" after preprocessing, that is to extract local features in time step. 1DCNN is typically employed to address time series related problems, convolution kernel only slides along an inflexible direction to automatically extract the hidden features and internal laws of data in the time direction. The extracted feature information sequence is input into LSTM network. By a large number of training data, the weights of input gate, forget gate and output gate in LSTM network are adjusted constantly, so that LSTM is capable to learn the time dependence relationship between feature information sequence and output.

As demonstrated in Fig.5, the original data is a multi-dimensional load features with time information. CNN which has an excellent feature extraction ability is the same as automatic encoder in Seq2Seq model. Furthermore, LSTM which has a brilliant prediction capacity of long time series is more like an automatic decoder in Seq2Seq model. This

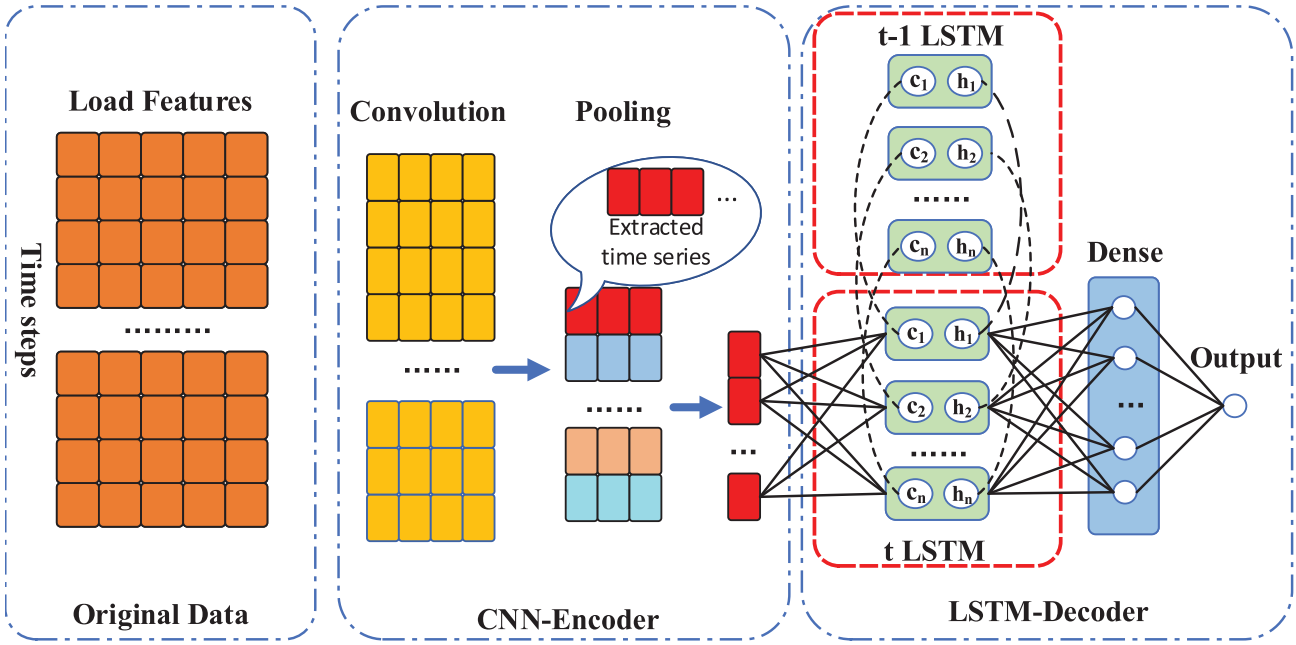


FIGURE 5. The overall design of the hybird CNN-LSTM model.

hybrid model CNN-LSTM which combines the advantages of both, improves the prediction accuracy. It's extensively operated in load forecasting, fault diagnosis, version classification and other fields in recent years.

III. DATA AND MODELING

A. DATA SOURCES AND PREPROCESSING

This public electrical load dataset from Australian New South Wales (NSW) which is downloaded at the Australian Energy Market Operator (AMEO) Official website https://www.aemo.com.au/energy-systems/electricity/national-electricity-market-nem/data-nem/aggregated-data. NSW power system is responsible for supplying electricity to almost 1.6 million users, mainly including three metropolises, Sydney, Newcastle and Wollongong. The original data is composed of settlement date, totol load demands, electricity price, period type. Obviously, they are insufficient as a complete basis for load forecasting. It is universally acknowledged that meteorological factors are particularly prevalent in affecting load variation, therefore, meteorological factors are supplemented at the website https://www.wunderground.com, including temperature (drybulb temperature, dewpoint temperature, wetbulb temperature), humidity, precipitation, wind speed, pressure and so on. Besides, seek for calendar to determine the date type due to the massive distinction between weekdays and weekends, holidays and non-holidays. Every 0.5h is a time step, so there are 48 time indexes in one day. The data is distributed from January 01, 2006 to December 31, 2010, with a total of 1826*48 rows of data, furthermore, the columns represent total load features (Historical load and influencing factors).

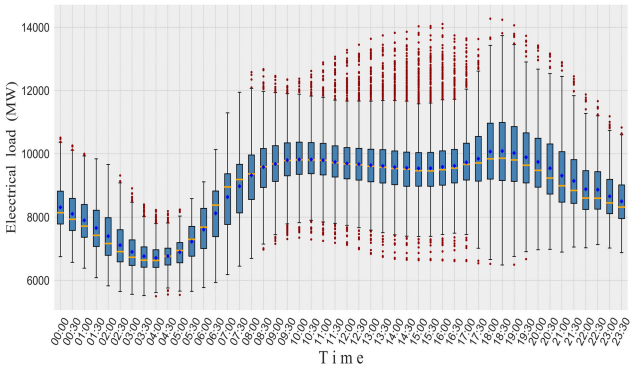


FIGURE 6. The distribution of whole load data in the dataset for per half hour.

