Detection and Classification of Refrigeration Units in a Commercial Environment: Comparing Neural Networks to Unsupervised Clustering

Jérôme Van Zaen, Christopher Moufawad El Achkar, Rafael E. Carrillo, and Andreas Hutter

CSEM
Rue Jaquet-Droz 1
Neuchâtel, Switzerland
Email: {jerome.vanzaen, christopher.moufawadelachkar, rafael.carrillo, andreas.hutter}@csem.ch

***Abstract*—Non-intrusive load monitoring aims to estimate the power consumption of individual appliances from a single meter. Several methods have been proposed to solve this blind source separation problem, such as clustering, hidden Markov models, or neural networks. We present two approaches for detecting and classifying refrigeration units in a commercial environment. The first one is based on unsupervised event detection and the second one on neural networks with convolutional layers. We show that both approaches can accurately recognize power cycles of refrigeration units. The extracted cycles can then be used to reduce total energy consumption or for predictive maintenance by identifying units with an increased risk of failure.**

I. Introduction

Non-intrusive load monitoring (NILM) [1], also known as energy disaggregation, is a blind source separation problem where the goal is to estimate the power consumption of individual appliances from a single meter which measures the combined consumption of several appliances. Common use cases include helping users to reduce their energy consumption, detecting faulty appliances as part of predictive maintenance, or investigating appliance usage.

The methods that have been proposed for energy disaggregation can be broadly separated in two categories: methods for high-frequency NILM (sampling at 1 kHz or above) and methods for low-frequency NILM (sampling around 1 Hz or below). Techniques developed for energy data sampled above 1 kHz are typically based on features extracted from high order harmonics and provide relatively accurate disaggregation results [2]. However, the hardware required to monitor power consumption at such high frequencies is also more expensive. Tools to monitor aggregated power at around 1 Hz are much cheaper and can be quickly installed. But low-frequency NILM is also more challenging as less information is available due to the lower sampling rate. Depending on the characteristics of the monitored appliances, it is sometimes difficult to detect transitions between different power levels. For instance, refrigeration units typically show a very short spike with a large amplitude when transitioning from the inactive state to the active state. As the spike is usually shorter than 1 second it can be completely missed by the monitoring hardware. Nonetheless, numerous machine learning approaches have been proposed over the years to tackle low-frequency NILM [3]. The large majority of methods relied on hand-engineered features. Typical examples include clustering [1], integer programming [4], factorial hidden Markov models [5]–[8], sparse coding [9], [10], or multi-label classification [11]. Recently, after improving several image processing benchmarks, neural networks were applied for energy disaggregation. In particular, architectures such as recurrent neural networks [12] and autoencoders [13] were applied to this problem. Convolutional neural networks were also used to predict a single point from a sequence [14] and to classify appliances [15].

In this paper, we investigate and compare two approaches for energy disaggregation of refrigeration units: an unsupervised method based on event detection and an approach using neural networks to predict power consumption. We show that both approaches are applicable to detect and classify refrigeration units and estimate their power cycles. We also discuss their respective advantages and drawbacks.

II. Methods

A. Dataset

The dataset used in this study was obtained from a commercial sandwich shop. Five refrigeration units were operating simultaneously along with other devices during opening hours. Focusing mainly on refrigeration, only night-time signals were used ensuring only these units were seen in the aggregated power. A total of 37 nights, 8 hours each, were thus used. The aggregated power was obtained at 1 Hz from a mains clamp and reference power for three of the refrigeration units were obtained from individual power meters. The meters for the two other units were defective and recorded only noise. The monitored units consisted of a display cabinet, an under counter fridge, and an upright fridge. The 37 nights were split into a training set of 24 nights and a test set of 13 nights.

B. Unsupervised Event Detection

The algorithm flow chart is shown in Figure 1. The aggregated power signal is first filtered with a median filter of order 15 to remove noise and sharp peaks. An event detection is then performed using successive signal partitioning based

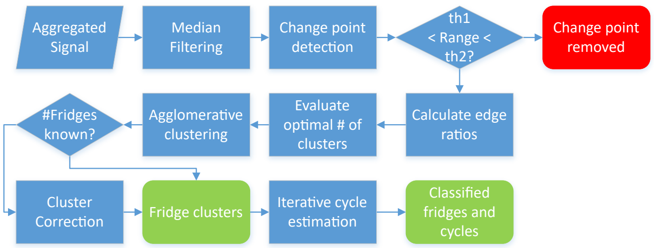

Fig. 1. Flowchart for unsupervised event detection.

