

RESEARCH

Open Access



A deep learning-based load forecasting algorithm for energy consumption monitoring system using dimension expansion

Wei-guo Zhang^{1*} , Qing Zhu^{2,3}, Lin-Lin Gu^{2,3} and Hui-Jie Lin^{2,3}

*Correspondence:
weiguoseu@126.com

¹ Southeast University,
Nanjing 210096, China

² Nari Technology Co., Ltd.,
Nanjing 211106, China

³ NARI Nanjing Control System
Co., Ltd., Nanjing 211106, China

Abstract

As a basic task in energy consumption monitoring system, load forecasting has great effects on system operation safety, generation costs and economic benefits. In this paper, a long-term load forecasting algorithm using data dimension expansion and deep feature extraction is proposed. First, the outliers of the meteorological measurements are removed by median filter method, and then the time information is encoded to form the fingerprint of the training data. Next, the full connected network (FCN) is used to expand the dimensions of the fingerprint, and the convolutional neural network (CNN) is used to extract the deep features which can obtain better feature representation. Finally, the FCN, the CNN and regression learning model are combined for jointly offline training. The optimal parameters of these network can be obtained under global solution. Experimental results show that the proposed algorithm has better load forecasting performance than existing methods.

Keywords: Data dimension expansion, Deep learning, Feature extraction, Load forecasting

1 Introduction

With the rapid development of power system, power load forecasting plays an important role in the system operation and planning. Accurate load forecasting can increase the power system operation safety, reduce power generation costs and improve economic benefits [1–3]. Thus, power load forecasting has received much attentions for both academic and industry. Since the power load is affected by weather change, social activities and festival types, it is can be considered as a non-stationary random process in time series. However, since the affected factors generally have a certain periodicity, such as weekly periodicity, monthly periodicity and annual periodicity, it provides a theoretical basic for effective power load forecasting realization.

Recently, previous works for power load forecasting can be divided into three main techniques. The first kind technique is called traditional load forecasting method by Kalman filter method, exponential smoothing method and gray forecasting method [4–6]. The second kind technique is called classical load forecasting method. It contains time series-based method and regression analysis-based method [7]. The last

kind method is called intelligent prediction method. It mainly uses the artificial neural network (ANN), fuzzy theory and machine learning technique for load forecasting [8–10].

With the development of machine learning, deep learning, as a better artificial intelligence technology, has solved many complex pattern recognition problems. It has achieved excellent results in the fields of computer vision, speech recognition, natural language processing, audio recognition and bioinformatics. Deep learning takes advantage of multiple processing layers with complex structures or multiple nonlinear transformations to describe data at a high level. Compared with artificial feature extraction, it can automatically obtain the internal features for better internal information description. Moreover, by learning the data features layer by layer through multi-layer model, it can achieve more effective feature expression [11]. However, little research works are concern on the training data preprocessing, especially for how to expand the data dimension for better feature representation of training data.

In order to solve this problem, in this paper, a long-term load forecasting method based on the data dimension expansion using full connection network is proposed. It can comprehensively use the meteorological and time information to predict the power load. By extracting better depth feature of measured data, the efficiency of offline learning can be improved. The main contributions of this paper are given as follows:

- (1) Different from previous works where the sequence of energy consumptions, the incremental sequence of the time day indices, the corresponding day of week indices and the corresponding binary holiday marks are used for load forecasting [14], in this paper, besides the time information, the meteorological information is considered for load forecasting. Since the meteorological information has great effects on the load consuming, the proposed load forecasting is more suitable for practical application.
- (2) Different from the data preprocessing where K-means clustering methods is used for training data clustering in large data set [12], in this paper, the median filter is used to remove the abnormal meteorological measurements in the training data and reduce the noise influence for prediction process. By integrating the encoding time information and meteorological measurements, the fingerprint of the training data is defined.
- (3) Different from previous work where the 1D CNN-LSTM hybrid model is used for feature extraction [13], in this paper, the full connection network is used to expand the fingerprint dimension for better description of fingerprint at first. And then, the deep learning network is used to extract the depth information of fingerprint automatically. Since the fingerprint is transformed from low-dimensional feature space to high-dimensional feature space, better feature representation can be extracted which can improve the efficiency of offline learning.
- (4) In the proposed algorithm, the full connected network dimension expansion, the deep learning network for feature extraction and the regression model for load forecasting are combined together for offline learning. Through this total learning, the global optimal solution for above three separate network optimization can be obtained. Thus, it can improve the prediction performance dramatically.