The distribution of the whole electrical data is presented in the Fig.6. From the box-whisker plot, it seems that there are numerous outliers from 9:00 to 17:00, but this is not necessarily a rotten matter, the difference of classification and difficulty level of prediction are highlighted. At the same time, it should attach importance to noted that the dimensions of power load data are all on the thousand or ten thousand scale, too large value retard speed of finding the optimal solution by the gradient descent method. To overcome this obstacle, normalization which converts the data to [0,1] is utilized to simplify calculation. Assuming that the maximum and minimum values of data x are $x_{\max}$, $x_{\min}$ respectively, the normalized data $\hat{x}$ calculation formula is as follows:

$$\hat{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{20}$$

TABLE 1. The table of segmental fuzzy mapping results of influencing factors.

Influencing factors	Original physical scale	Fuzzy mapping results
Drybult temp	$T_{\min} \leq T \leq T_{\max}$	$(T - T_{\min})/(T_{\max} - T_{\min})$
Humidity	$H_{\min} \leq H \leq H_{\max}$	$(H - H_{\min})/(H_{\max} - H_{\min})$
Wind speed	$W_{\min} \leq W \leq W_{\max}$	$0.5[(W - W_{\min})/(W_{\max} - W_{\min})]$
Precipitation	$Pc_{\min} \leq Pc \leq Pc_{\max}$	$0.5[1 - (Pc - Pc_{\min})/(Pc_{\max} - Pc_{\min})]$
Pressure	$Ps_{\min} \leq Ps \leq Ps_{\max}$	$0.5[(Ps - Ps_{\min})/(Ps_{\max} - Ps_{\min})]$
Date type	$weekdays$	1
Date type	$weekends$	0.5
...	...	...

Meteorology is the most ordinarily wielded influencing factors in STLF, since they have impact on economic activities (industrial, residential, commercial and agriculture), so as to indirectly affect electricity consumption. For instance, when the temperature drops low enough, more energies are required to increase the comfort index of human body (CIHB) which leads to boom load demand. In addition, agricultural electricity is the most sensitive to precipitation. However, These meteorological factors that keep different weighting impacts on load variation have diverse units, such as temperature (°C) and wind speed (mph). On weekends or holidays people live with more recreational activities, while on weekdays and non-holidays they tend to be obsessed with subsistence, which justifies why date types should be taken into account. It is also strenuous to input date types directly into our forecasting model as they have no units.

Through fuzzification of influencing factors, a kind of mapping is established to conquer the obstacle of no unit or different units. The result of fuzzy mapping is determined according to the Pearson correlation coefficient (PCC) which is an examination of the degree of correlation between two domains [36]. The PCC formula between n-dimensional data X and Y is defined by Eq.(21):

$$PCC(X, Y) = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n} (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^{n} (Y_i - \bar{Y})^2}} \tag{21}$$

where $\bar{X}$ and $\bar{Y}$ are the mean value of X and Y separately. PCC values range from -1 to 1, where its sign represent positive or negative correlations, and the magnitude of the absolute value depends on correlational strength. For the sake of clarity, PCC between load data and segmental factors is exhibited in heat map Fig.7.

It can be markedly noted that the absolute value of characteristic PCC scopes from 0.11 to 0.73 in the first column. The median absolute value of 0.42 is applied as a demarcation for the correlational strength. In other words, if the absolute value of PCC between load and a factor is greater than 0.42, this factor is defined as a strong correlation,such as the above drybulb temperature, humidity and date type, otherwise, the opposite is defined as a weak correlation, such as wind speed, precipitation and pressure.

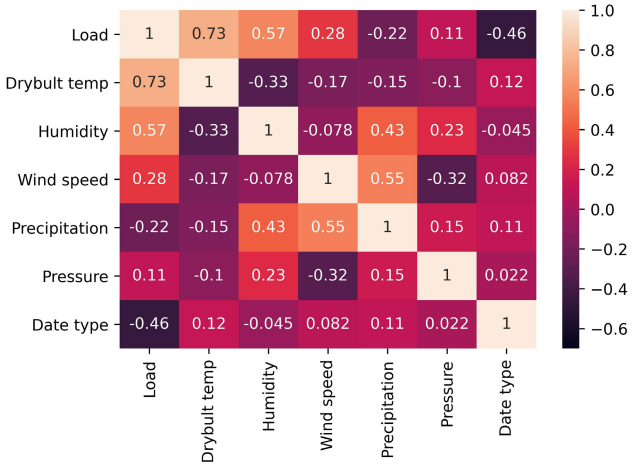


FIGURE 7. The heat map of PCC between load and segmental factors.

Furthest behind, the fuzzy mapping result is determined by the correlational strength and positive-negative correlation, in which the strong correlation maps the original physical scale to [0,1], the weak correlation maps to [0, 0.5]. Positive correlation mapping results are proportional to the original physical scale and negative correlation is inversely proportional. After fixing the two parameters, it refers numerically to the normalization. This fuzzy mapping approach not only conquers the lack of outright input influencing factors into the prediction model caused by different units or no units, but also avoids the slow gradient descent as normalization. The segmental fuzzy mapping table of influencing factors in this dataset is shown below Table.1.

B. DAILY LOAD CURVE CLUSTERING

Clustering technology can excavate the archetypal power consumption characteristics from enormous load data, and supply sovereign support for power grid companies to achieve load forecasting and demand side management. A significant step of daily load curve clustering is to determine the number

of clusters in the PSO-KFCM algorithm mentioned above. There are various methods to ascertain the number of clusters, for instance, within-cluster sum of squared errors (SSE), partitioning around medoids (PAM), gap statistic (GS) [37]. In this article, the comprehensive SSE method is adopted to accomplish this assignment. The analytical formula of SSE is as follows:

$$SSE(k) = \sum_{i=1}^{k} \sum_{x \in p_i} \|c_i - x\|_2^2 \tag{22}$$

where c_i is the i-th clustering center, p_i represents the muster of data points in the i-th cluster. With the increase of cluster number k, the sample partition will be more meticulous, and the aggregation degree of each cluster will gradually amend, so SSE will naturally become smaller. Theoretically, the smaller the SSE value is, the better the clustering effect will be. However, when k increases to a certain extent, the effect on the decrease of SSE is rare. Therefore, the k value near the inflection point of the curve is normally the appropriate number of clusters. In this clustering experiment, Set the particle population size $N = 100$, maximum iteration steps $M = 100$, RBF kernel parameters $\sigma = 150$, termination error $\delta = 1e-4$ and the range of k is programmed from 1 to 10, and the SSE broken line is exhibited in the below Fig.8. Evidently, the most appropriate number of clusters is recognized as 6, since there is no abrupt inflection point nearby and it tends to the minimum. After that, this novel algorithm begins to cluster the preprocessing load data with 48 time steps.

