

Article

Non-Intrusive Load Disaggregation Based on a Multi-Scale Attention Residual Network

Liguo Weng¹, Xiaodong Zhang¹, Junhao Qian¹, Min Xia^{1,*} , Yiqing Xu² and Ke Wang³

¹ Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology, Nanjing University of Information Science and Technology, Nanjing 210044, China; 002311@nuist.edu.cn (L.W.); 20181223103@nuist.edu.cn (X.Z.); 20181223052@nuist.edu.cn (J.Q.)

² College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China; yiqingxu@njfu.edu.cn

³ China Electric Power Research Institute, Nanjing 210003, China; wangke@epri.sgcc.com.cn

* Correspondence: xiamin@nuist.edu.cn

Received: 30 November 2020; Accepted: 17 December 2020; Published: 21 December 2020



Abstract: Non-intrusive load disaggregation (NILD) is of great significance to the development of smart grids. Current energy disaggregation methods extract features from sequences, and this process easily leads to a loss of load features and difficulties in detecting, resulting in a low recognition rate of low-use electrical appliances. To solve this problem, a non-intrusive sequential energy disaggregation method based on a multi-scale attention residual network is proposed. Multi-scale convolutions are used to learn features, and the attention mechanism is used to enhance the learning ability of load features. The residual learning further improves the performance of the algorithm, avoids network degradation, and improves the precision of load decomposition. The experimental results on two benchmark datasets show that the proposed algorithm has more advantages than the existing algorithms in terms of load disaggregation accuracy and judgments of the on/off state, and the attention mechanism can further improve the disaggregation accuracy of low-frequency electrical appliances.

Keywords: load disaggregation; multi-scale; attention mechanism; residual network

1. Introduction

Load disaggregation technology is a key technology in smart grids [1]. Traditional load monitoring adopts intrusive methods, which are able to obtain accurate and reliable data with low data noise [2], but they are difficult to be accepted by users due to their high implementation costs. Non-intrusive methods can provide detailed information for residents in time, and have the advantages of low cost and easy implementation. According to this technology, the power consumption behaviors of users can be analyzed, and users can be guided toward a reasonable consumption of electricity and hence reduce their power consumption costs. With the continuous development of power demand side management [3], big data analysis, and other technologies, non-intrusive load disaggregation is attracting more attention.

The microgrid is an important manifestations of the smart grid. With the development of clean energy, such as solar and wind energy, and energy internet technology, the microgrid has emerged. It is a small power system with distributed power sources, which can realize a highly reliable supply of multiple energy sources and improve the quality of the power supply [4]. As NILM technology becomes more mature, the intelligent dispatching of the microgrid can be realized through automation in the future to improve the effective utilization of power resources, ensure the stable economic operation of a power system, and avoid the unnecessary waste of power resources. Therefore, NILM technology is important.

The concept of non-intrusive load monitoring was firstly proposed by Hart [5]. It mainly uses non-intrusive load disaggregation (NILD). In this method, the total power consumption is disaggregated to each individual electrical appliance. Hart proposed the concept of “load characteristics”, which he defined as the information change of the electrical power of an appliance in operation. Hart further used the steady-state load characteristics [6] to design a simple NILD system to decompose power. However, effective features extracted by the algorithm were limited, and large disaggregation errors occurred easily.

At present, combinatorial optimization (CO) methods and pattern recognition algorithms are the main algorithms for realizing non-intrusive load disaggregation. Among them, NILD based on a combinatorial optimization algorithm [7] determines the power consumption value of each appliance by investigating load characteristics as well as error comparisons between power states of combined appliances and the total power. Chang [8] and Lin [9] used the Particle Swarm Optimization (PSO) algorithm to solve the disaggregation problem based on the steady state current on a few electrical appliances, but the disaggregation result error was large. In order to solve the NILD problem, Piga [10] proposed a sparse optimization method to improve the disaggregation accuracy. The combinatorial optimization method is essentially an NP-hard problem, so its efficiency is a challenge. In addition, the optimization theory could be only used to analyze discrete states of electrical appliances, so it is difficult to model loads with large load fluctuations.

With the development of machine learning, pattern recognition algorithms have been applied to NILD. Batra [11] solved the depolymerization problem of low-power appliances using K-nearest neighbor regression (KNN) [12], but the algorithm could not solve the problem of the large power difference between appliances. Kolter [13] used the sparse coding algorithm to learn the power consumption models of each electrical appliance and used these models to predict the power of each electrical appliance. Johnson [14] used unsupervised learning for NILD, and this model had a high training speed. However, compared to the supervised algorithms, the ability of Johnson’s method’s to identify complex continuous state loads was limited because of the lack of prior knowledge. Kim [15] used the multi-factor hidden Markov algorithm to disaggregate the continuous value of each electrical appliance according to the given total power data. Some excellent machine learning algorithms, such as the support vector machine [16] and the adaboost algorithm [17], achieved certain processes, but these methods shared the same problem: a large number of load characteristics were required for identification, a requirement that was often difficult to meet in practice. Different from traditional methods, the deep learning method [18,19] is able to automatically extract features from original data [20]. In Kelly’s [21] experiment, various NILD algorithms using deep learning were proposed, such as the Delousing AutoEncoder (DAE), the long-short term memory network (LSTM), the gated recurrent unit (GRU), the Factorial Hidden Markov model (FHMM), and the CO method. The DAE algorithm was proven to have good disaggregation results. Zhang [22] used two convolutional neural network algorithms for load disaggregation. Compared with Kelly’s method, the two CNN methods, sequence-to-sequence and sequence-to-point, achieved better performance [23], but their layer numbers were small, and hence there were unable to extract higher level load characteristics. In the above methods, the CO algorithm, the DAE, and the two CNN methods were all trained by low-frequency data from the REDD dataset, which was first processed by the NILM-TK toolkit. The sampling interval of the data was 3 s. With an improvement of the model structure, Yang [24] proposed a semisupervised deep learning framework based on BiLSTM and the temporal convolutional network for multi-label load classification. Akhilesh [25] proposed a multilayer deep neural network based on the sequence-to-sequence methodology, and the algorithm, by reading the daily load profile for the total power consumption, could identify the state of the appliances according to the device-specific power signature. Since the neural networks were only trained for each appliance and the computational cost was high, Anthony [26] proposed UNet-NILM for multi-task appliances’ state detection and power estimation, which had a good performance compared with traditional single-task deep learning.

The innovation of the algorithm proposed in this paper lies in the following: The multi-scale structure is used to extract different load information according to the characteristics of load disaggregation.

The attention mechanism is used to fuse the load information at different scales to further enhance the feature extraction ability of the network, especially for the extraction of electrical features that are not frequently used. The overall architecture uses a skip connection of the residual network [27] to improve network performance. Experimental results on two benchmark datasets show that our method is superior to other present methods.

2. Multi-Scale Attention Residual Network

2.1. Deep Residual Network

The depth of the neural network [28] is crucial to the performance of the model. Increasing the depth of the neural network is helpful to extract more complex features and improve the accuracy of the algorithm. However, at the same time, when the network reaches a certain depth, further deepening would make the training accuracy saturated or even reduced. Traditional methods dealing with gradient disappearance and gradient explosion, e.g., the activation function Relu and batch normalization, are able to alleviate these problems to a certain extent, but not fundamentally solve the problems.

The deep residual network (Resnet) [29] uses the idea of cross-layer connection. If those behind the deep network are identity mapping, the model would degenerate directly into a shallow network, and it is difficult to use the neural network stacked with hidden layers to fit into a potential identity mapping function $H(x) = x$. However, if the network is designed to be $H(x) = F(x) + x$, the procedure can be translated into learning a residual function $F(x) = H(x) - x$. When $F(x) = 0$, it presents an identity map $F(x) = H(x)$. The residual network is a structure that outputs the features of front layers to back layers. By introducing this structure, a neural network is able to perform well.

