BERT4NILM: A Bidirectional Transformer Model for Non-Intrusive Load Monitoring

Zhenrui Yue
Technical University of Munich
zhenrui.yue@tum.de

Camilo Requena Witzig
Technical University of Munich
camilo.requena@tum.de

Daniel Jorde
Technical University of Munich
daniel.jorde@tum.de

Hans-Arno Jacobsen
Technical University of Munich
jacobsen@in.tum.de

ABSTRACT

Non-intrusive load monitoring (NILM) based energy disaggregation is the decomposition of a system's energy into the consumption of its individual appliances. Previous work on deep learning NILM algorithms has shown great potential in the field of energy management and smart grids. In this paper, we propose BERT4NILM, an architecture based on bidirectional encoder representations from transformers (BERT) and an improved objective function designed specifically for NILM learning. We adapt the bidirectional transformer architecture to the field of energy disaggregation and follow the pattern of sequence-to-sequence learning. With the improved loss function and masked training, BERT4NILM outperforms state-of-the-art models across various metrics on the two publicly available datasets UK-DALE and REDD.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**; • **Information systems** → **Information systems applications**.

KEYWORDS

NILM, Non-Intrusive Load Monitoring, Energy Disaggregation, Deep Learning, Neural Network, Transformer

ACM Reference Format:
Zhenrui Yue, Camilo Requena Witzig, Daniel Jorde, and Hans-Arno Jacobsen. 2020. BERT4NILM: A Bidirectional Transformer Model for Non-Intrusive Load Monitoring. In *The 5th International Workshop on Non-Intrusive Load Monitoring (NILM'20), November 18, 2020, Virtual Event, Japan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3427771.3429390

1 INTRODUCTION

Massive energy consumption contributes significantly to the emission of carbon dioxide and consequently to the current climate change phenomenon. The objective to reduce this energy consumption suggests that there is wastage in the use of electric power that could first be identified in order to be minimized [4]. One approach to better understand energy consumption is non-intrusive load monitoring (NILM) by G. W. Hart [8]. NILM algorithms gain information on a device's energy usage solely from the aggregated electricity consumption without the need for additional sensors. The energy-saving potential through this method is estimated to be about 5% to 12% [5].

The main contributions of this paper are the BERT4NILM architecture that utilizes self-attention for energy disaggregation, our proposed loss function which improves model performance as well as a sequence-to-sequence (seq2seq) benchmark evaluation for prediction and classification with state-of-the-art models running on two publicly available datasets, the REDD [16] and the UK-DALE dataset [11]. Based on BERT [3, 13], we release the BERT4NILM PyTorch implementation in a public repository for reproducibility[1]. To the best of our knowledge, this is the first work that applies transformer architecture with a novel loss function to NILM.

2 RELATED WORK

The separation problem of NILM has been studied since its introduction [8]. Most of the decomposition techniques are based on signal processing [24, 26], machine learning [6], variants of hidden Markov models (HMM) [15, 17] and recent deep learning methods. This paper focuses on deep learning NILM approaches, among which we find successful applications on power predictions with recurrent neural networks (RNN) [12, 19], denoising autoencoders (DAE) [12], convolutional neural networks (CNN) [23, 25] and generative adversarial nets (GAN) [18]. In general, researchers approach NILM either as a seq2seq, sequence-to-subsequence (seq2subseq) or sequence-to-point (seq2point) problem with tasks of single- or multi-label status classification and energy usage prediction.

Based on the mentioned architectures, attention mechanisms are gaining popularity in the field of NILM with convolutional and recurrent neural nets [2, 20]. In the recent development of natural language processing (NLP), the bidirectional transformer (BERT) is introduced for language understanding after the transformer architecture with self-attention [22]. Based on encoder representations of transformers, BERT adopts layers of multi-head self-attention and position-wise feed forward nets that map attention to the value matrix with query-key pairs and process the output through linear transformation and non-linear activation [3]. BERT is first pre-trained on large corpora and then fine-tuned for individual tasks, it