The remainder of this paper is organized as follows. Section 2 describes the related works of the proposed algorithm. Section 3 describes the framework of the proposed algorithm. The offline phase description and the online phase description of the proposed algorithm are proposed in Sect. 4 and Sect. 5, respectively. Experiment and performance analysis are illustrated in Sect. 6 and conclusion is given in Sect. 7.

2 Related work

2.1 Traditional load forecasting method

In [4] a blind Kalman filtering algorithm is proposed for real-time load prediction. Through the experimental results, it can be shown that it has considerable advantages over some existing works. Exponential smoothing model is one of the main load forecasting models of power systems, the accuracy of the model depends on smoothing coefficient. In [5], the optimal smoothing coefficient which more weighting for near data and less weighting for far data is proposed for load forecasting. It can achieve good results in power load forecasting. In [6], a load forecasting method based on gray model and regression model with variable weight combination is proposed. It can extend the gray model to medium and long-term load forecasting. In [7], an autoregressive moving average (ARMA) method combined with back propagation neural network is proposed for load forecasting. Since it combines linear and nonlinear components at the same time, the good prediction results can be obtained. In [8], the ANN-based load forecasting method is proposed for short-term load forecasting. Since ANN can be adaptive to a large number of non-structural and inaccurate laws, it can obtain better prediction performance. At present, fuzzy theory is mainly applied to load prediction by fuzzy clustering method and fuzzy similarity priority ratio method. The authors of [9] used fuzzy inductive reasoning for short-term load prediction one day. In [10], a short-term load forecasting model based on an improved fuzzy c-means clustering algorithm, random forest and deep neural network is proposed.

2.2 Deep learning-based load forecasting method

In [12], a convolutional neural network (CNN) with K-means clustering-based load forecasting is proposed. The large training data set is clustered into subsets using K-means algorithm. And then the obtained subsets are used to train the convolutional neural network. The authors of [13] proposed a hybrid neural network combines elements of 1D-CNN and a long short memory network (LSTM) for load prediction. Multiple independent 1D-CNNs are used to extract load, calendar, and weather features while LSTM is used to learn time patterns. In [14], a LSTM recurrent neural network-based framework is proposed to solve the problem of load forecasting. Through the experiments on a publicly available set of real residential smart meter data, it can outperforms the other listed rival algorithms. Through the above analysis, it can be seen that o current deep learning-based research mainly focuses on how to select the appropriate deep learning technology to improve the forecasting performance.

3 Algorithm framework description

According to the block diagram of algorithm framework shown in Fig. 1, the proposed algorithm contains two main phase: offline training phase and online prediction phase. For offline phase, it includes (1) data preprocessing, (2) feature extraction

and (3) offline training. For another, the steps of online phase includes (1) data preprocessing (2) feature extraction and (3) load forecasting. In the following, each steps of the above two phases are described in detail.

4 Offline phase description of the proposed algorithm

4.1 Training data preprocessing

Since the meteorological information in the training data, such as temperature, humidity, pressure and wind speed, are obtained from the corresponding sensors, there will exist some abnormal measurements in the data collection. In order to reduce this affect, in this paper, median filter is used for data preprocessing.

Median filter is one of the main nonlinear signal processing technology using statistical theory, which can effectively remove the outlier. When the median filter is used, the current data in the data sequence is instead by the median value of the corresponding neighborhood [15].

For a given training data sequence of one sensor measurement X_j , the window length is defined as L ($L = 2q + 1$, q is a positive integer).

At time moment k , the training data measurements is written as $x_j(k)$, the data in the window can be described as

$$x_j(k - q), \dots, x_j(k), \dots, x_j(k + q) \tag{1}$$

Arranging the L measurements by ascending order at first, we can obtain the new training data sequence \hat{X}_j , the median value of \hat{X}_j is the filtering result of current data which can be described as

$$x_j(k) = Med(\hat{X}_j) \tag{2}$$

where Med is the median calculation.

