Author(s): Zhao, Qiang; Chang, Zheng; Min, Geyong

Title: Anomaly Detection and Classification of Household Electricity Data : A Time Window and Multilayer Hierarchical Network Approach

Year: 2022

Version: Accepted version (Final draft)

Anomaly Detection and Classification of Household Electricity Data: A Time Window and Multilayer Hierarchical Network Approach

Qiang Zhao, Zheng Chang, *Senior Member, IEEE,* Geyong Min, *Senior Member, IEEE*

***Abstract*—With the increasing popularity of the smart grid, huge volumes of data are gathered from numerous sensors. How to classify, store, and analyze massive datasets to facilitate the development of the smart grid has recently attracted much attention. In particular, with the popularity of household smart meters and electricity monitoring sensors, a large amount of data can be obtained to analyze household electricity usage so as to better diagnose the leakage and theft behaviors, identify man-made tampering and data fraud, and detect powerline loss. In this paper, the time window method is first proposed to obtain the features and potential periodicity of household electricity data. Combining the denoising ability of the autoencoder and the induction ability of the feedforward neural network, a multilayer hierarchical network (MLHN) is then established to detect anomalies in single sensor data and classify multiple groups of sensor data, respectively. The experimental results show that the accuracy of detecting abnormal data and data classification is significantly improved compared with the presented scheme.**

***Index Terms*—household electricity, multilayer hierarchical network, autoencoder, feedforward network, anomaly detection, classification.**

I. INTRODUCTION

A. Background and State-of-the-art

With the rapid development and wide deployment of Internet of Things (IoT) devices, the smart grid emerged as a novel power grid platform to replace the traditional power management system. In the smart grid, both computing and communication technologies are synchronized to analyze real-time grid data to realize strengthened grid management [1]. Thus, the success of the smart grid highly relies on the obtained data and data analytic capability. Consequently, abnormal data and information attacks can be harmful for the implementation of smart grid applications. Typical abnormal data in the power grid include data fraud, electricity stealing data, damage information, and virus. The abnormal data lead the smart grid to make fault decisions that not only cause huge economic losses but also become a risk factor that can collapse the whole power grid, leading to large-scale blackouts and other serious consequences. To prevent the damages caused by abnormal data, there are mainly different methods [2], among which, machine learning- and neural network (NN)-based schemes are of particular interests due to their low cost and high detection accuracy.

Through analyzing the data obtained from different meters, the smart grid can be better managed. In particular, the successful management of household electricity data is closely related to everyone's life. On the one hand, if the electricity consumption habits of users in the whole region are accurately analyzed, the power supply demand of the power grid can be accordingly adjusted [3]. On the other hand, if the data obtained from various electrical appliances of a single user is analyzed, the power consumption level of users can be predicted, and regulation and energy conservation can be carried out [4]. Therefore, it is important to develop the data analytic schemes to accurately analyze the electricity consumption situation in each household and optimize the electricity expenditure of each household and the whole power grid [5].

To identify the abnormal data, a variety of data anomaly detection methods has been applied for household appliances. The authors of [6] use data autocorrelation and clustering methods to determine data anomaly. The present scheme uses autocorrelation coefficients, partial autocorrelation coefficients, and quantity autocorrelations to calculate the autocorrelation in time series data and then cluster and identify data anomaly. In [7], the authors investigate the time window method and change the observation interval of the time series data to detect whether abnormal information is apparent in the time window. By combining the time domain and frequency domain features of the time series data, an anomaly detection method is presented under a support vector machine (SVM) classifier in [8]. The authors utilize the autoencoder to cluster the smart grid data and propose a corresponding scheme to predict the electricity consumption in [9]. In [10], a Bayesian-based method that is proposed to classify data from household appliances shows that by performing electrical appliance classification and abnormal data classification, the accuracy rate of data detection can be greatly improved.

In recent years, with the rapid development of neural network (NN) technology, a large number of NN-base schemes have been applied to the analysis of power grid data [11]. The existing stable NN models mainly include multilayer perception (MLP), the recurrent neural network (RNN), the convolution neural network (CNN), the autoencoder and generative adversarial networks (GAN) [12]. An improved artifi-

Q. Zhao is with the Department of Software Engineering, Shanxi University, Taiyuan, Shanxi, 037006 China e-mail: (hmoe@sxu.edu.cn). Z. Chang is with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China, and with the Faculty of Information Technology, University of Jyväskylä, P.O. Box 35, FIN-40014 Jyväskylä, Finland. G. Min is with the Department of Computer Science, College of Engineering, Mathematics, and Physical Sciences, University of Exeter, Exeter, EX4 4QF, U.K. This work is partly supported by NSFC (No. 62071105).

cial neural network (ANN) method is proposed in [13], which uses the optimization algorithm to improve the accuracy, and the performance is better than that achieved by the SVM. Due to the fact that the MLP model may lead to an increase in training complexity and a decrease in network induction ability [14], the CNN has been recently applied, which can use raw data (which means the unprocessed data is obtained directly from the sensor, and statistical calculation processing and other feature extraction methods are not needed) as the input and the convolution operation to extract data features. In [15], the authors utilize the image processing method to process the data obtained in the smart grid and combine routing information, meter recording, and node observation data together as input for training. Because of its short-term memory ability, the RNN network has unique advantages in processing time series data; the long short-term memory (LSTM) model is a particularly successful application in natural language processing [16]. The LSTM RNN network is also applied to analyze the power consumption data in [17] to forecast the electric load of a single energy user.

The purpose of different NN methods is to learn the potential rules from a large number of data samples and then realize the recognition and analysis of unknown samples by constantly correcting the weights in the network. There are two main methods to use data samples. This is the first method to directly use the raw data for data input, such as the CNN and the RNN. The convolution operation of the CNN only focuses on the relationship among local data, so it is difficult to find a variety of different periodic rules in smart grid data [15], and the RNN is difficult to deal with smart grid data without obvious context characteristics [16]. It is difficult to directly apply deep learning methods for data that do not have obvious periodicity and context. Previous researchers have tried to adjust the arrangement of energy consumption data and reorganize it in a fixed period [20]. However, it is not easy to observe other periodic rules by fixing the data period. There are also others have used the RNN method after compressing data to reduce the impact of short-term data fluctuation [24]. Clearly, directly using a large amount of raw data gathered from household appliances for training is not only computationally infeasible but also time-consuming. In addition, ignoring the internal relations of raw data or using a simple network model to find internal relations can reduce the accuracy of detection and classification processes. Therefore, it is necessary to preprocess the data by data cleaning, data selection, and migration, or data mining, to extract a series of features for next-step training.

Another way of data processing is to use the data after feature extraction, such as the SVM or back propagation. Based on the analysis of the energy consumption data in this study, the data itself has periodicity, while other abnormal data, such as electricity stealing information, have little regularity. Previous authors have used statistical methods to extract the feature of electricity data from both time and frequency domains and to facilitate the anomaly detection process [6] [10]. To find more potential relationships and reconstruct one-dimensional time series data into a two-dimensional sample data set, a time window method is introduced, which utilizes the periodicity of data, samples the data periodically, and organizes them into two dimensions. Then these data can be used in NN training and various machine learning methods [25]. However, the time window method only reconstructs one-dimensional time series data into a group of sample data with the same feature dimension. Further mining the potential relationship between features is still needed. Although deep neural network (DNN) can be used [5], it is obvious that convolution and an automatic induction ability can not directly apply to time window features.

To make better use of the features extracted by the time window method, a multilayer hierarchical network (MLHN) is proposed. Different from NN's automatic mining of data relations and from machine learning methods that directly use features for learning, the MLHN can group the features of samples according to the correlation degree. In this structure, the sample features can be divided into multiple groups for learning, and the important features can be repeatedly introduced into different groups, and then the proportion of output in the aggregated results can be adjusted according to the importance of each group. Finally, the aggregated results are used for classification and filtering tasks. Compared with the previous NN methods, the MLHN has higher flexibility and can adjust the number and type of module network used in the hierarchical structure according to the feature grouping and classification tasks. The MLHN also has better adaptability, which makes good use of the induction of the feedforward network and the filtering ability of the autoencoder, so it can be used for other machine learning tasks. The MLHN also has good expansibility and its modular structure enables it to be combined at will; It can even be integrated with the CNN.

In this paper, the MLHN is proposed to cooperate with the time window method. But this kind of hierarchical network can also be used in many other fields. For example, in network intrusion detection, some researchers have used clustering to group features and then input the grouped features into a hierarchical network [26]. In the analysis of financial data, some scholars have examined group learning and re-aggregating extracted features [27]. In the data analysis of the smart grid, this idea of a hierarchical network has also been used in many aspects, such as short-term load estimation [28] and energy consumption mode analysis [29].

