IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 10 September 2022, accepted 25 September 2022, date of publication 3 October 2022, date of current version 25 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3211427

RESEARCH ARTICLE

Real-Time Energy Disaggregation Algorithm Based on Multi-Channels DCNN and Autoregressive Model

LINTAO DENG[1], CHENGXIN PANG[1], (Member, IEEE), XINHUA ZENG[2], JUN ZHANG[3], AND CHIZHI HUANG[3]

[1]School of Electronics and Information Engineering, Shanghai University of Electric Power, Shanghai 200000, China
[2]Academy for Engineering and Technology, Fudan University, Shanghai 200433, China
[3]Distribution Network Operation and Maintenance Department, Nari Technology Nanjing Control Systems Company Ltd., Nanjing, Jiangsu 211106, China

Corresponding author: Chengxin Pang (chengxin.pang@shiep.edu.cn)

This work was supported by the State Grid Corporation of China through the Science and Technology Project ''Research on Service Driven Planning Method and Key Technologies of Electric Sensor Network'' under Grant SGSCJY00GHJS2000014.

ABSTRACT Energy disaggregation refers to the process of obtaining the energy consumption of several appliances in a house by disaggregating the aggregate power consumption measured by an electrical meter. Currently, deep learning methods are widely applied in this field. Real-time energy disaggregation is an important branch of energy disaggregation. Based on the Short Sequence-to-Point (Short Seq2point) (Odysseas) network structure, a real-time energy disaggregation algorithm based on multi-channels deep convolutional neural networks (MC-DCNN) and autoregressive model (AR) is proposed in this paper, which obtains theenergy consumption of appliances at the current time point by disaggregating the historical aggregate power consumption to achieve delivering disaggregation results in real-time. The proposed method takes the original aggregate power sequence and differential power signal as the input of the network, and extracts the information of different time lengths in the sequence using multi-channels deep convolutional neural networks with a modified concatenate layer, so that the network can adapt to different appliances with different operating modes. In addition, the traditional autoregressive model is added as the linear component for solving the problem that the scale of the output is insensitive to the scale of the input in the neural network model. Finally, the proposed method was tested on the UK-DALE and REDD datasets, and the experimental results show that the method has good disaggregation performance on both datasets, has a small number of parameters and achieves fast inference.

INDEX TERMS Energy disaggregation, short seq2point, MC-DCNN, autoregressive model.

I. INTRODUCTION

Energy disaggregation (also referred to as non-intrusive load monitoring (NILM)) was originally proposed by George Hart [1], [2] and refers to the process of extracting the energy consumption of individual appliances from the total energy consumption of all appliances in a residence. Compared with intrusive load monitoring, NILM does not require sensors to be installed on each appliance to monitor its operation, but only requires a sensing device to be installed at the entrance of the home to obtain the aggregate power information, which can then be disaggregated algorithmically to obtain the electricity consumption of each appliance, with the advantages of convenience and low cost [3].

There has been a lot of studies in NILM, and pattern recognition-based methods are one of the major research directions, which include supervised and unsupervised learning algorithms. Supervised learning algorithms require the labeling of each device to enable the energy disaggregation system to identify the devices, including artificial

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

neural networks (ANN) [4], decision trees [5], support vector machines (SVM) [6], [7], K-nearest neighbors (KNN) algorithms [8], [9], and so on. For unsupervised learning algorithms, the labeling process is not required for system modeling, but the accuracy is a little lower than supervised learning, and the main methods adopted are k-means clustering [10], Hidden Markov Model (HMM) [11], Expectation Maximization (EM) [12], etc. Both unsupervised learning algorithms and supervised learning methods require feature mapping and transformation techniques to extract device-independent features so as to obtain robust features for effectively modeling NILM systems.

Deep learning algorithms have recently been widely adopted in the fields of computer vision [13], [14], speech recognition [15], [16], and natural language processing [17], [18] with very excellent results, which are able to extract the intrinsic features of the original data without the need for specialized knowledge, which has prompted many researchers to carry out researches related to energy disaggregation based on deep learning. Kelly et al. pioneered the idea of sequence-to-sequence into this field and designed three deep neural network architectures on the basis of convolutional and recurrent neural networks, namely Long short-term memory, Denoising autoencoder and Rectangle that regresses on the beginning and ending times of activation and the average power consumption of every device. All these networks outperform the combinatorial optimisation (CO) and factorial hidden Markov model (FHMM) algorithms on the UK-DALE dataset [19]. The adjacent windows of the input sequences of the sequence-to-sequence model overlap each other, resulting in predicting every element of the output sequence many times; also, the model cannot use all nearby elements of the input sequence for predicting elements at the edges of the window. To address these problems, Zhang et al. proposed the sequence-to-point (seq2point) model, with the input being the aggregate power sequence of window length and the output being the target equipment power value at the middle point of the window, and the disaggregation performance is better than that of the sequence-to-sequence (seq2seq) model [20]. Yang et al. [21] proposed a sequence-to-point model based on temporal convolutional networks, using dilated convolution to obtain larger receptive field and introducing residual blocks to avoid degradation problems, which significantly improved the network performance and reduced the model parameters. Zhou et al. [22] proposed a multi-scale residual network, which consists of dilated convolutional residual blocks as the basic structural unit, residual blocks are sequentially connected into a residual block body, and multiple residual block bodies of different depths are connected in parallel to form multi-branches structure for learning mixed-data features. The results show that the model has improvement on disaggregation performance and model complexity across different devices. Antoine et al. proposed an energy disaggregation method based on the variational autoencoders framework, which consists of two parts, the encoder extracts the target device information from the input signal and the decoder reconstructs the power signal of the target device. The method achieved excellent performance on the UK-DALE and REFIT datasets [23]. Considering the difficulty of obtaining large amount of labeled training data, Cui et al. proposed a method for estimating power consumption via background filtering, which uses only synthetic aggregate data to train the neural network, reducing the difficulty of obtaining training data and obtaining better performance [24].

In order to select more effective features from numerous appliance features, several researchers have introduced attention mechanisms into NILM. Chen et al. proposed a novel neural network architecture called scale- and context-aware network (SCANet), which utilizes a multi-branch architecture to extract multi-scale feature, a self-attention module to integrate context information and adversarial loss and state augmentation to improve accuracy. The experimental results showed a significant improvement in model performance compared to the state-of-the-art models [25]. Based on bidirectional encoder representations from transformers (BERT), Yue et al. proposed a structure called BERT4NILM, which utilizes multi-head attention for energy disaggregation. With the proposed loss function and masking training procedure, the proposed method outperforms the state-of-the-art models in various metrics on the UK-DALE and REDD datasets [26].

In order to estimate the power value of the appliance at moment t, the above studies used future data as part of the input, which contains the future operation state of the appliance (e.g., the appliance is turned on or off; the appliance is operating in another mode), and this information can help the network to perform more accurate disaggregation, but it also means that it needs to wait for the meter to collect the future power data before disaggregation can be performed, and these are low-frequency sampled data, which leads to significant delay in disaggregation, so these schemes are not suitable for real-time disaggregation scenarios where the users need to receive the disaggregation results with the shortest delay as possible. For example, in a dynamically priced grid, a user may turn on a high energy-consuming appliance at a time when electricity is expensive. If real-time notification is given, the user could choose to postpone the use of the appliance for the purpose of saving money and reducing the network load during peak hours [27]. Christos et al. proposed a novel multi-class real-time identification system using high-frequency data sampled at 100 Hz as input, and the system updates power data every 6 seconds and identifies devices. Moreover, by using KNN classifiers, the system can add new devices without retraining [28]. However, high-frequency sampling requires complex hardware, which can lead to additional costs during the monitoring process [29]. In contrast, Odysseas et al. still used low-frequency data, but used the total power over a period of time before moment t as input for predicting the power consumption of the appliance at moment t. Three network architectures that use sliding windows for real-time energy disaggregation were proposed, namely, Long-Short Term Memory (LSTM)

networks, Gated Recurrent Units (GRU) networks and Short Sequence-to-Point (Short Seq2point) networks, which were more effective on multi-state appliances than on two-state appliances [30]. Similarly, Virtsionis et al. [31] took only past data as input. They proposed a lightweight deep neural network based on attentional mechanism, which is called Self-Attentive-Energy-Disaggregation (SAED), using attention mechanism to focus on the most important features. Additive and point-attention mechanisms are compared, and the results show that the performance of these two attention mechanisms are comparable. The network is capable of fast training and inference.

