

# Fault Detection of Smart Electricity Meters Based on 1D Convolution Twin Network

Hao XUE, Yiran LIU, Linkun ZHOU\*

**Abstract:** Timely detection and maintenance of smart electricity meter faults are essential for smart grid systems, but there is no high-accurate algorithm to detect the meter fault yet. So, in this paper, we propose a deep learning algorithm to detect the fault of the smart electricity meter. Our algorithm is based on a 1D convolution twin network, which can distinguish the meter data of different fault types with high precision. To realize the fault detection task, we design a twin classifier for counting the number of matches between the data to be predicted and each type of known data and select the type with the most counts as the predicted type. Our algorithm automatically detects the fault of the smart electricity meter while its accuracy reaches 94.52%, which can significantly improve the maintenance efficiency of the fault detection.

**Keywords:** fault detection; smart electricity meter; 1D convolution twin network

## 1 INTRODUCTION

As the essential equipment of the power grid system, the smart power meter completes the response to end-to-end demands and realizes the two-way communication required by the low-voltage side management of distribution. To ensure the power grid systems' operation, we should maintain the fault meter in time. Therefore, it is a crucial task to detect the fault of smart electricity meters quickly.

There are many machine learning algorithms to detect the fault meter. SVM combined with a genetic algorithm [1] and SVM combined with a deep confidence network [2] improve the detection accuracy, but it is very time-consuming. Extend random forest to a parallel algorithm enhances the accuracy and reduces a small amount of time [3]. Besides, integrating multiple machine learning algorithms improves accuracy to a lesser extent [4].

With the development of deep learning, more and more deep learning algorithms solve related tasks. Convolution neural network extracting the time-domain information realizes a high-precision transient steady-state power systems analysis [5]. In the study of transformer fault detection, deep learning machine [6], self-coding network [7], twin network [8], and other deep learning algorithms are used to improve detection accuracy. Among them, the twin network performs better, it has higher accuracy and shorter calculating time.

In the current study of the fault detection of smart electricity meters, most machine learning algorithms lack satisfactory accuracy and response speed, and few deep learning algorithms are used to solve this task. So, in this paper, we propose a deep learning algorithm based on a convolution twin network and a twin classifier to detect the fault of smart power meters. We process the fault dataset of smart electricity meters to get a better quality, build a convolution twin network, and post-process the network's output using a twin classifier. To evaluate the advantages of our algorithm, we test several machine learning and deep learning algorithms and compare them with our algorithm for accuracy and speed. The result shows that our algorithm speeds up training and improves detection accuracy. Our algorithm realizes fast detection of faults, improves detection accuracy, and provides guidance for meter maintenance, significantly improving maintenance efficiency.

## 2 THE FAULT DETECTION ALGORITHM BASED ON 1D CONVOLUTION TWIN NETWORK

### 2.1 1D Convolution Twin Network

The twin network shares the weight between two subnetworks [9]. The twin network is mainly used to calculate the similarity of two inputs and judge whether they are of the same type. Fig. 1 shows the basic structure of the twin network. Two subnetworks receive different inputs and extract features separately. Then the extracted features are calculated to obtain a value that can measure the similarity of the two inputs.

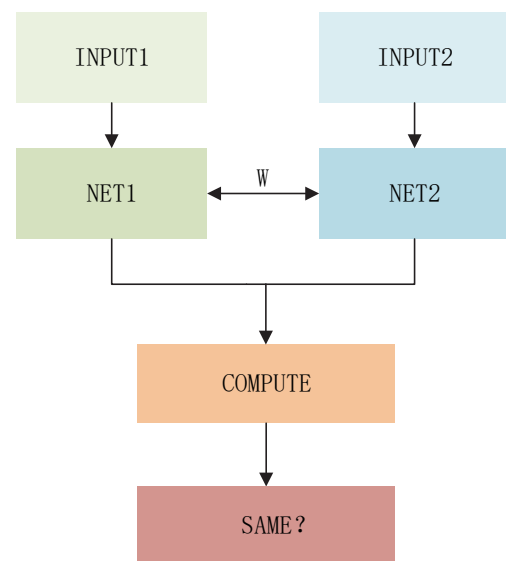


Figure 1 Schematic diagram of twin network structure

Fig. 2 shows the 1D convolution twin network used in this paper. Each subnetwork adopts a consistent structure of one convolution layer with three full connection layers to extract the input features.