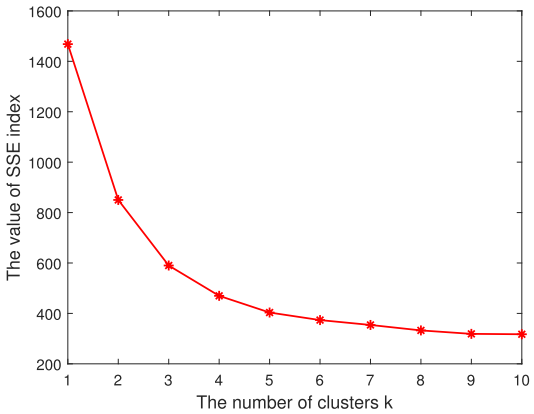


FIGURE 8. The gradient graph of SSE index with increasing cluster number k from 1 to 10.

The daily load curve after PSO-KFCM clustering is shown in the Fig.9, where numerous daily loads are represented by different color curves. In order to prove the superiority of this algorithm, FCM, KFCM, GA-KFCM algorithm are contrasted with proposed method. Four internal indexes [38], [39] are adopted to evaluate the validity of clustering as shown in Eq.(23)-Eq.(26).

1) Silhouette coefficient (SC): The range of coefficient is between [-1,1], and the closer to 1, the better the clustering performance. a_i is the average distance between the sample i and other points in the same category, and b_i is the minimum average distance from sample i to other clusters

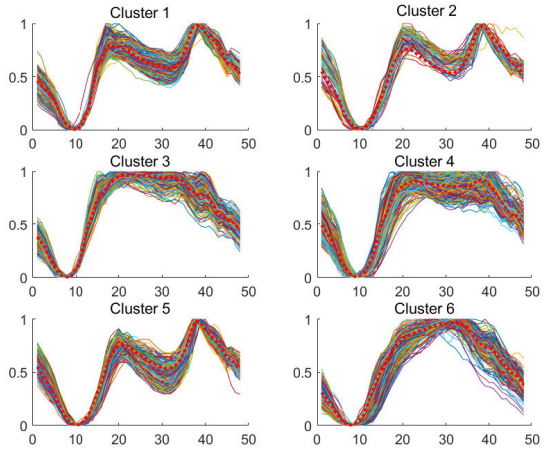


FIGURE 9. The daily load curve after PSO-KFCM clustering when the number of clusters is determined to be 6.

2) Davies-Bouldin index (DB): DB index describes the distance between the clustering centers and the within cluster divergence of samples. The smaller the index, the better the clustering effect. s_i represents the average distance between samples in cluster i.
3) Calinski-Harabasz index (CH): CH index is obtained by the ratio of compactness to separation. Thus, the larger the index, the more compact it is. $Tr(B_k)$ denotes the trace of between-clusters dispersion mean matrix and $Tr(W_k)$ represents the trace of within-cluster dispersion matrix.
4) Krzanowski-Lai index (KL): KL index can only be applied to calculate clusters of two categories and above. In order to achieve the best clustering effect, KL index should be as large as possible. W_k is the sum of the squares of the distances from the clustering interior point to clustering center.

$$SC = \frac{1}{n} \sum_{i=1}^{n} \frac{b_i - a_i}{\max(a_i, b_i)} \tag{23}$$

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{j=1 \ldots k, i \neq j} \frac{s_i + s_j}{\left\|c_i - c_j\right\|_2^2} \tag{24}$$

$$CH = \frac{Tr(B_k)}{Tr(W_k)} \times \frac{n-k}{k-1} \tag{25}$$

$$KL = \left| \frac{(k-1)^{\frac{2}{p}} W_{k-1} - k^{\frac{2}{p}} W_k}{k^{\frac{2}{p}} W_k - (k+1)^{\frac{2}{p}} W_{k+1}} \right| \tag{26}$$

Their comparison indicators are clearly displayed in Table.2 and Fig.10. It can be seen from the results that the three indices of SC, CH, KL are the highest and DB is the lowest with 0.496, 1120.465, 1.575, 1.001, respectively, proving that all the clustering validity is preferred over the other three proposed methods. In addition, the clustering

TABLE 2. The comparative table of four clustering validity indicators, SC, DB, CH and KL between FCM, KFCM, GA-KFCM and PSO-KFCM method.

Methods	SC	DB	CH	KL
FCM	0.376	1.342	801.314	1.045
KFCM	0.417	1.244	987.454	1.187
GA-KFCM	0.452	1.117	1055.001	1.236
PSO-KFCM	**0.496**	**1.001**	**1120.465**	**1.575**

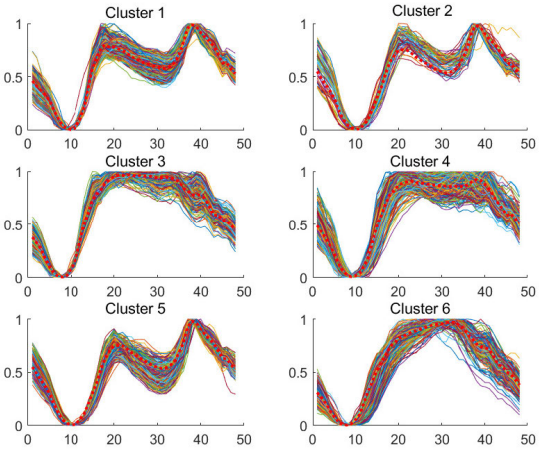


FIGURE 10. The comparative graph of four clustering validity indicators, SC, DB, CH and KL between FCM, KFCM, GA-KFCM and PSO-KFCM method.

