Journal of Network Intelligence ©2021 ISSN 2414-8105 (Online)
Taiwan Ubiquitous Information Volume **6**, Number **2**, May **2021**

Smart Meter Fault Prediction Based on One-dimensional Convolution Neural Network Integrated Model

Ye Chen

Electric Power Research Institute of Yunnan Power Grid Co
Ltd,Kunming 650217, China
363419628@qq.com

Wen-Long Xu

Qujing Power Supply Bureau of Yunnan Power Grid Co
Ltd,Qujing 655000,China
258553255@qq.com

Zhi-Hu Hong

Electric Power Research Institute of Yunnan Power Grid Co
Ltd,Kunming 650217, China
842956127@qq.com

Meng-Meng Zhu

Electric Power Research Institute of Yunnan Power Grid Co
Ltd,Kunming 650217, China
396923800@qq.com

Bo Li

Electric Power Research Institute of Yunnan Power Grid Co
Ltd,Kunming 650217, China
49923387@qq.com

Jianjun Li*

School of Computer Science and Technology
Hangzhou Dianzi University, China
lijjcan@gmail.com

Chin-Chen Chang*

Department of Information Engineering and Computer Science
Feng Chia University, Taiwan
alan3c@gmail.com

Received December 2020; revised March 2021

Abstract. As a product of the development of science and technology, smart meters have been widely used all over the world because of its rich functions and reliable use. The fault detection of smart meters has become more and more important. How to quickly determine the fault types of smart meters has become a key problem. In this paper, a fault detection method based on a one-dimensional convolution neural network and model integration is proposed. Firstly, the data and features which are highly related to the fault types of smart meters are screened out from the fault data set of smart meters. Secondly, the structure and model of a one-dimensional convolution neural network are established. In this paper, a one-dimensional convolution network structure with one batch normalization (BN) layer, two convolution-pooling layers, two fully connected layers, and one dropout layer is selected. Finally, we integrated the network by adding a dropout layer and the network performance is evaluated by the index, which is verified on the public data set and tested on the smart meter fault data set in the power grid system. The results show that the proposed has a significant improvement in the accuracy of fault classification after model integration.
Keywords: Smart meter, fault classification, one-dimensional convolution neural network, model integration.

---

1. **INTRODUCTION.** To further reduce the operation cost and facilitate the operation and maintenance of power grid departments [1], requirements can grid system through the analysis of the data of smart meters to send information to determine whether the smart meters failure and the failure types [2], which can directly send professional maintenance personnel point to point of fault meters for repair.

There are many existing fault classification methods for predicting the fault of smart meters. Use belief network to extract feature and classify with SVM [3] can improve the accuracy of fault diagnosis of metering automation terminal. Manually construct the overall structure of fault diagnosis [4] with a reliability function can simplify the cumbersome information. Unfortunately, manual fault prediction method [4–6] often introduce difficulties cause subjective judgments are always not the best scheme, while the conventional machine learning methods [3, 7–9] have trouble in fitting a complex data set which will cut down the accuracy of prediction.

Recent advances in machine learning have mainly been driven by deep convolution neural networks (CNNs [10]) due to their superior capabilities to automatically learn important features from data [11]. They have improved the state-of-the-art not only in many computer vision tasks [12, 13], but also in a number of power grid fault diagnosis [14–20]. Different from previous methods that use manual knowledge to build structure, CNN models learn information from a large amount of data set automatically. However, there is still no public data set for meter fault diagnosis. So that we request for and get nearly 70,000 smart meter fault information from a city's power grid.

In this paper, we propose a one-dimensional convolution neural network to predict meter failure. We make some change based on an architecture of CNN, which is simple but effective in non-Linear regression. We follow the idea of integration [7] to improve the original model. Compared with conventional machine learning methods, a one-dimensional CNN model has a better ability to fit data set and get 15% increase of accuracy, while reduces a lot of manual operation. Meanwhile, the integration of models gets more than 5% increase. It shows that integrate models with different parameters do can improve the accuracy, which can be a trick for further study.