Assuming that the input of a forward neural network is x , the dimension is $H \times W \times C$, and the expected output of the network is $H(x)$. The residual structure $F(x) + x$ can be realized by a forward neural network and a skip connection. When the number of input and output channels is the same, the dimension of $H(x)$ is also $H \times W \times C$. A skip connection means that one or more layers of the network are bypassed. A skip connection performs only one identity mapping and adds its outputs to the outputs of stack layers.

The residual structure is shown in Figure 1. Each structure has two layers, where W_1 and W_2 are the weight matrixes. The input is sequentially multiplied by the W_1 matrix of the first layer, activated by the relu function, and then multiplied by the W_2 matrix of the second layer to obtain the forward output. The forward neural network of the residual structure could be expressed as

$$F(x) = W_2\sigma(W_1x), \quad (1)$$

where σ is the activation function. The right hand is the skip connection x of the residual structure, and the final output is obtained after a summation operation. The formula is as follows:

$$y = F(x, W_i) + x, \quad (2)$$

where $F(x, W_i)$ is the residual mapping function that needs to be studied. W_i represents the weight matrix of the hidden layer. When the input and output dimensions need to be changed, a linear transformation $W \cdot x$ could be performed through the input at the skip connection residual structure. Thus, the same figure of input and output characteristics could be expressed as

$$H(x) = F(x, W_i) + W_s x. \quad (3)$$

According to Equation (3), it could be noted that for a deeper layer L , its relationship with the l layer could be expressed as

$$x_L = x_l + \sum_{i=1}^{L-1} F(x_l, W_l), \quad (4)$$

where layer x_L and layer x_l are the residual unit inputs of layer L and layer l . According to Equation (4), inputs of the residual unit in layer L could be expressed as the sum of inputs of a shallow residual unit and all the complex mappings. The calculation power needed of the sum is much less than that of the quadratics.

With a loss function, and according to the chain rule of back propagation, we can obtain

$$\frac{\partial \varepsilon}{\partial x_l} = \frac{\partial \varepsilon}{\partial x_L} \frac{\partial x_L}{\partial x_l} = \frac{\partial \varepsilon}{\partial x_L} \left(1 + \frac{\partial}{\partial x_l} \sum_{i=1}^{L-1} F(x_l, W_l)\right). \quad (5)$$

This means that continuous multiplications generated in the network are replaced by plus operations, and the problem of gradient disappearance is well solved. The use of residual structure is able to increase the network depth and extract deeper load characteristics from the data, thus improving the accuracy of disaggregation algorithms. Based on the residual network, in view of the characteristics of load disaggregation, we replaced the convolutional layer with the multi-scale structure and the attention structure. Therefore, we proposed the MSA-Resnet.

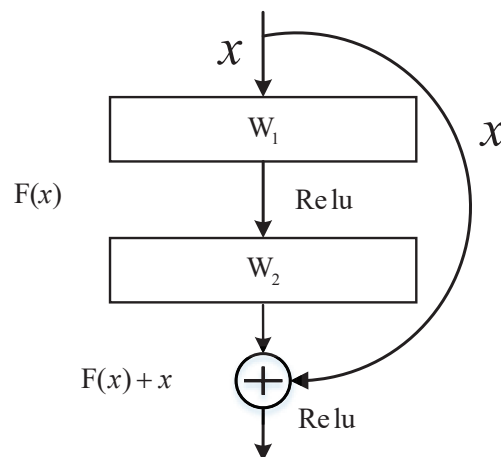


Figure 1. The residual structure.

2.2. Attention Mechanism

A convolution kernel, the core of a CNN, is generally regarded as aggregating information spatially and channel-wise on local receptive fields. A CNN is composed of a series of convolution layers, non-linear layers, and down-sampling layers, among others, and captures required features from global receptive fields [30].

However, in order to obtain better network performance, a squeeze and excitation attention mechanism is used in the network [31]. Its structure is shown in Figure 2. One of the novelties of the algorithm in this paper lies in the use of the attention mechanism in the residual structure to further improve the feature extraction ability of the network, especially for the extraction of electrical features that are not frequently used. This attention mechanism is different from the previous structure, as it improves performance from feature channels. The first 3D matrix in Figure 2 is composed of a feature graph C with a size of $H \times W$.

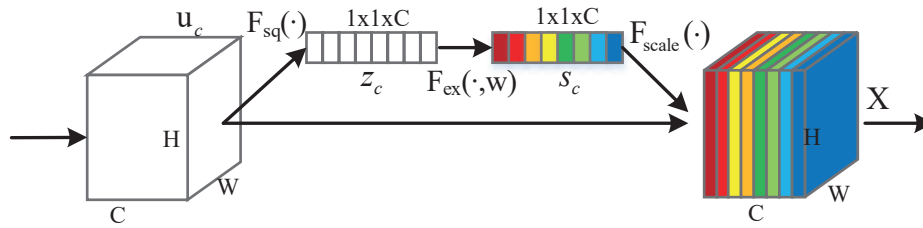


Figure 2. The attention block.

According to Figure 2, the spliced feature map of a multi-scale module is obtained through the global pooling layer to obtain attention vector z_c , which is compressed from the spatial dimension to obtain the global receptive field. z_c is a high-dimensional vector containing the low-order global feature information of the feature map, and its dimension is $1 \times 1 \times C$. The expression equation is as follows:

$$z_c = F_{sq}(u_c) = \frac{1}{h \times w} \left(\sum_h \sum_w u_c(i, j) \right). \quad (6)$$

Next, two fully connected layers are applied to establish correlations between channels as excitation and to output the same number of weights as the input feature. The equation is

$$s_c = F_{ex}(z_c, w_2) = \delta(w_2 \sigma(w_1 z_c)), \quad (7)$$

where the dimensions of w_1 and w_2 are all $\frac{C}{s} \times C$, s represents the scaling coefficient, and the attention vector s_c is obtained after being activated by the Relu function and the Sigmoid function. It is a high-dimensional vector of high-order global feature information obtained on the basis of attention vector z_c . The attention vector s_c further represents the change of load feature information in the channel dimension, and its dimension is also $1 \times 1 \times C$. A Sigmoid activation function is similar to a gating processing mechanism, which generates different weights for each feature channel of the attention vector s_c . Finally, the original three-dimensional matrix u_c is multiplied to complete the weight recalibration. Therefore, the importance of each load feature is obtained according to the feature map [32]. Finally, the useful load characteristics are enhanced and the less useful load characteristics are suppressed, which can improve the accuracy of load disaggregation of low-usage appliances. The equation is

$$X = F_{scale}(u_c, s_c) = u_c \cdot s_c. \quad (8)$$

2.3. Multi-Scale Attention Resnet Based NILD

From the point of view of neural networks, the problem of NILM can be interpreted in this way. Assuming that $Y(t)$ is the sum of all the active power consumption of appliances in the household, it can be expressed as the following formula:

$$Y(t) = \sum_i^I X_i(t) + e(t). \quad (9)$$

In the formula, $X_i(t)$ represents the power data of electrical equipment i at time t , I represents the number of electrical equipment, and e represents the model noise. Therefore, there is a pair of data (X, Y) , a model can be trained to represent the relationship between X and Y . $X = f(Y)$ is a non-linear regression problem, and the neural network is an effective method to learn the function f .