NILM'20, November 18, 2020, Virtual Event, Japan

ACM ISBN 978-1-4503-8191-8/20/11…$15.00
https://doi.org/10.1145/3427771.3429390

[1] https://github.com/Yueeeeeeee/BERT4NILM

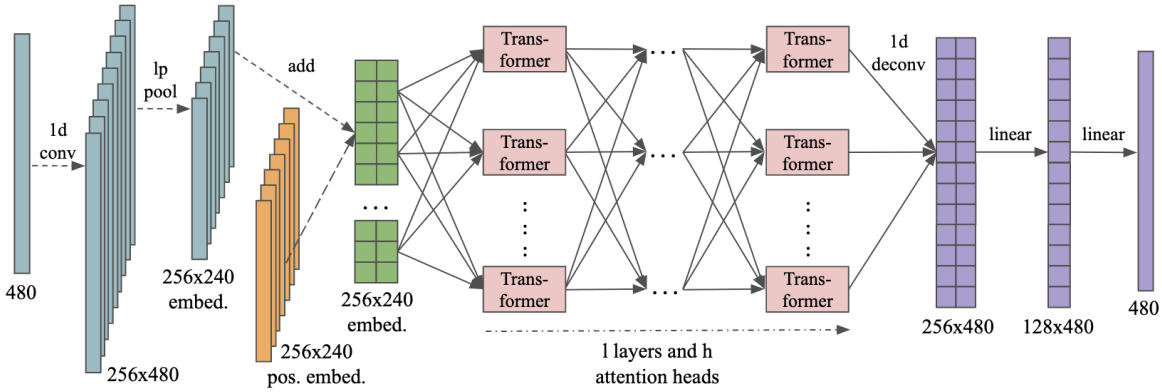

Figure 1: Architecture of BERT4NILM

achieves state-of-the-art performance in NLP and is also successfully adapted to different fields like recommender systems [10, 21].

3 METHODOLOGY

3.1 BERT4NILM Architecture

Based on BERT [3], the proposed architecture is depicted in Figure 1 and it contains an embedding module, transformer layers as well as a MLP output layer. The network takes sequential data with a fixed length and predicts individual energy usage with output of the same shape, while the appliance status is additionally computed by comparing the on-thresholds. Before passing the input data to transformer blocks, we first extract features and increase the hidden size of the one-dimensional input sequence by adopting a convolutional layer. The convolutional output with increased hidden size is then pooled by a learned L^2 norm pooling operation that applies squared-average pooling over the input sequence to preserve features while reducing the length by half [7]. The pooled input is then added to a learnable positional embedding matrix that captures sequence positional encoding, see Eq. 1.

$$Embedding(X) = LPPooling(Conv(X)) + E_{pose} \tag{1}$$

The final embedding matrix is fed to a bidirectional transformer that consists of l layers of transformers and h attention heads within each layer. The single-head self-attention (scaled dot-product attention) could be formulated with Q (Query), K (Key) and V (Value) matrices, obtained by linear transformation of the input matrix. Q and K are first multiplied and divided by the squared root of the hidden size, which is then processed by a softmax operation to construct soft attention before being multiplied with V and returns a weighted value matrix. Similarly, multi-head attention divides the hidden space into multiple subspaces with parameter matrices and performs the identical computation, resulting in multiple Q, K, V matrices. Each of them has an individual attention that can access information from different subspaces. Their results are concatenated and transformed to form the attentive output [22]:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2}$$

$$\begin{aligned} MultiHead(Q, K, V) &= Concat(head_1, head_2, ..., head_h)W^O \\ \text{where} \quad head_i &= Attention(QW_i^Q, KW_i^K, VW_i^V) \end{aligned} \tag{3}$$