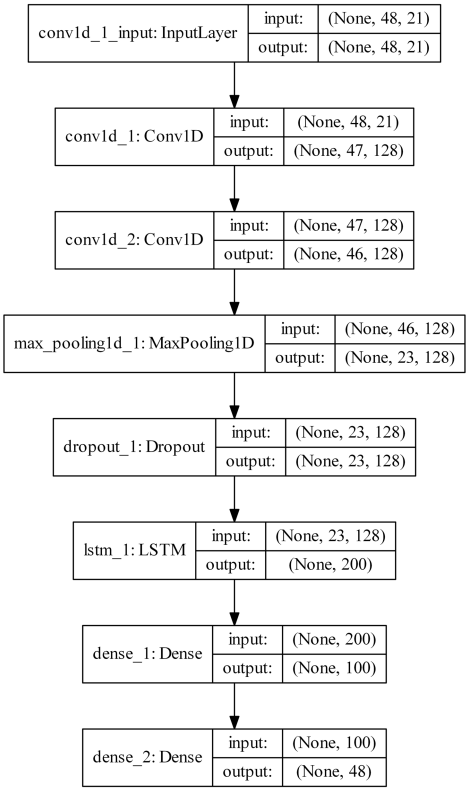


FIGURE 11. The network structure of the proposed multivariable and multi-step CNN-LSTM model.

effect is enhanced by 31.9% compared with the traditional FCM algorithm in term of SC. From the comparison results above, the effectiveness of the proposed method is verified.

C. PREDICTIVE MODELING

The dataset is divided into 80% training set (January 01, 2006 to December 31, 2009,1461 days) and 20% (January 01, 2010 to December 31, 2010,365 days) test set. Among them, the training set is employed for daily load curve clustering and training prediction model, the test set is used to determine the similar cluster and as the input in the trained model.

In this paper, the default prediction model is multivariable and multi-step CNN-LSTM (MMCNN-LSTM). The overall network composition is distinctly expressed in Fig.11. Each time step has a total of 21 dimensions of load features, such as the current moment L_t, the previous moment L_{t-1}, the first 2 moments L_{t-2}, the first 48 moments L_{t-48} of load, ..., the current temperature T_t, humidity H_t, wind speed W_t, ..., electricity price P_t etc. The input is a 48*21 matrix representing time steps and the number of load features. A double convolution layer with 128 convolution kernel of size 2 extracts features and the extracted time series are compressed by a max pooling layer of size 2. To avoid the model from overfitting, dropout layers are added with a probability of 0.1. Set the amount of LSTM hidden layer units to 200, since the load is only predicted one day ahead, the output displayed is 48 dimensions. These network parameters are continuously adjusted through predictive experiments before they are obtained.

With the purpose of certifying the hybrid model's superiority proposed in this paper, three indexes are selected for prediction evaluation, namely, root mean square error (RMSE), mean absolute error (MAE), mean absolute percent error (MAPE) [40]. Suppose the original data is y and the predicted data is $\hat{y}$, these calculation formulas are shown as Eq.(27)-Eq.(29) respectively.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{27}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right| \tag{28}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \tag{29}$$

IV. EXPERIMENTS AND RESULTS

A. EXPERIMENT I

The predictive experiment platform is Jupyter Notebook, the framework are Tensorflow (GPU) and Keras, and the device configuration is NVIDIA Titan xp and Intel(R) Xeon(R) CPU E5-2620 and RAM 16G.

In the first experiment, the PSO-KFCM clusering model and the model without pretreatment were compared while

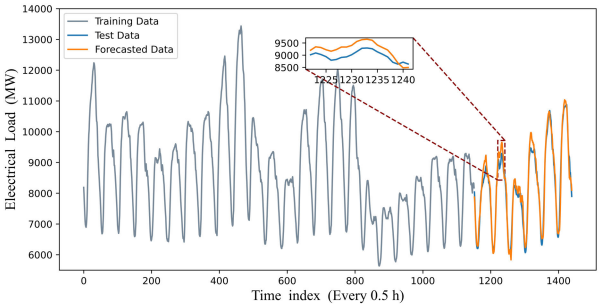
keeping the other variables the same. Selected local 30-day data, a total of 30*48 time steps as observations. When an unpreprocessed forecast is settled upon, the observations are the last 24 days of the training set and the first 6 days of the test set. When the PSO-KFCM clustering data are imported into prediction model, the observations are the last 24 days of a certain cluster in the training set and the first 6 days of a similar cluster based on cosine similarity in the test set. The clustering data has been given in mentioned Section III(B), and the cluster label "Cluster 5" is selected in this investigation.

The two vectors of cosine similarity are determined as influencing factor of the predicted day and average influencing factor of within-cluster. The prediction model's input is in form of daily maximum temperature $T_{\max}$, minimum temperature $T_{\min}$, average temperature T_{avg}, maximum humidity $H_{\max}$, minimum humidity $H_{\min}$, average humidity H_{avg} and so on, within-cluster is the average of all days in the same format. The determination of similar cluster is based on the maximum cosine similarity principle. Among them, the cosine similarity data operated to determine the similarity of two samples (Two of the six days) are shown in Table.3. According to the two largest cosine similarity, 0.925 and 0.897, respectively, it can be indeed confirm that the two test samples belong to "Cluster 5".