The overall network structure of the multi-scale attention residual network (MSA-Resnet) proposed in this paper is shown in Figure 3.

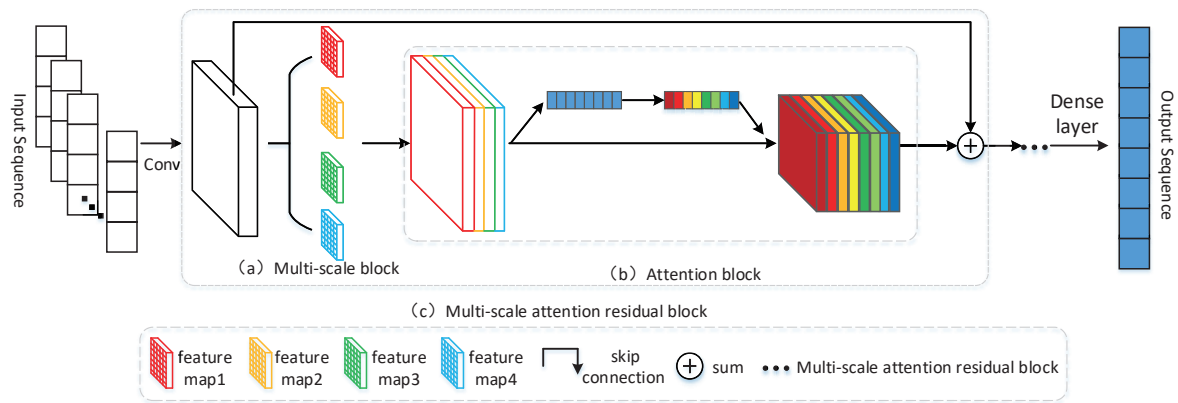


Figure 3. The MSA-Resnet overall structure.

(a) The multi-scale module is composed of convolution kernels with sizes of 3×1 , 5×1 , and 1×1 and the pooling layer [33]. By combining (b) the attention block and the residual structure, (c) the multi-scale attention residual block is formed. The structure of (a) is shown in Figure 4.

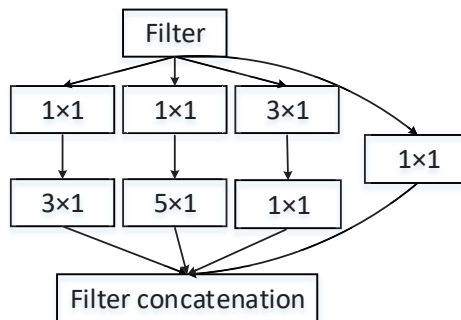


Figure 4. Multi-scale block.

All convolution cores of original residual elements are 3×1 in size, which makes it impossible for convolution layers to observe load data from multiple scales, and difficult to obtain more abundant load features. The multi-scale module first goes through a 3×1 convolution, followed by four branches. The first branch uses a 1×1 convolution to increase the load characteristics transformation [34], and a 3×1 convolution is then applied to obtain a feature map (map1). The second branch is convolved at 1×1 , and a 5×1 convolution is then added to obtain map2. The third branch is pooled at 3×1 to obtain map3. The fourth branch uses a 1×1 convolution to obtain map4. Finally, these feature maps are concatenated to input vectors for the attention module. Because the actual load power has a large number of different gear positions, switch starts and stops, and operating characteristics, the multi-scale feature method can improve the network's ability to extract load characteristics and increase the diversity of different scales of the network, thus improving the accuracy of non-intrusive load disaggregation.

In the network structure of MSA-Resnet, nine multi-scale attention residual blocks are used. Forty convolution cores are used in the first three blocks, 60 convolution cores are used in the fourth to sixth blocks, and 80 convolution cores are allocated in the last three. The first convolution layer and each output part of multi-scale attention residual block is activated by an activation function Leaky-Relu. Relu and Leaky-Relu [35] are shown in Figure 5:

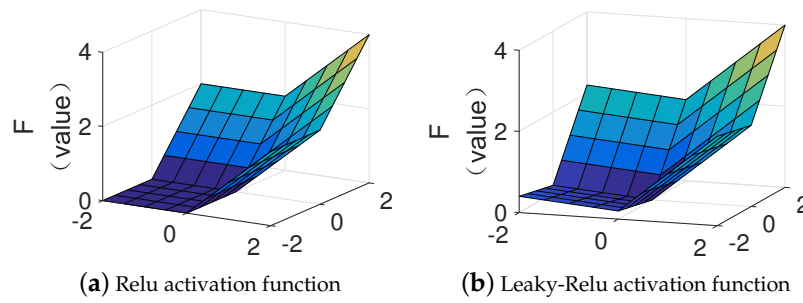


Figure 5. The Comparison of Relu and Leaky-Relu.

The Relu activation function represents the “modified linear element,” which could accelerate the convergence speed of the network. Its equation is

$$f(x) = \max(0, x). \quad (10)$$

When the input is positive, the derivative is not zero, so the learning is based on the gradient. However, when the input is negative, the learning speed of Relu slows down and can even inactivate the neurons, such that it cannot follow new weights. Equation (11) represents the Leaky-Relu activation function, where $\lambda \in (0, 1)$ modifies the data distribution and retains the value of the negative axis. As a result, the information retention ability is improved without losing more load characteristics, and the gradient is guaranteed not to appear.

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ \lambda x & \text{if } x < 0 \end{cases}. \quad (11)$$

3. Data Selection and Preprocessing

3.1. Data Sources

The experimental data in this paper is from the public dataset UK-DALE [36] and WikiEnergy [37]. The UK-DALE dataset is a public access dataset from the UK, and the sampling frequency is 1/6 Hz. The WikiEnergy dataset was produced by the Pecan Street company in the UK and contains the data of nearly 600 households. It includes the total power consumed by each household over a period of time and the power consumed by each individual electrical appliance. The sampling frequency of the dataset is 1/60 Hz [38]. Kettles, air conditioners, fridges, microwaves, washing machines, and dishwashers were chosen as non-intrusive load disaggregation tasks for the following reasons: (1) The power consumption of these electrical appliances is a large proportion of the total power consumption. They are representatives of electrical appliances. (2) Electrical appliances with low frequency and minimal power consumption used in the data are easily disturbed by noise and not easily disaggregated. (3) The power consumption of these six electrical devices includes mode disaggregation from simple to complex.

3.2. Data Preprocessing

Firstly, the NILM-TK toolkit was used to export the power data of the selected home appliances from the WikiEnergy database and UK-DALE database. We then created aggregate power profiles and used them as the experimental data. Secondly, different evaluation indexes often have different dimensions, and their values are quite wide ranged, and this may affect the analysis results. In order to eliminate the influence of these differences, data standardization is needed. Here, the maximum and minimum value normalization method was used to normalize the result between $[0, 1]$. The normalization equation is

$$x_t^* = \frac{x_t - X_{\min}}{X_{\max} - X_{\min}}, \quad (12)$$

where x_t is the total unstandardized power at time t , X_{max} is the maximum of the total power sequence, X_{min} is the minimum of the total power sequence, and x_t^* is the standardized result at time t .

3.3. Sliding Window

Deep learning training relies on a large amount of data. The NILM-TK toolkit was used to process the database. We selected the desired electrical data and sorted it into an Excel file. The top 80% of the processed data was taken as training data. The total power sequence X was taken as the input sequence, and the individual electrical appliance Y was taken as the target sequence. The remaining 20% of the processed data was taken as testing data. In order to increase the training data of the network and improve the expression ability of the data, the data was processed by using a sliding window [39].