All the above studies took the aggregate power series as the network input; however, some researchers found that the differential signal obtained from the original sequence through the differential process contains information about the state change of the electrical equipment, and using it as input could enhance the network disaggregation performance. The literature [32] proposed a composite deep LSTM based method to perform load disaggregation. It takes the aggregate power and differential power information as input, and then encodes, separates, and decodes them to achieve regression from one sequence to several sequences. Comparing to the single sequence to single sequence method, this method simplifies the procedure of disaggregation and enhances the disaggregation efficiency. In the literature [33], a NILM-based EMS and a convolutional neural network model that uses the differential signal as input are proposed. It is pointed out that the differential operation is performed implicitly in the neural network-based models that use raw data as input, but this is inaccurate and computationally expensive. Experimental results show that using differential sequences as input improves the disaggregation performance of the neural network, while the number of parameters of the network is greatly reduced.

In this paper, based on Short Sequence-to-Point network, we propose a real-time energy disaggregation algorithm based on multi-channels deep convolutional neural networks (MC-DCNN) [34] and autoregressive model (AR). First, the original total power sequence is differenced to obtain the differential signal, and then the differential signal and the original sequence are input to different channels of the network, so that the network can directly learn the on/off information of the equipment contained in the differential signal without simple and explicit differential operation; at the same time, the remaining useful information contained in the original sequence can be learnt. Then, feature extraction of the time series is performed using MC-DCNN to learn the amplitude and state change information of the appliances from the sequences of the two channels separately; furthermore, to compensate for the information loss caused by the max pooling layer and to adapt the network to different appliances, features of different time lengths extracted at different stages in the channels are concatenated as the input to the multilayer perceptron (MLP). Finally, a conventional autoregressive model is added for solving the scale insensitivity problem in the neural network model. The proposed method is validated on the UK-DALE and REDD datasets, and the results show that the proposed method has good performance. The main contributions of this paper are as following:

- MC-DCNN is adopted for solving this multivariate time series regression problem, where the aggregate power series and the differential series are fed into different channels, so that the amplitude and state change information of the electric appliance are learned from the sequences of the two channels respectively. The energy disaggregation performance is improved.
- The features of different time lengths extracted from two channels are fused to compensate for the information loss caused by the max pooling layer in the feature extraction process and enable the network to adapt to different appliances.
- The autoregressive linear model is used as the linear component for addressing the scale insensitivity problem in the neural network model.

II. PRELIMINARY

A. PROBLEM FORMULATION OF ENERGY DISAGGREGATION

Given that the total power consumption in time period T as $X = (x_1, x_2, \cdots, x_T)$, the power consumption of the i-th appliance as $Y^i = (y_1^i, y_2^i, \cdots, y_T^i)$. At each time step, the aggregate consumption can be expressed as the summary of the power consumption of all devices, as follows:

$$xt = \sum_{i=0}^{i=m} y_t^i + \varepsilon t \tag{1}$$

where ε_t denotes the Gaussian noise with zero mean and variance σ_t^2, and m denotes the sum of the number of appliances in the room. Assuming that we are only interested in household appliances that are widely used in most households, the power consumption from other appliances can be expressed as $S = (s_1, s_2, \ldots, s_T)$, and (1) can be rewritten as:

$$xt = \sum_{i=0}^{i=n} y_t^i + s_t + \varepsilon t \tag{2}$$

Energy disaggregation is obtaining the sequence of power consumption of the appliances $Y^1, Y^2, \ldots, Y^n$ through the aggregate power.

B. SHORT SEQUENCE TO POINT LEARNING

Seq2point takes a partial sequence of the total power sequence $Xt - w/2 : t + w/2$ as input to estimate the energy consumption of the target device at the intermediate time point y_t. Data after t moment are utilized, which is not suitable for online disaggregation scenarios. For this problem, Short Sequence to point takes the aggregate power sequence segments before the target moment $Xt - w/2 : t$ as input, and defines a neural network F, which maps the window sequence $Xt - \tau : t$ to the device power consumption at the target moment y_t:

$$y_t = F(Xt - \tau : t) + \varepsilon \tag{3}$$

where ε denotes Gaussian random noise.

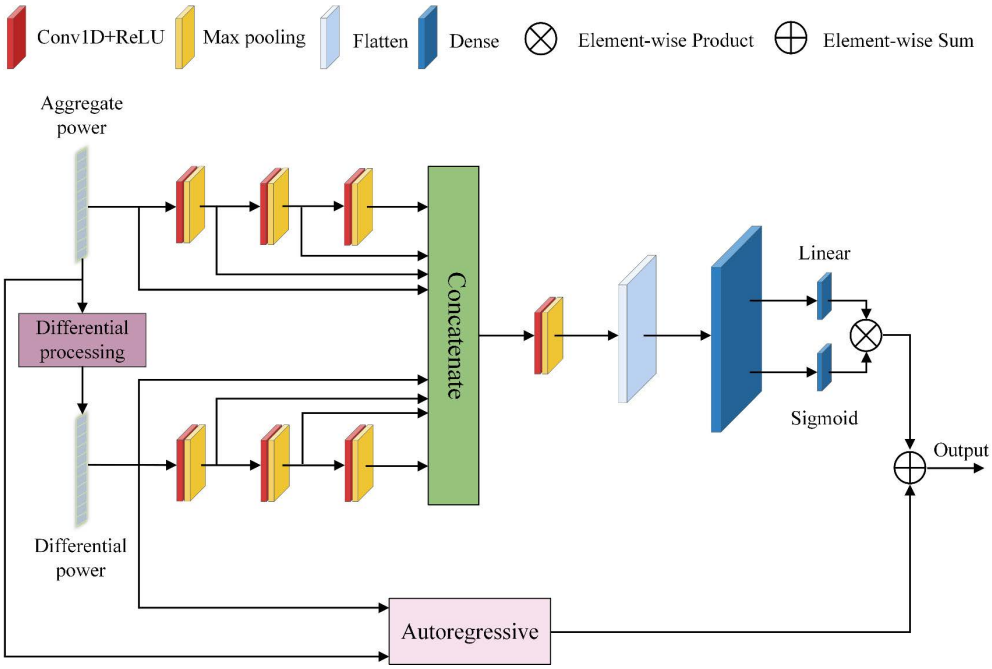


FIGURE 1. Overall structure of network.

In this paper, we take the aggregate power series and the differential series as input, i.e., the inputs are multivariate time series $M = \{m'_1, m'_2, \cdots, m'_T\}$, where $m'_1 \in \mathbb{R}^N$, N is the number of variables, and here $N = 2$. (3) can be reformulated as follows:

$$y_t = F(M_{t-\tau:t}) + \varepsilon \tag{4}$$

III. PROPOSED METHOD

In this section, we will give a complete explanation of the method proposed in this paper, and the overall network structure is shown in Fig. 1. It includes a nonlinear part and a linear part, where the nonlinear part is the MC-DCCN with feature fusion integrated, and the linear part is the autoregressive model.

A. MC-DCNN

MC-DCNN is designed to solve multivariate time series classification problems and has achieved excellent results among several multivariate time series datasets. Since the binary time series consisting of the aggregate power series and the differential power series are the network inputs, we treated the MC-DCNN as the backbone of the nonlinear part. The structure of the MC-DCNN network is shown in Fig. 2.

First, the aggregate power series and the differential power series are fed into two channels, one channel focuses on learning the amplitude information of the appliance contained in the series, and the other focuses on learning the state change information of the appliance among the series. In each channel, a feature extractor consisting of multiple stages learns hierarchical features from the univariate time series. Each stage consists of a one-dimensional convolutional layer with *RELU* as the activation function and a max pooling layer. Convolutional layers are used to obtain local time information of the sequence. The input of every convolutional layer is a time series $x_i^l \in \mathbb{R}^{len_i^l \times m_i^l}$, $1 \le i \le n$, where l denotes the layer from which the input comes, i denotes the channel to which it belongs, n represents the number of channels, i.e., the number of univariate time series, len_i^l and m_i^l denotes the length and dimensionality of the input series, respectively. The convolution layer contains k_i^l filters, the width of each kernel is equivalent to the dimensionality of the input m_i^l and the height is h_i^l. The j-th filter scans across the input matrix and generates:

$$x_{ij}^{l+1} = RELU(W_j^{l+1} * x_i^l + b_j^{l+1}) \tag{5}$$

where $*$ indicates the convolution operator, and x_{ij}^{l+1} denotes the output. $RELU\ (x) = max\ (0, x)$ is used as the activation function. After each convolution operation, as the size of the output matrix decreases, the output matrix will lose information at a large number at the edge positions, and subsequent convolution operations will be adversely affected. Thus, we decided to zero-fill the input matrix x_i^l, so that the output matrix has the same length as the input matrix. The size of the output matrix of the convolution layer x_i^{l+1} is $len_i^l \times k_i^l$.