Then, the time information in the training data is converted into the fingerprint information by encoding. Usually, the meteorological data are measured 24 h a day. In this paper, the proposed time coding method is shown in Fig. 2. Starting from 0 time of each day, each hour encodes one code with an integer. The output range of the encoder is [0, 23].

After the above data preprocessing, we can obtain the training data which is shown in Fig. 3. In this paper, the fingerprint of the training data includes temperature, humidity, wind speed, pressure and time code with size of 1×5 . The label is the current load.

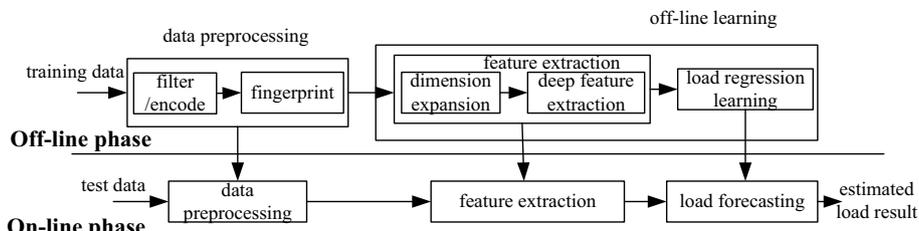


Fig. 1 Block diagram of the proposed algorithm

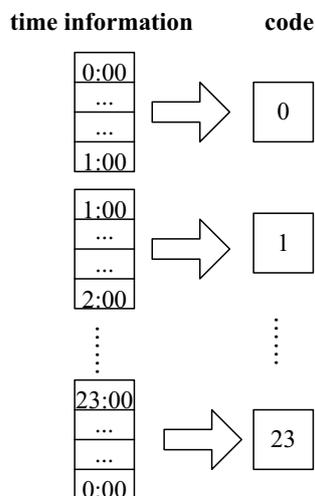


Fig. 2 Schematic diagram of time information coding

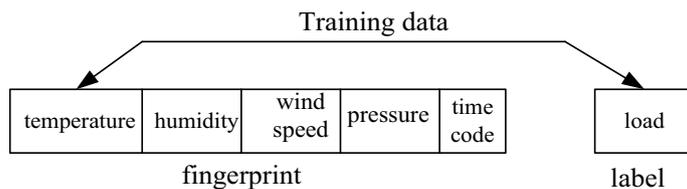


Fig. 3 Format of training data after data preprocessing

4.2 Offline learning

4.2.1 Feature extraction by dimension expansion and deep learning network

In this section, the proposed feature extraction contains two main steps: (1) data dimension expansion by full connected network and (2) feature extraction by deep learning network.

First, according to the block diagram shown in Fig. 4, a full connected network is used to perform data expansion. In this network, the input data of each fully connected layer is transmitted to the next layer by activation function process. In this step, the chosen activation function is the ReLu function which can be defined as [13]

$$f(x) = \max(0, x) \tag{3}$$

If $x > 0$, the output of function is x , otherwise, the output is 0.

For the network design, the number of neurons of the first layer should be equal to the dimension of the initial training data fingerprint. Moreover, the number of neurons of the last layer is the fingerprint dimension after dimension expansion. Two full connected layers are selected to for dimension expansion. In this paper, the dimension is increased from 5 to 64.

Then, the CNN, one of the deep learning network, is used to extract the depth feature of the expanded fingerprint. Figure 5 describes the process of feature extraction. The fingerprint is processed by multiple convolutional layers and pooling layers in turn at first.