As shown in Figure 6a, the overlap sliding window [40] was used to process the total power sequence and the target sequence in the training data to increase the data samples. Assuming that the length of power sequence is M , a window of length N was cut from the original data with a sliding step of 1, and the sliding operation was then carried out to obtain $M - N + 1$ samples. However, as shown in Figure 6b, the non-overlap sliding window was used to process the testing data to save time. Assuming that the sequence length is H , $\frac{H}{N}$ samples could be obtained.

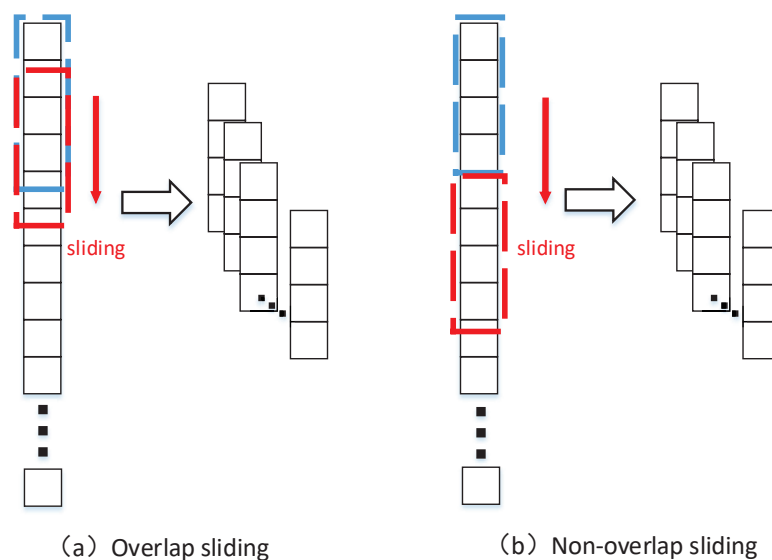


Figure 6. Sliding operation.

4. Result

This experiment used the Keras neural network framework. The computer processor was AMD2600, and the graphics card was 1060 6G. After data was standardized, the length of the sliding window was set to 200, the learning rate of the network was set to 0.001, and the Adam optimizer was selected as the network optimizer.

Kelly's experiments indicate that the DAE algorithm performs well in NILD, and Zhang C's work also shows a good performance of CNNs in sequence-to-sequence and sequence-to-point load disaggregation. From the WikiEnergy data, we selected the air conditioner, fridge, microwave, washing machine, and dishwasher from Household 25. From the UK-DALE dataset, the kettle, fridge, microwave, washing machine, and dishwasher of Household 5 were selected. In order to verify the effectiveness and stability of the algorithm proposed in this paper, four approaches were compared with the MSA-Resnet: the KNN, the DAE, the CNN sequence-to-sequence learning (CNN s-s), and the CNN sequence-to-point learning (CNN s-p). Firstly, the WikiEnergy dataset was tested. Figure 7 shows the disaggregation effect diagrams of five appliances of WikiEnergy from Household 25, and the actual power consumption

data of these appliances. The figure compares the four disaggregation methods with the MSA-Resnet proposed in this paper.

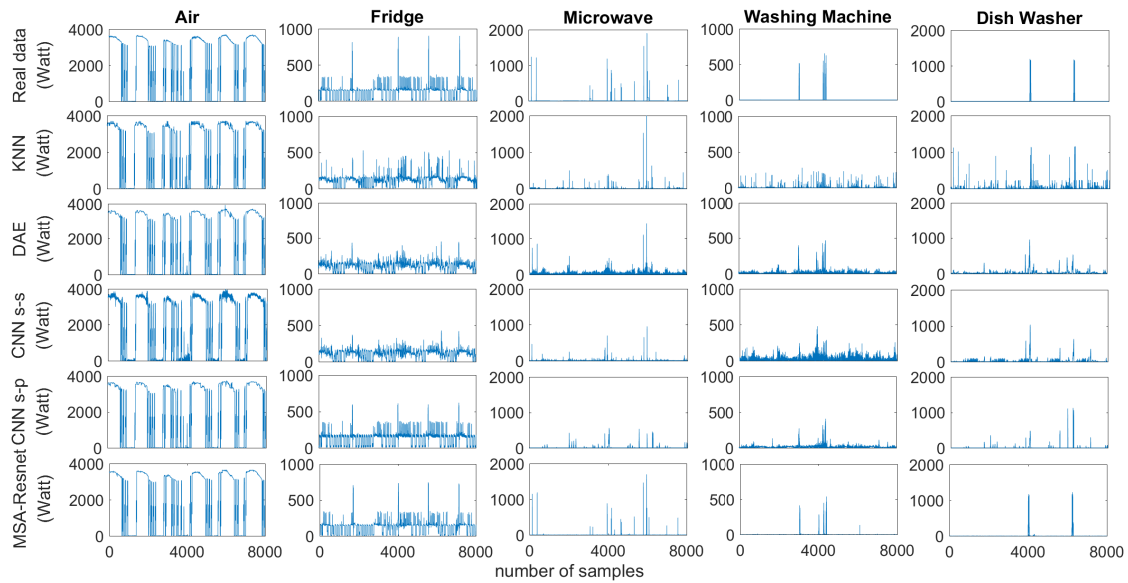


Figure 7. Comparison load disaggregation results of Household 25 in the WikiEnergy dataset.

In order to verify the effectiveness of the proposed method, two evaluation indexes were selected to evaluate the performance of the algorithm: the Mean Absolute Error (*MAE*) and the Signal Aggregate Error (*SAE*). The *MAE* evaluation index was used to measure the average error of power consumption and the actual power consumption of individual electrical appliances disaggregated at each moment. The *MAE* is expressed as the following:

$$MAE = \frac{1}{T} \sum_{t=1}^T |p_t - g_t|, \quad (13)$$

where g_t represents the actual power consumed by an appliance at time t , p_t represents the disaggregation power of the appliance at time t , and T represents the number of time points.

Equation (14) is the expression of the *SAE*, where \hat{e} and e represent the power consumption predicted by disaggregation within a period of time and the real power consumption within a period of time. This index is helpful for daily electricity reports.

$$SAE = \frac{|\hat{e} - e|}{e}. \quad (14)$$

Figure 7 describes disaggregations of Household 25 in the WikiEnergy dataset. It can be seen that the above algorithms can basically achieve effective load disaggregation for the air conditioner. In the load disaggregation diagram of the fridge, the DAE and CNN s-s algorithms fluctuate greatly in the mean area of the appliance, compared with other algorithms. The KNN algorithm has the worst load disaggregation effect on the last three kinds of electrical appliances, so it could not realize an effective disaggregation of mutation points. For these three low-frequency electrical appliances, the load disaggregation of CNN s-s and CNN s-p algorithms are stable compared with the other two algorithms, but the load disaggregation of the CNN s-p method fluctuates greatly in the region of low power consumption. In summary, compared with other methods in load disaggregation, the MSA-Resnet shows the best performance on each electrical appliance, based on the power consumption curve.