2. **Analysis of meter data.** The data of the meter fault data set studied in this paper comes from nearly 70,000 smart meter fault information collected by a power grid in a city, which records the voltage, current, and power values collected during the fault of

each meter, as well as information such as city ID, work order identification, putting into use time and other information.

In this paper, the fault types in the smart electricity meter fault data are firstly counted statistically. Then, based on the observation and analysis of various data indicators, a total of ten data are selected to classify the fault types, and the data is per-processed to ensure data input can be used normally into the network.

2.1. **Smart meter fault type statistics.** This paper makes a statistical analysis of the fault types of smart electricity meter fault data. There are nine fault types, whose specific names and sample numbers are shown in Fig.1.

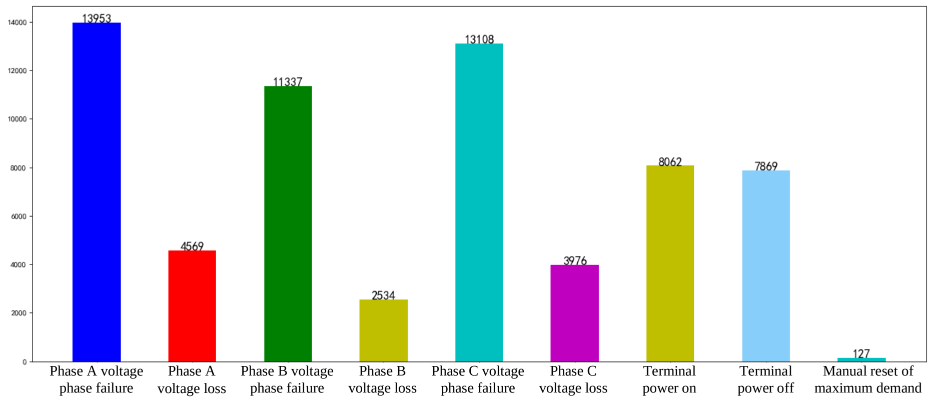

FIGURE 1. Fault type name and sample number distribution

The number of samples of each fault type is basically in an order of magnitude, so it can be used as the condition of the final fault classification. In this paper, nine fault types are numbered from 1 to 9 in Tab 1. Subsequently, the number is taken as the output of the classification model to facilitate the calculation of the loss function.

TABLE 1. Correspondence table of fault type names and numbers

Fault type name	Serial number
A phase voltage break	1
A phase voltage loss (phase loss)	2
B phase voltage phase break	3
B phase voltage loss (phase loss)	4
C phase voltage phase break	5
C phase voltage loss (phase loss)	6
Terminal powered on	7
Terminal powered off	8
Maximum demand manual reset	9

2.2. **Smart meter fault data selecting.** After counting the number of various fault types, it is necessary to filter out the data items that are highly correlated with the fault type and have a significant impact on the fault type from the entire grid data as the basis for subsequent fault classification. According to observation, analysis, and logical judgment, a total of ten data items are selected as the classification basis in the paper, including phase A voltage, phase B voltage, phase C voltage, phase A current, phase B current, phase C current, total forward active power electric energy, work order identification, running energy meter ID, and collection object identification code.

2.3. **Smart meter fault data per-process.** As there are absent data in the smart meter fault data set, which need to be filled in. According to the actual working conditions, for each phase current and voltage data and total energy data, we choose to fill with zero; As for work order identification, running energy meter ID, and collection object identification code, we choose to fill with the average value of the entire data set.

3. One-dimensional convolution neural network model and integration method.

3.1. **One-dimensional convolution neural network structure.** Different from BP neural network, the convolution neural network consists of a convolution layer, pooling layer, and full connection layer. Since the target task is to classify fault types according to a series of data values, the one-dimensional convolution structure is suitable for this task. The neural network structure used in this paper includes a BN layer, two convolution layers, two pooling layers, and two full connection layers. The specific structure is shown in Fig.2.