B. Contribution

In this paper, the anomaly detection and classification problem for the household electricity data are concentrated. A new time series data feature extraction algorithm is introduced in this paper to find the periodic features of the raw data. In addition, the multilayer hierarchical network is proposed, which uses the autoencoder and feedforward network for training. Accordingly, the classification of data obtained by multiple meters is implemented, and the abnormal data detection and early warning capability (e.g. whether the corresponding electrical appliance is normal or damaged) can be realized. The major contribution of this paper can be summarized as follows:

- A new time series data feature extraction algorithm is introduced in this paper, in which one-dimensional

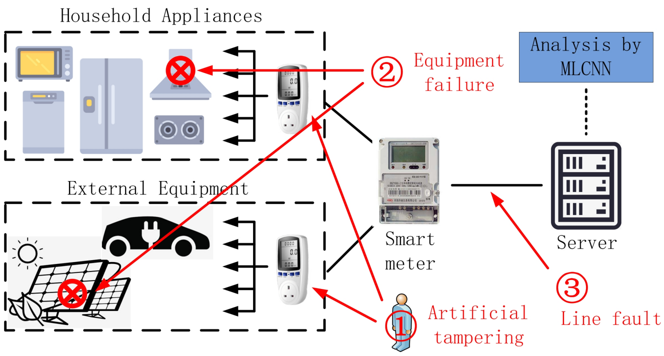


Fig. 1: Detecting household electricity data

time series data are transformed into a group of two-dimensional data with periodic features. This set of two-dimensional data and the corresponding electricity meter label are used as the input samples of NN.

- The MLHN, which uses the autoencoder and feedforward network for training is proposed. The feedforward network is used to explore the potential relation between single window features, and the autoencoder is utilized to filter the noise among the input samples. The classification of data obtained by multiple meters is implemented and the abnormal data detection and early warning capability (e.g. whether the corresponding electrical appliance is normal or damaged) can be realized.
- Extensive simulation studies are conducted. Through applying the proposed scheme to the real and high volume data set, the advantages of the proposed detection and classification scheme can be observed. Compared to the existing methods, the classification accuracy has been significantly improved.

II. System Model

Examples of different types of abnormal household electricity data are shown in Figure 1. These types of anomalies include smart meters failure (from equipment aging, equipment downtime, equipment replacement, etc), abnormal data caused by changing circuit or tampering with smart meter, and abnormal transmission data caused by line failure (e.g. line aging and private connections). In view of these abnormal data, multiple meters are used to monitor and collect information from different household appliances, and then a server, such as an edge computing node, can analyze the abnormal data through analyzing long-term statistics. Analytic or machine learning technologies such as the MLHN in this work can be applied to analyze and identify abnormal data and data that has possibly been tampered with, find the source of anomaly through classification and warn about possible electricity theft and damage.

III. Methodology

A. Time Window

The raw household electricity data that are obtained from the smart meters may be from one day or of one-year duration. Therefore, it is necessary to set a reasonable time window for further data processing. The time window is the maximum length of time that data can be read after the start time is determined, and the observation results are attenuated according to the sequence in which the time window appears. For example, a week's worth of household electricity data are obtained from the meters, and the size of the time window can be set to one day. Then each window will only observe the data within one day after the start time. The time window is continuously moved for reading, and all the previously obtained window data are multiplied by the attenuation coefficient. Finally, an extracted feature sequence is obtained. Obviously, the time window has two main characteristics: the window size and the moving step length of the window.

To find the possible periodic nature of household electricity data, different sizes of time windows can be used to control the size of the observation period. If one year of electricity data is available, then the time window could be set to 7 days (week), 30 days (month), or 90 days (quarter) and so on. Then, the features of multiple groups of observation data can be obtained and compared. According to the length of the data, the size of the time windows can be flexibly set to attain more potential regular features from the data. The fixed window can only observe the data relation within and between windows, and the moving window can compare the periodic features of the whole time series. Thus, a reasonable setting of the moving distance of the window not only improves the probability of discovering periodic laws but also reduces the raw data scale and the number of calculations. A group of windows with different sizes are synchronously moved by a set of distances, and then the data relations are compared among those windows. This set of moving distances can further improve the coverage of observation and increase the possibility of discovering periodic laws. If the size of the window is one day, the moving step can be set to minutes or hours so that the sampling results at different starting times can be observed and the integrity of data can be ensured.

1) Statistical features in single window: To better represent the data in a time window and reduce the size of data for further analysis, statistical features are used to represent the observed data. Once the window size is set, the time series data in a selected time window can be expressed as $\mathcal{X} = \{x_1, x_2, x_3, ..., x_n\}$, and (1) can be applied to calculate the features of regularity and difference within this window. The mean value μ and mode m are used as the regularity features, and the standard deviation σ and coefficient of variation ξ represent the difference features. Then, μ is the mean value of the observed data in the time window, which is used to reflect the general situation of electricity usage in this time period. Also, m highlights the most common data in the time window to reflect the most common electricity usage during the period, and σ represents the fluctuation of data in this time window, which reflects the change of electricity usage. The coefficient of variation ξ indicates the intensity of data fluctuation, which shows the abnormal electricity usage during the period. In particular, $\mu = 0$ means the electricity data is not obtained, and its coefficient of variation is $\xi = 0$. Finally, the observed data $\mathcal{X}$ of a single window is counted as $w = \{\mu, \sigma, m, \xi\}$.

$$\begin{cases} \mu = \frac{\sum_{i=1}^{n} x_i}{n} \\ m = \mu - 3(\mu - \frac{x_{(n/2)}+x_{(n/2+1)}}{2}) \\ \sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i-\mu)^2}{n}} \\ \xi = \sigma/\mu \end{cases} \tag{1}$$

2) The difference between different windows: In addition to the statistical features of a single window, with the change of window size and the movement of window, multiple windows exist and the relations between windows should be analyzed. To clearly describe the regular changes of data between time windows, the characteristic distance S, characteristic deviation R, similarity degree C, and correlation coefficient P are also presented to describe the relationship between windows, as shown in (2). S_{ij} represents the overall degree of deviation between two windows, and R_{ij} is the fluctuation deviation between two windows. C_{ij} represents the overall approximation of two sets of window data, and P_{ij} represents the degree of similarity between the fluctuations of two windows. In particular, when the mean value of the two sets of data is zero, then $\mu_i = \mu_j = 0$, and $C_{ij} = \frac{\mu_i R_{ij}}{2\mu_i} = \frac{R_{ij}}{2}$. When the mean value is not zero, but the variance is zero, then $P_{ij} = \frac{S_{ij}}{\mu_i+\mu_j}$. When the mean and variance are both zero, then$C_{ij} = 0$ and $P_{ij} = 1$. Finally, the relations between two windows is counted as $r = \{S, R, C, P\}$, where

$$\begin{cases} S_{ij} = \sqrt{\mu_i^2 + \mu_j^2} \\ R_{ij} = \sqrt{\sigma_i^2 + \sigma_j^2} \\ C_{ij} = \frac{S_{ij} R_{ij}}{\mu_i+\mu_j} \\ P_{ij} = \frac{C_{ij}}{\sigma_i+\sigma_j} \end{cases} \tag{2}$$

3) Window selection and feature extraction: The data obtained from the meters are continuous time series data, and the defined statistical features shown in Table I can be used to extract them. The selection of window size should be as different as possible to ensure the comprehensiveness of the data observation. In addition, the set standard window size should be appropriate with a practical period. For example, if the data obtained are from one meter for a whole year, then the window can observe the data according to the size of the week, month, and quarter so that it will be easier to find meaningful features. In this way, a group of window data can be obtained by observing a whole year of data at a certain time as $\mathcal{W} = \{w_{day}, w_{week}, w_{month}, w_{quarter}\}$.

As for the moving distance, generally, when the window moving distance is shorter, the periodicity of observed data can be better observed at the cost of bigger data volume. When the window moving distance is long, less attention is paid to the periodic laws of the data. Therefore, the moving distance can be gradually changed as the window moves. In this paper, the minimum window size is chosen as the initial moving distance, and then the moving distance is increased to get different window data. For example, if the minimum time window is one day, then a group of windows can be obtained by moving the steps for the day (7 times), week (4 times), or half month (2 times), and the differences between the original window and these moving windows are calculated. Finally,

Type	Statistic	Notation
Single window	**Mean**	μ
	Std.	σ
	Mode	m
	C. V.	ξ
Between windows	**Magnitude**	S
	Radius	R
	Approx. Covariance	C
	Correlation Coefficient	P

TABLE I: Statistical features based on time window segmentation

the complete features corresponding to the initial samples are $\mathcal{F} = \{W_0, W_1, r_{W_{01}} ..., W_7, r_{W_{07}}, W_{15}, r_{W_{015}}, W_{30}, r_{W_{030}}\}$.