Table 1 shows comparisons of *MAE* and *SAE* indexes of Household 25 load disaggregation in the WikiEnergy dataset. It can be seen that MSA-Resnet has obvious advantages in the disaggregations of the air conditioner, fridge, microwave, washing machine, and dishwasher. According to the *MAE* index, the MSA-Resnet performs better than the other four methods. For the *SAE*, the MSA-Resnet achieves the lowest value on the fridge, washing machine, and dishwasher, and accurate disaggregation of energy is achieved over a period of time. Combined with Figure 7 and Table 1, it can be inferred that the shallow CNN s-s and CNN s-p have difficulty accurately disaggregating the total power into the appliances with lower frequency. Compared with KNN and MSA-Resnet, the disaggregation errors of CNN s-s and CNN s-p are larger, because the structure of shallow CNNs is not able to extract deeper and more effective load characteristics, and their disaggregation effect is not as good as that of MSA-Resnet. There are two reasons for this: firstly, the residual is used to deepen the network and better enhance the ability to learn unbalanced samples; secondly, the ability to deal with low frequency appliances by multi-scale convolutions is strong. As can be seen in Figure 7, the overall disaggregation effect of the KNN on the washing machine is not good, but the disaggregation error is small in terms of two indicators. To explain this phenomenon, certain interval periods are selected for comparative analysis, as shown in Figure 8, the disaggregation comparison diagram shows each algorithm on each electrical appliance with a finer scale. The figure reflects the ability of the KNN to detect peak values. It can be seen in Figure 8b,c that the KNN is not able to accurately disaggregate mutation points, but it could process regions with a power close to 0.

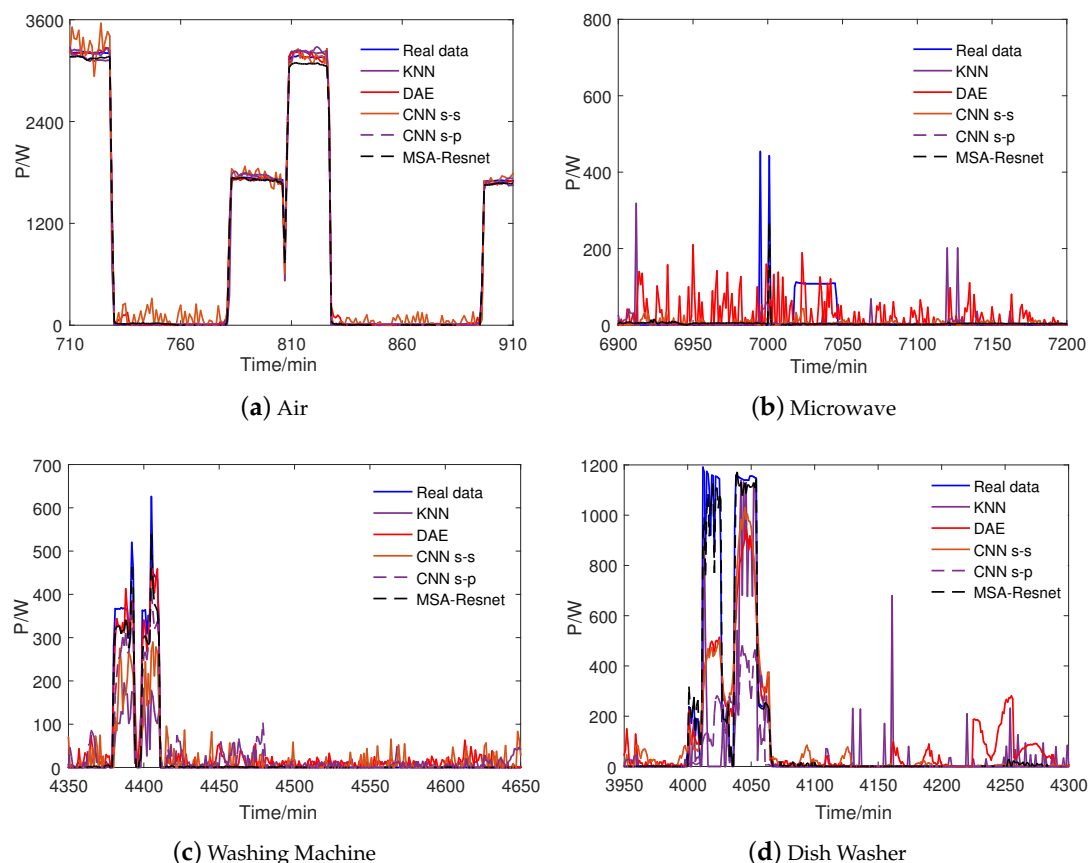


Figure 8. Load disaggregation comparison for Household 25 of the WikiEnergy dataset.

Table 1. Comparison of load disaggregation index of Household 25 of the WikiEnergy dataset.

Index	Method	Air	Fridge	Microwave	Washing Machine	Dish Washer
MAE	KNN	38.484	34.014	6.928	6.677	10.630
	DAE	36.964	39.520	17.015	12.081	25.107
	CNN s-s	61.129	38.413	9.973	18.497	19.084
	CNN s-p	39.635	13.760	13.155	11.959	11.624
	MSA-Resnet	36.388	10.440	4.862	2.161	2.013
SAE	KNN	0.0006	0.026	0.060	0.323	0.121
	DAE	0.0001	0.071	2.317	2.835	1.405
	CNN s-s	0.013	0.051	0.060	3.925	0.886
	CNN s-p	0.006	0.074	0.319	2.467	0.098
	MSA-Resnet	0.014	0.025	0.143	0.152	0.052

After the disaggregation of load, power thresholds of electrical appliances were used to distinguish the on/off states, so as to calculate their evaluation indexes. The thresholds of the air conditioner, fridge, microwave, washing machine, and dishwasher were set to 100 W, 50 W, 200 W, 20 W, and 100 W, respectively. *Recall* rate, *precision* rate, *accuracy* rate, and *F1* values [41] were used to further evaluate the performance of the different algorithms in their on/off states.

Recall represents the probability of predicting correctly in the instance with a positive label:

$$Recall = \frac{TP}{TP + FN'} \quad (15)$$

where True Positive (*TP*) represents the number of predicted states that are disaggregated as “on” when their ground truth state is “on”, and False Negative (*FN*) denotes the number of predicted states that are “on” when their ground truth state is “off”. There are two possibilities: one is to predict the original positive class as a positive class (*TP*), and the other is to predict the original positive class as a negative class (*FN*).

Precision refers to the proportion of samples that are predicted to be in an “on” state and are indeed in an “on” state:

$$Precision = \frac{TP}{TP + FP'} \quad (16)$$

where False Positive (*FP*) represents the number of states that are actually “off” when their predicted states are “on”. *Accuracy* refers to the ratio of the number of samples correctly predicted to the number of the total dataset:

$$Accuracy = \frac{TP + TN}{P + N}, \quad (17)$$

where *P* is the number of positive samples, and *N* is the number of negative samples. *F1* can be expressed as

$$F1 = 2 \times \frac{precision \times recall}{precision + recall}. \quad (18)$$

Table 2 is a comparison of the evaluation indexes for judging “on” or “off” states of Household 25 electrical appliances. It can be seen from Table 2 that for *Accuracy* and *F1*, MSA-Resnet achieves the best performance in various electrical appliances. The disaggregation diagrams of the microwave, the washing machine, and the dishwasher are in Figure 8, which shows that, in the actual power consumption of these three electrical appliances, their proportion of “on” states is significantly lower than that of the first two electrical appliances. In such unbalanced sample data with a small sample size, the “on” states of the washing machine cannot be effectively predicted using the CNN s-s and CNN s-p, whereas the MSA-Resnet presents better results.

Table 2. Performance comparison of different algorithms for electrical on/off judgement of Household 25 in the WikiEnergy dataset.

