2019 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)

Non-Intrusive Load Disaggregation Using Semi-Supervised Learning Method

Nan Miao
The Key Laboratory of Embedded System and Service Computing of Ministry of Education Tongji University
Shanghai, China
miaonan_s19@163.com

Shengjie Zhao*
School of Software Engineering The Key Laboratory of Embedded System and Service Computing of Ministry of Education Tongji University
Shanghai, China
shengjiezhao@tongji.edu.cn

Qingjiang Shi
School of Software Engineering Tongji University
Shanghai, China
qing.j.shi@gmail.com

Rongqing Zhang
School of Software Engineering Tongji University
Shanghai, China
rongqing.labs@gmail.com

2019 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC) 978-1-7281-5928-7/20/$31.00 ©2020 IEEE 10.1109/SPAC49953.2019.237865

Abstract—**With the emerging of smart metering around the world, there is a growing demand to analyse the residential energy usage. In this paper, we propose a Deep Neural Network (DNN)-based approach for non-intrusive load monitoring (NILM), which can achieve effective and efficient estimation of individual appliance usage according to a single main meter reading in a non-intrusive manner. Considering practical situations, two training methods are provided. The first training approach is fully supervised learning, which requires a ground truth of label, indicating the state of the appliance (ON/OFF), to build a prediction model. The second training approach is semi-supervised learning, leading to better performance by F-Measure metric while only requiring some more unlabelled training data. Experimental results on the low-sample rate REDD dataset demonstrate the superior performance of our proposed DNN-based method compared with Hidden Markov Model (HMM)-based and Graph Signal Processing (GSP)-based approaches.**

Index Terms—**Energy Disaggregation, Non-Intrusive Load Monitoring, Deep Neural Network, Machine Learning, Semi-supervised Learning.**

I. Introduction

Nowadays, energy saving is a great challenging problem, with energy demands increasing exponentially. Numerous researchers are trying to find useful solutions to addressing this problem. In China, residential users consume approximately 13.04% of the total electrical energy usage (roughly 756 billion kWh per year). Therefore, the residential part of energy saving will have a significant impact on the overall energy demand reduction. Many researchers think that real-time feedback would be a very useful mechanism, but current electricity metering and billing infrastructure cannot solve this task. We need to monitor real-time appliance consumption and provide real-time actionable feedback to consumers. Through such feedback, households can know what appliances, when and how much they use [1]. As a result, households would be allowed to actively plan their energy use to reduce their monthly energy bills. According to the study, energy efficiency awareness combined with this kind of feedback can trigger positive behavior changes [2] and lead to 10-15 percent energy savings [3].

Developing demand side management and demand response strategies requires concrete information on appliances in operation. Traditionally, real-time appliance energy consumption is obtained by deploying sensors at an appliance level granularity, which is called Intrusive Load Monitoring method (ILM). While the ILM approach is accurate, deploying such a sensing infrastructure is expensive and complex. Taking privacy and cost into consideration, ILM methods cannot be widely adopted. Another approach is to analyse the aggregate household mains power consumption in the house and disaggregate this into the load of individual devices, which is called Non-Intrusive Load Monitoring (NILM) method.

The concept of NILM appeared in the research literature in the 1980's [4]. Since then, many NILM algorithms have been proposed, which improve the performance of initial model and adapt to the advancement of sensor technology, generally requiring a sampling rate of kHz or MHz. With the popularity and large scale deployments of smart meters, more and more researchers have been interested in NILM algorithms that work at a lower sampling rate of seconds or minutes. Many factors drive the widespread deployment of low-sampling smart meters, including the cost of sensors, computation and storage. However, there are no widely available NILM methods with high precision and low complexity at a low sampling rate.