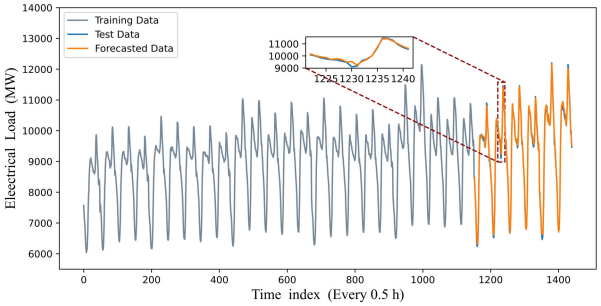
TABLE 3. The cosine similarity belonging to 6 clusters and determination of similar cluster.

Sample	Cosine Similarity						Similar Cluster
	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	
Sample1	0.884	0.852	0.567	0.426	**0.925**	0.354	5
Sample2	0.876	0.815	0.516	0.482	**0.897**	0.315	5

The comparative graph of 6-day forecast data between the model without pretreatment and the PSO-KFCM clustering model are clearly depicted in Fig.12(a) and Fig.12(b) respectively. Obviously, the data after PSO-KFCM clustering present a regular periodicity, which relatively alleviate the predictive hindrance. Through local amplification, especially the prediction accuracy of peak and valley values have been greatly upgraded. The detailed RSME, MAE and MAPE of the two methods are presented in Table.4(a) and Table.4(b) respectively. It should be noted that the maximum MAPE value of the clustering model 0.83% is also fewer than the minimum of the model without pretreatment 1.81% within these six days. On the whole, the three indexes of the model after pretreatment are 82.71, 61.86 and 0.67% on average, which is almost one-third of the unpreprocessed model. Compared with the unpreprocessed model, the average MAPE value of the assembled model is acutely dropped by 1.51%, which amply verifies the superiority of PSO-KFCM pretreatment and method for determining similar clusters by cosine similarity. In order to further observe the prediction accuracy, experiments II and III were carried out.



(a) The forecast data of moedel without pretreatment



(b) The forecast data of PSO-KFCM clustering model

FIGURE 12. The comparative graph of 6-day forecast data between the model without pretreatment and the PSO-KFCM clustering model.

TABLE 4. The comparative table of three predictive evaluation indexes, RMSE, MAE, MAPE between the model without pretreatment and the PSO-KFCM clustering model.

(a) The predictive evaluation indicators of model without pretreatment.

Indexs	Day1	Day2	Day3	Day4	Day5	Day6	Average
RMSE	228.00	221.15	218.47	240.18	183.29	250.87	223.66
MAE	180.39	176.48	178.56	207.94	153.46	174.32	178.85
MAPE	2.27%	2.11%	2.42%	2.49%	1.81%	2.00%	2.18%

(b) The predictive evaluation indicators of PSO-KFCM clustering model.

Indexs	Day1	Day2	Day3	Day4	Day5	Day6	Average
RMSE	87.94	95.53	66.42	81.02	93.91	71.70	82.71
MAE	65.61	69.67	46.91	57.72	75.60	55.66	61.86
MAPE	0.77%	0.78%	0.52%	0.61%	0.83%	0.55%	0.67%

B. EXPERIMENT II

To certify the supremacy of multivariate and multi-step input approach on the foundation of original clustering logic, compare with the other three input modes, namely, univariate and single-step CNN-LSTM (USCNN-LSTM), univariate and multi-step CNN-LSTM (UMCNN-LSTM), multivariable and single-step CNN-LSTM (MSCNN-LSTM). The univariate model represents only a single historical load data as a theoretical support without taking into account all influencing factors let alone fuzzy mapping. Similarly, just one step data will be predicted in single-step model, even insufficient outputs result in overlapping predictions. In order to preserve the consistency of other settings, set the optimizer to 'Adam', the learning rate is '0.001' and the maximum number of iterations is 100.

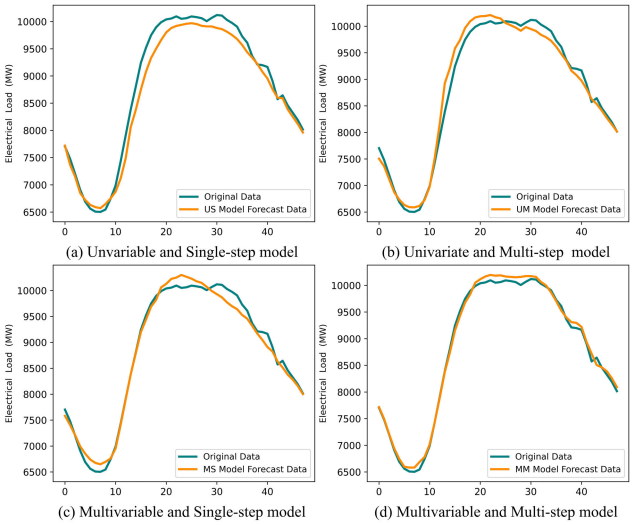


FIGURE 13. The comparative graph of CNN-LSTM model between four different input modes on January 14, 2010.

TABLE 5. The comparative table of three predictive evaluation indexes, RMSE, MAE, MAPE between four different input modes on January 14, 2010.

Models	RMSE	MAE	MAPE
US Model	211.98	168.71	1.85%
UM Model	163.21	128.27	1.44%
MS Model	136.34	111.96	1.26%
MM Model	**76.99**	**65.75**	**0.74%**

January 14, 2010 was selected as the prediction date. Using the above clustering data and calculating cosine similarity, the label of similar cluster is determined as "Cluster 6". Fig.13 indicates the comparisons between four different input modes on that day and the complicated indexes are shown in Table.5 and Fig.14. From the chart above, it can be clearly observed that the outcomes of multivariate model are more likely to superior than those of the univariate model in case of keeping output step size constant. From the perspective of elaborate MAPE value, The MS model decreases by 0.59% over the US model and the largest accurate enhancement is in MM model, which is a 0.7% dramatic decline compared to UM model. This justifies seamlessly why it is necessary to consider numerous influencing factors and fuzzy mapping, which provides with a admirable theoretical basis. Maintaining the input variables as constant, US model and UM model, MS model and MM model are compared, and the precise improvement is between 0.41% and 0.52% in term of MAPE. Multi-step models are acknowledged due to precise accuracy and prominent applicability to the STLF domain for multi-time forecasting.