Index	Method	Air	Fridge	Microwave	Washing Machine	Dish Washer
Recall	KNN	0.998	0.996	0.759	0.290	0.561
	DAE	0.999	0.996	1	0.451	0.833
	CNN s-s	0.999	0.990	0.949	0.290	0.868
	CNN s-p	0.999	1	0.987	0.129	0.596
	MSA-Resnet	1	0.986	0.880	0.806	1
Precision	KNN	0.987	0.870	0.198	0.236	0.336
	DAE	0.987	0.853	0.050	0.229	0.281
	CNN s-s	0.939	0.847	0.033	0.428	0.391
	CNN s-p	0.995	0.996	0.050	0.047	0.414
	MSA-Resnet	0.999	0.988	0.795	0.962	0.884
Accuracy	KNN	0.991	0.889	0.967	0.993	0.978
	DAE	0.991	0.872	0.812	0.992	0.967
	CNN s-s	0.958	0.864	0.729	0.995	0.978
	CNN s-p	0.997	0.980	0.816	0.986	0.982
	MSA-Resnet	0.999	0.981	0.997	0.999	0.998
F1	KNN	0.993	0.928	0.314	0.260	0.421
	DAE	0.993	0.919	0.095	0.304	0.421
	CNN s-s	0.968	0.913	0.064	0.346	0.539
	CNN s-p	0.997	0.986	0.096	0.069	0.489
	MSA-Resnet	0.999	0.987	0.835	0.877	0.938

In order to prove the effectiveness of the Leaky-Relu function, under the same conditions, comparative experiments are conducted with WikiEnergy's Household 25 using the Relu function. According to the experimental results in Table 3, at the two indicators of the *MAE* and the *SAE*, the algorithm using the Leaky-Relu function is better.

Table 3. Comparison of activation function of Household 25 of the WikiEnergy dataset.

Index	Function	Air	Fridge	Microwave	Washing Machine	Dish Washer
MAE	Relu	46.029	10.799	6.954	3.403	4.539
	Leaky-Relu	36.388	10.440	4.862	2.161	2.013
SAE	Relu	0.015	0.034	0.234	0.287	0.157
	Leaky-Relu	0.014	0.025	0.143	0.152	0.052

For further verification, we selected five electric appliances from Household 5 in the UK-DALE dataset for additional experiments. Figure 9 shows the results of disaggregation. The figure shows that all of the above algorithms are able to achieve effective disaggregation for the kettle, an electrical appliance that is used often. For the fridge, the KNN and the DAE work worse than the CNN s-s, the CNN s-p, and the MSA-Resnet. For the microwave, the washing machine, and the dishwasher, which are infrequently used and have a low power consumption, the MSA-Resnet has better disaggregation results than the other two deep learning algorithms, mainly because it could better detect peaks and state changes.

Table 4 shows Household 5's load disaggregation evaluation index in the UK-DALE dataset. Table 4 shows that the MSA-Resnet does better in *MAE* and *SAE* compared with other methods. For the *MAE*, the MSA-Resnet performs better with respect to the kettle, the fridge, the washing machine, and the dishwasher. The MSA-Resnet has smaller *SAE* values in the kettle, the fridge, and the washing machine.

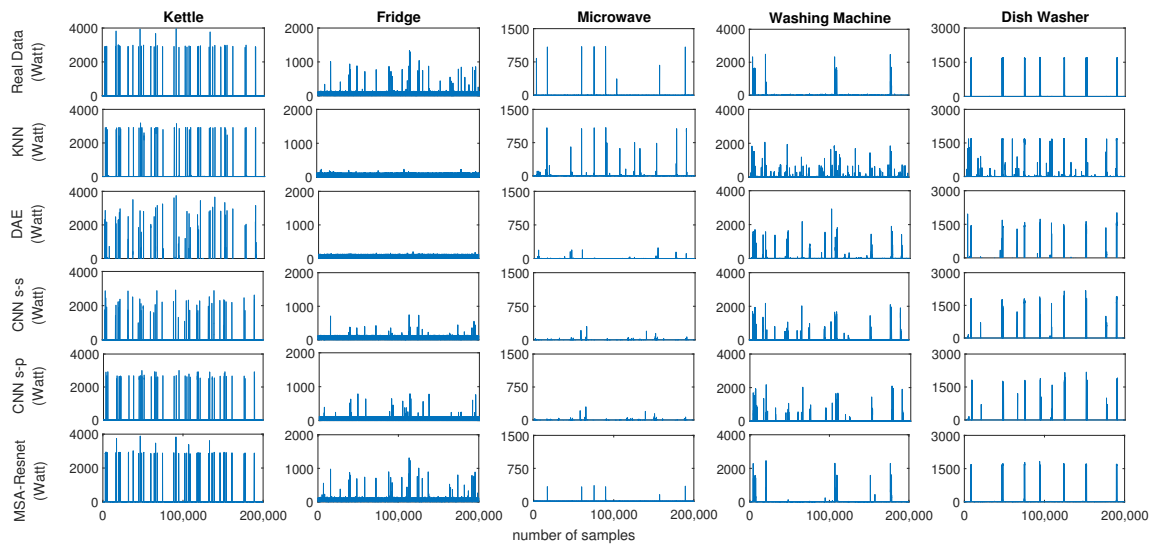


Figure 9. Comparison load disaggregation results of Household 5 in the UK-DALE dataset.

Table 4. Comparison of the load disaggregation indexes of Household 5 of the UK-DALE dataset.

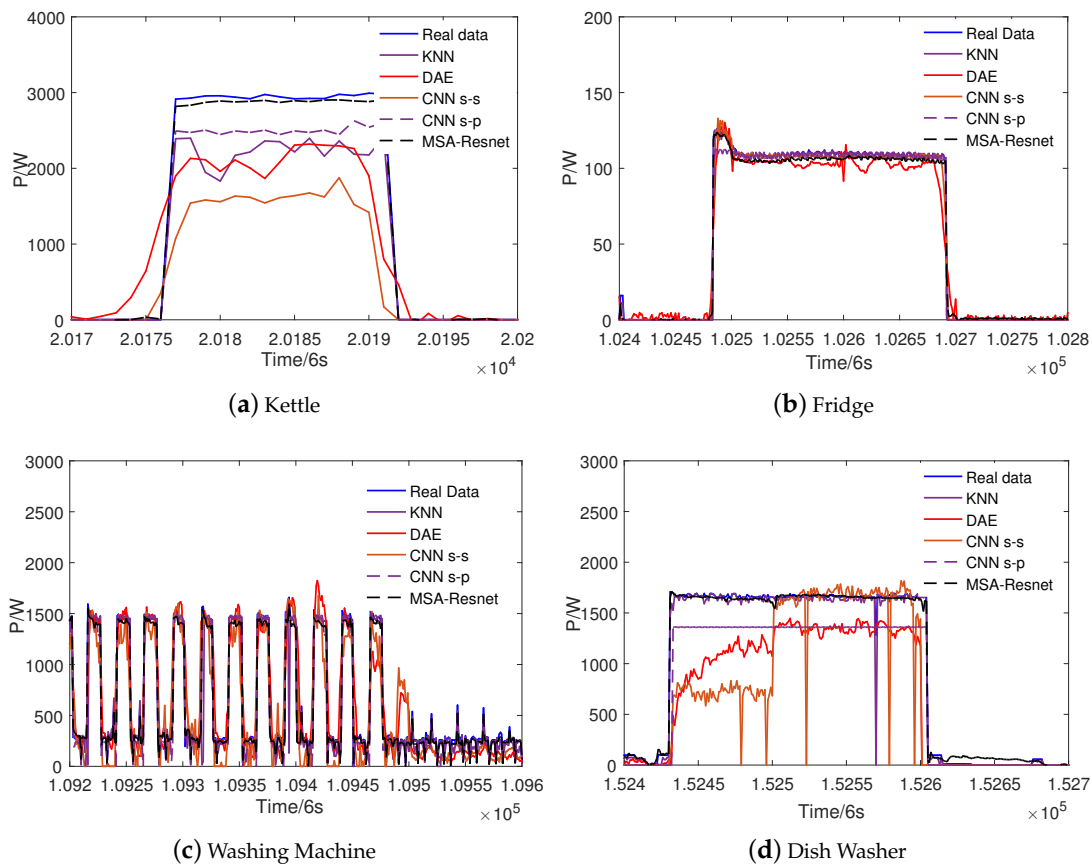
Index	Method	Kettle	Fridge	Microwave	Washing Machine	Dish Washer
MAE	KNN	1.413	2.407	0.378	4.032	3.274
	DAE	8.867	8.218	1.226	14.920	12.756
	CNN s-s	8.829	3.866	1.125	20.696	9.101
	CNN s-p	4.002	4.517	1.159	23.881	9.747
	MSA-Resnet	0.804	2.136	0.906	3.618	2.601
SAE	KNN	0.076	0.015	0.054	0.018	0.001
	DAE	0.377	0.021	0.748	0.006	0.340
	CNN s-s	0.522	0.032	0.880	0.315	0.213
	CNN s-p	0.242	0.024	0.845	0.302	0.154
	MSA-Resnet	0.001	0.013	0.720	0.0007	0.050