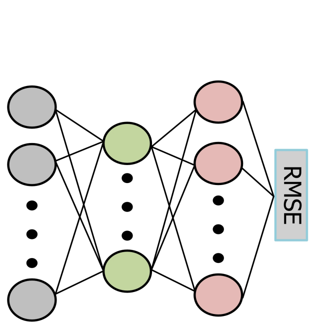
In addition, by changing the start time of the time window, different features ($\mathcal{F}$) can be obtained. In this paper, the starting points are selected with a fixed step interval, and then a set of features $\{F_i\}$ with the same dimension is obtained. To be used as the input feature of the neural network model, the corresponding dimension is numerically standardized. The data are standardized by using the Z-score (standard deviation standardization) method here, and the final calculation is as shown as follows:

$$F_i'(j) = (F_i(j) - mean(\sum_i^n F_i(j)))/std(\sum_i^n F_i(j)). \tag{3}$$

B. Multilayer hierarchical network

The traditional neural network takes all the features as the input data of the input layer, sets different numbers of hidden layers, and obtains the results from the output layer. Through the feedback and learning process, the optimal solution is finally obtained. Such a network model is simple and easy to understand. However, when all the features are taken into the same model without any difference, the relations among the features induced by training are unstable, the convergence ability of the network is poor, and the classification efficiency is low [22]. The CNN can be trained to investigate the relations among the input features, but it brings a high amount of computation, and the best parameters of the network model are difficult to set, which means it is hard to apply to real-time data classification. Therefore, the MLHN is applied here by using the data denoising ability of the autoencoder and the feature induction ability of the feedforward network. First, the features are input into different feedforward networks by groups to observe the intra-relations, and then all the outputs are used as the input of an autoencoder for anomaly detection or passed to another feedforward network for classification. After several inductions and transmission, the data can be used to obtain stable convergence results.

1) Feature encoder: A feature encoder is used to process the features of input data. Two types of feature encoders are used in this paper; one is a feedforward network for feature induction (Fig. 2a), and the other is the autoencoder for feature denoising (Fig. 2b). After inputting the characteristic data into the feedforward network, the network is iteratively induced by the loss between the output and the expectation. The



(a) autoencoder (b) feedforward

Fig. 2: Two types of feature encoders

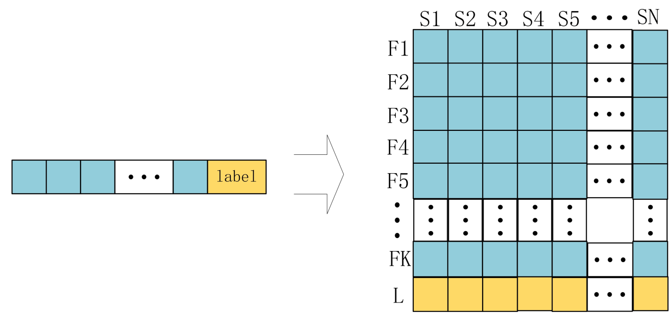


Fig. 3: Feature recombination for encoder

hidden layer of the feedforward network can be multilayer, and the number of neurons in the hidden layer is usually larger than the number of neurons in the input and output layers. In this case, the relation of different features can be better extracted. The autoencoder reduces the noise of the data by continuously training the input data. The number of neurons of the input layer is the same as that of the output layer, and only one hidden layer is in between. The number of neurons in the hidden layer is less than that in the input layer. The autoencoder iterates through the loss between the output data and the input data to reconstruct the features.

To achieve the data classification and abnormal information filtering through the obtained featured data, it is necessary to use two feature encoders to build a multilayer hierarchical network model. In this model, multiple feedforward networks are set based on the relation between the time window featured data, and further induction is carried out for the summarized results. Based on the autoencoder, the root mean square error (RMSE) is set as the loss function to make the distance between input and output closer. Through continuously training the data, a stable RMSE interval can be obtained, which can be used to detect abnormal data.

In particular, the number of neurons of the hidden layer of the autoencoder is set to be half of that of the input layer, while the number of neurons in the feedforward network is set to be 2 times of that of the input layer. For the autoencoder, the number of neurons of output layer is the same as that of the input layer, but the number of neurons in the output layer of the feedforward network should be set according to the proportion of each network. As shown in Figure 2, n neurons are in the input layer, and o neurons can be used after throwing k neurons in the hidden layer.

2) Feature recombination for encoder: The electricity data are collected from different meters, and the monitoring targets of different meters are not the same. It is necessary to detect whether the data has been tampered with; meanwhile, it is of interest to decompose the data to discern the occurrence time of electric larceny or other attacks. However, the information directly obtained from the meters is time series data, and this kind of data is difficult for further classification. To train and learn the data using the MLHN , it is necessary to first extract and label the data. Then, a set of two-dimensional data with the same number of features and labels is reconstructed according to the different starting time of observation, as shown in Figure 3.

As shown in Figure 3, the time series data is divided into n samples (called S1 to SN) according to the different window start times with a same interval, and the corresponding features (F1 to FK) of each sample are derived through the time window feature extraction method proposed in this paper. Finally, the one-dimensional time series data are transformed into a group of samples with the same label and the same feature dimension. After feature recombination, the meter recorded data are converted into a set of two-dimensional data, and each row of the data is taken as sample data of the meter recording. At the same time, the recombination feature is standardized to be used as the input of the feature encoders by the standardizing equation (3).

3) Unsupervised MLHN and anomaly detection: An unsupervised MLHN model is used to detect abnormal data in meter readings, such as electricity stealing data. The features of these abnormal data are not in line with the periodic law of normal electricity consumption data, and the abnormal data can cause sudden fluctuations of meter readings. Accordingly, the unsupervised MLHN uses the autoencoder as the last layer and several feedforward networks as the first layer of the input model. The evaluation standard of the network model is the difference in RMSE, which is determined from the output of the autoencoder through a number of similar samples. Through the cyclic training of the input data, the RMSE iteratively converges, and finally, a stable RMSE value is obtained by training a large number of raw data. Then the model is used to detect the abnormal data that falls outside the threshold range of RMSE after passing through the network.

The whole network structure is shown in Figure 4. The input data is the feature of each sample obtained by feature recombination. Then, according to the relationship between features, such as a single window feature and an inter-window feature, different features are input into a group of parallel feedforward networks to better understand the potential connections between them. Finally, the output of each feedforward network is aggregated as the input of an autoencoder for unsupervised learning, and the RMSE interval is obtained.

The training process of the unsupervised MLHN model is a self-iterative process, and the RMSE between each training result and input data itself is calculated. This model can make the same type of input data iteratively converge to a stable RMSE value, and the RMSE of the abnormal data after passing

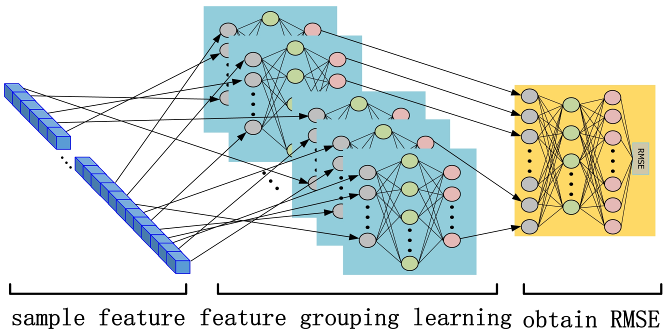


Fig. 4: Unsupervised MLHN structure

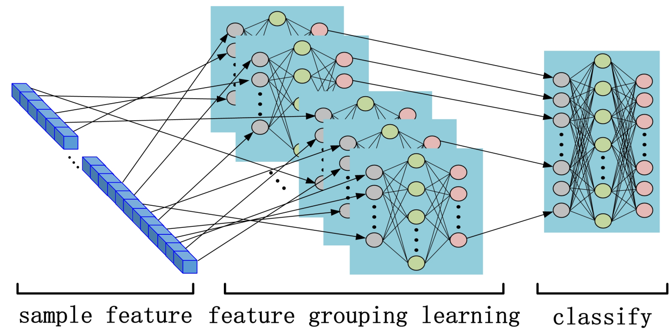


Fig. 5: Supervised MLHN structure

through the network will be much larger than that of the normal data. It is easy to find that the unsupervised model needs a large number of correct samples to train the model, and then it can be used for detection. Once an abnormal value is detected in the unsupervised MLHN, the starting time of the window corresponding to this set of data can be obtained, and the statistical characteristics of the windows with different sizes can also be obtained. By comparing the characteristics of data in different windows, it observes which window the differentiated data locate, and then the time interval of abnormal data can be found.