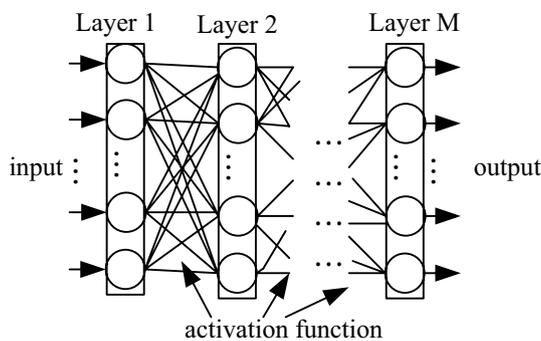


Fig. 4 Schematic diagram of data dimension expansion by full connected network

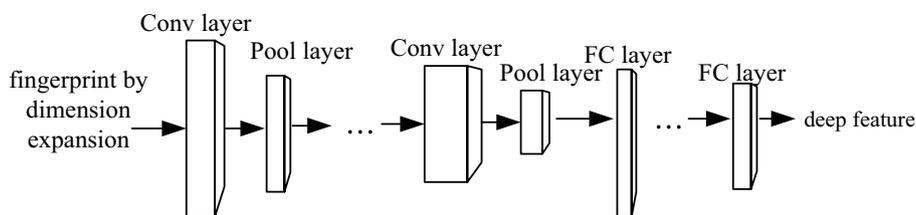


Fig. 5 deep feature extraction process of training data

And then multiple fully connected layers is used to obtain the depth information of the fingerprint.

For the network design, the convolution layer uses convolution kernels to obtain feature maps by convolution operations with the input. Each convolution kernel corresponds to a feature map. And the neurons in the same feature map share the weights and the bias in the filter. At the same time, nonlinear factors are added through the activation function. The pooling layer extracts the main features which compresses the obtained feature map and decreases the computational complexity of the network. The fully connected layer solves the overfitting problem in offline learning and increases the robustness by removing some neurons in the neural network.

In this paper, we transform the 1 * 64 fingerprint into 8 * 8 fingerprint matrix as the input of the convolutional neural network. After 2 convolution layers, 1 pooling layer, and 2 fully connected layers, we obtain 1 * 64-dimensional depth features. The parameters of each layers is summarized in Table 1.

4.2.2 Regression learning

In this section, the linear activation function is chosen to training the relationship between the feature of fingerprint and the load, the regression learning model can be written as

$$q_n = \sum_{i=1}^{\eta} w_i F_{n,i} + b \tag{4}$$

Table 1 The parameters of each layers for feature extraction

	Size of kernel	Number of kernel	stride	padding	active function
Conv layer 1	(1, 1)	32	1	Same	ReLu
Pool layer 1	(2, 2)	–	1	Same	/
Conv layer 2	(3, 3)	64	2	Same	ReLu

where $F_{n,i}$ is the i -th dimension of the n th fingerprint depth feature. w_i is the corresponding weight coefficient. b is the bias, and η is the number of deep feature dimension. q_n is the label (load) of n th training data.

For this model, the mean square error (MSE) is selected as the loss function which is defined as [5]

$$J = \frac{1}{N} \sum_{n=1}^N (q_n - \hat{q}_n)^2 \quad (5)$$

where \hat{q}_n is the estimated load using regression learning model. N is the number of training data.

In offline learning, the full connected network for data dimension expansion, the deep learning network for feature extraction and the regression learning model are jointly trained. At last, optimal parameters of the above network are obtained for online estimation.

5 Online phase description of the proposed algorithm

When each step of the offline phase is achieved, the optimal network parameters of dimension expansion of the training, fingerprint feature extraction and regression learning model are obtained. Thus, the aim of the online phase is to use these optimal models for load forecasting. The steps can be concluded as follows.

First, similar to the data preprocessing in offline phase, the median filter is used to delete the abnormal meteorological measurements. The current time information is encoded with the same method of offline phase. The fingerprint for load forecasting can be described as (temperature, humidity, wind speed, pressure and time code).

Second, the obtained fingerprint is used for load forecasting. The fingerprint is used as the input for the training network. Through the full connected network for data dimension expansion, the deep learning network for feature extraction and regression model, the output is final load prediction result.

6 Experiment and performance analysis

6.1 Experimental setup and environment

In this experiment, the actual load and meteorological data of a residential area in Suzhou Jiangsu Province are chosen for training and testing. These data are measured 24 h a day with an interval of 15 min. 69,304 data from 2015-01-01 to 2016-12-31 (a total of 731 days) are used for training data set. Moreover, from January 1, 2017 to December 31, 2017 (365 days in total) 35,040 data were used for testing data set.