The two inputs of the network have the same shapes (vector with a length of 17). Therefore, the two subnetworks can share the weight. Sharing weight makes the subnetworks carry out the same operation to extract features. We initialize the weight with the same value and adjust synchronously to ensure the weight is shared.

When judging the similarity of the extracted feature, Euclidean Distance is used, and the formula can be expressed as follows:

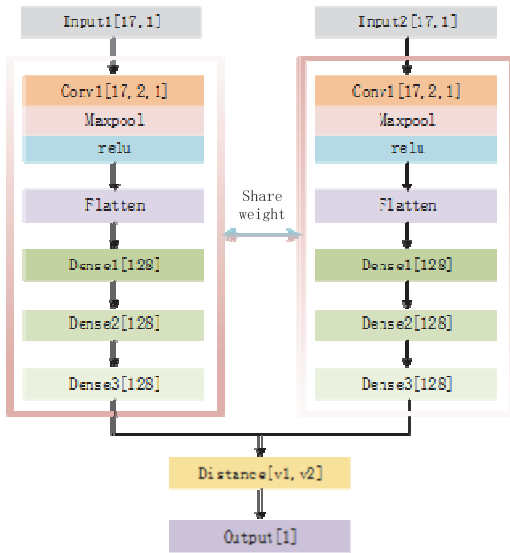


Figure 2 Our 1D convolution twin network

$$D_w = \sqrt{[G_w(X_1) - G_w(X_2)]^2} \quad (1)$$

where:  $G_w(X_1)$  and  $G_w(X_2)$  both are the extracted features.

## 2.2 Contrast Loss Function

In the training process, the loss function needs to reflect the ability to discriminate inputs. If the two inputs are of the same type, the loss function's value should be small. If not, the value should be large.

In this paper, we choose the contrast loss function [8], and the formula can be expressed as follows:

$$loss = \frac{1}{2}(1-Y)(D_w)^2 + \frac{1}{2}Y[\max(0, m - D_w)]^2 \quad (2)$$

Here, the value of  $Y$  is 0 or 1, which represents whether the two inputs are of the same type. When 0 is taken, the inputs are similar. When 1 is taken, they are different. The  $m$  is called marginal value, while the value of it is greater than 0 and it takes effect when  $Y$  is 1.

When the two inputs are of the same type ( $Y$  is 0), the loss function degenerates to Eq. (3). In this case, if the Euclidean Distance between the two inputs is small, which shows that the fault classification is good, the value calculated by the loss function should be small. If the Euclidean Distance is large, which means that the network judges the two as different types, the loss function needs to punish the network, and the value should be large.

$$loss = \frac{1}{2}(D_w)^2 \quad (3)$$

When the two inputs are different ( $Y$  is 1), the loss function degenerates to Eq. (4). If the Euclidean Distance is greater than the marginal value ( $m$ ), it indicates that the network successfully distinguishes different inputs. So, the  $m - D_w$  is negative and the loss function is 0. If the Euclidean Distance is not large enough, the differentiation degree is not precise, and the network needs to be punished.

$$loss = \frac{1}{2}[\max(0, m - D_w)]^2 \quad (4)$$

## 2.3 Twin Classifier

Essentially, the twin network can distinguish whether the two inputs are of the same type, but it cannot directly classify them in multi-classification tasks. So, we design a twin classifier in this paper. After pairing data, we put the input to be classified into the twin network and the twin classifier to get the predicted type.