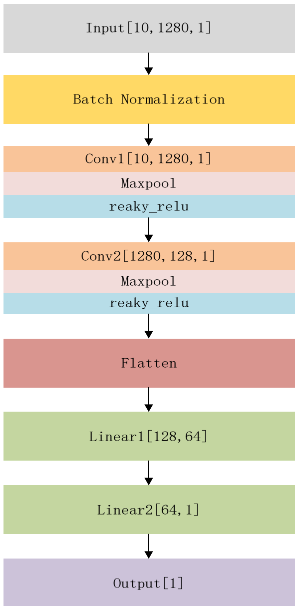

FIGURE 2. Neural network structure diagram

The BN layer is used to normalize the input small batch data to accelerate the training speed and improve the generalization ability of the model. Two convolution layers are responsible for extracting features from the input data, and two pooling layers are located behind the two convolution layers respectively for data compression and parameter reduction. After that, the extracted features are flattened and input into the fully connected layer, and the value of the model classification is output, that is, the code of the calculated fault type. In order to integrate the model, a Dropout layer is added before each of the two fully connected layers, which can form different network structures and further improve the generalization ability of the network.

3.2. **Model integration approach.** The purpose of the integrated approach is to supply a maximum flexibility in choosing a family of models and is a method for estimating uncertainty. Neural network model has a great ability to fit data set, but there are some factors which will probably make the model converge to a local optimal solution, such as initial value of weight and order of training data. For just one training process, the model does not necessarily get the global optimal solution. However, if there are enough well-trained models, the sum of them will be closer to the real optimal solution. Integrate models with different parameters will correct errors, reduce and cancel bias so as to improve the classification accuracy. In this paper, the method of model integration is adopted to train with different data set, and then the same test data is classified. Finally, the classification results are integrated to determine the final category.

After the data set was divided into a training set and test set in a ratio of 7:3, the whole training set was named as TR and the test set as TE. On the premise of keeping the proportion of TR categories unchanged, we randomly selected m sub sets from the training set TR and denoted as $TR_i(i = 1, 2, ..., m)$. The starting model is named as M_0, which is trained in the sub-training sets $TR_i(i = 1, 2, ..., m)$ separately and m well-trained models were obtained, denoted as $M_i(i = 1, 2, ..., m)$. Then we randomly selected n test data from the test sets $TE_j(j = 1, 2, ..., n)$ and feed them into all the trained models $M_i(i = 1, 2, ..., m)$ in turn, and denoted the test result of test data of q_{th} tested in the p_{th} model as y_{pq}. Therefore, the output matrix of Y with n rows and m columns can be obtained, in which, each row is the classification value of a certain data in different models, and each column is the classification value of different data in the same model.

Finally, the output matrix Y is summed up by the number of rows n, denoted as $\bar{Y}$, which is a one-dimensional vector with length n. In other words, the mean value of the output values of n data through m models, and is used as the classification value of the test data of the model $TE_j(j = 1, 2, ..., n)$.

3.3. **Evaluation index of the neural network classification model.** In this paper, the evaluation index of the model classification results is selected as R^2score, and its value is calculated by the sum of regressive squares (SSR) and the sum of residual squares (SSE).

If the real observed value, namely the tag value, is denoted as y_i, the average tag value is denoted as $\bar{y}$, and the classification value is denoted as $\hat{y}_i$, then the calculation formula of the regressive sum of squares is shown in formula(1), namely the error between the classification value and the average label value. This value reflects the sum of squared deviations of the correlation between the independent variable and the dependent variable.

$$SSR = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 \tag{1}$$

The calculation formula of the residual sum of squares SSE is shown in formula(2), which reflects the fitting degree of the model.

$$SSE = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \tag{2}$$

R^2score, as known as the determination coefficient, is calculated by formula(3), which reflects the proportion of dependent variables that can be explained by independent variables. Its value is close to 1, indicating that independent variables can better fit into dependent variables and have better classification performance.

$$R^2score = 1 - \frac{SSE}{SSE + SSR} \tag{3}$$

4. **Neural network training process.** In the process of neural network training in this paper, the data set is divided into training set and test set with the ratio of 7:3. The mini-batch gradient descent method is adopted and each training process selects a certain

size of mini-batch data from the training set, inputs it into the neural network, and the classification result is obtained.