In order to better load forecasting of the proposed algorithm, three different machine learning methods, the ELM method [16], the SVM method [17], the CNN method [18] are used for algorithm comparison.

6.2 Performance index

In this paper, the average absolute percentage (MAPE), mean absolute error (MAE), root mean square error (RMSE) and cumulative error distribution function (CDF) are used to evaluate the load forecasting performance. MAPE, RMSE and MAE which are defined as (6)-(8) [5]. MAPE is a percentage value which is easier to understand than other statistics. RMSE represents the fit standard deviation of the regression system. MAE describes the average absolute error between the predicted value and the actual value. According to Eq. (9), CDF describes the probability of errors occurring in an interval.

$$MAPE = \frac{100\%}{N} \sum_{n=1}^N \left| \frac{\hat{q}_n - q_n}{q_n} \right| \quad (6)$$

$$RMSE = \sqrt{\frac{\sum_{n=1}^N (\hat{q}_n - q_n)^2}{N}} \quad (7)$$

$$MAE = \frac{1}{N} \sum_{n=1}^N |\hat{q}_n - q_n| \quad (8)$$

where q_n, \hat{q}_n are the actual load and predicted load, respectively. N is the number of load to be predicted.

$$F_X(x) = P(X \leq x) \quad (9)$$

where X is the real number.

6.3 Performance analysis

6.3.1 Offline training performance

First, the offline training performance of the proposed algorithm is described. In the experiment, the hardware parameters of computer configuration is described as: CPU: Intel(R) Core(TM) i7-8750H, GPU: Nvidia GTX 1050Ti 4G, memory: 8G × 2. The software is Pycharm (Python 3.5) + TensorFlow 1.8.0 + Keras 2.1.5. According to the offline training performance shown in Fig. 6, as expected, when the number of iteration increases, the MSE of the training error decreases. We also find that when the number of iteration is 100, minimum MSE is obtained. In this condition, the training process is achieved and the learned model can be used for online estimation.

6.3.2 Hardware platform porting experiment

Figure 7 describes the hardware platform of the experiment. The Tensorflow and Keras learning framework are installed in the raspberry pi in advance. Then, the environment

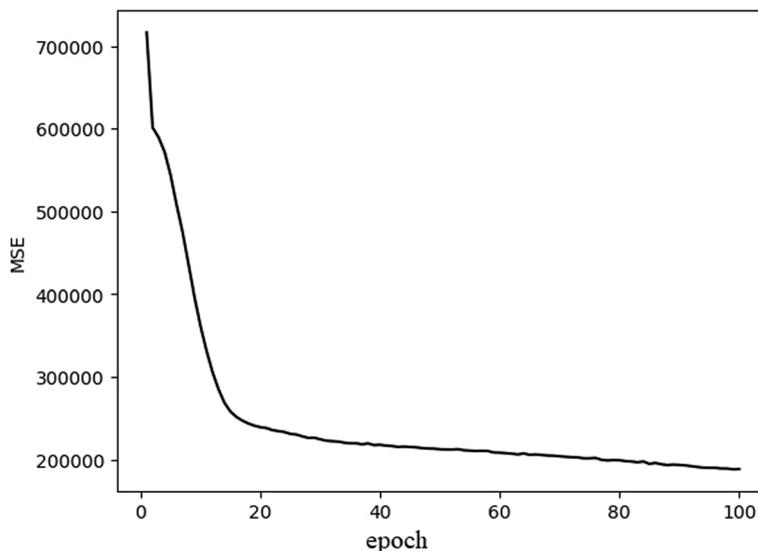


Fig. 6 Offline training performance description



Fig. 7 The photo of the hardware for algorithm running

and libraries required for the experiment is configured. At last, the pre-trained load forecasting model and test data are ported to achieve the load forecasting.

Taking the actual load 1679.4543 as an example, the predicted load is 1552.582. The error is only 126.8723 which is accepted for practical application.