Over the last decade, there is a growing number of datasets developed specifically for energy disaggregation issues. Thus a wide variety of supervised methods were applied to the problem, such as Hidden Markov Model (HMM), Graph Signal Processing (GSP) and so on. In addition, with the big success of Deep Neural Network (DNN) in image classification, whether Deep Neural Network (DNN) can be applied to addressing energy disaggregation problem has caused a strong discussion.

Inspired by this, in this paper, we propose a new low-rate Deep Neural Network (DNN)-based NILM approach, which combines sequence-to-point neural network architecture and dilated convolutional module. At the same time, aware of the value of a large amount of unlabelled data, we try to combine DNN architecture and semi-supervised training method to improve the performance.

The contributions of this paper can be summarized as: (1) We propose a new sequence-to-point neural network architecture to estimate the ON/OFF state of appliance. (2) We employ two training methods, namely fully supervised and semi-supervised learning approach, to develop our model and demonstrate that semi-supervised training method is more accurate and feasible in realistic situations. (3) We compare our model with two state-of-the-art NILM methods, and numerical results verify that our model results in superior performance at most cases.

The rest of the paper is structured as follows. We review related work briefly in Section II, and give a fair overview of our proposed DNN-based approach in Section III. We discuss experimental results on the REDD dataset in section IV. In Section V, conclusion and future work are discussed.

II. Related Works

Research on NILM started with the work of Hart [4] in the 1980s. Hart proposed a method which only examined the appliance specific power signatures. There are three groups of signatures for home appliances [4]: single-state, continuously varying and multi-state. Fig. 1 shows several examples of each group of signatures for home appliances. After Hart's work, a number of research projects have emerged in NILM to meet practical challenges. NILM is a very important part of Smart Grid technology now.

The main differences between NILM methods lie in models and features used for energy disaggregation. According to features, NILM methods can be broadly grouped into two categories: transient-state methods and steady-state methods. Transient-state methods use transient signatures, including transient power [5], shape of transient waveform [6], high frequency voltage noise [7] and so on. Steady-state methods use features extracted under steady-state operation of electrical appliances, e.g., active power [8], reactive power, voltage and current waveform [9]. Transient-state methods require a sampling rate of kHz or MHz, thus provide more obvious features than steady-state methods, which leads to higher accuracy. Also some non-traditional signatures are usually combined with traditional features to improve the performance of learning algorithms.

There are many inductive learning algorithms employed to NILM. Learning algorithms are supervised, semi-supervised or unsupervised methods. Supervised learning methods need class labels to build a prediction model. We can further divide supervised algorithms into parametric models and non-parametric methods. The parametric model assumes a probability distribution a priori, and determines the best matching parameters between the model and the training data. Non-parametric models make no assumption about the overall distribution of the data and directly analyse the samples. Non-parametric methods include k-nearest neighbor (KNN) [10] and some neural network approaches [11]. Deep Neural Networks (DNN) [12], Hidden Markov Model (HMM) and its variants [13], Graph Signal Processing (GSP) [14] were also

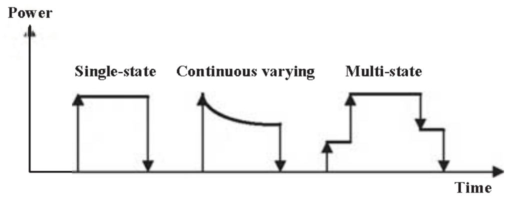


Fig. 1. Different types of appliances.

examined. On the contrary, unsupervised learning algorithms discover the regularity of data and group them according to their common features. The main benefit of unsupervised approaches is the fact that they do not require a ground truth. However, unsupervised methods usually achieve less accuracy than supervised methods. Clustering [15] and bayesian inference [16] were used to build databases for classification.

In realistic situations, labelled data are difficult to obtain. Unsupervised learning methods are obviously more suitable, but it is better to make use of the great accuracy of supervised learning. There is a bridge between them. Semi-supervised learning refers to using a small number of labelled data to improve the performance of large unlabelled data sets. Li et al. explored this idea for Hidden Markov Models in [17]. Wong et al. associated correlated wide side-channel information with actual labels for contextual supervision learning [18].