The training processing is as a normal CNN network, the optimization algorithm and loss function are calculated by comparing classification results and labels, the calculation result of the loss function is back propagated, and the network parameters are updated time to time. After the reciprocating training reaches a certain number of times, the loss function value remains stable, the training ends, and the obtained network structure is tested on the test set.

4.1. **Batch size.** In this paper, the mini-batch gradient descent method is adopted during training, that is, the batch size of data is randomly selected from the training set for each training and put into the network model for training. If the batch setting is too small, the model convergence is too slow; If the batch setting is too large, it will be affected by machine memory.

4.2. **Learning rate.** During the learning process, setting an efficient learning rate is a very key step in calling the optimization function is called and its value will have an impact on the rate of gradient descent. The initial learning rate setting is too small and the model converges slowly. If the initial learning rate is set too large, the model cannot converge to the optimal solution.

4.3. **Loss function.** The loss function is used to measure the deviation between the classified value and the true value in the training process, and now many types have been developed, mainly including L1 Loss, MSE L1 Loss, and Smooth L1 Loss. L1 Loss is also known as an average absolute error, and its calculation formula is shown in formula(4).

$$loss = |x_i - y_i| \tag{4}$$

MSE L1 Loss also becomes a mean square error, and the calculation formula is shown in formula(5).

$$loss = (x_i - y_i)^2 \tag{5}$$

Smooth L1 Loss is the smooth version of L1 Loss. We usually adopt Smooth L1 Loss when the error is less than 1 and L1 Loss when the error is greater than 1. The calculation equation is shown in formula(6).

$$loss = \begin{cases} 0.5(x_i - y_i)^2 & |x_i - y_i| < 1 \\ |x_i - y_i| & |x_i - y_i| \geq 1. \end{cases} \tag{6}$$

4.4. **Optimization algorithm.** The function of the optimization algorithm is to adjust the network weight during the training to minimize or maximize the selected loss function. Common optimization algorithms include SGD, ASGD, Adam, etc. SGD needs to adjust the learning rate artificially, so this algorithm is not used in the experiment. SGD optimization is to update the weight w_t(denoted as $\nabla Q(w)$) with the gradient of the loss function in each iteration. The updating formula is shown in formula(7).

$$w_{t+1} = \bar{w_t} - \eta \nabla Q(w) \tag{7}$$

Where η is the learning rate, and the calculation equation $\bar{w_t}$ is formula(8).

$$\bar{w_t} = w_t \tag{8}$$

Compared with SGD, the weight updating formula of ASGD is similar to formula(7), but the calculation equation $\bar{w_t}$ is different, as shown in formula(9).

$$\bar{w}_t = \begin{cases} w_t & t < t_0 \\ \frac{1}{t-t_0+1}\sum_{\tau=t_0}^{t} w_\tau & t \geq t_0. \end{cases} \tag{9}$$

Adam algorithm, as known as the adaptive time estimation method, can calculate its adaptive learning rate for each parameter. The weight updating equation is formula(10).

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\bar{\upsilon}_t} + \epsilon}\bar{m}_t \tag{10}$$

Where η is the initial learning rate, ϵ is the set minimum value, $\bar{m}_t$ and $\bar{\upsilon}_t$ are shown in formula(11) and formula(12) respectively.

$$\bar{m}_t = \frac{m_t}{1-\beta_1^t} \tag{11}$$

$$\bar{\upsilon}_t = \frac{\upsilon_t}{1-\beta_2^t} \tag{12}$$

Where β_1 and β_2 are per-set.

5. **Experimental results and analysis.** The selection of a series of super parameters, algorithms, and functions during the neural network model and model training will have an impact on the network classification performance. To obtain the best network classification performance, the smart electricity meter fault data should be used to conduct uni-variate experiments on these super parameters, algorithms and functions.

5.1. **Model validation.** Before selecting the optimal hyper-parameter, the classification performance of the one-dimensional convolution neural network used in this paper is verified on the common data set. In this paper, four common data-sets are selected from the third-party library Sklearn, and the names and number of categories of all kinds of data sets are shown in Tab 2.