In this entire procedure, the time taken by the US model (Training and prediction), UM model (Training and prediction), MS model (Clustering, training and prediction) and MM (Clustering, training and prediction) model are 20s, 22s, 30s and 33s, respectively. Although the present model spends more time in training and clustering process than

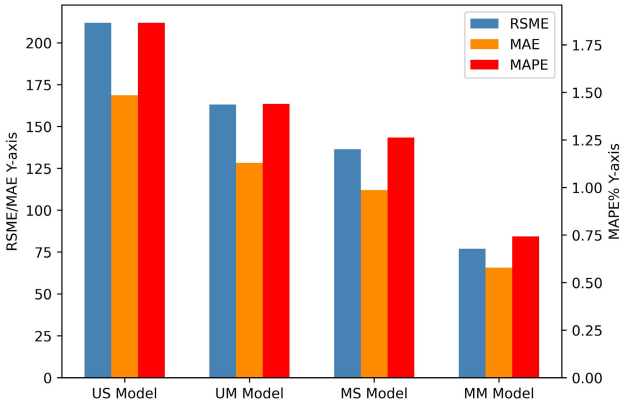


FIGURE 14. The comparative graph of three predictive evaluation indexes, RMSE, MAE, MAPE between four different input modes on January 14, 2010.

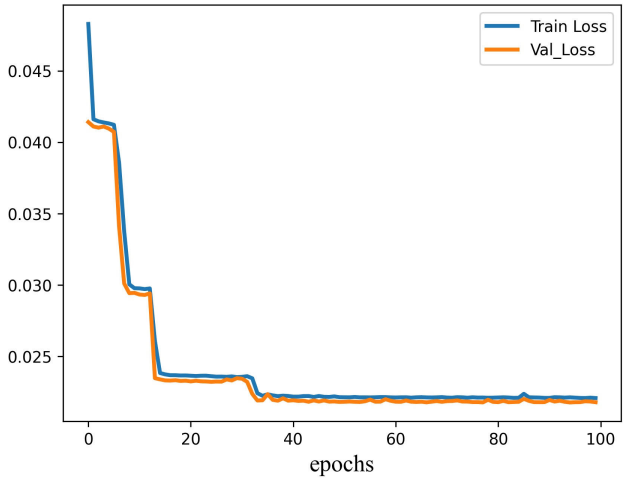


FIGURE 15. The curve chart of train loss and validation loss in MMCNN-LSTM model.

its partner, the high accuracy criterion still better meets the realistic demands. The mean square error (MSE) loss curve of MMCNN-LSTM model is exhibited in Fig.15. The train loss and validation loss are evidently decline and eventually stabilize. Combined with the fitted curves, which are almost identical in magnitude and direction, these are distinguished signs that it has outstanding fitting ability and appropriate network parameter settings. In a nutshell, the results clearly indicate that the proposed MMCNN-LSTM model is more capable of finishing the STLF task than its counterparts regardless of precision and adaptability.

C. EXPERIMENT III

In the first two experiments, the advantages of clustering preprocessing and the optimal input pattern have been demonstrated. The overall effect will verified by comparing with alternative unclustered DNN-based models in the following step. Multilayer perceptron (MLP), gate recurrent unit (GRU), bidirectional recurrent neural network (Bi-RNN), extreme gradient boosting (XGBoost) and

TABLE 6. The comparative table of max MAPE, min MAPE and average MAPE of the entire test set between the proposed model and other DNN baseline models.

Models	Max MAPE	Min MAPE	Average MAPE
MLP Model	3.82%	1.88%	2.73%
GRU Model	6.88%	2.59%	4.65%
Bi-RNN Model	4.51%	1.83%	3.39%
LSTM Model	5.46%	3.06%	4.36%
XGboost Model	3.08%	1.15%	2.16%
Proposed	**2.40%**	**0.48%**	**1.34%**

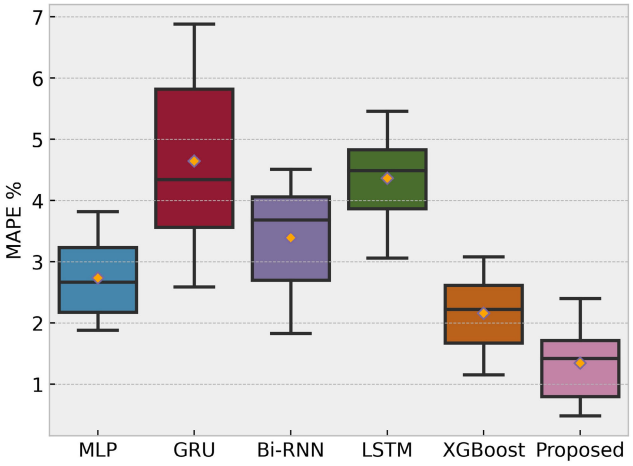


FIGURE 16. The MAPE distribution graph of the entire test set between the proposed model and other DNN baseline models.

conventional LSTM with certain impacts in the domain of time series forecasting have been handpicked. Taken as a whole, the entire test set was determined to be predicted since MAPE existed for each day.

Table.6 indicates the detailed evaluation parameters of the six models in terms of max MAPE, min MAPE, average MAPE and Fig.16 provides a visual representation of the MAPE distribution. From these six groups of results, it is obvious that poor presentation of the single LSTM model and GRU model with maximum mape higher than 5%. As the more prevalent XGboost model whose average mape reaches 2.16% in deep learning over recent years has a 20.9% heightened accuracy over the traditional MLP model, which is second only to the novelty admixture. As can be see, the proposed method produces better prediction results, with a decent in average MAPE value from 0.82% to 3.31%. The results show that the proposed method significantly upgrades predictive accuracy in comparison with unprocessed DNN baseline models. In addition, the difference between the maximum and minimum is only 1.92%, however, the GRU and Bi-RNN model are 4.29%,2.68% respectively, which proves to play prettily in the stability of predictions as well.