4) Supervised MLHN and classification: The purpose of an unsupervised multilayer network is to detect abnormal information in single meter recording, while the supervised multilayer network is designed to identify different meter data and distinguish deceptive data. In general, although there are few differences between fraud information and normal data, some work can differentiate the artificial and normal data. The last layer of a supervised multilayer network is a feedforward network, which takes the cross-entropy loss between the output value of training samples and the real label of samples as the feedback data to train the network model. The cross-entropy loss is given as

$$L = \sum_{i=1}^{n} y^{(i)} log\hat{y}^{(i)} + (1 - y^{(i)} log(1 - \hat{y}^{(i)})). \tag{4}$$

To summarize the feature and highlight part of the feature, the supervised MLHN also uses multiple feedforward networks to induce the features in the first layer. Classification and induction can be accomplished in two ways: one is to transmit and aggregate the features of each window and the associated features between windows through multiple feedforward network; the other is to transmit and aggregate the features of each window and the associated features of this window with other windows through multiple feedforward networks. Only the mapping method from the feature to the first combined layer is different between the two methods. At the same time, the same feature can be repeatedly forwarded into different feedforward networks in the first combined layer to better analyze the intrinsic relationship between features. Such an MLHN structure can better summarize the relations among the features, accelerate the operation speed of the neural network, and better train and classify (as shown in Figure 5). MLHN can also flexibly adjust the hierarchical structure to adapt to different tasks.

The number of input nodes of each feedforward network in the first layer of the supervised multilayer neural network is determined by the relevant input features, while the number of output nodes is determined by the importance ratio of these features. This means that the larger the proportion of output nodes to the final layer of MLHN, the more important this group of features is. It is obvious that the statistical characteristics of a single window account for a large proportion in distinguishing multiple meter readings. When distinguishing the abnormal readings of the same meter, the relation among the windows should take a larger proportion. To eliminate the outliers of the data itself in the training process, the autoencoder can be used to denoise the features in the first layer of MLHN, and then the feedforward networks can be combined in the next layer. The free combination of the autoencoder and the feedforward network can construct different MLHN models. These models can give full play to the ability of the autoencoder to stabilize the distance between the input and the output and the ability of the feedforward network to summarize the features.

5) Complexity analysis: MLHN is composed of multiple feedforward networks and autoencoders. A hierarchical network is faster than a single network, such as MLP. For a single feedforward network, it is assumed that the number of neurons in the input layer is n (same as the characteristic dimension of the input sample), the number of neurons in the hidden layer is x times that of the input layer, and the number of neuron nodes in the output layer is m (same as the number of categories of the sample). Then the time complexity of one sample is $O(n*x*n+x*n*m)$. The larger the x, the stronger the learning ability of the network, but the higher the complexity; thus, a general x value is 2 to 3 times of n, and the final complexity is $O(n^2+n*m)$.

When using the hierarchical structure, the input features are divided into p groups, and each group is input into a feedforward network. The complexity of each feedforward network is $O((\frac{n}{p})^2+\frac{n}{p}*q)$, where q is the number of output elements of each neural network. Then the output of each network is aggregated as the input of a new network whose time complexity is $O(pq*pq+pq*m)$. Generally, the number of networks p and the type of samples m are small constants, so the time complexity of the hierarchical network

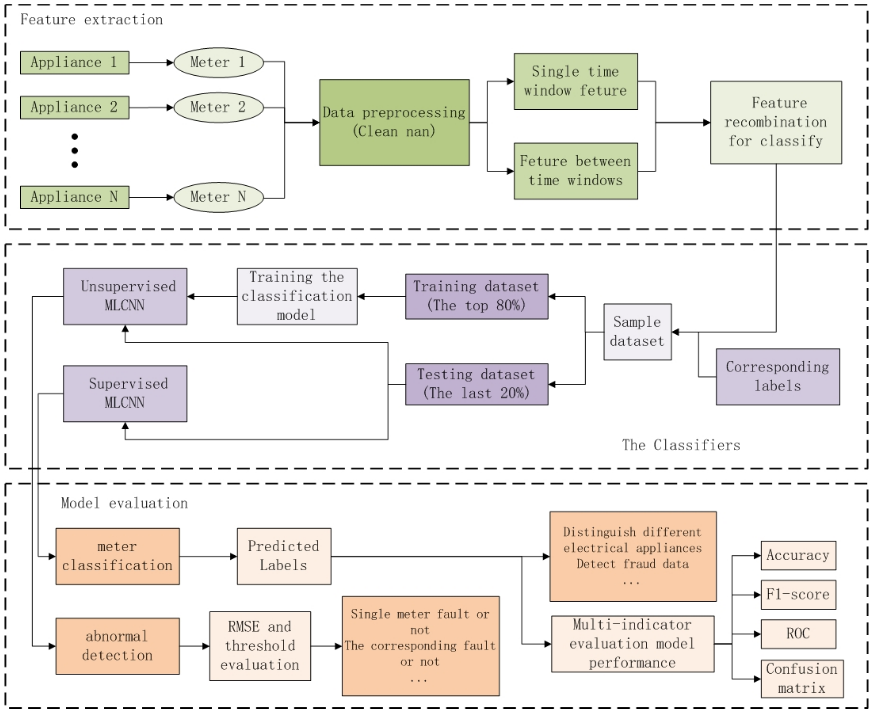


Fig. 6: Overall framework of the proposed model

is $O(q^2+q)$. The time complexity of one hierarchical layer is $O(n^2+n*q+q^2+q)$. With the increase of the number of layers, the final time complexity is $O(n^2+n*q+\sum_{i=1}^{k}(q_i^2+q_i*q_{i+1}+q_{i+1}^2+q_{i+1}))$. Since the number of neurons in the output layer of each feedforward and autoencoder is less than or equal to the number of neurons in the input layer, so $p*q \leq n$ and $q_{i+1} \leq q_i$. Moreover, the number of layers k used in MLHN is also constant, so the worst-case time complexity of MLHN is the same as that of the single feedforward network, which is $O(n^2)$.

To sum up, to fully explore the electricity data of the household appliances, the feature extraction method is first used for the preprocessed data, which takes advantage of the combination of the time domain and frequency domain features. The time window is used to analyze the time series data, and the statistical characteristics in a single window and the differences between multiple windows are obtained. Then, combined with the statistical characteristics above and the labels of the household electrical meter, the sampled data for the NN model are constructed. Second, through this set of data, the automatic encoder and RMSE values are combined to conduct unsupervised clustering training on the training set data so that the network model can produce RMSE values that are significantly different from normal data to abnormal data. Third, the MLHN is constructed using the feature induction ability of the feedback network for the input data, which is used to induce the input feature data again. Then, the potential relations between the features of two single windows and among different windows can be found to improve the recognition accuracy of the network models for different electrical appliance data so that the model can distinguish normal meter data and fraud or other abnormal data. Finally, the accuracy, recall rate, and confusion matrix are used to evaluate the classification effect of the model, and it is able to find which house appliance may have electricity-related problems. The proposed method can be explicitly described in the framework shown in Figure 6.

IV. Performance Evaluations

To evaluate the presented scheme, the experiments are conducted on a realistic electricity consumption dataset from [18], containing several weeks of power data for six different homes and high-frequency current/voltage data for the main power supply of two of these homes. The so-called Redd data set records the readings per second of each meter and the total readings of all meters in six rooms. The readings per second of these meters are separately recorded in their own files so that different readings can be easily marked. Here, energy consumption data collected in Room1 is used, which includes two site meters and several electrical appliances. Each electrical appliance and its corresponding label are shown in the Table II, in which “membership” means that the equipment is subordinate to other electrical appliances. For example, the data of the meter (label 1) contains all the appliance data with membership value of 1.