The previous matrix is then additionally fed to a position-wise feed-forward network (PFFN, see Eq. 4) after the multi-head attention in each transformer layer. This layer processes input elements with linear transformations and GELU activation [9]. Note that after both attention module and feed-forward module, residual connections are applied to preserve input features, followed by layer normalization (LayerNorm) that stabilizes the hidden state dynamics between various layers [1], this operation can be formulated as: $LayerNorm(x + Dropout(Module(x)))$

$$PFFN(X) = GELU(0, XW_1 + b_1)W_2 + b_2 \tag{4}$$

After passing the values through transformer layers, the output MLP (Eq. 5) that includes a deconvolutional layer and two linear layers follows. The deconvolutional layer first expands the output to its original length with transposed convolution. Subsequently, a two-layer MLP with Tanh activation in between would restore the input hidden size to the desired output size. Output values (ideally between interval $[0, 1]$) are multiplied with the maximal device power and clamped to construct reasonable energy prediction, while appliance status is obtained additionally by matching corresponding on-thresholds:

$$Out(X) = Tanh(Deconv(X)W_1 + b_1)W_2 + b_2 \tag{5}$$

3.2 Objective Function

For accurate energy prediction and simultaneous status classification, we specifically developed a novel loss function for BERT4NILM in Eq. 6, where $\boldsymbol{x}, \hat{\boldsymbol{x}} \in [0, 1]$ represent the ground truth and prediction of power usage sequence divided by maximum power limit and $\boldsymbol{s}, \hat{\boldsymbol{s}} \in \{-1, 1\}$ are the appliance state label and prediction. T stands for total time steps (sequence length) and O refers to the set of time steps when either the status label is on or the prediction is incorrect. In this equation, we also introduce hyperparameters τ and λ for tuning sequence softmax temperature and reduction of absolute error.

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{s}) = \frac{1}{T}\sum_{i=1}^{T}(\hat{\boldsymbol{x}}_i - \boldsymbol{x}_i)^2 + D_{KL}(softmax(\frac{\hat{\boldsymbol{x}}}{\tau})||softmax(\frac{\boldsymbol{x}}{\tau})) + \frac{1}{T}\sum_{i=1}^{T} log(1 + exp(-\hat{\boldsymbol{s}}_i \boldsymbol{s}_i)) + \frac{\lambda}{T}\sum_{i \in O}|\hat{\boldsymbol{x}}_i - \boldsymbol{x}_i| \tag{6}$$

Based on a mean squared error (MSE) loss, the second term specifies KL divergence loss, where the tempered softmax results of output and ground truth are observed as probability distributions for evaluating the divergence between the prediction and label. As electrical appliances are turned off most of the time, we set the temperature parameter to be 0.1 to address the difference between on- and off-loads and improve model performance on error metrics for rarely used appliances like the kettle. To reduce misclassification at the same time, we introduce soft-margin loss to take status predictions into consideration and penalize inconsistent predictions and label status. Lastly, to close the gap between predicted energy values and ground truth, we adopt a L^1 term that provides backward gradients when the device is on or the state incorrectly classified, as we observed that energy predictions are often much lower compared to true label, even when the state is correctly classified. By applying various λ values depending on the scenario (see Table 1), we find the model performs better on most appliances, although λ might cause oscillating accuracy during training.

4 EXPERIMENTS

4.1 Dataset and Preprocessing

REDD was one of the first datasets to provide residential data and measured the power usage of six houses in the United States [16]. It consists of the electricity data of the main channel and of device-specific usages. We utilize the low frequency data and train on 4 specific appliances: refrigerator, washer and dryer, microwave and dishwasher. UK-DALE measures similar values from five homes in the United Kingdom and again, we focus on the low frequency data and the same appliances, we also include the kettle from UK-DALE along as an additional target [11].

The original data from both REDD and UK-DALE is sampled at least every six seconds on the aggregate channel and less frequent on individual appliances. Similar to the preprocessing from neural NILM [12], we align the timestamps of main channels and individual appliances, resample every six seconds and forward fill all time gaps shorter than three minutes. The raw data is clamped according to Table 1, which specifies the maximum power, on-threshold, minimum on- and off-duration of each appliance in our experiments. The status ground truth of each appliance is acquired by simple comparison between the data and the on-status thresholds, only status changes that last longer than the minimum on- and off-duration are considered valid. After that, we normalize the aggregated input sequence and reserve relevant statistics like mean and standard deviation for evaluation on test data.