V. CONCLUSION

Short-term load forecasting is a basic work for daily operation of power grid. This paper presents a STLF method based on PSO-KFCM daily load curve clustering and CNN-LSTM model. This comprehensive technique taken historical load data and influence factors (meteorology, date type, economy and others) into account, where historical load data were normalized and influencing factors were fuzzy mapped according to the Pearson correlation coefficient. The novel PSO-KFCM algorithm clusterd the preprocessed daily load curves, which not only solved the problem of sensitivity of the initial clustering center, but also greatly improved the clustering quality. The clustering experiment shown that the number of clusters was determined as 6 by sum of squared error index and the 31.9% improvement in Silhouette coefficient over conventional FCM algorithm. Besides the Silhouette coefficient, Davies-Bouldin index, Calinski-Harabasz index and Krzanowski-Lai index were operated in clustering validity indicators as well. The cosine simlarity mainly for multidimensional positive space was selected as the indelible bridge between clustering label and prediction model. Multivariate and multi-step CNN-LSTM was focused on predicting load data for the next 24h in half-hourly steps and the accuracy was verified by root mean square error, mean absolute error, mean absolute percent error. This hybrid prediction integrates the advantages of both, feature extraction capability and long time series processing potential. Finally, contrasted with the model without clustering, three other input models, and five DNN baseline models, the extensive comparative results have confirmed the high-precision and excellent practicality and stability of the proposed model.

The PSO-KFCM method and CNN-LSTM model proposed in this paper are not only limited to short-term load forecasting, but also can be applied to other deep learning contents, such as bearing fault diagnosis, signal pattern recognition, intelligent visual sorting, etc.

REFERENCES

[1] S. K. Panda, A. K. Jagadev, and S. N. Mohanty, "Forecasting methods in electric power sector," *Int. J. Energy Optim. Eng.*, vol. 7, no. 1, pp. 1–21, Jan. 2018.
[2] A. H. Sanstad, S. Mcmenamin, A. Sukenik, G. L. Barbose, and C. A. Goldman, "Modeling an aggressive energy-efficiency scenario in long-range load forecasting for electric power transmission planning," *Appl. Energy*, vol. 128, pp. 265–276, Sep. 2014.
[3] Y. Chen, P. Xu, Y. Chu, W. Li, Y. Wu, L. Ni, Y. Bao, and K. Wang, "Short-term electrical load forecasting using the support vector regression (SVR) model to calculate the demand response baseline for office buildings," *Appl. Energy*, vol. 195, pp. 659–670, Jun. 2017.
[4] G. Hartless, J. G. Booth, and R. C. Littell, "Local influence of predictors in multiple linear regression," *Technometrics*, vol. 45, no. 4, pp. 326–332, Nov. 2003.
[5] S. S. Pappas, L. Ekonomou, P. Karampelas, D. C. Karamousantas, S. K. Katsikas, G. E. Chatzarakis, and P. D. Skafidas, "Electricity demand load forecasting of the hellenic power system using an ARMA model," *Electric Power Syst. Res.*, vol. 80, no. 3, pp. 256–264, Mar. 2010.
[6] C.-M. Lee and C.-N. Ko, "Short-term load forecasting using lifting scheme and ARIMA models," *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5902–5911, May 2011.
[7] Ö. Ö. Bozkurt, G. Biricik, and Z. C. Tayi, "Artificial neural network and SARIMA based models for power load forecasting in turkish electricity market," *PLoS ONE*, vol. 12, no. 4, Apr. 2017, Art. no. e0175915.
[8] H. Cui and X. Peng, "Summer short-term load forecasting based on arimax model," *Dianli Xitong Baohu Yu Kongzhi/Power Syst. Protection Control*, vol. 43, no. 4, pp. 108–114, 2015.

[9] J. Hua, W. Xiong, and Y. Zhou, "Short-term load forecasting of local power grid based on support vector machine," in *Proc. 7th Int. Conf. Educ., Manage., Comput. Soc. (EMCS)*, 2017, pp. 1–6.

[10] D. L. Li, X. F. Zhang, M. Z. Qiao, and G. Cheng, "A short-term load forecasting method of warship based on PSO-SVM method," *Appl. Mech. Mater.*, vol. 127, pp. 569–574, Oct. 2011.

[11] J. Wang, R. Ran, Z. Song, and J. Sun, "Short-term photovoltaic power generation forecasting based on environmental factors and GA-SVM," *J. Electr. Eng. Technol.*, vol. 12, no. 1, pp. 64–71, Jan. 2017.

[12] A. Kavousi-Fard, H. Samet, and F. Marzbani, "A new hybrid modified firefly algorithm and support vector regression model for accurate short term load forecasting," *Expert Syst. Appl.*, vol. 41, no. 13, pp. 6047–6056, Oct. 2014.

[13] A. Zhang, P. Zhang, and Y. Feng, "Short-term load forecasting for microgrids based on DA-SVM," *COMPEL Int. J. Comput. Math. Electr. Electron. Eng.*, vol. 38, no. 1, pp. 68–80, Jan. 2019.

[14] A. Yang, W. Li, and X. Yang, "Short-term electricity load forecasting based on feature selection and least squares support vector machines," *Knowl.-Based Syst.*, vol. 163, pp. 159–173, Jan. 2019.

[15] D. C. Park, M. A. El-Sharkawi, R. J. Marks, L. E. Atlas, and M. J. Damborg, "Electric load forecasting using an artificial neural network," *IEEE Trans. Power Syst.*, vol. 6, no. 2, pp. 442–449, May 1991.

[16] J. Moon, J. Kim, P. Kang, and E. Hwang, "Solving the cold-start problem in short-term load forecasting using tree-based methods," *Energies*, vol. 13, no. 4, p. 886, Feb. 2020.

[17] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—A novel pooling deep RNN," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 5271–5280, Sep. 2018.