The sizes of the groups of time windows that have been used in this paper are shown in Table III, and the features of the meter readings are counted by these windows. In addition, the groups of moving steps of these windows are also set as shown in Table III. As shown in Table I, each single window has four statistic features, and the number of features between a pair of windows is four. Then, according to the window size in Table III, 24 features $(6*4)$ can be obtained. With each move step, six new windows and five windows pairs (the first

name	site meter	site meter	electric oven	electric oven	fridge	dishwasher	sockets	sockets	light	washer dryer
label	1	2	3	4	5	6	7	8	9	10
membership			1	1	1	1	1	2	1	1
name	microwave	unknown	electric space heater	electric stove	sockets	sockets	light	light	unknown	microwave
label	11	12	13	14	15	16	17	18	19	20
membership	1	1	1	1	3	4	2	3	2	1

TABLE II: The electrical appliances and their corresponding Redd labels

name	values
autoencoder	(322,160,322)
Unsupervised MLHN	{(168,320,80),(154,320,80)},{(160,80,160)}
feedforward network	(322,600,20)
supervised MLHN	{(168,320,80),(154,320,80)},{(160,320,20)}
window size	{1,30,60,480,720,1440}minutes
move steps	{1,5,15,30,60,720,1440}minutes

TABLE III: The main parameters in the experiment

window acts as a unit window, and its inter window features are not needed) between the corresponding windows will be generated. Then, according to the move steps in Table III, 308 features $(7*(6*4+5*4))$ are obtained. Finally, a set of sample data with dimension 323 ($24+308+1$, where 1 means label) can be obtained by setting the start time of the window with 1 minute intervals and is standardized by (3). This work can also be compared with [19] and [23] to verify the advantages of the presented feature extraction method and MLHN .

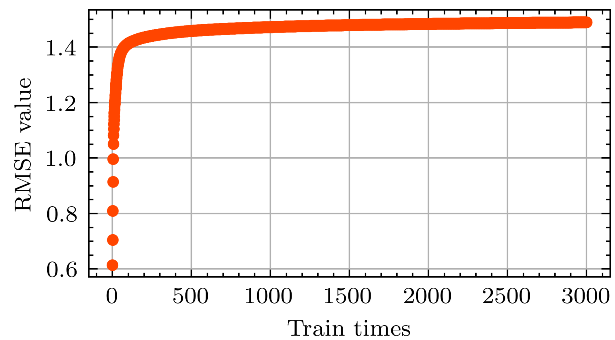
A. Unsupervised anomaly detection

First, the data obtained from a single meter are used for training and predicting through the unsupervised MLHN. Here, the data from the site meter (label 1) are selected. The ratio of the training set to test set is set at $7:3$.

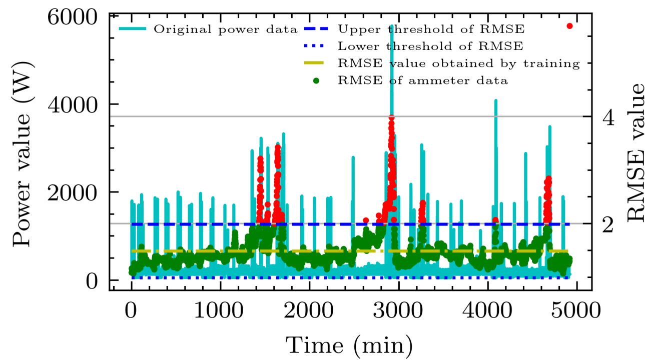
1) Autoencoder: When no correlation between the data features is observed, the MLHN formed by a single autoencoder can be used for training. In this experiment, the input layer has 322 neurons, a hidden layer has 160 neurons, and the output layer has the same number of neurons as the input layer. The numbers of neurons in the experiment are shown in Table III, where "(a, b, c)" represents the number of neurons in the input, hidden, and output layers, respectively. The "$\{\}$" represents multiple sub network modules in parallel. Finally, sigmoid is used as the activation function in the experiment.

In addition, the weight matrix between the hidden layer and the output layer of the autoencoder uses the same transposition of the weight matrix between the input layer and the hidden layer, which can accelerate the training speed of the training process. During the training process, the RMSE between the output and the input is continuously calculated and recorded. According to the theoretical analysis in the previous section, after training with a single meter data, a stable RMSE with a threshold range can be obtained. Then, after the abnormal data are input into the network, the RMSE value will exceed the threshold. Then, the abnormal electricity data in a single meter can be distinguished. The specific experimental results are shown in Figure 7.

Figure 7a shows that with the increase of training times, the RMSE value becomes stable after approximately 1000 training times. It shows that the data features of a single meter can



(a) RMSE change of single autoencoder



(b) Training and detection results of single autoencoder

Fig. 7: RMSE change of single autoencoder and detection effect

be stable at a specific RMSE value through the autoencoder network. After that, the RMSE value and a threshold value of 0.5 are used to detect the test data, and the results are shown in Figure 7b. As long as the RMSE value of the test data is within the range of 1.45 ± 0.5, it is regarded as normal data. If it exceeds the range, it is possibly abnormal electricity consumption data. In Figure 7b, it can be seen that the meter data are in normal periodic fluctuation in most cases, and the RMSE value obtained is within the reasonable range of 1.45 ± 0.5.

When dense data are higher than the normal cycle, the RMSE value will increase, as shown in the red dot in the figure. This shows that during the abnormal RMSE period, the use of high-power electrical appliances suddenly appeared in the electrical appliances monitored by the site meter (label 1). By comparing the electric oven data (label 3), the abnormal RMSE period is similar with the use time of the electric oven. It can be observed that the autoencoder network can detect a group of data with small continuous changes or values with large fluctuations. However, it is not sensitive to abnormal electricity data with a short duration and smaller changes.

2) Unsupervised MLHN: There are 322 neurons in the input layer of a single autoencoder network, and there is no connection between the input features. To solve this problem, an unsupervised multilayer neural network model is used. First, the featured data is input into different feedforward networks according to the relationship between the features, and then the results of each feedforward network are input into an autoencoder in the last layer. Because the correlation of features in a single window can better represent the electricity consumption data, the corresponding features between different windows can represent the difference of the electricity data in different intervals. Therefore, the data representing a single window in the eigenvector and the relationship data between different windows are input into two feedforward networks, respectively. Then the results of the two feedforward networks are used as the input of the autoencoder to construct the multilayer hierarchical network. After training and forecasting the electricity data through the model, the detection effect is basically consistent with the model using a single autoencoder, with a faster training speed.

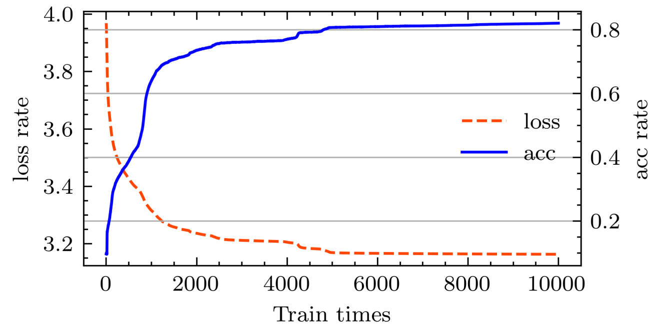
B. Supervised classification

For the data from different meters, it is necessary to distinguish different data types to identify the data sources. It is also aimed at finding the forged information in the data and classifying the forged data that does not belong to the meter. The electricity data of 20 meters are selected for feature extraction in this experiment. At the same time, the training and testing are conducted with the ratio of the training set and a test set of 8 : 2.

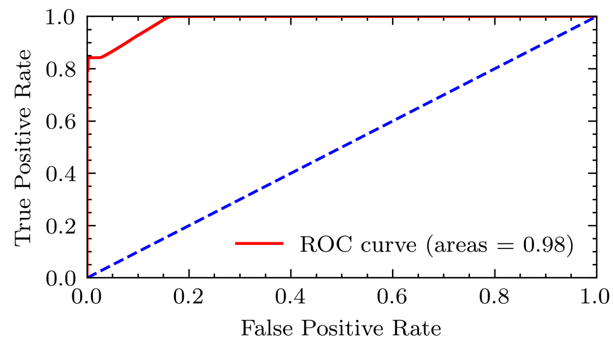
1) Feedforward network: Similar to the experiment on unsupervised multilayer hierarchical networks, a single feedforward network is used to construct the network firstly. The number of input neurons of the network is 322, the number of neurons in the middle layer is 600, and the number of neurons in the output layer is set to 20 according to the number of data categories. The experimental results are shown in Figure 8.

It can be seen from Figure 8a that the value of loss decreases with the increase of training times, which proves that the training process converges. The training accuracy rate is also increased to more than 80% with the training times. Meanwhile, using the trained model to predict and verify the test set, the classification results of 20 kinds of data and the average ROC chart are shown in Figure 8b. It can be observed that a high level of consistency is achieved.

The classification results of each category are shown in Table IV, and the confusion matrix of the prediction test set is shown in Figure 9. Most of the meter types can be accurately predicted, and the recall rate and accuracy rate of each meter data are both high. However, the accuracy level of meter numbers 3, 4, 15, 16, 19, and 20 are only about 50%, and their recall rate is poor. Note that meters 15 and 16 are sub-devices of meters 3 and 4, respectively. From the results, it can be found that the feature extraction method and single feedforward network can achieve more accurate classification. Therefore, it is difficult to identify the data for the electric oven (3, 4) and its socket (15, 16), the microwave (20), and the unknown equipment (19). Even the recognition preparation rates of the dish washer (6), the washer dryer (10), and the light (17) are less than 80%.