III. DNN-based NILM Approach

A. Problem Formulation

The goal of energy disaggregation is to recover the energy consumption of different individual appliances from the total power usage in the household. Suppose that we have observed the aggregate active power x_i in a household at time t_i. Let $\mathcal{M}$ be the set of all known appliances in the household. For appliance a, its power usage at time t_i is denoted by p_i^a.

The aggregate power x_i is assumed to be the sum of each individual appliance and some Gaussian noise ϵ_i. We are only interested in appliances in $\mathcal{M}$. The other appliances can be regarded as an unknown factor u_i. Then the full model can be represented in (1):

$$x_i = \sum_{a \in \mathcal{M}} p_i^a + u_i + \epsilon_i \tag{1}$$

In order to solve the load disaggregation problem, we construct a neural network F that maps sliding windows $X_{i:i+W}$ of the input mains power to the corresponding midpoint $y^a_{i+\frac{W}{2}}$ of the output state of appliance a. That is,

$$y^a_{i+\frac{W}{2}} = F(X_{i:i+W}) \tag{2}$$

For training, the task is to estimate $y^a_{i+\frac{W}{2}}$ the state of appliance a (ON/OFF) at time $t_{i+\frac{W}{2}}$ through input mains power sequence $X_{i:i+W}$. First of all, we carry out some preprocessing on the raw data. Secondly, sequence-to-point neural network architecture and dilated convolutional module are combined to build the classification model. Thirdly, we employ fully supervised training method and semi-supervised training method to train the prediction model. At last, two state-of-the-art NILM methods have been used for comparison to evaluate the performance of our proposed approaches.

Define Thr_a as a power threshold for appliance a, which is set during preprocessing. If the appliance power usage is larger than Thr_a, we infer that the appliance is ON.

$$y_i^a = \begin{cases} 1, & for \quad p_i^a \geq Thr_a \\ 0, & for \quad p_i^a < Thr_a \end{cases} \tag{3}$$

Thr_a is usually set to be half of the mean power value of appliance a. We list power thresholds used for each appliance in Table I.

B. Network Architecture

The neural network architecture used is shown in Fig. 2. All of the convolutional layers except the last two layers have a filter width of 3, as this is the smallest filter that can consider past, present and future. The initial convolutional layer is followed by a series of 3 dilated convolutional layers. Yu et al. [19] developed a new convolutional network module for dense prediction, which first used dilated convolutions to aggregate multi-scale contextual information. We use dilated convolutions in our neural network architecture in order to exponentially expand the receptive field of the network without the loss of resolution or coverage. Let F_i be discrete functions, which refer to convolutional layers, where $i = 1, 2, 3...n$, n is the number of convolutional layers. Let F_0 be input data. Define the receptive field of an element p in F_i as the set of elements in F_0 that modify the value of $F_i(p)$. It is easy to see that the receptive field of the initial convolutional layer is 3, which is same as 1-dilated convolution. A 2-dilated convolutional layer follows the initial convolutional layer, and each element in it has a receptive field of 7. The receptive field is an exponentially increasing size. This is illustrated in Fig. 3. Dilated convolutional layers we used in our neural network model in turn have a dilation rate of 2, 4 and 8. Following the dilated convolutional layers, a convolutional layer with 40 filters is used to further refine prior layers. The last convolutional layer has only one filter in order to reduce the output of the network to a single channel. Fully connected layer is followed to get the classification result.

C. Model Training

We employ two training methods for our proposed network model, namely fully supervised method and semi-supervised method.

TABLE I
POWER THRESHOLDS FOR DIFFERENT APPLIANCES.