A max pooling layer is connected after each convolutional layer, which subsamples the output matrix of the convolutional layer x_i^{l+1}:

$$s_i^l = MaxPooling(x_i^{l+1}) \tag{6}$$

where *MaxPooling* denotes the 1-D max pooling layer, s_i^l denotes the output matrix with size$slen_i^l \times k_i^l$, $slen_i^l = \lfloor len_i^l / stride \rfloor$ and *stride* denotes the stride length of the max pooling layer.

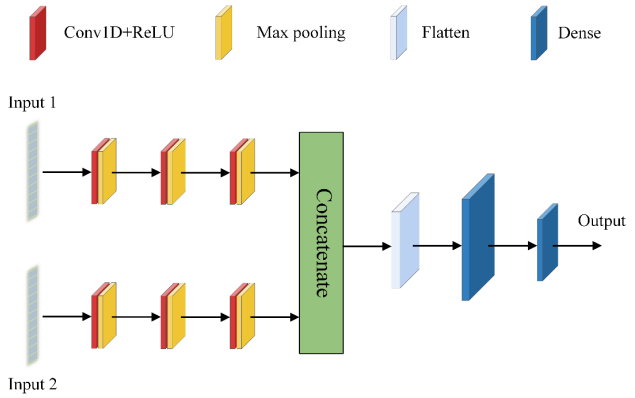


FIGURE 2. Architecture of MC-DCNN.

Then, the learned features of each channel are concatenated together. In particular, the method proposed in this paper uses a feature fusion module instead of the original simple feature concatenate layer to fuse information of different time lengths in the channel.

Lastly, the obtained features are input to the fully connected layer to obtain the output of nonlinear part y_t^N.

B. FEATURE FUSION

This module is integrated into the MC-DCNN to fuse the input series and the features extracted at different stages in the two channels.

Usually, time series data can be considered as a one-dimensional or flattened image, and convolutional neural networks(CNN) are utilized for extracting signals or characteristics from the time series. It is pointed out in the literature [35] that the data sophistication of time series is usually much lower compared to images, and the effective variables are much less, and the pooling layer shrinks the parameter dimensionality in the process of down-sampling the data, which may result in losing too much useful information. The experimental results indicate the model performance always decreases after introducing pooling layers, which proves that pooling layers have negative effects. However, eliminating the pooling layer would result in too many elements of the feature map to be processed, and more convolutional layers would need to be stacked to make the output features of the last convolutional layer contain the overall information of the input, which will make the model very large. In addition, the operating states and running times of different household appliances can vary, among which, kettle and microwave are short duration type appliances, fridge is medium duration type appliances, and washing machine and dishwasher are long duration type appliances. If only the features obtained in the last stage are taken as the input of the fully connected layer for regression, the network will not be able to learn information of different time length, which will result in not being able to take into account the operating characteristics of different household appliances.

To solve these problems, we decided to concatenate the output matrixes of each stage (convolutional and pooling layers) in the channel s_i^l and the original input sequences of the network without removing the pooling layer. The deeper the level, the larger the observation window, the information over a larger time length range can be extracted. The output matrix s_i^l of different stages contains information in the receptive fields of different sizes, representing patterns of different time lengths in the time series. The model is able to adapt to different appliances with different operating modes by learning features of different time lengths, at the same time, is able to learn from the output of the previous stages some of the useful information that is lost due to down-sampling.

The length of the output matrix varies from stage to stage. In order to concatenate the features obtained at different stages and the original input sequence, the output matrix is padded with zeros at the end to make its length $slen_i^l$ equal to the length of the input sequence len. The sequence padding and concatenation process is as follows:

$$pad_i^l = padding(s_i^l) \tag{7}$$

$$con = concatenate(m1, pad_1^1, pad_1^2, \ldots, pad_1^l, \ldots, mi, pad_i^1, pad_i^2, \ldots, pad_i^l) \tag{8}$$

where $pad_i^l \in \mathbb{R}^{len \times k_i^l}$, $con \in \mathbb{R}^{len \times (i+\sum_{a=1}^{a=i}\sum_{b=1}^{b=l} k_a^b)}$, len denotes the length of the input sequence.

These feature sequences have different effects on the disaggregation results, so they should have different weights, which need to be learned through training.

A CNN kernel is used to scan the concatenated features to catch the dependent patterns between different time series. The width of the kernel w is equal to the dimension of the fusion feature sequence con, and the height of the kernel is h. Specifically, the k-th convolution filter sweeps over the input matrix con and obtain:

$$Rk = RELU(Wk * con + bk) \tag{9}$$

where the output vector of the filter is Rk.We pad the input matrix with zeros, and the output matrix of the convolutional layer is $R \in \mathbb{R}^{len \times q}$, q denotes the number of filters.

The max pooling layer is connected after the convolutional layer, compressing the sequence and extracts the very long patterns:

$$u = Maxpooling(R) \tag{10}$$

where, $u \in \mathbb{R}^{\lfloor len/stride \rfloor \times q}$.

C. AUTOREGRESSIVE

Owing to the nonlinear properties of convolutional neural networks, the model suffers from the disadvantage that the scale of the output is insensitive to the scale of the input, resulting in a substantial reduction in the prediction accuracy of the model on datasets where the scale of the inputs is changing in an acyclic manner [36]. To address this issue, in the literature [36], [37], the researchers have incorporated a conventional autoregressive model to the nonlinear neural network and demonstrated that it can make the model more robust to time series that are in violation of scale changes.

The aggregate power series and the differential series do not have significant periodicity, and thus similar AR models are introduced in this paper as well. The autoregressive model is formulated as follows:

$$y_{t,i}^{L} = \sum_{k=1}^{k=window} W_{k,i}^{L} x_{t-k,i} + b_i^{L} \tag{11}$$

$$y_t^{L} = \sum_{i=1}^{i=n} y_{t,i}^{L} \tag{12}$$

where the output of the autoregressive model is $y_t^L \in \mathbb{R}$, and the autoregressive coefficient of the model is $W^L \in \mathbb{R}^{window}$, and the deviation is $b^L \in \mathbb{R}$, *window* denotes the size of the time window of the AR model (also referred to as the order of the model).

The final output values are derived through integrating the output of the neural network with that of the AR model:

$$\hat{Y}_t = y_t^N + y_t^L \tag{13}$$

where $\hat{Y}_t$ indicates the final output of the model at moment t.

IV. EXPERIMENTS

The hardware environment for this study is a 64-bit computer with 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz, 16G RAM and NVIDIA GeForce RTX 3050 Laptop GPU. The software platform is WINDOWS 10 Professional OS, Python 3.8.12 (64-bit) and TensorFlow-gpu 2.4.0 deep learning framework. In the proposed model structure, the convolution kernel size is 3 and the stride size is 1; the pooling size is 2. When training the model, the batch size is set to 128, the mean square error is adopted as the loss function, and the Adam optimizer is used with a learning rate of 0.001.

A. DATA SET

We evaluated our proposed method on two datasets, UK-DALE [38] and REDD [39]. The UK-DALE dataset collects electricity consumption data from five UK houses, while the REDD dataset collects data from six US houses.

The UK-DALE dataset was created by Kelly and Knottenbelt in 2015, which contains electricity consumption data for five UK houses. All data was recorded at 6-second intervals from November 2012 to January 2015 and contain total power consumption and measurements for 4-54 devices. Five appliances such as kettle, microwave, fridge, dishwasher and washing machine were selected for disaggregation in the experiment.

The REDD dataset was created by Kolter and Johnson in 2011. The data for different households spanned 23-48 days, with appliance and mains readings being recorded every 3 seconds and 1 second, respectively. Three appliances such as microwave, fridge, and dishwasher were selected for disaggregation in the experiment.

We divided the dataset using the same way as in the literature [19], [20], [30], and the houses that were used to train and test are listed in Table 1.

TABLE 1. Building used for training and testing.