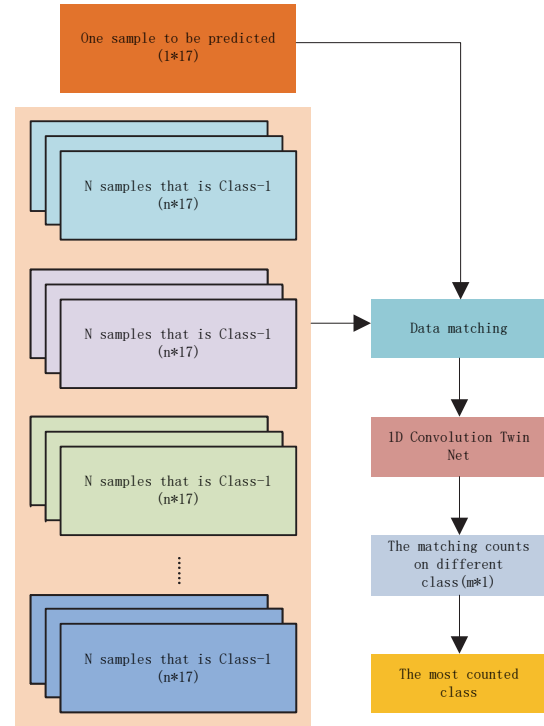


Figure 3 The structure of the twin classifier

Fig. 3 shows the structure of the twin classifier. We need to select samples from each fault type in the training set and pair the data to be classified with the selected samples. Then we put the paired data into the twin network to judge the similarity. Finally, we take the highest counted type as the predicted type.

We assume that the number of classifications is  $n$ , the input to be classified belongs to Type  $x$ . The number of selected samples is  $m \times n$  ( $m$  samples from each type), and the probability that the twin network correctly distinguishes two inputs is  $p$  (recorded as TDP). Then we can calculate the probability classifying the input correctly (recorded as CA).

We take  $n = 4$ ,  $m = 5$ ,  $p = 0,8$  and  $n = 10$ ,  $m = 5$ ,  $p = 0,8$  as examples. Then we calculate the probability distribution of the number with correct classification and CA. Tab. 1 shows the details in the calculation process.

Eq. (5) indicates the probability that the number of correct classifications is  $i$ .  $C_m^i$  means the number of situations that choose  $i$  correct predictions in  $m$  total predictions.  $p^i$  is the probability of these correct predictions, while  $(1-p)^{m-i}$  is the probability of the incorrect predictions.

**Table 1** Derivation process of twin classifier

class	Samples from each class	TDP	The probability ( $p$ ) distribution of the number ( $i \in \{1, 2, \dots, m\}$ ) with the correct classification							CA
$n$	$m$	$p$	Eq. (5)							Eq. (6)
4	5	0.8	$i$	0	1	2	3	4	5	0.9207
			$P$	0.0003	0.0064	0.0512	0.2048	0.4096	0.3277	
10	5	0.8	$i$	0	1	2	3	4	5	0.8352
			$P$	0.0003	0.0064	0.0512	0.2048	0.4096	0.3277	

Eq. (6) calculates the probability that the prediction number of the correct type is greater than the prediction number of other incorrect types. In the first summation sign, we assume that the prediction number of the Type  $x$  (the input belongs to) is  $i$ , and the probability of  $i$  is  $P(X = i)$ . In the second summation sign, we calculate the probability that the prediction number of other types is lower than  $i$ .

$$P(X = i) = C_m^i p^i (1-p)^{m-i} \quad (5)$$

$$P(N_T > N_F) = \sum_{i=1}^{i=n} \left[ P(X = i) \times \left( \sum_{q=1}^{q=i} P(X = n - q + 1) \right)^n \right] \quad (6)$$

### 3 DATA PROCESSING AND DATA MATCHING

#### 3.1 Data Processing

In this paper, our dataset is collected by a provincial power grid. The dataset records the information of the fault smart power meter in three cities. The dataset has 130 000 data with eight fault types, including meter power on, meter down, phase A voltage open, phase A voltage loss, phase B voltage open, phase B voltage loss, phase C voltage open, and phase C voltage loss. The dataset records 17 pieces of data, including the voltage and the current of phase A, B, and C, forward direction active total electric energy, meter ID, area ID, City ID, start-up time, acquisition time, work order identification, alarm identification, alarm additional items. The above 17 pieces of data are all used as the judgment basis of fault detection [10].