6.3.3 Algorithm performance description and comparison

Figures 8 and 9 describe the load forecasting and the error for different algorithms, respectively. From the experiment results, it can be concluded that the performance of traditional machine learning methods, such as ELM method [16], SVM method [17], is worse than that of the proposed algorithm. The reason can be attributed to the proposed feature extraction technique. Since better feature description has been obtained, more accurate load forecasting result can be estimated.

In order to show the algorithm performance comparison more clearly, Table 2 gives the statistical analysis of load forecasting error for different algorithms. As expected, the proposed algorithm has the best forecasting performance among these approaches. Taking the RMSE as an example, the proposed algorithm decreases 245.31, 529.01 and 15.8 for ELM method [16], SVM method [17] and CNN method

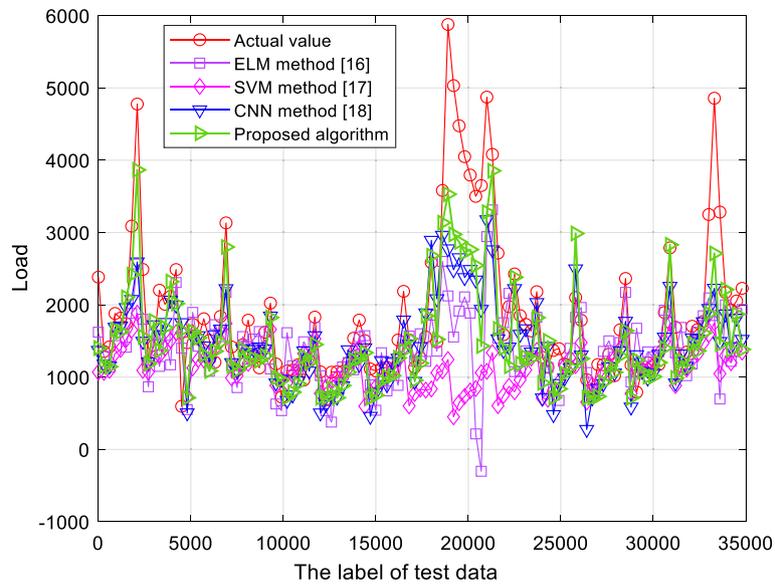


Fig. 8 Load forecasting result for different algorithms

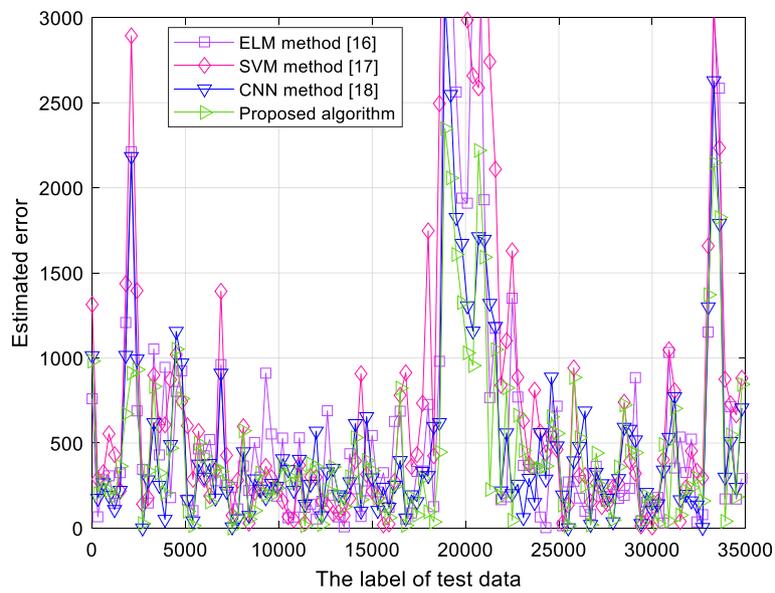


Fig. 9 Load forecasting error description for different algorithms

Table 2 The estimated error statistical characteristic comparison with different machine learning algorithms

	MAPE (%)	MAE	RMSE
ELM method[16]	38.25	632.03	966.54
SVM method[17]	38.98	792.30	1250.24
CNN method[18]	36.11	491.99	737.03
Proposed	33.63	488.33	721.23