Appliance	UK-DALE		REDD	
	Training	Testing	Training	Testing
Dish Washer	1,2	5	2,3,4,6	1
Fridge	1,2,4	5	2,3,5,6	1
Kettle	1,2,3,4	5	-	-
Microwave	1,2	5	2,3,5	1
Washing machine	1,5	2	-	-

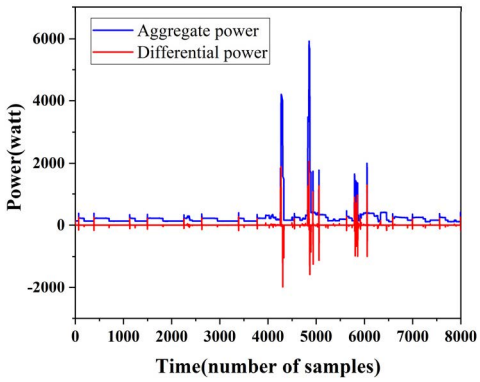


FIGURE 3. Aggregate and differential power.

B. DIFFERENTIAL PROCESSING

In the energy disaggregation, the aggregate power of the instant load is taken as the observed sequence. The aggregate power value of the current point in time is subtracted from the aggregate power value of the previous point in time to obtain the differential value. The aggregate power and differential power are shown in Fig. 3. Each non-zero value in the differential signal represents a state change of the appliances. The literature [33] points out that existing neural network models that use raw data as input perform the differencing process implicitly and automatically, however, there is an error between the calculated differential value and the actual value, thus, it is inaccurate for the neural network to perform the differential operation. Therefore, in this paper, we take the raw data and the differential signal of the raw data as the input, so that the power variation of the target device can be extracted more easily. The differential signal is calculated as follows:

$$\Delta Xt = X\mathrm{t} - Xt - 1 \tag{14}$$

where Xt indicates the aggregate power consumption at moment t, $Xt - 1$ indicates the aggregate power consumption at moment $t - 1$, and ΔXt denotes the result of differential operation.

C. SLIDING WINDOW PROCESSING

The aggregate power and differential signals are processed using sliding windows, taking the sequence segments in the range $[t - w, t]$as input and the appliance energy consumption at moment t as output. The window sizes of every appliance and network are shown in Table 2.

TABLE 2. Sliding window sizes for each appliance and network.

Appliance	MC-DCNN	Proposed
Dish Washer	100	100
Fridge	50	100
Kettle	100	100
Microwave	50	100
Washing machine	200	200

D. DATA NORMALIZATION

Normalizing the data can eliminate the effect of the magnitude between indicators. In this experiment, the original data are processed using min-max normalization to constrain the size of the data to [0, 1]:

$$x* = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{15}$$

where x^* denotes the normalized value, $x_{\max}$ denotes the max value, $x_{\min}$ is the min value.

Appliance activations are extracted using NILMTK, the obtained related arguments of appliance are presented in the Table 3.

E. METRICS

For comparing the performance of these methods, appropriate evaluation metrics should be chosen for evaluation. We adopted the mean absolute error (MAE), the relative error in total energy, Energy F-score and total energy correctly assigned (TECA) as the energy disaggregation evaluation index:

$$\textit{mean absolute error} = \frac{1}{T}\sum_{t=1}^{t=T}\left|\hat{y}_i(t) - y_i(t)\right| \tag{16}$$

$$\textit{relative error in total energy} = \frac{\left|E' - E\right|}{\max(E', E)} \tag{17}$$

$$\textit{Energy F-score} = 2\frac{P^{(E)}R^{(E)}}{P^{(E)} + R^{(E)}} \tag{18}$$

$$TECA = 1 - \frac{\sum_{t=1}^{t=T}\sum_{i=1}^{i=N}\left|\hat{y}_i(t) - y_i(t)\right|}{2\sum_{t=1}^{t=T}\sum_{i=1}^{i=N} y_i(t)} \tag{19}$$

$$P^{(E)} = \frac{1}{N}\sum_{i=1}^{i=N} P_i^{(E)} \tag{20}$$

$$R^{(E)} = \frac{1}{N}\sum_{i=1}^{i=N} R_i^{(E)} \tag{21}$$

$$P_i^{(E)} = \frac{\sum_{t=1}^{t=T}\min(\hat{y}_i(t), y_i(t))}{\sum_{t=1}^{t=T}\hat{y}_i(t)} \tag{22}$$

$$R_i^{(E)} = \frac{\sum_{t=1}^{t=T}\min(\hat{y}_i(t), y_i(t))}{\sum_{t=1}^{t=T} y_i(t)} \tag{23}$$

$$\hat{y}_i(t) = \text{estimated value of appliance } i \text{ at time point } t \tag{24}$$

$$y_i(t) = \text{true value of appliance } i \text{ at time point } t \tag{25}$$

$$T = \text{total amount of predicted time points} \tag{26}$$

$$E = \text{total energy consumed} \tag{27}$$

$$E' = \text{total predicted energy consumed} \tag{28}$$

$$N = \text{total umber of appliances} \tag{29}$$

The ability to accurately identify the on/off state of an appliance is another aspect of measuring the performance of an energy disaggregation network. When the power disaggregation is completed, we can discriminate the status of on/off through the threshold value of the device. Four event detection evaluation indexes were chosen, namely: recall, precision, F1-score, accuracy:

$$recall = \frac{TP}{TP + FN} \tag{30}$$

$$precision = \frac{TP}{TP + FP} \tag{31}$$

$$F1\text{-}score = 2 \times \frac{precision \cdot recall}{precision + recall} \tag{32}$$

$$accuracy = \frac{TP + TN}{T} \tag{33}$$

$$TP = \text{number of true positives} \tag{34}$$

$$TN = \text{number of true negatives} \tag{35}$$

$$FP = \text{number of false positives} \tag{36}$$

$$FN = \text{number of false negatives} \tag{37}$$

$$T = \text{the total number} \tag{38}$$

Among them, mean absolute error (MAE), relative error of total energy, recall, precision, F1-score and accuracy are general metrics used for classification/regression problems, while energy F-Score and TECA are metrics proposed specifically for energy disaggregation.

V. RESULTS

The models without using future data (LSTM [30], GRU [30], Short Seq2point [30], SAED-dot [31], SAED-add [31], MC-DCNN [34]) and the model with using future data (BERT [26]) are adopted for comparison. We report the results of the evaluation metrics for the UK-DALE and REDD datasets in Table 4, Table 5, Table 6 and Table 7. On the UK-DALE dataset, washing machine was not tested in the same house as other appliances, so TECA value cannot be calculated and we do not report it in Table 5. The best result is highlighted in bold in the tables. We tried to replicate the experiments in [26], [30], and [31], but could not achieve the results they reported, and to respect their work, we directly used the results in the reference, and we emptied metrics that are not reported in these references.

In UK-DALE, for event detection performance, BERT outperforms other models on dishwasher, fridge and kettle, SAED-add performs better on microwave compared to other models, and the proposed model in this paper has the best performance on washing machine. For energy disaggregation performance, BERT has the smallest MAE value on

TABLE 3. Arguments of appliance.

Appliance	Max power (watt)	On power threshold (watt)	Mean power (watt)	Standard deviation of Power(watt)	Min. on Duration (secs)	Min. off Duration (secs)
Dish washer	3000/3964	10	700	1000	1800	1800
Fridge	200/3323	50	200	400	60	12
kettle	3000	2000	700	1000	12	0
Microwave	3000/3969	200	500	800	12	30
Washing machine	2500	20	400	700	1800	160

a/*b*: *a* and *b* represent the device parameters in UK-DALE and REDD respectively.

TABLE 4. Results on the general evaluation metrics for the UK-dale dataset.