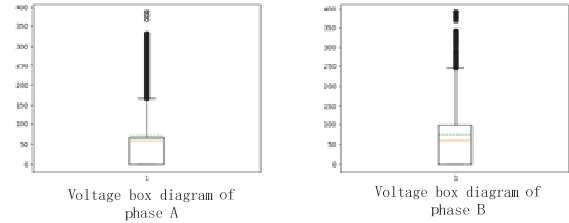
**Table 2** Results of data preprocessing

	Phase A Current		Phase B Current		Phase C Current		Total electric energy of positive active power	
	mean	variance	mean	variance	mean	variance	mean	variance
Before data cleaning	0.36497	2.54757	0.40517	3.55044	0.89288	16.74959	8 083.63323	1 196 615 915
After data cleaning	0.19386	0.18475	0.17506	0.19420	0.25707	0.58687	1 612.13104	2 900 116
After normalization	0.03786	0.00705	0.02910	0.00537	0.01986	0.00350	0.01614	0.00291

#### 3.2 Data Matching

The twin network is used to detect and classify the dataset, so we must generate data pairs. What's more, to calculate the loss function and the TDP of the twin network, each pair of data needs a label that marks whether the two data are of the same type.

In this paper, 70% of the samples are the training set. We record the least number of samples among the eight fault types as  $a$ . Then, data pairs are generated from fault types 1 to 8 in the order of one positive data pair (two data are of the same type) and one negative data pair (two data are different).  $2(a - 1)$  pairs are generated from each fault



**Figure 4** Examples of box diagram (The black hollow circle is the abnormal value. The black thin horizontal line is the upper edge. The red dotted line is the median while the green dotted line is the mean value)

Before using the dataset, we draw box charts to observe the original data distribution, as shown in Fig. 4. We find out that the original data has much abnormal value (black hollow circle) caused by meter disturbance, abnormal operation of electrical appliances or transmission line fault.

So, we pre-process the original data by data cleaning and normalization to get a high-quality dataset. Firstly, we replace the abnormal value with the upper edge (black thin horizontal line) to clean the original dataset. Then, considered the numerical difference of different pieces of information, we subtract the minimum and divide by the range to eliminate dimensional effects.

We select four pieces of data to show the mean and the variance before data cleaning, after data cleaning, and after normalization in Tab. 2. Tab. 2 shows that the variance has decreased after data cleaning, and the normalized mean falls between 0 and 1. After preprocessing, the effect of dimensions has been eliminated, and the new dataset is more convenient for calculation [11].

type and  $8 \times 2(a - 1)$  pairs are generated in total. The label following the data pair is an alternation of 1 (same) or 0 (different). The label's length is  $8 \times 2(a - 1)$  too.

The remaining 30% samples are validating set. Similar operations are carried out to generate data pairs.

## 4 EXPERIMENT AND ANALYSIS

### 4.1 Experiments on 1D Convolution Twin Network

Our experiments on training the 1D convolution twin network use RMS prop as the optimization algorithm, and the batch size is 128. Fig. 5 shows the learning curve of TDP and training epochs on the training set. The learning

curve shows that the TDP has reached the maximum value in a few epochs. After that, with the increase of epochs, the TDP is almost unchanged.

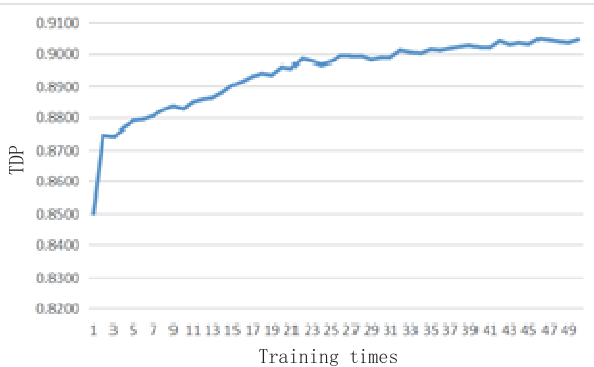


Figure 5 Twin network learning curve

Table 3 Speed comparison on different network

	mnist			Grid fault		
	Time / s	epoch	TDP	Time / s	epoch	TDP
CNN	9.463	36 864	0.9821	15.385	4 049	0.7349
our Net	6.215	2 541	0.9901	2.286	768	0.9044

We respectively train the CNN and our 1D convolution twin network on the mnist dataset (a handwritten digital image classification dataset) and the power grid fault dataset. Tab. 3 shows the value of time, epoch, and TDP when the accuracy reaches the maximum. Compared with CNN, our 1D convolution twin network gets higher accuracy on identification task.