Table 2. Name of public data-set and number of classification categories

Data set name	The number of categories
Iris	3
wine	3
cancer	2
MNIST	10
Meter failure data set	9

In this paper, all kinds of data sets are divided into training sets and test sets according to 7:3. After 100 times of training on the training set, the proposed CNN network is tested, and the test results are shown in Tab 3.

Table 3. Test results of network model on the common data set

Data set	L1Loss	$R^2 score$
Iris	0.121	0.953
wine	0.043	0.936
cancer	0.034	0.974
MNIST	0.236	0.928

Among them, various super parameters and algorithms are temporarily selected as: batch size is equal to 64, the initial learning rate is set as 0.01, the optimization algorithm is Adam, and the loss function is L1 Loss.

It can be seen from the experimental results that the performance of the neural network is good, and the value on the relatively standard classification data set can get more than 0.9, indicating that the neural network model can complete the classification task.

5.2. **Hyper-parametric training.** After verifying the classification performance of the neural network structure on the public data set, the input data set is replaced by the smart electricity meter fault data set for experiments, and various super-parameters, optimization algorithms, and loss functions are trained to select the best super-parameters and functions.

Taking the optimization algorithm as an example to facilitate the experiment, three optimization algorithms that do not need to manually select the learning rate are tested in this paper. By comparing the performance of the network model on the test set after the training is completed, the experimental results are listed in Tab 4.

TABLE 4. Test results of network model using different optimization algorithms on ammeter data set

Optimization algorithm	L1 Loss	R^2score
ASGD	1.216	0.663
RMSprop	1.104	0.711
Adam	1.097	0.715

It can be seen from the test results that RMSprop and Adam have significantly better test results than the former, ASGD. To determine which one to choose between RMSprop and Adam, the paper makes a judgment by observing the curves of the two optimization algorithms in the training process, as shown in Fig.3.

It can be seen from Figure 4 that although the convergence value of RMSprop is close to Adam, the RMSprop curve has a large burr during the whole training process, which affects the performance of the model. Therefore, the Adam optimization algorithm is selected in this paper. In the same way, experiments can be carried out on other super parameters and functions and the best choice can be made.

After experiments, it can be concluded that the network model has the best performance when the batch size is 1280, the initial learning rate is 0.008, the optimization algorithm is Adam, and the loss function is Smooth L1 Loss.

5.3. **Model integration.** Finally, the model integration is carried out based on the original neural network model, and the Dropout layer is added before each fully connected layer. After that, the new network model is trained and the test set is tested after training.

When testing, data is randomly selected from the test set as input feed to the network model, and the Dropout layer is enabled to iterative classify the selected inputs, meaning that a small part of the network model is disabled for each test and the inputs are categorized using a different network model.

output result of each iteration was recorded, the mean value was taken as the final classification result, and the index R^2score was calculated. Tab 5 lists the test results of the network model on the common data-set and the smart meter fault data-set before and after model integration.

It can be seen from the test results in the table that the evaluation indexes are further improved and the network performance is improved after adopting the model fusion method.