Appliance	Model	Recall	Precision	Accuracy	F1 Score	Relative error in total energy	MAE
Dish	LSTM	0.26	0.68	0.97	0.38	0.58	25
washer	GRU	0.42	0.62	0.97	0.5	0.07	24
	Short Seq2point	**0.43**	0.47	0.96	0.45	0.07	21
	SAED-dot	-	-	-	0.25	0.46	43.3
	SAED-add	-	-	-	0.52	0.37	44.48
	BERT	-	-	0.97	**0.67**	-	16.18
	MC-DCNN	0.38	0.68	0.97	0.49	0.13	20
	Proposed	0.39	**0.78**	**0.98**	0.52	**0.003**	**16**
Fridge	LSTM	0.51	0.45	0.6	0.47	0.21	51
	GRU	0.75	0.46	0.6	0.57	0.26	51
	Short Seq2point	0.74	0.42	0.54	0.53	0.29	51
	SAED-dot	-	-	-	0.49	0.29	50.89
	SAED-add	-	-	-	0.5	0.33	51.39
	BERT	-	-	**0.81**	**0.77**	-	**25.49**
	MC-DCNN	**0.78**	0.44	0.56	0.56	0.33	48
	Proposed	0.66	**0.64**	0.74	0.65	**0.06**	39
Kettle	LSTM	0.46	0.41	0.99	0.44	0.59	25
	GRU	0.74	0.76	0.99	0.75	0.15	9
	Short Seq2point	0.81	**0.95**	0.99	0.88	**0.07**	**4**
	SAED-dot	-	-	-	0.273	0.27	12.2
	SAED-add	-	-	-	0.305	0.18	10.9
	BERT	-	-	0.99	**0.90**	-	6.82
	MC-DCNN	0.83	0.76	0.99	0.79	0.11	6.9
	Proposed	**0.85**	0.78	0.99	0.82	0.1	7.5
Microwave	LSTM	0.45	0.01	0.93	0.02	0.1	86
	GRU	0.75	0.02	0.93	0.04	**0.07**	97
	Short Seq2point	0.79	0.01	0.91	0.03	0.16	103
	SAED-dot	-	-	-	0.21	0.58	56.93
	SAED-add	-	-	-	**0.22**	0.51	56.36
	BERT	-	-	**0.99**	0.014	-	**6.57**
	MC-DCNN	0.74	0.04	0.96	0.07	0.3	79
	Proposed	**0.85**	**0.06**	0.98	0.11	0.32	80
Washing	LSTM	0.56	0.16	0.95	0.24	0.35	25
machine	GRU	0.54	0.22	0.96	0.31	0.58	30
	Short Seq2point	0.55	0.26	0.97	0.35	0.28	17
	SAED-dot	-	-	-	0.56	0.36	51.39
	SAED-add	-	-	-	0.38	**0.24**	40.06
	BERT	-	-	0.97	0.32	-	**6.98**
	MC-DCNN	0.86	0.54	**0.99**	0.68	0.45	12
	Proposed	**0.87**	**0.8**	**0.99**	**0.83**	0.43	9

TABLE 5. Results on the specialized evaluation metrics for the UK-dale dataset.

Metric	LSTM	GRU	Short Seq2point	SAED-dot	SAED-add	BERT	MC-DCNN	Proposed
Energy F-score	-	-	-	-	-	-	0.48	**0.51**

microwave, washing machine and fridge, Short Seq2point has the smallest prediction error on kettle, and the proposed method in this paper has the smallest MAE value and relative error in total energy on dishwasher, and it has higher Energy F-score compared to MC-DCNN.

In REDD, for the event detection performance, the model proposed in this paper has the best performance on all three appliances. For energy disaggregation performance, BERT has the smallest MAE value on dishwasher and fridge, the proposed method in this paper has the smallest MAE value

TABLE 6. Results on the general evaluation metrics for the redd dataset.

Appliance	Model	Recall	Precision	Accuracy	F1 Score	Relative error in total energy	MAE
Dish washer	LSTM	0.47	0.55	0.96	0.51	0.37	22
	GRU	0.45	0.60	0.97	0.51	0.42	22
	Short Seq2point	0.57	0.63	0.97	0.60	0.50	19
	SAED-dot	-	-	-	0.41	0.19	27
	SAED-add	-	-	-	0.18	**0.13**	36.16
	BERT	-	-	0.97	0.52	-	20.5
	MC-DCNN	0.56	0.84	**0.98**	0.67	0.57	17
	Proposed	**0.58**	**0.89**	**0.98**	**0.70**	0.51	**15**
Fridge	LSTM	**0.99**	0.55	0.79	0.71	0.15	43
	GRU	0.84	0.51	0.76	0.64	**0.11**	42
	Short Seq2point	0.83	0.61	0.82	0.70	0.21	39
	SAED-dot	-	-	-	0.52	0.29	51.35
	SAED-add	-	-	-	0.52	0.22	50.52
	BERT	-	-	0.84	0.76	-	**32.3**
	MC-DCNN	0.87	0.57	0.79	0.69	0.18	42
	Proposed	0.85	**0.71**	**0.88**	**0.77**	0.26	35
Microwave	LSTM	0.42	0.63	**0.99**	0.51	0.32	22
	GRU	0.51	0.60	**0.99**	0.55	0.22	18
	Short Seq2point	0.44	0.36	0.98	0.39	0.16	23
	SAED-dot	-	-	-	0.34	0.2	25.67
	SAED-add	-	-	-	0.34	**0.15**	25.13
	BERT	-	-	**0.99**	0.48	-	**17.6**
	MC-DCNN	**0.58**	0.56	**0.99**	0.57	0.23	21
	Proposed	0.57	**0.65**	**0.99**	**0.61**	0.37	20

TABLE 7. Results on the specialized evaluation metrics for the redd dataset.

Metric	LSTM	GRU	Short Seq2point	SAED-dot	SAED-add	BERT	MC-DCNN	Proposed
Energy F-score	0.48	0.52	0.52	-	-	-	0.55	**0.56**
TECA	0.57	0.59	0.60	-	-	-	0.61	**0.65**

TABLE 8. Number of parameters for each model.

Appliance	LSTM	GRU	Short Seq2point	SAED-dot	SAED-add	BERT	MC-DCNN	Proposed
Dish Washer			5.14M				26K	208K
Fridge			2.58M				13K	208K
Kettle	1.26M	263K	5.14M	39.8K	39.9K	1.94M	26K	208K
Microwave			2.58M				13K	208K
Washing machine			10.26M				52K	413K

TABLE 9. Inference time for testing set and for each sample.

Model	Inference time for testing set(s)			Inference time for each sample(ms)		
	Dish washer	Fridge	Microwave	Dish washer	Fridge	Microwave
LSTM	138.73	134.86	132.30	0.704	0.684	0.671
GRU	136.02	142.92	136.73	0.690	0.725	0.694
Short Seq2point	2.97	2.46	2.33	0.015	0.014	0.012
SAED-dot	4.71	4.65	4.67	0.024	0.023	0.023
SAED-add	6.29	6.22	6.29	0.032	0.031	0.032
BERT	61.19	61.47	60.88	37.36	37.53	37.17
MC-DCNN	3.73	3.31	3.34	0.018	0.016	0.017
Proposed	4.65	4.89	4.84	0.023	0.025	0.024

on dishwasher and the largest Energy F-score and TECA, and SAED-add has the smallest relative error in total energy on microwave and dishwasher.

Generally, the proposed method performs slightly worse than BERT on UK-DALE, but better than other models; it performs well on REDD dataset, especially for event detection performance. As mentioned earlier, the reason for the better performance of BERT is the use of future data, the equipment operation state in the future period helps model perform a more accurate disaggregation. However, the proposed method has much less number of parameters than BERT and is therefore more suitable for deployment in smart meters.

We measured the inference time for each model on the REDD dataset by using time.time() function, and the inference times for the test set and for each sample are shown in the Table 9. The window size of the model input affects the total sample size of the test set, which affects the inference time of each model on the test set. Therefore, we mainly compare the