## 4.2 Experiments on Twin Classifier

To validate the effectiveness of the twin classifier, we randomly select samples from the validating set while we set the number of samples selected from each type to 10. Then we put the paired data into a well-trained twin network and calculate the CA.

Table 4 Validating results of twin classifier

	mnist	Grid fault
TDP	0.9901	0.9044
CA	0.9986	0.9452

The result recorded in Tab. 4 shows that the CA is not lower than the TDP, which means that the twin classifier

Table 6 CA for every fault type

CA	Phase A open	Phase B open	Phase C open	Phase A loss	Phase B loss	Phase C loss	terminal power on	terminal power off
Training set	0.9622	0.9677	0.9449	0.8570	0.8511	0.8139	0.9260	0.9043
Validation set	0.9438	0.9557	0.9486	0.8708	0.8445	0.8050	0.9043	0.9246

Tab. 6 shows the accuracy in the training set and the validating set in detail. In Tab. 6, the twin network performs well in the fault type of three-phase voltage open, terminal power on and off, and the accuracy is above 92%. For three-phase voltage loss, which has relatively low accuracy, the accuracy is also above 80%. These results show that the 1D convolution twin network proposed in this paper can learn the power grid fault dataset. Our network can detect and classify the fault type of power meters using the information collected. Our algorithm can

successfully use a high-precision identification network to complete a high-precision classification task.

## 4.3 Algorithm Comparison and Analysis

Besides, we implement SVM, Decision Tree, and CNN based on the mnist dataset and the power grid fault dataset to verify our advantages.

Ablation experiments are carried out independently for each algorithm, and the super parameters are adjusted to obtain the highest accuracy. Tab. 5 records the highest accuracy.

The super parameters of each algorithm determined on the pre-processed power grid fault dataset are set as follows:

**SVM:** radial basis function, standard difference gamma is 0.01, penalty coefficient C is 400, weight ratio is set as balanced, and polynomial power is 5.

**Decision Tree:** the maximum depth is 4, the node dataset is divided into information entropy and Gini-coefficient to judge, the minimum number of samples in the node data set is 25, the minimum number of samples in leaf node is 15, the maximum number of leaf nodes is 5.

Table 5 The highest accuracy (CA) of different algorithms

	SVM	Decision Tree	CNN	Our Net
mnist	0.9268	0.8873	0.9821	0.9986
Grid fault	0.6472	0.6285	0.7349	0.9452

**Decision Tree:** the maximum depth is 4, the node data set is divided into information entropy and Gini coefficient to judge, the minimum number of samples in the node data set is 25, the minimum number of samples in leaf node is 15, the maximum number of leaf nodes is 5.

**CNN:** batch size is 1 920, learning rate is 0.12, the optimizer is Adam, and the loss function is cross entropy loss function.

**Our 1D Convolution Twin Net:** batch size is 256, learning rate is 0.08, the optimizer is Adam, and the loss function is cross entropy loss function.

Tab. 5 shows that the other four algorithms can fit the mnist dataset well and achieve high accuracy. However, it is challenging to train SVM and decision tree on power grid fault sets. Compared with them, CNN enhances the accuracy, but there is still room for improvement. The 1D convolution twin network proposed in this paper achieves the highest accuracy among the four validated algorithms.

remotely obtain the possible meter faults, such as missed detection, error detection, open circuit, and so on. Power grid systems can enhance reliability and stability using our algorithm.