Appliance	Power Thresholds (W)
MW	771
WD	1424
DW	257
ST	731
REFR	96

The loss of fully supervised training method is negative log-likelihood:

$$\overline{y_i^a} = F(X^i) \tag{4}$$

$$loss_{ful} = \sum_i^M -log\overline{y_i^a}[y_i^a] \tag{5}$$

where X^i is the input mains power window to predict the state of appliances at time t_i, y_i^a is the state of appliance a at time t_i and $\overline{y_i^a}$ is the predicted label of appliance a at time t_i. 0 means OFF and 1 for ON. M is the batch size, and we use a batch size of 16 to train our model. We construct an independent neural network for every appliance. That is to say, we construct 5 networks in total.

Miyato et al. [20] proposed a new regularization method that trains the output distribution to be isotropically smooth around each input data. The notion of virtual adversarial direction was introduced to quantify the idea. Virtual adversarial direction refers to the direction of the perturbation that can most greatly change the output distribution from the perspective of distributional divergence. Unlike adversarial direction introduced by Goodfellow et al. [21], virtual adversarial direction can be defined on an unlabelled data point. With the definition of virtual adversarial direction, Miyato et al. proposed a novel training method called virtual adversarial training (VAT), which uses an efficient approximation for the purpose of maximizing the likelihood of the model while promoting the model's local distributional smoothness (LDS) on each training input data point. Following the approach used by Miyato et al. [20], we apply virtual adversarial training (VAT) to semi-supervised learning task on REDD dataset. VAT is a training method with the regularizer $\mathcal{R}_{vadv}$. The regularization term $\mathcal{R}_{vadv}$ is the average of local distributional smoothness (LDS) over all input data points. Algorithm 1 summarizes the procedure we use for mini-batch stochastic gradient descent (SGD) with semi-supervised training method. VAT updates the model by the weighted sum of the gradient of the likelihood and the gradient of $\mathcal{R}_{vadv}$ computed with Algorithm 1.

$$\mathcal{R}_{vadv}^i = D[p(\overline{y_i^a}|X^i), p(\overline{y_i^a}|X^i + r_{vadv}^i)] \tag{6}$$

where D is Kullback-Leibler (KL) divergence and θ are the parameters of the neural network F. KL divergence is a natural way to measure the difference between two probability distributions. Then, the loss of semi-supervised training method is defined as:

$$loss_{semi} = \sum_i -log\overline{y_i^a}[y_i^a] + D[p(\overline{y_i^a}|X^i), p(\overline{y_i^a}|X^i + r_{vadv}^i)] \tag{7}$$

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Datasets

We evaluate our proposed DNN-based model on the REDD [22] dataset. REDD dataset is collected for energy disaggregation research. It is freely available, containing several weeks of active power data for 6 different houses and high-frequency

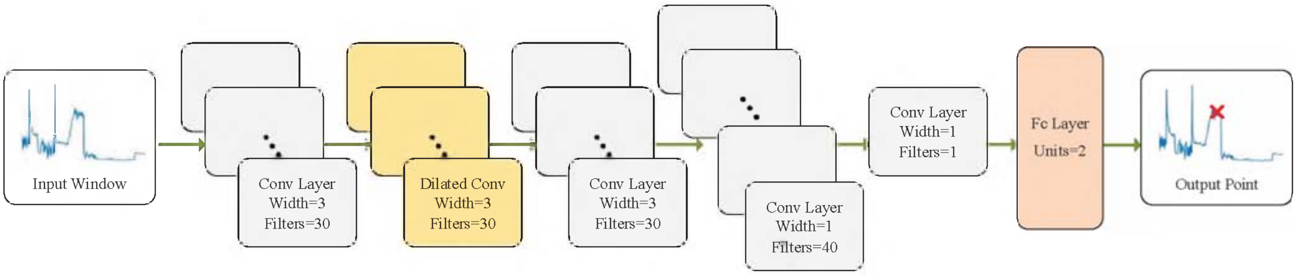


Fig. 2. Neural network architecture we used in load disaggregation problem.

Algorithm 1 Mini-batch SGD for semi-supervised training method.