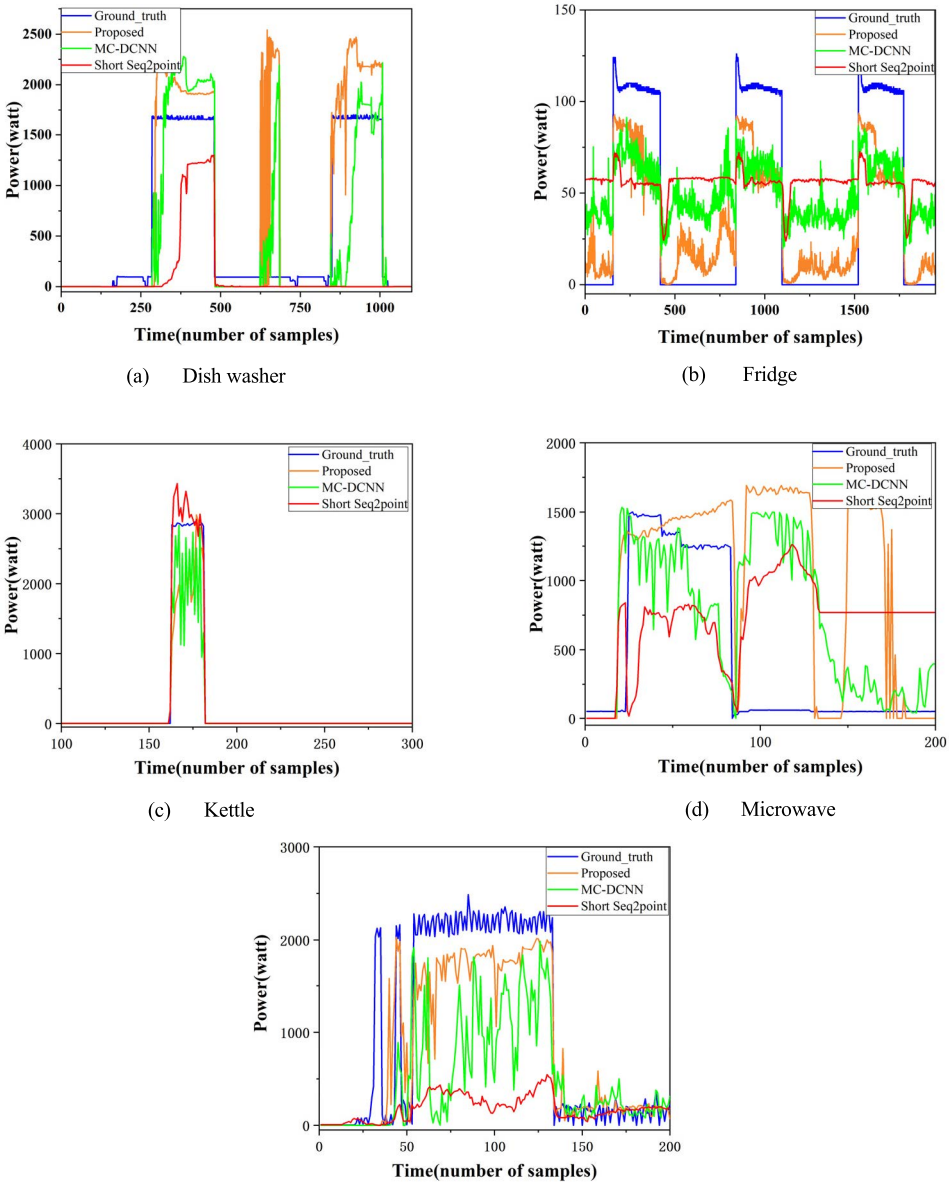


FIGURE 4. Comparison results of power disaggregation.

inference time for each sample. Compared with LSTM, GRU and BERT, our method is much faster and the inference time differs very slightly from the remaining methods. Moreover, BERT needs to wait until the future data is collected before disaggregating, and the delay is significant. On the contrary, other methods can disaggregate a sample in much less than the sampling period as soon as the data at the target moment is collected, which fully meets the requirement of real-time disaggregation to deliver disaggregation results with short delay.

We further compare the three convolutional neural network-based models, Short Seq2point, MC-DCNN and the proposed method. The power disaggregation comparison results of these five target appliances for these three methods on UK-DALE is illustrated in Fig. 4. As can be seen in Fig. 4, for the dishwasher, the proposed method and MC-DCNN more accurately identify the entire operating cycle, where the proposed method more accurately estimates the time of the appliance state change; however, Short Seq2point only identifies the previous activation of the appliance. For the fridge, the estimated power of MC-DCNN and Short Seq2point fluctuate above and below the threshold value (50 watt), and the estimated power of the proposed method are much closer to the actual power values and fit much better. For the kettle, all three methods are able to identify the device activation relatively accurately, and the Short Seq2point fits slightly better. For the microwave, the predicted power value of the device operating by the method proposed in this paper

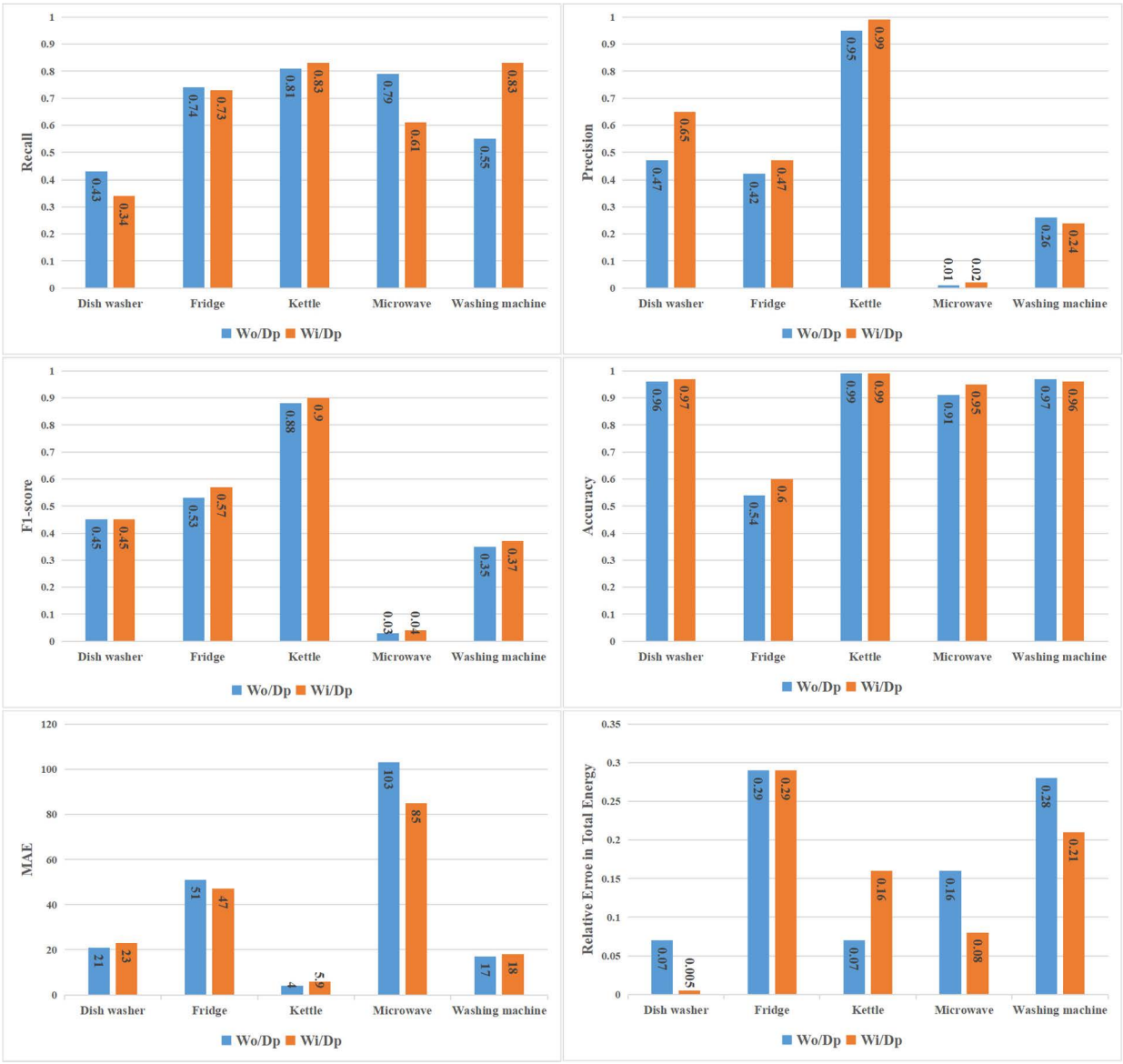


FIGURE 5. Comparison results of different inputs.

is closer to the actual value than the other two methods; however, after the appliance stops working, all the methods mistakenly assume that the appliance is working again and classify the negative samples as positive, which leads to a large FP value and correspondingly Precision is very low. For the washing machine, Short Seq2point is able to predict that the device is working, but the predicted value of power is only slightly above the threshold (20 watt), which is much lower than the actual value. The predicted values of the other two methods are closer to the actual values, but the power values of MC-DCNN fluctuate drastically. It can be found that the comprehensive performance of the two methods (MC-DCNN and the proposed method) that incorporate differential power as input is superior, and we believe that the on/off state information contained in the differential signal plays a role, as we will further demonstrate in section VI. In addition, the number of parameters in Short Seq2point is tens and hundreds of times higher than the first two methods, respectively, indicating that doubling the number of filters in the single-channel model does not achieve the same results as in the multi-channels model, and is also unnecessary.

In summary, the proposed approach achieves good performance on both datasets, which illustrates the generalization capability of the model structure. In terms of model capacity, it is a lightweight model that can be easily deployed on smart meters. In terms of speed, the inference speed of the model can fully meet the demand of real-time disaggregation.