## 5 CONCLUSIONS

In this paper, to speed up the maintenance efficiency of meter faults, we propose a deep learning algorithm to detect the smart electricity meters' fault automatically. Our

algorithm is based on a 1D convolution twin network, which can distinguish the meter data of different fault types with high precision. To realize the fault detection task, we design a twin classifier for counting the number of matches between the data to be predicted and each type of known data and select the type with the most counts as the predicted type. Compared with other algorithms, our algorithm automatically detects the fault of the smart electricity meter with over 90% accuracy, which has practical application value. The power grid system can dispatch professional personnel to repair the meter faults in time using our algorithm, which will ensure the reliability and stability of the power grid system.

## 6 REFERENCES

- [1] Huang, X., Jiang, W., Zhu, Y., & Tian, Y. (2020). Transformer Fault Prediction based on Time Series and Support Vector Machine. *High Voltage Engineering*, 46(7), 2530-2538. <https://doi.org/10.13336/j.1003-6520.hve.20191344>
- [2] Chen, J. (2018). A Diagnosis Method of Metering Automation Terminal Fault Based on Deep Belief Network and Support Vector Machine. *Guangxi electric power*, 41(5), 16-19. <https://doi.org/10.16427/j.cnki.issn1671-8380.2018.05.004>
- [3] Yue, Y. & Li, K. (2020). Day Ahead V2G Power Capacity Prediction based on Distribution Internet of things Technology and Parallel Random Forest Algorithm. *Power Demand Side Management*, 22(4), 31-34.
- [4] Chen, Y., Han, T., Wei, L., Yu, X., & Li, X. (2020). Prediction on fault classification of smart meters based on multi-classification integration model. *Electrical Measurement & Instrumentation*.
- [5] Gao, K., Yang, S., Liu, S., & Li, X. (2019). Transient Stability Assessment for Power System Based on One-dimensional Convolutional Neural Network. *Automation of Electric Power Systems*, 43(12), 18-26. <https://doi.org/10.7500/AEPS20180911006>
- [6] Wang, C. & Zhu, Y. (2019). Transformer fault diagnosis based on deep denoising extreme learning machine. *Electrical Measurement & Instrumentation*, 56(15), 143-147. <https://doi.org/10.19753/j.issn1001-1390.2019.015.022>
- [7] Xu, Q., Ji, X., Zhang, Y., Li, J., & Yu, Y. (2018). Application of stacked de-noising auto-encoder network in fault diagnosis of power transformer. *Electrical Measurement & Instrumentation*, 55(17), 62-67. <https://doi.org/10.3969/j.issn.1001-1390.2018.17.011>
- [8] Zhu, R., He, D., & Hu, S. (2021). Transformer fault diagnosis based on convolution twin network with small sample size. *Proceedings of the CSU-EPSCA*, 33(1), 64-69, 84. <https://doi.org/10.19635/j.cnki.csu-epsa.000503>
- [9] Wang, X., Ke, L., & Ren, J. (2019). Chemical industrial process fault detection based on sample reconstruction multi-scale siamese CNN. *Chinese Journal of Scientific Instrument*, 40(11), 181-188. <https://doi.org/10.19650/j.cnki.cjsi.J1905469>
- [10] He, N. (2016). Research on big data analysis of intelligent meters. *Management & Technology of SME*, (7), 142-145. <https://doi.org/10.3969/j.issn.1673-1069.2016.19.087>
- [11] Hao, X., Zhang, C., Pei, T., Wang, W., & Liu, F. (2020). Power load data preprocessing based on adaptive PFCM clustering. *Electrical Measurement & Instrumentation*, 57(21), 40-46. <https://doi.org/10.19753/j.issn1001-1390.2020.21.006>

## Contact information:

**Hao XUE**, assistant research  
China Academy of Railway Sciences Corporation Limited,  
2 Daliushu Road, Haidian District, Beijing, China  
E-mail: XUEHAO1989BEIJING@163.com

**Yiran LIU**, assistant research  
China Academy of Railway Sciences Corporation Limited,  
2 Daliushu Road, Haidian District, Beijing, China  
E-mail: 13681053804@163.com

**Linkun ZHOU**, postgraduate  
(Corresponding author)  
Beijing University of Posts and Telecommunications,  
10 Xitucheng Road, Haidian District, Beijing, China  
E-mail: 3215681611@qq.com