4.2 Training

The masked training process is known as the masked language model (MLM) from the pre-training of BERT [3]. The input sequence is processed with random masking, where a proportion p of input elements is randomly masked with a special token and only output results from such positions are used to compute the loss. In this way, the model is forced to learn from context in order to predict the masked item, which enhances the model's capability to capture important patterns from the entire input sequence.

With fixed input length of 480 and $p = 0.25$ as masking portion, we initialize BERT4NILM with $l = 2$ layers of transformers, $h = 2$ attention heads and 256 as maximum hidden size using truncated normal distributions. The convolution operation has a kernel size of 5 and replicate padding length of 2 on both sides, while deconvolution has a kernel size of 4, stride of 2 and padding length of 1. The learned L^2 norm pooling layer is initialized using kernel size and stride of 2, with dropout rate chosen to be 0.1, BERT4NILM is trained with a learning rate of 10^{-4} and Adam for optimization, where betas of 0.9 and 0.999 and zero weight decay are adopted [14]. The house numbers for training, validation, and testing are shown in Table 2, test data is specifically chosen from entirely unseen data to test the model's generalization. Evaluation data is normalized with the mean and standard deviation from training data and passed to models without masking.

Table 1: Appliances for REDD (upper) and UK-DALE (lower)

Appliance	λ Value	Max. Limit	On-Thres.	Min. On-Duration	Min. Off-Duration
Fridge	10^{-6}	400	50	60s	12s
Washer	10^{-3}	500	20	1800s	160s
Micro.	1	1800	200	12s	30s
Dishw.	1	1200	10	1800s	1800s
Kettle	1	3100	2000	12s	0s
Fridge	10^{-6}	300	50W	60s	12s
Washer	10^{-2}	2500	20W	1800s	160s
Micro.	1	3000	200	12s	30s
Dishw.	1	2500	10	1800s	1800s

Table 2: Training and evaluation data by house number

Name	Train	Validation	Test
REDD	2, 3, 4, 5, 6;	2;	1;
UK-DALE	1, 3, 4, 5;	1;	2;

5 EVALUATION

For our experimentation, we adopt four widely used metrics in NILM research: accuracy, F1 score, mean relative error (MRE) and mean absolute error (MAE) [12, 19, 25]. To evaluate the BERT4NILM model, we adopted several state-of-the-art architectures in order to evaluate the improvement, including a regularized bidirectional GRU (GRU+) and LSTM (LSTM+) [19] as well as a seq2seq CNN (CNN) model [25]. Networks are modified for the identical input length and maximum hidden size and trained in a similar setting until convergence. They are tested on the unseen households of each dataset. The results in Tables 3 and 4 show the scores achieved