VI. ABLATION STUDY

To validate the effectiveness of our proposed method, we performed a careful ablation study on UK-DALE dataset. First, to verify that adding differential power as input enables

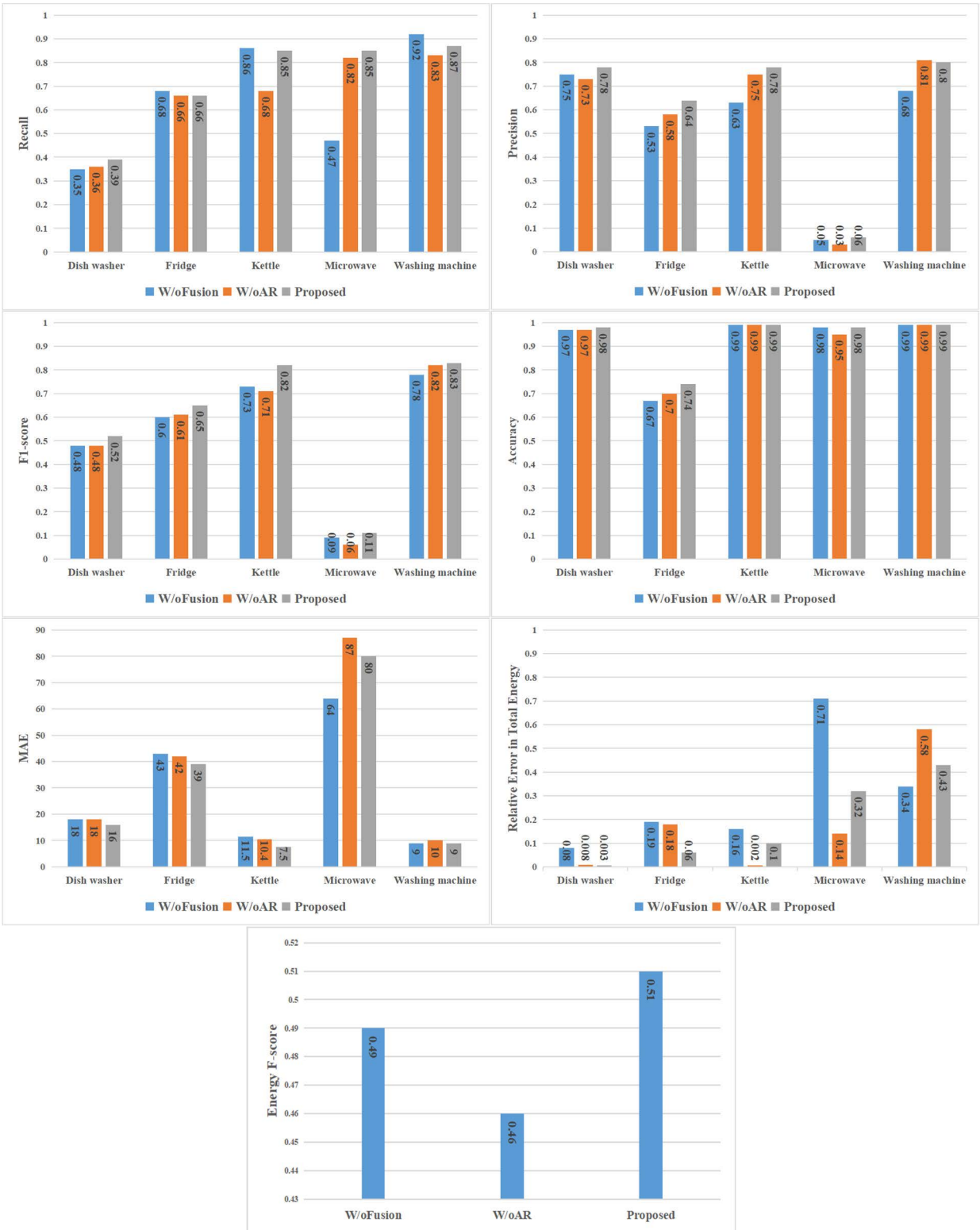


FIGURE 6. Comparison results of removing different part.

the network to directly learn the information about state changes of devices to improve the disaggregation performance, we took the Short Seq2point model as an example and compared the performance in both cases with(Wi/Dp) and without(Wo/Dp) adding differential signals, and the comparison results are shown in Fig. 5. We do not compare the Energy F-score, because it was not reported in [30].

As seen in the figure, the performance in most of the appliances is slightly improved after the addition of the differential signal, where the ability of identifying switching states is improved on all appliances. It indicates that the information of appliance on/off status contained in the differential sequence does help to improve the disaggregation performance of the network, and the difference operation of the neural network is inaccurate, which is consistent with the view of Yuanmeng Zhang et al [33].

Moreover, comparing the performance of the Short Seq2point model with the addition of differential signals, MC-DCNN and the proposed method on UK-DALE, it can be found that the latter two multi-channels structure models possess superior performance, which indicates that it is necessary and effective to process the two signals separately.

Then, we removed one part at a time in the model structure of the proposed method. We name the models with different parts removed with the following names:

- Wo/Fusion: The model without feature fusion which fuses the information of different lengths of time.
- Wo/AR: The model without Autoregressive mod-el(AR).

The comparison results are shown in Fig. 6. From these results it is clear that:

- The disaggregation performance of the method proposed in this paper is better than the other two models on most appliances, where the ability to identify on/off states is optimal on all appliances.
- Removing the AR component from the completed model(wo/AR) results in a degradation of disaggregation performance on most appliances. It is shown that the AR component plays a key role in the over-all.
- Not fusing information of different time lengths leads to worse results of the model(Wo/Fusion) on most appliances, which demonstrates the importance of learning patterns of different time lengths and shows that the feature fusion part does allow the model to learn some of the information lost due to the pooling layer.

Overall, this ablation study clearly demonstrates the necessity of adding differential information as the input of the network, as well as the effectiveness of our model design, with all components contributing to enhance the performance of the model.

VII. CONCLUSION

In this paper, a novel lightweight model is proposed, which takes aggregate power sequences and differential power sequences as the inputs to the network, and combines multi-channels deep convolutional neural networks with feature fusion integrated and autoregressive model as the nonlinear and linear components, respectively. By conducting experiments on the UK-DALE and REDD datasets, the results show that the method proposed in this paper has good performance on both datasets, and is able to deliver results in real time. In addition to this, we demonstrated the efficiency of the proposed model architecture through an in-depth analysis.

In the next step, we are ready to implement our system in a real scenario. As mentioned earlier, this is a lightweight model, so we plan to embed the model into smart meters. In addition, considering that when adding new appliances, a large amount of data needs to be collected and a new model needs to be trained, we plan to store the collected data and retrain the model with the help of cloud servers.

REFERENCES

[1] G. W. Hart, *Prototype Nonintrusive Appliance Load Monitor: Progress Report 2*. Cambridge, MA, USA: MIT Energy Laboratory, 1985.

[2] G. W. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec. 1992.

[3] A. Ridi, C. Gisler, and J. Hennebert, "A survey on intrusive load monitoring for appliance recognition," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 3702–3707.

[4] H.-H. Chang, K.-L. Chen, Y.-P. Tsai, and W.-J. Lee, "A new measurement method for power signatures of nonintrusive demand monitoring and load identification," *IEEE Trans. Ind. Appl.*, vol. 48, no. 2, pp. 764–771, Mar. 2012.

[5] J. Liao, G. Elafoudi, L. Stankovic, and V. Stankovic, "Power disaggregation for low-sampling rate data," in *Proc. 2nd Int. Non-Intrusive Appliance Load Monitor. Workshop*, Austin, TX, USA, vol. 1, 2014, p. F1.

[6] T. Hassan, F. Javed, and N. Arshad, "An empirical investigation of V-I trajectory based load signatures for non-intrusive load monitoring," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 870–878, Mar. 2013.

[7] M. Wenninger, D. Stecher, and J. Schmidt, "SVM-based segmentation of home appliance energy measurements," in *Proc. 18th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2019, pp. 1666–1670.

[8] T. Saitoh, T. Osaki, R. Konishi, and K. Sugahara, "Current sensor based home appliance and state of appliance recognition," *SICE J. Control, Meas., Syst. Integr.*, vol. 3, no. 2, pp. 86–93, Mar. 2010.

[9] Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, "Smart non-intrusive appliance identification using a novel local power histogramming descriptor with an improved k-nearest neighbors classifier," *Sustain. Cities Soc.*, vol. 67, Apr. 2021, Art. no. 102764.