(a) Changes of loss and accuracy during training



(b) Average ROC curve of 20 classification

Fig. 8: Training process and prediction effect of single feedforward network

class	precision	recall	f1-score	support
1	0.90242	0.99901	0.94826	1009
2	0.87683	1.00000	0.93437	413
3	0.56579	0.51190	0.53750	420
4	0.54591	0.47414	0.50750	464
5	0.81720	0.99346	0.89676	459
6	0.95367	0.59518	0.73294	415
7	0.75086	0.99772	0.85686	438
8	0.87683	1.00000	0.93437	420
9	0.75698	1.00000	0.86168	461
10	0.57428	0.74941	0.65026	423
11	0.80712	0.99538	0.89142	433
12	0.66423	0.79825	0.72510	456
13	0.81136	1.00000	0.89586	400
14	0.77593	0.82379	0.79915	454
15	0.94215	0.49891	0.65236	457
16	0.87698	0.51276	0.64714	431
17	0.96407	0.71875	0.82353	448
18	0.85822	1.00000	0.92370	454
19	0.95475	0.48394	0.64231	436
20	0.77003	0.49552	0.60300	446

TABLE IV: Prediction results of 20 kinds of electricity meter data under single feedforward network

2) Supervised MLHN: To improve the classification accuracy, the multiple feedforward networks are also set as the first layer of MLHN to give priority to the grouping induction of features. First, 168 eigenvalues about the single window feature are taken as the input of the first feedforward network, and then the other correlation data between windows are used as the input of the second feedforward network. The output

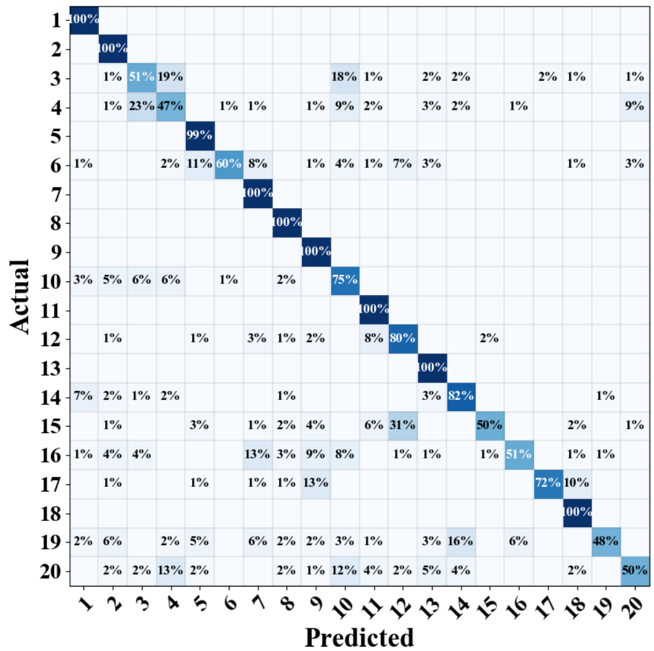
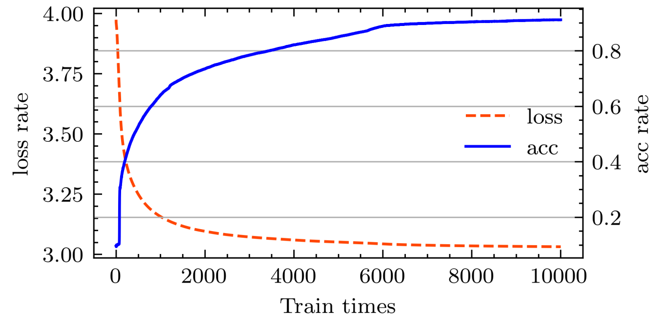
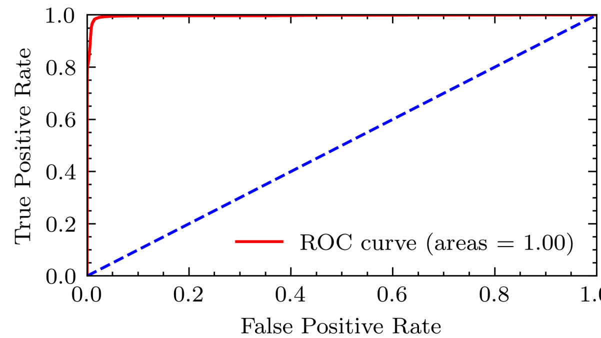


Fig. 9: Confusion matrix of single feedforward network



(a) Loss and accuracy in training process



(b) Average ROC chart of classification results

Fig. 10: Training process and prediction results of supervised MLHN

neurons of the two feedforward networks are set to 80. Then, the output of the two encoders are taken as the input of the last feedforward network, and the number of output neurons in the last layer of MLHN is set to 20. Each of the three feedforward encoders has one hidden layer, and the number of neurons of the hidden layer is 320. The same training data and test data with the single feedforward network are used. The training results are shown in Figure 10.

Compared to the result achieved by using a single feedforward network in Figure 8, it can be seen from the figure that under the same training times, the changes of the loss value (loss) and accuracy rate (acc) in the training process are smoother and the convergence speed of the loss is faster. At the same time, the training accuracy of the supervised MLHN can reach more than 90%. The average accuracy of the 20 classifications is better. After many repeated experiments, the average prediction accuracy of the supervised MLHN is 0.891340, the average prediction accuracy is 0.895282, the average recall rate is 0.891340, the average micro value of F1 is 0.891340, the average macro value of F1 is 0.876949, and the average kappa coefficient is 0.884887, which is consistent with the prediction. To see the classification result of each category in the classification process more intuitively, the confusion matrix of 20 categories is drawn in Figure 11.

From the confusion matrix, it can be seen that the data recognition accuracy levels of meter numbers 3 and 15 are poor, and the accuracy levels of the other meters are higher. Meters 15 is a sub-devices of meters 3. By using the MLHN, the classification accuracy can be improved. Even the recognition preparation rate of the electric oven (4) and its sockets (16) is more than 80%. The speed of training and the speed of loss convergence are faster than that of a single feedforward network.

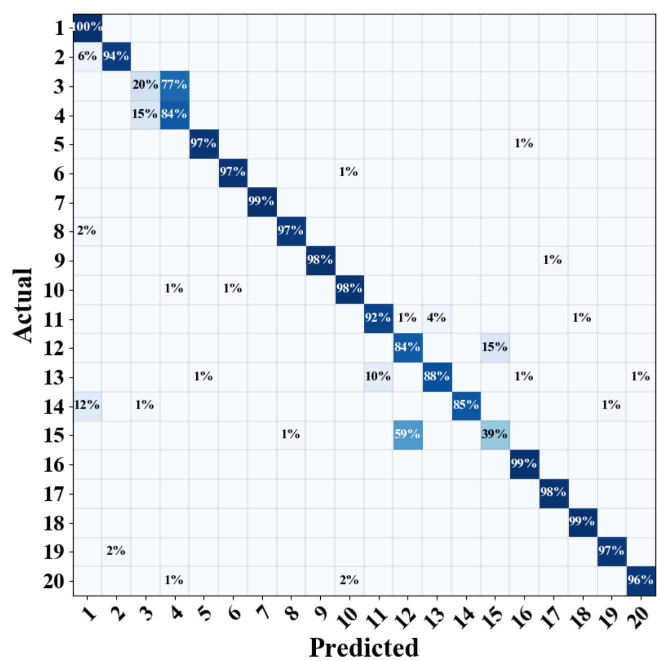
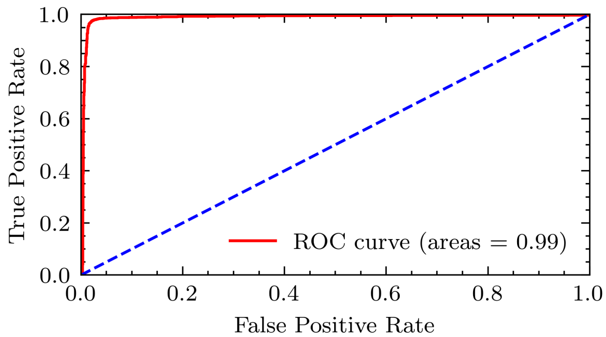
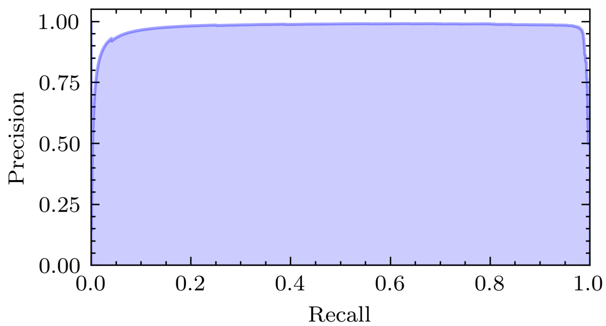