Table 5 shows the judgement results of “on” and “off” states of Household 5 in the UK-DALE dataset. The thresholds of the kettle, the fridge, the microwave, the washing machine, and the dishwasher were set to 100 W, 50 W, 200 W, 20 W, and 100 W, respectively. Table 5 shows that the Recalls of the washing machine and the dishwasher using the CNN s-s and the CNN s-p are low, the number of positive samples is small, and its ability to predict the “on” state is poor. If the task of judging the electrical state is considered as classification, appliances with a high utilization rate have better classification results.

Figure 10 shows load disaggregation comparisons of these five methods over a period of time. It can be seen from the figure that, compared with other algorithms, the MSA-Resnet could better disaggregate equipments, whereas the KNN and the DAE have the worst decomposition abilities. For the low-frequency washing machine and dishwasher, the MSA-Resnet could still well fit the power curve, because of its network structure. It uses multi-scale convolutions to obtain rich load characteristics, and it improves the performance of the network through the attention mechanism and the residual structure.

Table 5. Performance comparison of different algorithms for electrical on/off judgement of Household 5 in the UK-DALE dataset.

Index	Method	Kettle	Fridge	Microwave	Washing Machine	Dish Washer
Recall	KNN	0.987	0.988	0.944	0.911	0.968
	DAE	0.985	0.944	0	0.921	0.938
	CNN s-s	0.969	0.990	0	0.857	0.904
	CNN s-p	0.993	0.923	0	0.838	0.928
	MSA-Resnet	1	0.994	0.951	0.927	0.946
Precision	KNN	0.998	0.974	0.933	0.617	0.799
	DAE	0.650	0.932	0	0.471	0.813
	CNN s-s	0.946	0.944	0	0.663	0.835
	CNN s-p	1	0.968	0	0.701	0.829
	MSA-Resnet	0.996	0.996	1	0.672	0.850
Accuracy	KNN	0.999	0.986	0.999	0.981	0.996
	DAE	0.997	0.955	0.999	0.967	0.996
	CNN s-s	0.999	0.975	0.999	0.983	0.996
	CNN s-p	0.999	0.961	0.999	0.984	0.996
	MSA-Resnet	1	0.996	0.999	0.985	0.997
F1	KNN	0.992	0.981	0.939	0.736	0.875
	DAE	0.783	0.938	Nan	0.623	0.871
	CNN s-s	0.957	0.967	Nan	0.748	0.868
	CNN s-p	0.996	0.945	Nan	0.764	0.876
	MSA-Resnet	0.998	0.995	0.975	0.779	0.895

**Figure 10.** Load disaggregation comparison for Household 5 of the UK-DALE dataset.

In order to prove the effectiveness of the Leaky-Relu function, a comparative experiment with the Relu function was also done on the UK-DALE dataset. Table 6 can prove that the Leaky-Relu function is still the best.

Table 6. Comparison of the activation function of Household 5 of the UK-DALE dataset.

Index	Function	Kettle	Fridge	Microwave	Washing Machine	Dish Washer
MAE	Relu	2.449	4.506	1.371	25.126	4.706
	Leaky-Relu	0.804	2.136	0.906	3.618	2.601
SAE	Relu	0.286	0.061	0.950	0.253	0.036
	Leaky-Relu	0.001	0.013	0.720	0.0007	0.050

5. Conclusions

Load disaggregation is an important part of smart grids. At present, existing non-intrusive load disaggregation methods based on deep learning have some problems; for example, they easily lose features and have difficulty in detection, they do not identify low-use electrical appliances well, and their networks degrade easily due to gradient disappearance. The disaggregation effects of traditional methods are also very poor. In order to solve these problems, the MSA-Resnet is proposed for NILD. The residual network deepens the network structure, avoids the gradient, and reduces the optimization difficulty. Multi-scale convolutions obtain richer load characteristics and avoids feature simplification. The attention mechanism is used to enhance the ability of the network to learn load characteristics and improves the performance of the network. With its excellent performance on the WikiEnergy and UK-DALE datasets, the MSA-Resnet is shown to be an effective way to solve non-intrusive load disaggregation. In future work, we will conduct further experiments on public datasets such as REDD and real household data of the State Grid to verify the generalization performance of the model.

Author Contributions: Conceptualization, L.W., X.Z., and J.Q.; methodology, M.X. and Y.X.; software, K.W.; validation, X.Z. and M.X.; formal analysis, L.W., X.Z., and J.Q.; investigation, X.Z. and M.X.; resources, M.X. and K.W.; data curation, X.Z. and M.X.; writing—original draft preparation, X.Z.; writing—review and editing, L.W. and M.X.; visualization, X.Z.; supervision, M.X. and Y.X.; project administration, L.W.; funding acquisition, M.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This work was supported by the State Grid Corporation of China Project ‘Fundamental Theory of Dynamic Demand Response Control Based on Large-Scale Diversified Demand Side Resources’.

Conflicts of Interest: The authors declare that there is no conflict of interest regarding the publication of this paper.

References

1. Wong, Y.F.; Şekercioğlu, Y.A.; Drummond, T.; Wong, V.S. Recent approaches to non-intrusive load monitoring techniques in residential settings. In Proceedings of the 2013 IEEE Computational Intelligence Applications in Smart Grid, Singapore, 16–19 April 2013; pp. 73–79.
2. Prada, J.; Dorronsoro, J.R. General noise support vector regression with non-constant uncertainty intervals for solar radiation prediction. *J. Mod. Power Syst. Clean Energy* **2018**, *6*, 268–280. [\[CrossRef\]](#)
3. Liu, H.; Zeng, P.; Guo, J.; Wu, H.; Ge, S. An optimization strategy of controlled electric vehicle charging considering demand side response and regional wind and photovoltaic. *J. Mod. Power Syst. Clean Energy* **2015**, *3*, 232–239. [\[CrossRef\]](#)
4. Tostado-Véliz, M.; Arévalo, P.; Jurado, F. A Comprehensive Electrical-Gas-Hydrogen Microgrid Model for Energy Management Applications. *Energy Convers. Manag.* **2020**, *228*, 113726. [\[CrossRef\]](#)
5. Hart, G.W. Nonintrusive appliance load monitoring. *Proc. IEEE* **1992**, *80*, 1870–1891. [\[CrossRef\]](#)
6. Rahimpour, A.; Qi, H.; Fugate, D.; Kuruganti, T. Non-intrusive energy disaggregation using non-negative matrix factorization with sum-to-k constraint. *IEEE Trans. Power Syst.* **2017**, *32*, 4430–4441. [\[CrossRef\]](#)
7. Zoha, A.; Gluhak, A.; Imran, M.A.; Rajasegarar, S. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors* **2012**, *12*, 16838–16866. [\[CrossRef\]](#)