1) Choose M samples of $X^i (i = 1, ..., M)$ from dataset $\mathcal{D}$ at random.

2) Generate a random unit vector d^i using an idd Gaussian distribution.

3) Calculate r^i_{vadv} via taking the gradient of D with respect to r^i on $r^i = \xi d^i$ on each input mains power window X^i:

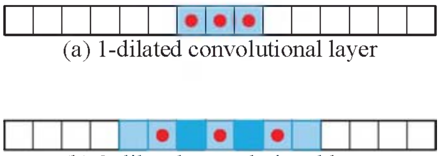
$g^i = \nabla_{r^i} D[p(\overline{y^a_i}|X^i), p(\overline{y^a_i}|X^i + r^i)]$

$r^i_{vadv} = g^i / ||g^i||_2$

4) Calculate $\mathcal{R}^i_{vadv}$:

$\mathcal{R}^i_{vadv} = D[p(\overline{y^a_i}|X^i), p(\overline{y^a_i}|X^i + r^i_{vadv})$

5) Return $\nabla_\theta(\sum_i^M -log\overline{y^a_i}[y^a_i] + \mathcal{R}^i_{vadv})$.



(a) 1-dilated convolutional layer

(b) 2-dilated convolutional layer

Fig. 3. Dilated convolutions support exponential expansion of the receptive field. (a) F_1 is produced from F_0 by a 1-dilated convolution; each element in F_1 has a receptive field of 3. (b) F_2 is produced from F_1 by a 2-dilated convolution; each element in F_2 has a receptive field of 7.

current data for the main power. We are only interested in low-sample rate data, that is to say, active power readings. Low-sample rate REDD dataset contains average power readings for mains and the individual circuits of the house. The data is logged at a frequency of about once a second for a mains and once every three seconds for the circuits. There are nearly twenty types of different appliances readings recorded. We pick some common appliances, which have adequate data to train and test. In this experiment, we use low-sample rate data of the five types of appliances: Microwave (MW), Washer_dryer (WD), Dishwaser (DW), Stove (ST) and Refrigerator (REFR). Each appliance signature can be seen in Fig. 4.

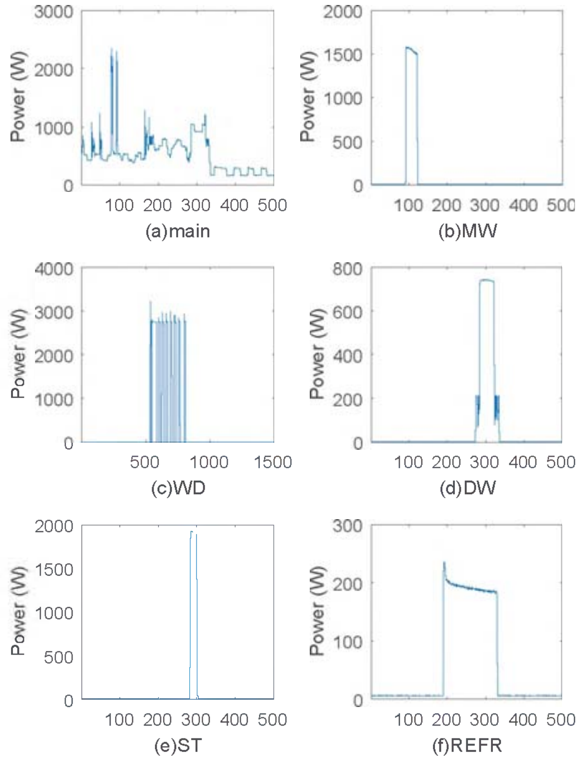


Fig. 4. Signatures of the five types of appliance in REDD.

As shown in Table II, the house ID for different appliances in training and test sets are listed. We divide the full data set into 3 parts. We think of 40% data as labelled training data, another 40% data as unlabelled training data, and the left 20% data as test sets.