Fig. 11: Confusion matrix of supervised MLHN

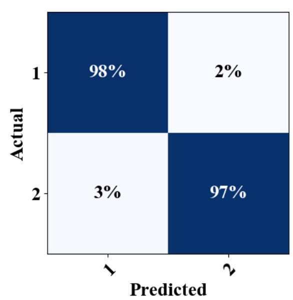
To further verify the advantages of the MLHN, the data for meters 3 and 15, which have the worst recognition accuracy, are selected for the binary classification with MLHN. The ROC curve, the PR curve, and the confusion matrix are shown in Figure 12 and the accuracy of the MLHN for two of these tasks can reach up to 98%. Therefore, the feature extraction



(a) Average ROC chart of binary classification

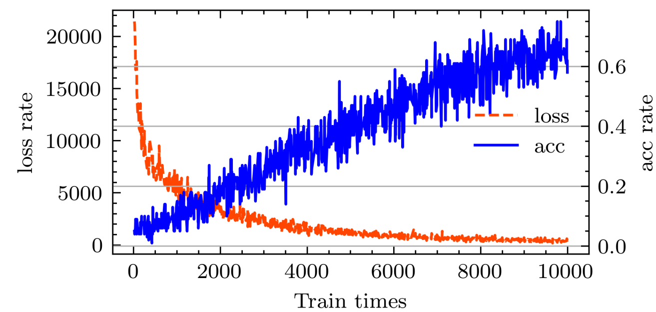


(b) Average PR chart of binary classification

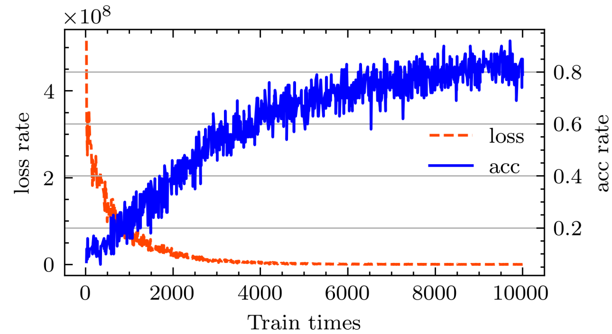


(c) Confusion matrix of binary classification

Fig. 12: Binary classification result of supervised MLHN



(a) Training process with original data



(b) Training process with extracted features

Fig. 13: Training process of original data and extracted features in CNN

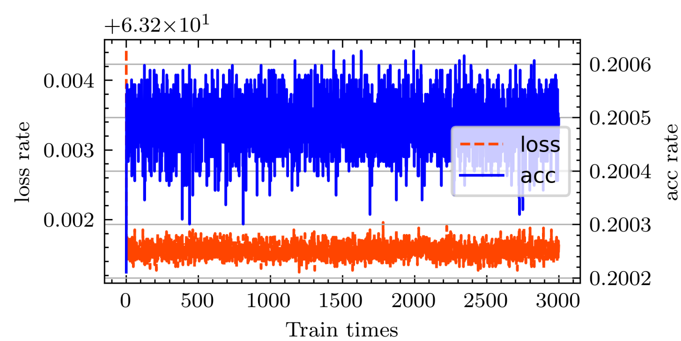


Fig. 14: Training result of original data in MLHN

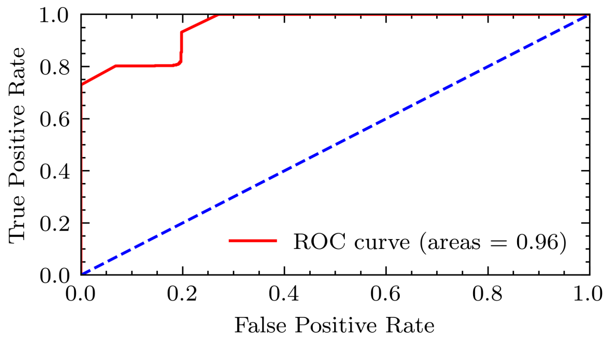
method and the MLHN can accurately distinguish the data from different meters.

C. Comparative experiment

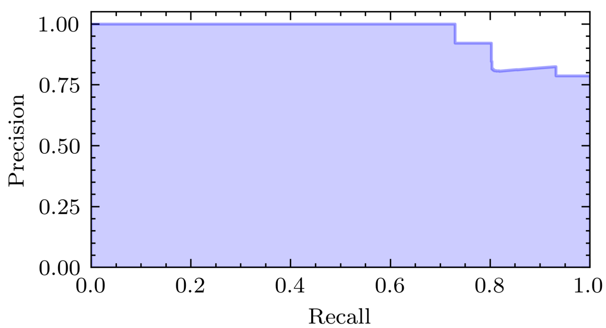
To further demonstrate the advantages of the proposed model, a number of comparative experiments are shown here. For comparison purpose, in the comparative experiment, the raw data are used in the CNN proposed in [19] to further detect the abnormal data, and the MLHN is used for classification training for comparison. Then the data obtained by the feature extraction algorithm is used as input, and the CNN and the SVM are used for classification training. First, the training process with 20 meter readings and the CNN is shown in Figure 13.

It can be seen that when using raw data for training, the convergence process is slower than that using the extracted feature. Under the same training times, the training accuracy of the raw data is also lower than that of the extracted feature. The average accuracy of the model trained with the raw data is finally about 68.9547%, while the average accuracy of the model using extracted features is 83.2951%. In addition, it can be seen from Figure 14 that when using the raw data and the same MLHN, the loss still fails to converge after 3000 times of training, and the training accuracy fluctuates around 20%. This proves that the periodicity and the correlation characteristics of the original data cannot be directly observed by the feedforward network and that some preprocessing is needed. Therefore, it is impossible to use raw data and the MLHN for classification training.

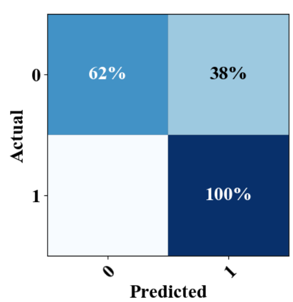
Moreover, the proposed scheme is also compared with the SVM method to see the accuracy improvement for data classification. From Figure 15, it can be seen that the accuracy of binary classification results under the SVM is lower than that under the MLHN. It is easy to find that the accuracy of data classification for meter number 3 is quite low, which also proves the effectiveness of the proposed scheme.



(a) Average ROC chart of binary classification



(b) Average PR chart of binary classification



(c) Confusion matrix of binary classification

Fig. 15: Binary classification result of SVM

V. Conclusion

With the popularity of household smart meters and electricity monitoring sensors, a large amount of data can be obtained to analyze household electricity usage so as to better diagnose the leakage and theft behaviors, identify man-made tampering and data fraud, and detect the powerline loss. In this paper, the time window method is first proposed to obtain the features and potential periodicity of household electricity data. Combining the denoising ability of the autoencoder and the induction ability of the feedforward neural network, a multilayer hierarchical network is then established to detect anomalies in a single meter data and classify multiple groups of meter data, respectively. Through extensive experiments, it is found that the accuracy of detecting abnormal data and data classification can be significantly improved with the presented scheme. In the future, we will leverage the machine learning-based method to further develop the abilities to detect faults in household appliances, estimate household energy prices, and warn consumers about household energy waste or theft.

References

[1] Y. Wang, Q. Chen, T. Hong and C. Kang, "Review of Smart Meter Data Analytics: Applications, Methodologies, and Challenges," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 3125–3148, 2019.

[2] R. Jiang, et al, "Energy-Theft Detection Issues for Advanced Metering Infrastructure in Smart Grid," *Tsinghua Science and Technology*, vol. 19, no. 2, pp. 105–120, 2014.

[3] M. S. Eduardo, V. S. Andrea, M. G. Alicia, "Forecasting the Electricity Hourly Consumption of Residential Consumers with Smart Meters using Machine Learning Algorithms," *2019 IEEE Milan PowerTech*, Milan, Italy. 334-341, 2019.