[10] M. M. Eskander and C. A. Silva, "A complementary unsupervised load disaggregation method for residential loads at very low sampling rate data," *Sustain. Energy Technol. Assessments*, vol. 43, Feb. 2021, Art. no. 100921.

[11] T. Y. Ji, L. Liu, T. S. Wang, W. B. Lin, M. S. Li, and Q. H. Wu, "Non-intrusive load monitoring using additive factorial approximate maximum *a posteriori* based on iterative fuzzy *c*-means," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6667–6677, Nov. 2019.

[12] M. Figueiredo, B. Ribeiro, and A. de Almeida, "On the optimization of appliance loads inferred by probabilistic models," in *Proc. 2nd Int. Workshop Non-Intrusive Load Monit.*, 2014.

[13] H. Xu, C. Li, M. M. Rahaman, Y. Yao, Z. Li, J. Zhang, F. Kulwa, X. Zhao, S. Qi, and Y. Teng, "An enhanced framework of generative adversarial networks (EF-GANs) for environmental microorganism image augmentation with limited rotation-invariant training data," *IEEE Access*, vol. 8, pp. 187455–187469, 2020.

[14] W. Zhao, W. Ma, L. Jiao, P. Chen, S. Yang, and B. Hou, "Multi-scale image block-level F-CNN for remote sensing images object detection," *IEEE Access*, vol. 7, pp. 43607–43621, 2019.

[15] S. Yang, Z. Gong, K. Ye, Y. Wei, Z. Huang, and Z. Huang, "EdgeRNN: A compact speech recognition network with spatio-temporal features for edge computing," *IEEE Access*, vol. 8, pp. 81468–81478, 2020.

[16] T.-W. Sun, "End-to-end speech emotion recognition with gender information," *IEEE Access*, vol. 8, pp. 152423–152438, 2020.

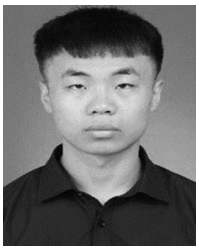
[17] Z. Hu, J. Luo, C. Zhang, and W. Li, "A natural language process-based framework for automatic association word extraction," *IEEE Access*, vol. 8, pp. 1986–1997, 2019.

[18] D. Wang, J. Su, and H. Yu, "Feature extraction and analysis of natural language processing for deep learning English language," *IEEE Access*, vol. 8, pp. 46335–46345, 2020.

[19] J. Kelly and W. Knottenbelt, "Neural NILM: Deep neural networks applied to energy disaggregation," in *Proc. 2nd ACM Int. Conf. Embedded Syst. Energy-Efficient Built Environ.*, Nov. 2015, pp. 55–64.

[20] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018, pp. 2604–2611.

[21] W. Yang, C. Pang, J. Huang, and X. Zeng, "Sequence-to-point learning based on temporal convolutional networks for nonintrusive load monitoring," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–10, 2021.
[22] G. Zhou, Z. Li, M. Fu, Y. Feng, X. Wang, and C. Huang, "Sequence-to-sequence load disaggregation using multiscale residual neural network," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–10, 2020.
[23] A. Langevin, M.-A. Carbonneau, M. Cheriet, and G. Gagnon, "Energy disaggregation using variational autoencoders," *Energy Buildings*, vol. 254, Jan. 2022, Art. no. 111623.
[24] G. Cui, B. Liu, W. Luan, and Y. Yu, "Estimation of target appliance electricity consumption using background filtering," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 5920–5929, Nov. 2019.
[25] K. Chen, Y. Zhang, Q. Wang, J. Hu, H. Fan, and J. He, "Scale- and context-aware convolutional non-intrusive load monitoring," *IEEE Trans. Power Syst.*, vol. 35, no. 3, pp. 2362–2373, May 2019.
[26] Z. Yue, C. R. Witzig, D. Jorde, and H.-A. Jacobsen, "BERT4NILM: A bidirectional transformer model for non-intrusive load monitoring," in *Proc. 5th Int. Workshop Non-Intrusive Load Monitor.*, Nov. 2020, pp. 89–93.
[27] A. Harell, S. Makonin, and I. V. Bajic, "Wavenilm: A causal neural network for power disaggregation from the complex power signal," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 8335–8339.
[28] C. L. Athanasiadis, T. A. Papadopoulos, and D. I. Doukas, "Real-time non-intrusive load monitoring: A light-weight and scalable approach," *Energy Buildings*, vol. 253, Dec. 2021, Art. no. 111523.
[29] A. Moradzadeh, B. Mohammadi-Ivatloo, M. Abapour, A. Anvari-Moghaddam, S. G. Farkoush, and S.-B. Rhee, "A practical solution based on convolutional neural network for non-intrusive load monitoring," *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 10, pp. 9775–9789, Oct. 2021.
[30] O. Krystalakos, C. Nalmpantis, and D. Vrakas, "Sliding window approach for online energy disaggregation using artificial neural networks," in *Proc. 10th Hellenic Conf. Artif. Intell.*, Jul. 2018, pp. 1–6.
[31] N. V. Gkalinikis, C. Nalmpantis, and D. Vrakas, "Attention in recurrent neural networks for energy disaggregation," in *Proc. Int. Conf. Discovery Sci.* Cham, Switzerland: Springer, 2020, pp. 551–565.
[32] M. Xia, W. Liu, K. Wang, W. Song, C. Chen, and Y. Li, "Non-intrusive load disaggregation based on composite deep long short-term memory network," *Expert Syst. Appl.*, vol. 160, Dec. 2020, Art. no. 113669.
[33] Y. Zhang, G. Yang, and S. Ma, "Non-intrusive load monitoring based on convolutional neural network with differential input," *Proc. CIRP*, vol. 83, pp. 670–674, Jan. 2019.
[34] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, "Exploiting multi-channels deep convolutional neural networks for multivariate time series classification," *Frontiers Comput. Sci.*, vol. 10, no. 1, pp. 96–112, Feb. 2016.
[35] S. Liu, H. Ji, and M. C. Wang, "Nonpooling convolutional neural network forecasting for seasonal time series with trends," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 8, pp. 2879–2888, Aug. 2020.
[36] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long- and short-term temporal patterns with deep neural networks," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 95–104.
[37] W. Song and S. Fujimura, "Capturing combination patterns of long- and short-term dependencies in multivariate time series forecasting," *Neurocomputing*, vol. 464, pp. 72–82, Nov. 2021.
[38] J. Kelly and W. Knottenbelt, "The U.K.-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five U.K. Homes," *Sci. Data*, vol. 2, no. 1, pp. 1–14, Dec. 2015.
[39] J. Z. Kolter and M. J. Johnson, "REDD: A public data set for energy disaggregation research," in *Proc. Workshop Data Mining Appl. Sustainability (SIGKDD)*, San Diego, CA, USA, 2011, vol. 25, pp. 59–62.



LINTAO DENG was born in 1998. He received the B.E. degree in electronic information engineering from Nanchang Hangkong University, Nanchang, China, in 2020. He is currently pursuing the M.E. degree in information and communication engineering with the Shanghai University of Electric Power (SUEP). His research interest includes non-intrusive load monitoring (NILM).

CHENGXIN PANG (Member, IEEE) received the Ph.D. degree. He is currently a Shanghai High-Level Distinguished Expert and a Professor with the Shanghai University of Electric Power (SUEP). His research topics have been in magneto-optics isolator with the Institut for Electronic Fundamentals (IEF), Paris-Sud University, Orsay, France; in hybrid silicon photonics with Orange Labs, France; and in LCF with the Institut d'Optique Graduate School (IOGS), Palaiseau, France. His research interests include power the Internet of Things, NILM, intelligent information perception, and machine vision.

XINHUA ZENG received the Ph.D. degree. He is currently the Deputy Director with the Network and System Center, Institute of Technology, Fudan University, and the Director of the Hefei Institute of Technological Innovation, Chinese Academy of Sciences. His current research interests include biological information collection, intelligent information perception and processing, visual image, and deep learning.

JUN ZHANG is currently a Senior Engineer with Nari Technology Nanjing Control Systems Company Ltd., Nanjing, China. His research interests include grid dispatch and power the Internet of Things sensing technology.

CHIZHI HUANG is currently an Engineer with Nari Technology Nanjing Control Systems Company Ltd., Nanjing, China. His research interests include power equipment online monitoring and power the Internet of Things sensing technology.