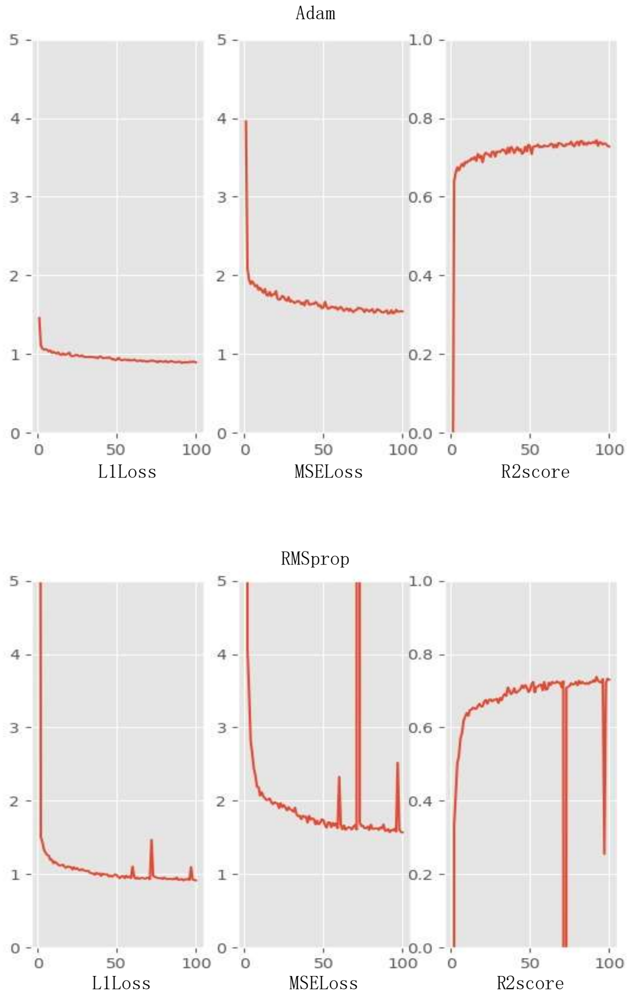

Figure 3. Neural network structure diagram

Table 5. Index test values before and after model fusion

Data-set	norm model's R^2score	our model's R^2score	Indicators
Iris	0.953	0.972	0.019
wine	0.936	0.968	0.032
cancer	0.974	0.985	0.011
MNIST	0.928	0.965	0.037
Meter fault	0.692	0.753	0.061

5.4. **Algorithm contrast.** To reflect the one-dimensional convolution neural network algorithm used in this paper on the power grid failure data sets of advantage, this paper implements a number of commonly used algorithms for fault diagnosis in the power system. Besides, the experimental results on failure data sets of the real power grid, and compared with the results of a one-dimensional convolution neural network algorithm, the experimental results are recorded in Tab 6.

As can be seen from the experimental data in Table 6, both the traditional image classification data set MNIST and the power grid fault data set studied in this paper, the classification performance and evaluation indexes of the one-dimensional convolution network algorithm are superior to those of the classical machine learning algorithms.

TABLE 6. Experimental results of algorithm comparison

	SVM	Decision tree	Random forests	1-D CNN
MNIST	0.9518	0.7403	0.9621	0.9921
Meter fault	0.3698	0.4805	0.5328	0.7528

6. **CONCLUSIONS.** In this paper, a one-dimensional convolution neural network algorithm is adopted to train the smart meter fault data collected by the smart grid, and realize the classification, prediction, and judgment of the fault types of smart meters, to improve the diagnosis and maintenance speed of the power grid system after the failure of smart meters. Firstly, We select key data and fill the missing data items to solve data quality problems. After that, we build a one-dimensional convolution neural network integrated model, verify the classification task of this model on the common data set. Finally, we train model parameters using the smart electricity meter fault data set and compare the performance with other algorithms. It can be seen from the results that the one-dimensional convolution neural network is better than traditional methods. Besides, the model integration do can improve the performance of the original model. There are wide development at the smart meter fault prediction in the future. Researchers can attempt to use a more complex model to get the highest accuracy and apply it at the power grid system.

Acknowledgements. This work was supported in part by National Science Fund of China no.61871170; Key Research and Development Plan of Zhejiang: No.2021C03131; The Basic Research Program of KY2017210A001; Key Laboratory of Brain Machine Collaborative Intelligence of Zhejiang Province.

REFERENCES.

[1] S. J. Wang, K. Chen, Automatic testing system design of smart meter based on computer vision, *Electrical Measurement & Instrumentation*, vol. 53, no. 4, pp. 63–68, 2016.

[2] N. He, Research on big data analysis of intelligent meters, *Management & Technology of SME*, no. 19, pp. 142-145, 2016.

[3] J. Chen, A Diagnosis Method of Metering Automation Terminal Fault Based on Deep Belief Network and Support Vector Machine, *GUANGXI ELECTRIC POWER*, vol. 41, no. 5, pp. 16-19, 2018.

[4] X. L. Song, G. Sun, J. Xu, X. J. Zeng, J. B. Li and J. T. Lei, Research on fault diagnosis technology of electricity acquisition system for electric power users, *Electrical Measurement & Instrumentation*, vol. 56, no. 19, pp. 97-101, 2019.