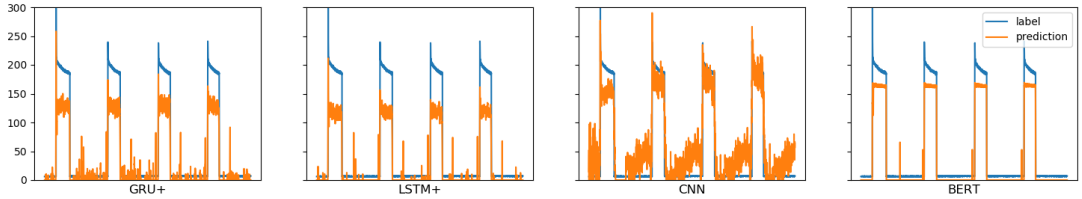

Figure 2: Sample output of refrigerator on REDD

Table 3: Model performances on REDD

Device	Model	Acc.	F1	MRE	MAE
Fridge	GRU+	0.794	0.705	0.829	44.28
	LSTM+	0.789	0.709	0.841	44.82
	CNN	0.796	0.689	0.822	35.69
	BERT	**0.841**	**0.756**	**0.806**	**32.35**
Washer	GRU+	0.922	0.216	0.090	**27.63**
	LSTM+	0.989	0.125	**0.020**	35.73
	CNN	0.970	0.274	0.042	36.12
	BERT	**0.991**	**0.559**	0.022	34.96
Micro-wave	GRU+	0.988	0.574	0.059	17.72
	LSTM+	**0.989**	**0.604**	0.058	**17.39**
	CNN	0.986	0.378	0.060	18.59
	BERT	0.989	0.476	**0.057**	17.58
Dish-washer	GRU+	0.955	0.034	0.042	25.29
	LSTM+	0.956	0.421	0.056	25.25
	CNN	0.953	0.298	0.053	25.29
	BERT	**0.969**	**0.523**	**0.039**	**20.49**
Average	GRU+	0.915	0.382	0.255	28.73
	LSTM+	0.933	0.465	0.244	30.80
	CNN	0.926	0.410	0.244	28.92
	BERT	**0.948**	**0.579**	**0.231**	**26.35**

on REDD and UK-DALE, Figure 2 presents sample output for the refrigerator on the REDD dataset by different models, where BERT appears to be more stable and precise while also manages to improve the prediction consistency compared to other models.

Table 3 shows the evaluation results of multiple models on the REDD dataset. The results of BERT4NILM outperform all other models on average and have advantages on most appliances. In Table 4, we see that BERT4NILM performs equally well and its scores are in line with the REDD dataset. Nevertheless, BERT4NILM has disadvantages in rarely used appliances that could be traced back to the masked training which reduces the model learning from device's on-state, suggested by relatively low F1 scores of the microwave in both datasets. It's likely that more training data and an improved masking strategy could further strengthen the performances for little-used appliances, given the model's complex training process. In our experiments, compared to the MSE loss, the proposed loss function helps to raise metrics on almost all appliances. With properly chosen λ, it significantly enhances performances on less often used electrical devices, such as the kettle. Considering the sampled output results and the complete performances, BERT4NILM provides consistent and accurate predictions and while other models might have better results in certain metrics, BERT4NILM performs the best in the overall comparison.

Table 4: Model performances on UK-DALE

Device	Model	Acc.	F1	MRE	MAE
Kettle	GRU+	0.993	0.425	0.008	23.22
	LSTM+	0.994	0.531	0.007	21.26
	CNN	0.997	0.850	0.003	9.64
	BERT	**0.998**	**0.907**	**0.002**	**6.82**
Fridge	GRU+	0.636	0.401	0.901	39.54
	LSTM+	0.573	0.174	0.956	43.74
	CNN	0.772	0.718	0.758	29.20
	BERT	**0.813**	**0.766**	**0.732**	**25.49**
Washer	GRU+	0.342	0.018	0.662	68.65
	LSTM+	0.938	0.150	0.067	15.66
	CNN	0.913	0.173	0.094	11.90
	BERT	**0.966**	**0.325**	**0.040**	**6.98**
Micro-wave	GRU+	**0.996**	0.266	0.014	6.41
	LSTM+	0.995	0.060	0.014	6.55
	CNN	0.995	**0.341**	**0.014**	**6.36**
	BERT	0.995	0.014	0.014	6.57
Dish-washer	GRU+	**0.977**	0.639	0.035	38.42
	LSTM+	0.976	0.605	**0.033**	36.36
	CNN	0.947	0.560	0.069	25.43
	BERT	0.966	**0.667**	0.049	**16.18**
Average	GRU+	0.789	0.350	0.324	35.25
	LSTM+	0.895	0.304	0.215	24.71
	CNN	0.925	0.528	0.188	16.51
	BERT	**0.948**	**0.536**	**0.167**	**12.41**

6 CONCLUSIONS AND FUTURE WORK

The self-attention mechanism and bidirectional transformer model is effective for NILM tasks, as we successfully manage to adapt the architecture for energy disaggregation. Based on the proposed loss function and masked training process, the proposed BERT4NILM architecture outperforms state-of-the-art models in most scenarios. Future work could focus on light-weight BERT model to accelerate training and inference as well as a more efficient optimization process that improves prediction quality on multi-staged appliances and unbalanced dataset.

REFERENCES

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).