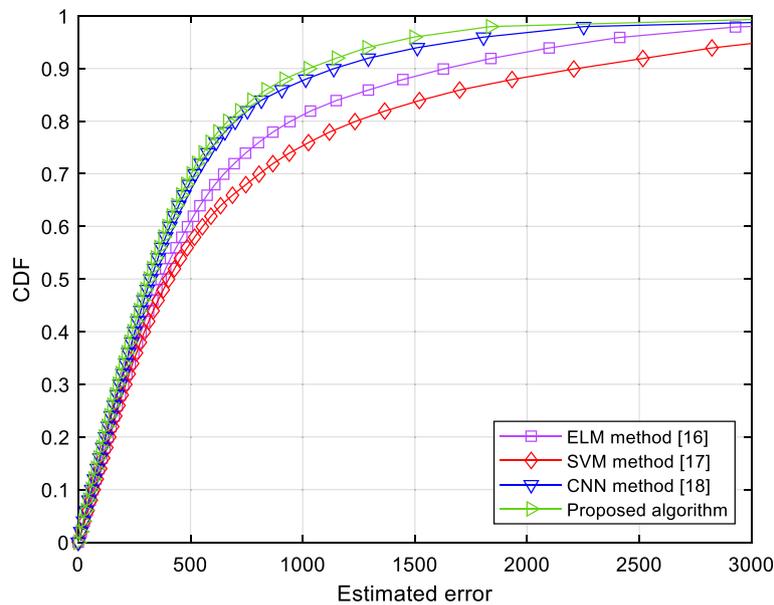


Fig. 10 CDF comparison for different algorithms

Table 3 The CDF index comparison with different machine learning algorithms

	50%	60%	70%
ELM method[16]	369.65	487.66	649.45
SVM method[17]	402.82	554.53	807.89
CNN method[18]	323.49	408.33	521.27
Proposed	312.63	396.52	504.19

[18], respectively. Figure 10 and Table 3 illustrate the CDF comparison for different algorithms. Considering the 50% load forecasting error, the ELM method [16], the SVM method [17], the CNN method [18] and the proposed algorithm are 369.65, 402.82, 323.49 and 312.63. Thus, the proposed algorithm can the minimum forecasting error among the chosen approaches.

7 Conclusion

In this article, a long-term load forecasting algorithm based on dimension expansion and depth feature extraction is proposed. The load can be estimated by the meteorological measurements and time information. The fingerprint of training data is constructed by the median filter preprocessing and time information encode. Then, the full connected network is used to transform the fingerprint from low-dimensional feature space to high-dimensional feature space. The deep learning network is used for depth information extraction automatically. Thus, better feature representation of fingerprint can be obtained. Finally, the full connected network, the deep feature extraction network and load regression model are combined for offline learning which improve the learning efficiency and prediction performance. Experiments show that the proposed algorithm has more accurate load prediction performance than other existing methods.

With the development of the AI technique, we will continue to study how to use the new deep learning algorithm and learning framework for load forecasting algorithm under different conditions in future. For example, in order to protect data privacy, the federated learning framework is proposed for load forecasting. Moreover, the hardware platform design for practical application is another research topic. For example, how to use the AI chip for real-time load estimation.

Abbreviations

ANN	Artificial neural network
CNN	Convolutional neural network
LSTM	Long short memory network
MSE	Mean square error
MAPE	Average absolute percentage
MAE	Mean absolute error
RMSE	Root mean square error
CDF	Cumulative error distribution function

Acknowledgements

Not applicable.

Author contributions

Wei-guo Zhang provides research ideas, oversight, and leadership responsibility for the research activity planning and execution. Qing Zhu, Lin-Lin Gu, and Hui-Jie Lin analyzes data. All authors read and approved the final manuscript.

Funding

The work was supported by the Science and technology project of State Grid Corporation of China (No.5100-202118566A-0-5-SF).