TABLE II
THE HOUSE ID FOR DIFFERENT APPLIANCES IN TRAINING AND TEST SETS.

Appliance	House ID
MW	1, 2, 3, 5
WD	1, 2, 3, 4, 5, 6
DW	1, 2, 3, 4, 5, 6
ST	1, 2, 4, 6
REFR	1, 2, 3, 6

B. Preprocessing

Before training and evaluating our proposed model, we carry out some preprocessing on the raw data. The data is cleaned and resampled as described below.

To reduce the amount of data, readings are only recorded from monitors when the power usage changes. Because of this, gaps less than or equal to one hour are assumed that there are no changes and forward filled. There are some longer gaps of missing data between readings due to signal propagation issues.

Gaps longer than one hour are assumed to be due to missing data and are removed from training. Before training and testing, all sensor readings are resampled to a sampling interval as Table III.

Following this, the data is standardized to be distributed normally, as neural networks tend to learn more efficiently if the input is scaled to have zero mean and unit standard deviation.

Similar to the way used by Kelly and Knottenbelt [12], we divide each input sequence by its own mean, and divide it by the standard deviation of all inputs from the training set.

A window of the mains power is used as input sequence. The window length is set to 79 to ensure enough receptive field of each filter. The training windows are obtained by sliding the mains readings (input) and appliance state (output) one timestep once.

Instead of training a network to predict a window of appliance states, we propose to train a neural network to only predict the state of midpoint of the input window. This makes the prediction problem easier and hopefully yields more accurate results, as the neural network is allowed to focus its representational power on the midpoint of the window. Given a full mains power sequence $X = (x_1, ..., x_T)$, we first pad the sequence with $[W/2]$ zeros at the beginning and the end to deal with the endpoints of the sequence, where T is the full sequence length and W is the input window length.

One of the advantages of our model is that there is only a single prediction for every time point, rather than a weighted sum of predictions for each output window. The output points do not overlap and the input windows are partially overlapped.

C. Training

We use Tensorflow [23] to develop our model and train the model by Adam optimizer [24] with a learning rate of 1×10^{-4} and a batch size of 16. Training is terminated after 151 epochs.

We use 40% labelled data to train fully supervised model and 40% more unlabelled data to train semi-supervised model. The last 20% data is left for testing.

D. Performance Evaluation

We use F_M metric, which is commonly used in NILM models, to compare fully supervised training approach with semi-supervised training approach and evaluate the performance of our proposed neural network model.

F-Measure(F_M) is defined as below:

$$PR = TP/(TP + FP) \tag{8}$$

TABLE III
SAMPLING INTERVALS OF DIFFERENT APPLIANCES.

Appliance	Sampling Interval (Second)
MW	6
WD	6
DW	60
ST	6
REFR	6

$$RE = TP/(TP + FN) \tag{9}$$

$$F_M = 2 * (PR * RE)/(PR + RE) \tag{10}$$

where TP is true positive, which refers to the amount of test examples whose predicted label is ON(1) and actual state of the appliance is ON(1), FP is false positive, which refers to the amount of test examples whose predicted label is ON(1) and actual state of the appliance is OFF(0), and FN is false negative, which refers to the amount of test examples whose predicted label is OFF(0) and actual state of the appliance is ON(1). PR is precision, which captures the correctness of classification. RE is recall, which means the percentage of correct classification of test examples whose actual state of the appliance is ON(1). F_M balances PR and RE.

E. Results

In this subsection, we compare two training approaches for our proposed neural network model by F_M metric: (1) fully supervised training method (Solution 1); (2) semi-supervised training method (Solution 2). We use 20% samples to test our model. As shown in Table IV, Solution 2 leads to better F_M performance for all the five appliances than Solution 1. It demonstrates that semi-supervised training method provides better performance than fully supervised training method while only needing some more unlabelled training data. Thus, we can infer that semi-supervised training approach is more practical than fully supervised training method in real situations.