[2] Kunjin Chen, Yu Zhang, Qin Wang, Jun Hu, Hang Fan, and Jinliang He. 2019. Scale-and Context-Aware Convolutional Non-Intrusive Load Monitoring. *IEEE Transactions on Power Systems* 35, 3 (2019), 2362–2373.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805

[4] K. Ehrhardt-Martinez, K. Donnelly, and J. Laitner. 2010. Advanced metering initiatives and residential feedback programs: a meta-review for household electricity-saving opportunities. *American Council for an Energy-Efficient Economy* (2010).

[5] Corinna Fischer. 2008. Feedback on household electricity consumption: A tool for saving energy. *Energy Efficiency* 1 (2008). https://doi.org/10.1007/s12053-008-9009-7

[6] F. Gong, N. Han, Y. Zhou, S. Chen, D. Li, and S. Tian. 2019. A SVM Optimized by Particle Swarm Optimization Approach to Load Disaggregation in Non-Intrusive Load Monitoring in Smart Homes. In *2019 IEEE 3rd Conference on Energy Internet and Energy System Integration (EI2)*.

[7] Caglar Gulcehre, Kyunghyun Cho, Razvan Pascanu, and Yoshua Bengio. 2014. Learned-norm pooling for deep feedforward and recurrent neural networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 530–546.

[8] George William Hart. 1992. Nonintrusive appliance load monitoring. *Proc. IEEE* 80, 12 (1992).

[9] Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415* (2016).

[10] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 197–206.

[11] Jack Kelly and William Knottenbelt. 2015. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Scientific Data* 2, 150007 (2015). https://doi.org/10.1038/sdata.2015.7

[12] Jack Kelly and William J. Knottenbelt. 2015. Neural NILM: Deep Neural Networks Applied to Energy Disaggregation. *CoRR* abs/1507.06594 (2015). arXiv:1507.06594

[13] Junseong Kim. 2018. BERT. https://github.com/codertimo/BERT-pytorch.

[14] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[15] J. Z. Kolter and T. Jaakkola. 2012. Approximate inference in additive factorial HMMs with application to energy disaggregation. *In AISTATS* (2012).

[16] J. Z. Kolter and Matthew J. Johnson. 2011. REDD: A Public Data Set for Energy Disaggregation Research. In *In SUSTKDD*.

[17] L. Mauch and B. Yang. 2016. A novel DNN-HMM-based approach for extracting single loads from aggregate power signals. In *2016 ICASSP*.

[18] Y. Pan, K. Liu, Z. Shen, X. Cai, and Z. Jia. 2020. Sequence-To-Subsequence Learning With Conditional Gan For Power Disaggregation. In *2020 ICASSP*.

[19] H. Rafiq, H. Zhang, H. Li, and M. K. Ochani. 2018. Regularized LSTM Based Deep Learning Model: First Step towards Real-Time Non-Intrusive Load Monitoring. In *2018 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*.

[20] Antonio Maria Sudoso and Veronica Piccialli. 2019. Non-Intrusive Load Monitoring with an Attention-based Deep Neural Network. *arXiv preprint arXiv:1912.00759* (2019).

[21] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *CIKM '19*.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*.

[23] Yandong Yang, Jing Zhong, Wei Li, T Aaron Gulliver, and Shufang Li. 2019. Semi-Supervised Multi-Label Deep Learning based Non-intrusive Load Monitoring in Smart Grids. *IEEE Transactions on Industrial Informatics* (2019).

[24] B. Zhang, S. Zhao, Q. Shi, and R. Zhang. 2019. Low-Rate Non-Intrusive Appliance Load Monitoring Based on Graph Signal Processing. In *2019 IEEE ICSPAC*.

[25] Chaoyun Zhang, Mingjun Zhong, Zongzuo Wang, Nigel H. Goddard, and Charles A. Sutton. 2018. Sequence-to-point learning with neural networks for nonintrusive load monitoring. In *AAAI*.

[26] B. Zhao, K. He, L. Stankovic, and V. Stankovic. 2018. Improving Event-Based Non-Intrusive Load Monitoring Using Graph Signal Processing. *IEEE Access* (2018).