[18] J. Liu, X. Wang, Y. Zhao, B. Dong, K. Lu, and R. Wang, "Heating load forecasting for combined heat and power plants via strand-based LSTM," *IEEE Access*, vol. 8, pp. 33360–33369, 2020.

[19] C. Li, G. Tang, X. Xue, A. Saeed, and X. Hu, "Short-term wind speed interval prediction based on ensemble GRU model," *IEEE Trans. Sustain. Energy*, vol. 11, no. 3, pp. 1370–1380, Jul. 2020.

[20] X. Tang, Y. Dai, Q. Liu, X. Dang, and J. Xu, "Application of bidirectional recurrent neural network combined with deep belief network in short-term load forecasting," *IEEE Access*, vol. 7, pp. 160660–160670, 2019.

[21] Z. Zhong, C. Yang, W. Cao, and C. Yan, "Short-term photovoltaic power generation forecasting based on multivariable grey theory model with parameter optimization," *Math. Problems Eng.*, vol. 2017, pp. 1–9, Jan. 2017.

[22] Y. Li and T. Fang, "Wavelet and support vector machines for shortterm electrical load forecasting," in *Proc. Int. Conf. Wavelet Anal. Appl.*, 2015, p. 399.

[23] N. Pin-Lei, F. Dong, W. Hong-Jie, and S. Tao, "Short-term power load forecasting based on EMD-BP neural network," *Control Instrum. Chem. Ind.*, vol. 43, no. 3, pp. 305–307, 2016.

[24] Y.-Y. Cheng, P. P. K. Chan, and Z.-W. Qiu, "Random forest based ensemble system for short term load forecasting," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Jul. 2012, pp. 52–56.

[25] J. C. Bezdek, R. Ehrlich, and W. Full, "Fcm: The fuzzy C-means clustering algorithm," *Comput. Geosci.*, vol. 10, nos. 2–3, pp. 191–203, 1984.

[26] F. Benabbas and M. T. Khadir, "Fuzzy C-Means clustering and kohonen maps for the identification of regional electricity load day types," *Int. J. Hybrid Intell. Syst.*, vol. 8, no. 2, pp. 81–92, May 2011.

[27] J. Hwang and S. Miyamoto, "Kernel functions derived from fuzzy clustering and their application to kernel fuzzy c-means," *J. Adv. Comput. Intell. Intell. Inform.*, vol. 15, no. 1, pp. 90–94, 2011.

[28] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*. Berlin, Germany: Springer, 2003.

[29] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. MHS 6th Int. Symp. Micro Mach. Hum. Sci.*, Oct. 1995, pp. 39–43.

[30] P. N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, 1st ed. Reading, MA, USA: Addison-Wesley, 2005.

[31] S. Pattnaik and A. K. Nayak, "Summarization of odia text document using cosine similarity and clustering," in *Proc. Int. Conf. Appl. Mach. Learn. (ICAML)*, May 2019, pp. 143–146.

[32] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol. Cybern.*, vol. 36, no. 4, pp. 193–202, Apr. 1980.

[33] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[35] W.-S. Jhang, S.-E. Gao, C.-M. Wang, and M.-C. Hsieh, "Share price trend prediction using attention with LSTM structure," in *Proc. 20th IEEE/ACIS Int. Conf. Softw. Eng., Artif. Intell., Netw. Parallel/Distrib. Comput. (SNPD)*, Jul. 2019.

[36] S.-M. Jung, S. Park, S.-W. Jung, and E. Hwang, "Monthly electric load forecasting using transfer learning for smart cities," *Sustainability*, vol. 12, no. 16, p. 6364, Aug. 2020.

[37] H. He, Y. Tan, and K. Fujimoto, "Estimation of optimal cluster number for fuzzy clustering with combined fuzzy entropy index," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2016, pp. 697–703.

[38] M.-D. Yang, C.-H. Hsu, and T.-C. Su, "Optimal cluster numbers of unsupervised classification in minkowski spaces," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2007, pp. 2044–2047.

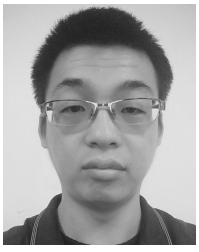
[39] C. Tomasini, L. Emmendorfer, E. N. Borges, and K. Machado, "A methodology for selecting the most suitable cluster validation internal indices," in *Proc. 31st Annu. ACM Symp. Appl. Comput.*, Apr. 2016, pp. 901–903.

[40] X. Shao and C. S. Kim, "Multi-step short-term power consumption forecasting using multi-channel LSTM with time location considering customer behavior," *IEEE Access*, vol. 8, pp. 125263–125273, 2020.

CHUAN SHANG received the B.S. degree in automation from Shandong Jiaotong University, Jinan, China, in 2019. He is currently pursuing the M.S. degree in control engineering with Qingdao University, Qingdao, China. His research interests include intelligent systems and intelligent control.

JUNWEI GAO received the B.S. degree in electrical engineering from the Shandong University of Technology, Zibo, China, in 1995, the M.S. degree in control theory and control engineering from the Lanzhou University of Technology, Lanzhou, China, in 2000, and the Ph.D. degree in traffic information engineering and control from the China Academy of Railway Sciences, Beijing, China, in 2003. From 2004 to 2011, he was an Associate Professor with the College of Automation, Qingdao University, Qingdao, China, where he has been a Professor, since 2012. His current research interests include intelligent systems and intelligent control.

HUABO LIU (Member, IEEE) received the B.S. degree in automation and the M.S. degree in automation and detection technology and automation equipment from Chongqing University, Chongqing, in 2001 and 2005, respectively, and the Ph.D. degree in control science and engineering from Tsinghua University, Beijing, in 2016. He currently serves with the School of Automation, Qingdao University, Qingdao. His current research interests include large-scale networked systems, hybrid systems, robust state estimation, and their applications to practical engineering problems.



FUZHENG LIU received the B.S. degree in automation from Qingdao University, Qingdao, China, in 2016, where he is currently pursuing the M.S. degree in control science and engineering. His research interests include intelligent systems and intelligent control.