Availability of data and materials

Not applicable.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 28 February 2023 Accepted: 11 October 2023

Published online: 18 October 2023

References

1. T. Hong, P. Wang, Artificial intelligence for load forecasting: history, illusions, and opportunities. *IEEE Power Energy Mag.* **20**(3), 14–23 (2022)
2. Y. Zhang, H. Chiang, Enhanced ELITE-load: a novel CMPSOATT methodology constructing short-term load forecasting model for industrial applications. *IEEE Trans. Industr. Inf.* **16**(4), 2325–2334 (2020)
3. A. Ghasempour, J. Lou, Advanced metering infrastructure in smart grid: Requirements challenges architectures technologies and optimizations. *Smart Grids: Emerg. Technol. Chall. Future Directions* **1**, 77–127 (2017)
4. S. Sharma, A. Majumdar, V. Elvira, É. Chouzenoux, Blind kalman filtering for short-term load forecasting. *IEEE Trans. Power Syst.* **35**(6), 4916–4919 (2020)
5. P. Ji, D. Xiong, P. Wang and J. Chen, A study on exponential smoothing model for load forecasting, *2012 Asia-Pacific Power and Energy Engineering Conference*, 1–4 (2012).
6. F. Zhang and X. Zhou, Gray-regression variable weight combination model for load forecasting, 2008 International Conference on Risk Management & Engineering Management, 311–316 (2008).
7. W. Jian-jun, N. Dong-Xiao and L. Li, An ARMA cooperate with artificial neural network approach in short-term load forecasting, 2009 Fifth International Conference on Natural Computation, 60–64 (2009)
8. S. Singh, S. Hussain and M. A. Bazaz, Short term load forecasting using artificial neural network, 2017 Fourth International Conference on Image Information Processing (ICIIP), 1–5 (2017)
9. V.H. Hinojosa, A. Hoese, Short-term load forecasting using fuzzy inductive reasoning and evolutionary algorithms. *IEEE Trans. Power Syst.* **25**(1), 565–574 (2010)
10. F. Liu, T. Dong, T. Hou, Y. Liu, A hybrid short-term load forecasting model based on improved fuzzy C-means clustering, random forest and deep neural networks. *IEEE Access* **9**, 59754–59765 (2021)
11. Z. Yang, T. Dan, Y. Yang, Multi-temporal remote sensing image registration using deep convolutional features. *IEEE Access* **6**, 38544–38555 (2018)
12. X. Dong, L. Qian and L. Huang, Short-term load forecasting in smart grid: A combined CNN and K-means clustering approach, *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 119–125 (2017).

13. H. H. Goh, B. He;H. Liu, D. Zhang, W. Dai, T. A. Kurniawan, K. C. Goh, Multi-convolution feature extraction and recurrent neural network dependent model for short-term load forecasting, *IEEE Access*, 9, 118528–118540 (2021)
14. W. Kong, Z.Y. Dong, Y. Jia, D.J. Hill, Y. Xu, Y. Zhang, Short-term residential load forecasting based on LSTM recurrent neural network. *IEEE Trans. Smart Grid* **10**(1), 841–851 (2019)
15. A. F. López Lopera, H. Darío Vargas Cardona, G. Daza-Santacoloma, M. A. Álvarez and Á. Á. Orozco, Comparison of preprocessing methods for diffusion tensor estimation in brain imaging, *2014 XIX Symposium on Image, Signal Processing and Artificial Vision*, 1–5 (2014)
16. G. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **42**(2), 513–529 (2012)
17. C.C. Chang, C.J. Lin, LIBSVM: a library for support vector Machines. *ACM Trans. Intell. Syst. Technol.* **2**(3), 1–27 (2011)
18. M.T. McCann, K.H. Jin, Michael unser, convolutional neural networks for inverse problems in imaging. *IEEE Signal Process. Mag.* **34**(6), 85–95 (2017)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Wei-guo Zhang was born in 1980. He is currently a PHD candidate at Southeast University. His research interests include virtual power plant, demand response and integrated energy optimization control.

Qing Zhu was born in 1981. He received Ph.D. degree in electrical engineering from Southeast University, Nanjing, China. His research interests include Virtual power plant, artificial intelligence in power system.

Lin-Lin Gu was born in 1984. She is an engineer. Her research interests include demand response, virtual power plant control technology, block chain smart contracts.

Hui-Jie Lin was born in 1997. Her research interests include machine learning algorithm and artificial intelligence in power system.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