We compare the performance of Solution 2 (semi-supervised training method) with two state-of-the-art NILM approaches, namely HMM-based approach [13] and GSP-based approach [14], which is shown in Table V for REDD dataset.

The experiment results of HMM-based approach and GSP-based approach are reused from [14]. Because we need a lot of data to train and test our neural network, we use data in all houses (1~6) from REDD dataset. We average the results of HMM-based approach and GSP-based approach in house 1, 2 and 6 from [14] and compare it with our result (Solution 2).

Solution 2 outperforms the other two methods in most cases except the refrigerator, which shows the superiority of our proposed DNN-based approach. Solution 2 classifies more accurately than GSP-based method for all the five types of appliances and HMM-based method for MW, WD, DW and ST. For REFR, Solution 2 equals HMM-based method, which usually works well for REFR due to continuous operation. The results for REDD dataset demonstrate the competitive performance of our proposed DNN-based NILM method.

TABLE IV
COMPARISON BETWEEN SOLUTION 1 AND SOLUTION 2 FOR REDD.

Appliance	MW	WD	DW	ST	REFR
Solution 1	0.90	0.93	0.83	0.91	0.86
Solution 2	0.96	0.95	0.85	0.97	0.91

TABLE V
COMPARISON BETWEEN SOLUTION 2, HMM-BASED AND GSP-BASED METHODS FOR REDD.

Appliance	MW	WD	DW	ST	REFR
Solution 2	0.96	0.95	0.85	0.97	0.91
HMM	0.32	0.00	0.08	0.10	0.91
GSP	0.84	0.89	0.70	0.89	0.83

V. CONCLUSION AND FUTURE WORK

In this paper, we presented a new sequence-to-point neural network model for low-rate NILM problem, which combines sequence-to-point neural network architecture and dilated convolutional module. We employed two training methods for our proposed model, namely fully supervised training method and semi-supervised training method, in order to make better use of unlabelled data and improve the performance of classification. And we found that semi-supervised training method achieved superior accuracy while only requiring some more unlabelled training data. This result demonstrated the significance of semi-supervised methods in realistic situations. In addition, we compared the experimental results of this network model with two existing state-of-the-art approaches, namely HMM-based approach and GSP-based method, and found that our proposed model performed better by F_M metric at most cases, which showed the competitiveness of the new low-rate NILM method we presented.

In this topic, future work will include further exploration of multi-label classifiers for NILM. In particular, we will investigate both supervised multi-label classifiers and semi-supervised multi-label classifiers, which are new in NILM.

ACKNOWLEDGMENT

This work is supported in part by the National Natural Science Foundation of China under Grant 61936014 and 61901302, the National Key Research and Development Project under Grant 2019YFB2102300, 2019YFB2102301 and 2017YFE0119300, and the Fundamental Research Funds for the Central Universities No. 0800219369.

REFERENCES

[1] R. Dong, L. Ratliff, H. Ohlsson, and S. S. Sastry, "A dynamical systems approach to energy disaggregation," in *Proc. IEEE Conference on Decision and Control*, Florence, Italy, Dec. 2013, pp. 6335–6340.

[2] N. Batra, A. Singh, and K. Whitehouse, "If you measure it, can you improve it? exploring the value of energy disaggregation," in *Proc. ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, Seoul, South Korea, Nov. 2015, pp. 191–200.

[3] S. Darby *et al.*, "The effectiveness of feedback on energy consumption," *A Review for DEFRA of the Literature on Metering, Billing and Direct Displays*, vol. 486, Jan. 2006.

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

[5] S. R. Shaw, S. B. Leeb, L. K. Norford, and R. W. Cox, "Nonintrusive load monitoring and diagnostics in power systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 7, pp. 1445–1454, 2008.