[4] S. Rahman, G. R. Alam, M. Rahman, "Deep Learning based Ensemble Method for Household Energy Demand Forecasting of Smart Home," *22nd International Conference on Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh. 314-317, 2019.

[5] B. Yildiz, J. I. Bilbao, J. Dore, A. Sproul, "Optimizing Consumer-Side Electricity Usage in a Smart Household," *24th International Conference on Methods and Models in Automation and Robotics (MMAR)*, Miedzyzdroje, Poland. 256-264, 2019.

[6] A. M. Alonso, F. J. Nogales, C. Ruiz, "Hierarchical Clustering for Smart Meter Electricity Loads Based on Quantile Autocovariances," *IEEE Transactions on Smart Grid*, vol. 11, no. 5, pp. 4522–4530, 2020.

[7] D. Ignatiadis, G. Henri, R. Rajagopal, "Forecasting Residential Monthly Electricity Consumption using Smart Meter Data," *2019 North American Power Symposium (NAPS)*, Wichita, KS, USA,.

[8] S. Yan, K. Li, F. Wang, X. Ge, et al., "TimeFrequency Feature Combination Based Household Characteristic Identification Approach Using Smart Meter Data," *IEEE Transactions on Industry Applications*, vol. 56, no. 3, pp. 2251–2262, 2020.

[9] S. Ryu, H. Choi, H. Lee and H. Kim, "Convolutional Autoencoder Based Feature Extraction and Clustering for Customer Load Analysis," *IEEE Transactions on Power Systems*, vol. 35, no. 2, pp. 1048-1060, March 2020

[10] D. Yan, Y. Jin, H. Sun, B. Dong, et al., "Household appliance recognition through a Bayes classification model," *Sustainable Cities and Society*, vol. 46, no. 3, pp. 1–11, 2019.

[11] M. Ozay, I. Esnaola, F. T. Yarman Vural, S. R. Kulkarni and H. V. Poor, "Machine Learning Methods for Attack Detection in the Smart Grid," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 8, pp. 1773-1786, Aug. 2016.

[12] N. Li, L. Wang, X. Li and Q. Zhu, "An effective deep learning neural network model for shortterm load forecasting," *Concurrency Computation*, vol. 32, no. 7, pp. 32–42, 2019.

[13] H. Lida, W. Yong., "A whale optimization algorithm-trained artificial neural network for smart grid cyber intrusion detection," *Neural Computing and Applications*, vol. 32, no. 13, pp. 9427–9441, 2020.

[14] C. He, M. Ma, P. Wang, "Extract interpretability-accuracy balanced rules from artificial neural networks: A review," *Neurocomputing*, vol. 387, no. 28, pp. 346–358, 2020.

[15] L. Li, O. Kaoru, M. Dong, "Everything is image: CNN-based short-term electrical load forecasting for smart grid," *Proceedings of 14th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN 2017)*, Exeter, United kingdom. 344-351, 2017.

[16] A. N. Mohamed, M. Karar, "Accurate photovoltaic power forecasting models using deep LSTM-RNN," *Neural Computing and Applications*, vol. 31, no. 7, pp. 2727–2740, 2019.

[17] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu and Y. Zhang, "Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841-851, Jan. 2019

[18] Z. Kolter and M. J. Johnson. "REDD: A public data set for energy disaggregation research," *In proceedings of the SustKDD workshop on Data Mining Applications in Sustainability,* 2011.

[19] Z. Zheng, Y. Yang, X. Niu, H. Dai, "Wide and Deep Convolutional Neural Networks for Electricity-Theft Detection to Secure Smart Grids," *IEEE Transactions on Industrial Informatics*, vol. 4, no. 14, pp. 1606–1615, 2018.

[20] S. Wang, S. Bi, Y. A. Zhang, "Locational Detection of the False Data Injection Attack in a Smart Grid: A Multilabel Classification Approach," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8218–8227, 2020.

[21] V. Chandola, A. Banerjee, V. Kumar, "Anomaly detection," *ACM Computing Surv.*, vol. 41, no. 3, pp. 1–58, 2009.

[22] J. Valls, R Aler, I. Galvn, D Camacho, "Supervised data transformation and dimensionality reduction with a 3-layer multi-layer perceptron for classification problems," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 1, pp. 1–13, 2021.

[23] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng and Z. Han, ”Detecting Stealthy False Data Injection Using Machine Learning in Smart Grid,” *IEEE Systems Journal*, vol. 11, no. 3, pp. 1644-1652, Sept. 2017.

[24] N. H. Md., T. R. Nishat, N. Abdullah-Al and M. Islam, “Electricity theft detection in smart grid systems: A CNN-LSTM based approach,” *Open Access Journal*, vol. 12, no. 17, pp. 118, 2019.

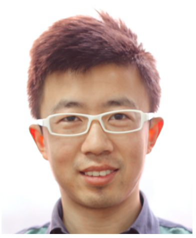
[25] B. T. Souza, “Evolution of Machine Learning in Smart Grids,” *International Conference on Smart Energy Grid Engineering (SEGE 2020)*, Oshawa, ON, Canada, Jan. 82-87, 2019.

[26] Y. Mirsky, T. Doitshman, Y. Elovici and A. Shabtai, ”Kitsune:An Ensemble of Autoencoders for Online Network Intrusion Detection,” *In proceedings of the Network and Distributed System Security Symposium*, 2018.

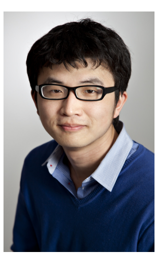
[27] K. Wang, C. Xu, Y. Zhang, S. Guo, et al, ”Robust Big Data Analytics for Electricity Price Forecasting in the Smart Grid,” *IEEE Transactions on Big Data*, vol. 5, no. 1, pp. 3445, 2019.

[28] J. Shi, C. Shi, L. Han and M. Xu, ”Smart grid short-term load estimation model based on BP neural network,” *Inderscience*, vol. 11, no. 2, pp.123136, 2020.

[29] Y. Jin, D. Yan, X. Zhang, M. Han, et al., ”District household electricity consumption pattern analysis based on auto-encoder algorithm,” *IOP Conference Series: Materials Science and Engineering*, Bari, Italy. 177-184, 2019.



Qiang Zhao was born in Shanxi, China, in 1987. After graduating from Yanshan University with a Bachelors Degree in 2011 and a Doctor’s Degrees in 2017, he works as a teacher at Department of Software and Engineering, Shanxi University. His research interests include image processing and machine learning.



Zheng Chang (S’10-M’13-SM’17) received the B.Eng. degree from Jilin University, Changchun, China in 2007, M.Sc. (Tech.) degree from Helsinki University of Technology (Now Aalto University), Espoo, Finland in 2009 and Ph.D degree from the University of Jyväskylä, Jyväskylä, Finland in 2013. Since 2008, he has held various research positions at Helsinki University of Technology, University of Jyväskylä and Magister Solutions Ltd in Finland. He was a visiting researcher at Tsinghua University, China, from June to August in 2013, and at University of Houston, TX, from April to May in 2015. He has been awarded by the Ulla Tuominen Foundation, the Nokia Foundation and the Riitta and Jorma J. Takanen Foundation for his research excellence. He has been awarded as 2018 IEEE Communications Society best young researcher for Europe, Middle East and Africa Region.

He has published over 100 papers in Journals and Conferences, and received best paper awards from IEEE TCGCC and APCC in 2017. He serves as an editor of IEEE Wireless Communications Letters, Springer Wireless Networks and International Journal of Distributed Sensor Networks, and a guest editor for IEEE Network, IEEE Wireless Communications, IEEE Communications Magazine, IEEE Internet of Things Journal, IEEE Transactions on Industrial Informatics, Physical Communications, EURASIP Journal on Wireless Communications and Networking, and Wireless Communications and Mobile Computing. He was the exemplary reviewer of IEEE Wireless Communication Letters in 2018. He has participated in organizing workshop and special session in Globecom’ 19, WCNC’18-21, SPAWC’19 and ISWCS’18. He also serves as TPC member for many IEEE major conferences, such as INFOCOM, ICC, and Globecom. His research interests include IoT, cloud/edge computing, security and privacy, vehicular networks, and green communications.

Geyong Min is a Professor of High Performance Computing and Networking in the Department of Computer Science within the College of Engineering, Mathematics and Physical Sciences at the University of Exeter, United Kingdom. He received the PhD degree in Computing Science from the University of Glasgow, United Kingdom, in 2003, and the B.Sc. degree in Computer Science from Huazhong University of Science and Technology, China, in 1995. His research interests include Computer Networks, Wireless Communications, Parallel and Distributed Computing, Ubiquitous Computing, Multimedia Systems, Modelling and Performance Engineering.