[5] L. H. Ni, J. Y. Wen, G. Y. Lv and Q. W. Xu, Power grid fault diagnosis method of hierarchical fuzzy petri net based on comprehensive variable weight, *Electrical Measurement & Instrumentation*, vol. 57, no. 12, pp. 111-117, 2020.

[6] C. M. Wang, Y. L. Zhu, Transformer fault diagnosis based on deep denoising extreme learning machine, *Electrical Measurement & Instrumentation*, vol. 56, no. 15, pp. 143-147, 2019.

[7] Y. Chen, T. Han, L. Wei, X. L. Yu and X. X. Li, Prediction on fault classification of smart meters based on multi-classification integration model, *Electrical Measurement & Instrumentation*, https://kns.cnki.net/kcms/detail/23.1202.TH.20200806.1627.012.html, 2020.

[8] C. C. Wei, X. L. Deng, S. H. Jia and L. Y. Fang, Fast smart grid fault diagnosis scheme based on multi-source data and multi-dimensional fault diagnosis space, *Electrical Measurement & Instrumentation*, https://kns.cnki.net/kcms/detail/23.1202.TH.20200604.1034.002.html, 2020.

[9] C. C. Liu, W. Wang, T. Xia and B. B. Luo, Verification of uncertainty of verification and measurement of the verification device of electric energy meter, *Electrical Measurement & Instrumentation*, vol. 55, no. S1, pp. 55-58, 2018.

[10] A. Krizhevsky, I. Sutskever and G. E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, *COMMUNICATIONS OF THE ACM*, vol. 60, no. 6, pp. 84-90, 2017.

[11] J. M. Wu, M. Tsai, S. Xiao and Y. Liaw, A deep neural network electrocardiogram analysis framework for left ventricular hypertrophy prediction, *Journal of Ambient Intelligence and Humanized Computing*, DOI: 10.1007/s12652-020-01826-1 , 2020.

[12] E. K. Wang, C. C. Chen, M. M. Hassan and A. Almogren, A deep learning based medical image segmentation technique in Internet-of- Medical-Things domain, *Future Generation Computer Systems*, vol. 108, pp. 135-144, 2020.

[13] C. Payer, D. Štern, H. Bischof and M. Urschler, Integrating spatial configuration into heatmap regression based CNNs for landmark localization, *Medical Image Analysis*, vol. 54, pp. 207-219, 2019.

[14] J. L. Jia, T. Yu, Z. J. Wu and X. H. Cheng, Fault diagnosis method of transformer based on convolutional neural network, *Electrical Measurement & Instrumentation*, vol. 54, no. 13, pp. 62-67, 2017.

[15] K. L. Gao, S. Yang, S. Y. Liu and X. W. Li, Transient Stability Assessment for Power System Based on One-dimensional Convolutional Neural Network, *Automation of Electric Power Systems*, vol. 43, no. 12, pp. 18-26, 2019.

[16] Y. Li, C. C. Feng, Fault Detection of Industrial Process Based on Deep Learning of One-Dimensional Convolution Neural Network, *Measurement and control technology*, vol. 38, no. 9, pp. 36-40, 2019.

[17] K. Wang, P. Xu, C. M. Chen, S. Kumari and M. Alazab, Neural Architecture Search for Robust Networks in 6G-enabled Massive IoT Domain, *IEEE Internet of Things Journal*, DOI: 10.1109/JIOT.2020.3040281 , 2020.

[18] E. K. Wang, S. P. Xu, C. M. Chen and N. kumar, Neural-Architecture-Search-Based Multiobjective Cognitive Automation System, *IEEE Systems Journal*, DOI: 10.1109/JSYST.2020.3002428, 2020.

[19] Q. W. Xu, X. Q. Ji, Y. Z. Zhang, J. Li and Y. J. Yu, Application of stacked de-noising auto-encoder network in fault diagnosis of power transformer, *Electrical Measurement & Instrumentation*, vol. 55, no. 17, pp. 62-67, 2018.

[20] H. Wang, H. R. Jiang and Z. D. Wang, Fault diagnosis of transformers based on multi-classified probability output model, *Electrical Measurement & Instrumentation*, vol. 55, no. 17, pp. 101-106, 2018.