[6] Y.-H. Lin and M.-S. Tsai, "Non-intrusive load monitoring by novel neuro-fuzzy classification considering uncertainties," *IEEE Transactions on Smart Grid*, vol. 5, no. 5, pp. 2376–2384, 2014.

[7] M. Hazas, A. Friday, and J. Scott, "Look back before leaping forward: Four decades of domestic energy inquiry," *IEEE Pervasive Computing*, vol. 10, no. 1, pp. 13–19, 2011.

[8] C. Dinesh, B. W. Nettasinghe, R. I. Godaliyadda, M. P. B. Ekanayake, J. Ekanayake, and J. V. Wijayakulasooriya, "Residential appliance identification based on spectral information of low frequency smart meter measurements," *IEEE Transactions on Smart Grid*, vol. 7, no. 6, pp. 2781–2792, 2016.

[9] J. Liang, S. K. Ng, G. Kendall, and J. W. Cheng, "Load signature studyłpart i: Basic concept, structure, and methodology," *IEEE Transactions on Power Delivery*, vol. 25, no. 2, pp. 551–560, 2010.

[10] K. Chahine, K. E. K. Drissi, C. Pasquier, K. Kerroum, C. Faure, T. Jouannet, and M. Michou, "Electric load disaggregation in smart metering using a novel feature extraction method and supervised classification," *Energy Procedia*, vol. 6, pp. 627–632, 2011.

[11] H.-H. Chang, K.-L. Lian, Y.-C. Su, and W.-J. Lee, "Energy spectrum-based wavelet transform for non-intrusive demand monitoring and load identification," in *Proc. IEEE Industry Applications Society Annual Meeting*, Lake Buena Vista, FL, USA, Oct. 2013, pp. 1–9.

[12] J. Kelly and W. Knottenbelt, "Neural nilm: Deep neural networks applied to energy disaggregation," in *Proc. ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, Seoul, South Korea, Nov. 2015, pp. 55–64.

[13] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "Non-intrusive load monitoring using prior models of general appliance types," in *Proc. AAAI conference on Artificial Intelligence*, Toronto, Ontario, Canada, Jul. 2012.

[14] K. He, L. Stankovic, J. Liao, and V. Stankovic, "Non-intrusive load disaggregation using graph signal processing," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 1739–1747, 2018.

[15] F. Jazizadeh, B. Becerik-Gerber, M. Berges, and L. Soibelman, "An unsupervised hierarchical clustering based heuristic algorithm for facilitated training of electricity consumption disaggregation systems," *Advanced Engineering Informatics*, vol. 28, no. 4, pp. 311–326, 2014.

[16] M. J. Johnson and A. S. Willsky, "Bayesian nonparametric hidden semi-markov models," *Journal of Machine Learning Research*, vol. 14, pp. 673–701, Feb. 2013.

[17] D. Li, K. Sawyer, and S. Dick, "Disaggregating household loads via semi-supervised multi-label classification," in *Proc. North American Fuzzy Information Processing Society*, Redmond, WA, USA, Aug. 2015, pp. 1–5.

[18] Y. F. Wong, T. Drummond, and Y. Şekercioğlu, "Real-time load disaggregation algorithm using particle-based distribution truncation with state occupancy model," *Electronics Letters*, vol. 50, no. 9, pp. 697–699, 2014.

[19] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. ICLR*, San Juan, Puerto Rico, 2016.

[20] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: a regularization method for supervised and semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1979–1993, 2018.

[21] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. ICLR*, San Diego, CA, USA, May. 2015.

[22] J. Z. Kolter and M. J. Johnson, "Redd: A public data set for energy disaggregation research," in *Proc. SustKDD Workshop on Data Mining Applications in Sustainability*, San Diego, CA, USA, Aug. 2011, pp. 59–62.

[23] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. USENIX conference on Operating Systems Design and Implementation*, Savannah, GA, USA, Nov. 2016, pp. 265–283.

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, San Diego CA, USA, May. 2015.