8. Chang, H.H.; Lin, L.S.; Chen, N.; Lee, W.J. Particle-swarm-optimization-based nonintrusive demand monitoring and load identification in smart meters. *IEEE Trans. Ind. Appl.* **2013**, *49*, 2229–2236. [\[CrossRef\]](#)
9. Lin, Y.H.; Tsai, M.S. Development of an improved time–frequency analysis-based nonintrusive load monitor for load demand identification. *IEEE Trans. Instrum. Meas.* **2013**, *63*, 1470–1483. [\[CrossRef\]](#)
10. Piga, D.; Cominola, A.; Giuliani, M.; Castelletti, A.; Rizzoli, A.E. Sparse optimization for automated energy end use disaggregation. *IEEE Trans. Control Syst. Technol.* **2015**, *24*, 1044–1051. [\[CrossRef\]](#)
11. Batra, N.; Singh, A.; Whitehouse, K. Neighbourhood nilm: A big-data approach to household energy disaggregation. *arXiv* **2015**, arXiv:1511.02900.
12. Tsai, M.S.; Lin, Y.H. Modern development of an adaptive non-intrusive appliance load monitoring system in electricity energy conservation. *Appl. Energy* **2012**, *96*, 55–73. [\[CrossRef\]](#)
13. Kolter, J.; Batra, S.; Ng, A. Energy disaggregation via discriminative sparse coding. *Adv. Neural Inf. Process. Syst.* **2010**, *23*, 1153–1161.
14. Johnson, M.J.; Willsky, A.S. Bayesian nonparametric hidden semi-Markov models. *J. Mach. Learn. Res.* **2013**, *14*, 673–701.
15. Kim, H.; Marwah, M.; Arlitt, M.; Lyon, G.; Han, J. Unsupervised disaggregation of low frequency power measurements. In Proceedings of the 2011 SIAM International Conference on Data Mining, Society for Industrial and Applied Mathematics, Mesa, AZ, USA, 28–30 April 2011; pp. 747–758.
16. Saitoh, T.; Osaki, T.; Konishi, R.; Sugahara, K. Current sensor based home appliance and state of appliance recognition. *SICE J. Control Meas. Syst. Integr.* **2010**, *3*, 86–93. [\[CrossRef\]](#)
17. Hassan, T.; Javed, F.; Arshad, N. An empirical investigation of VI trajectory based load signatures for non-intrusive load monitoring. *IEEE Trans. Smart Grid* **2013**, *5*, 870–878. [\[CrossRef\]](#)
18. Xia, M.; Wang, K.; Zhang, X.; Xu, Y. Non-intrusive load disaggregation based on deep dilated residual network. *Electr. Power Syst. Res.* **2019**, *170*, 277–285. [\[CrossRef\]](#)
19. Xia, M.; Zhang, X.; Weng, L.; Xu, Y. Multi-Stage Feature Constraints Learning for Age Estimation. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 2417–2428. [\[CrossRef\]](#)
20. Kuo, P.H.; Huang, C.J. A high precision artificial neural networks model for short-term energy load forecasting. *Energies* **2018**, *11*, 213. [\[CrossRef\]](#)
21. Kelly, J.; Knottenbelt, W. Neural nilm: Deep neural networks applied to energy disaggregation. In Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments, Seoul, Korea, 4–5 November 2015; pp. 55–64.
22. Zhang, C.; Zhong, M.; Wang, Z.; Goddard, N.; Sutton, C. Sequence-to-point learning with neural networks for nonintrusive load monitoring. *arXiv* **2016**, arXiv:1612.09106.
23. Liu, Y.; Liu, Y.; Liu, J.; Li, M.; Ma, Z.; Taylor, G. High-performance predictor for critical unstable generators based on scalable parallelized neural networks. *J. Mod. Power Syst. Clean Energy* **2016**, *4*, 414–426. [\[CrossRef\]](#)
24. Yang, Y.; Zhong, J.; Li, W.; Gulliver, T.A.; Li, S. Semi-Supervised Multi-Label Deep Learning based Non-intrusive Load Monitoring in Smart Grids. *IEEE Trans. Ind. Inform.* **2019**, *16*, 6892–6902. [\[CrossRef\]](#)
25. Yadav, A.; Sinha, A.; Saidi, A.; Trinkl, C.; Zörner, W. NILM based Energy Disaggregation Algorithm for Dairy Farms. In Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring, Yokohama, Japan, 18 November 2020; pp. 16–19.
26. Faustine, A.; Pereira, L.; Bousbiat, H.; Kulkarni, S. UNet-NILM: A Deep Neural Network for Multi-tasks Appliances State Detection and Power Estimation in NILM. In Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring, Yokohama, Japan, 18 November 2020; pp. 84–88.
27. Jin, Y.; Guo, H.; Wang, J.; Song, A. A Hybrid System Based on LSTM for Short-Term Power Load Forecasting. *Energies* **2020**, *13*, 6241. [\[CrossRef\]](#)
28. de Paiva Penha, D.; Castro, A.R.G. Home appliance identification for NILM systems based on deep neural networks. *Int. J. Artif. Intell. Appl.* **2018**, *9*, 69–80. [\[CrossRef\]](#)
29. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
30. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual attention network for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3156–3164.
31. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.

32. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
33. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
34. Lin, M.; Chen, Q.; Yan, S. Network in network. *arXiv* **2013**, arXiv:1312.4400.
35. Wang, B.; Li, T.; Huang, Y.; Luo, H.; Guo, D.; Horng, S.J. Diverse activation functions in deep learning. In Proceedings of the 2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Nanjing, China, 24–26 November 2017; pp. 1–6.
36. Nalmpantis, C.; Vrakas, D. Machine learning approaches for non-intrusive load monitoring: From qualitative to quantitative comparison. *Artif. Intell. Rev.* **2019**, *52*, 217–243. [\[CrossRef\]](#)
37. Kelly, J.; Batra, N.; Parson, O.; Dutta, H.; Knottenbelt, W.; Rogers, A.; Singh, A.; Srivastava, M. Nilmtk v0.2: A non-intrusive load monitoring toolkit for large scale data sets: Demo abstract. In Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings, Memphis, TN, USA, 4–6 November 2014; pp. 182–183.
38. Biansongnorn, S.; Plungklang, B. Non-intrusive appliances load monitoring (nilm) for energy conservation in household with low sampling rate. *Procedia Comput. Sci.* **2016**, *86*, 172–175. [\[CrossRef\]](#)
39. Xia, M.; Wang, K.; Song, W.; Chen, C.; Li, Y. Non-intrusive load disaggregation based on composite deep long short-term memory network. *Expert Syst. Appl.* **2020**, *160*, 113669. [\[CrossRef\]](#)
40. Krystalakos, O.; Nalmpantis, C.; Vrakas, D. Sliding window approach for online energy disaggregation using artificial neural networks. In Proceedings of the 10th Hellenic Conference on Artificial Intelligence, Patras, Greece, 9–12 July 2018; pp. 1–6.
41. Xia, M.; Tian, N.; Zhang, Y.; Xu, Y.; Zhang, X. Dilated multi-scale cascade forest for satellite image classification. *Int. J. Remote Sens.* **2020**, *41*, 7779–7800. [\[CrossRef\]](#)

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).