on root mean squared residual error. When the residual stops improving by a certain threshold, the detection stops and the change points are returned. These points correspond to the start (positive change) and stop (negative change) of a refrigeration cycle. The magnitude of the change is computed and change points outside of the range specified by thresholds th1 and th2 are rejected. The magnitudes of all the rising edges are then divided by those of the falling edges to obtain the edge ratios. Based on this parameter, a Davies-Bouldin evaluation of optimal number of clusters is performed [16]. Briefly, this is a criterion that ensures a large inter-class and a low intra-class separation of clusters based on the distance measure (Euclidean in this case). Each rising edge is assigned to a cluster by agglomerative clustering, which is a bottom up hierarchical clustering technique. Euclidean distance was used with Ward linkage, providing minimum intra-cluster variance. If the number of refrigeration units is known and the number of clusters evaluated by the Davies-Bouldin criterion are not the same, the algorithm provides a cluster correction by merging points from extra clusters with the nearest cluster according to Euclidean distance. Finally, to complete the refrigeration cycles, an iterative search is performed starting from the last rising edge and backwards. This strategy looks at consecutive rising edges from the same cluster and finds the closest matching falling edge based on the edge ratios. This implicitly means that no refrigeration unit can start a cycle without having finished a previous one.

C. Neural Networks

We selected a feedforward neural network architecture similar to the one presented in [14]. Namely, the aggregated power signal is windowed and each window is used to estimate the consumption signal of a refrigeration unit at the center point. As a single output is estimated for each window, transitions and edges are better preserved compared to an approach estimating the whole window. However, it incurs a higher computational load. The window size is set to 4096 samples (corresponding to approximately 1 hour and 8 minutes) and successive windows are shifted by one sample. The neural network is composed of four one-dimensional convolutional layers followed by two dense layers with 1024 units. The ReLU activation is used in all layers. Each convolutional layer uses a kernel of size 5, halves the window size with max pooling of stride 2, and doubles the number of channels staring from 8 channels. Dropout is used during training before each dense layer with a rate of 0.5 [17]. A final dense linear layer outputs an estimate for the point in the middle of the input window. The full architecture is given in Table I.

TABLE I
NEURAL NETWORK ARCHITECTURE FOR PREDICTING THE POWER CONSUMPTION OF REFRIGERATION UNITS.

Layer	Output dimensions (samples $\times$ channels)
Input window	4096×1
Convolutional layer 1	2048×8
Convolutional layer 2	1024×16
Convolutional layer 3	512×32
Convolutional layer 4	256×64
Dense layer 1	1024×1
Dense layer 2	1024×1
Output linear layer	1×1

Before feeding the aggregated power signal to the neural network, it is filtered with a median filter of size 15 to remove spikes caused by refrigeration unit activations. This step is necessary as these spikes are sometimes missing due to the low sampling rate. In addition, both the aggregated power signal and the reference power consumption signal were centered and scaled with their mean and standard deviation computed over the training set to facilitate training.

Three sets of network parameters were trained for 50 epochs by minimizing the mean squared error (MSE) between the predicted consumption and the reference for the three refrigeration units. The minimization was performed using stochastic gradient descent (SGD) with a learning rate of 0.002, a momentum of 0.2, and a mini-batch size of 100. Furthermore, when the MSE did not improve for two consecutive epochs, the learning rate was divided by two. This resulted in three different neural networks sharing the same architecture for the three considered refrigeration units.

D. Validation

To validate the approach based on unsupervised event detection, each cluster is assigned a label based on the magnitude of the rising edge. Then, the clusters are compared to the reference refrigeration signals for every detected rising edge. A confusion matrix is built and the precision and recall are calculated for each refrigeration unit. In terms of detection, cycles are compared to the reference by checking the onset and end of a cycle for both aggregated power and reference power. If both the detected falling and rising edges fall within 10 seconds of the reference, the detection is validated. The result is reported as a detection accuracy value, which is the sum of all correctly detected cycles divided by the total number of detected cycles.

The neural network approach is validated similarly. However, as the network yields an estimated signal, it is necessary to extract cycles. Since the estimations include a significant amount of jitter, a total variation filter is applied to the

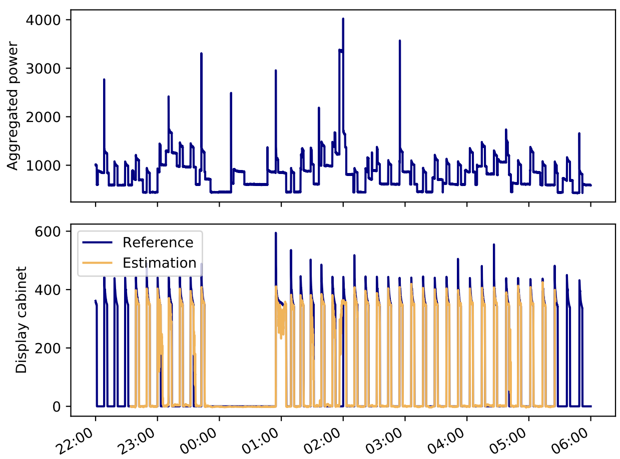

Fig. 2. Example of estimated power consumption for the display cabinet obtained with the neural network. The predicted signal is missing at the beginning and at the end of the night since our approach predicted only the center point of the window.

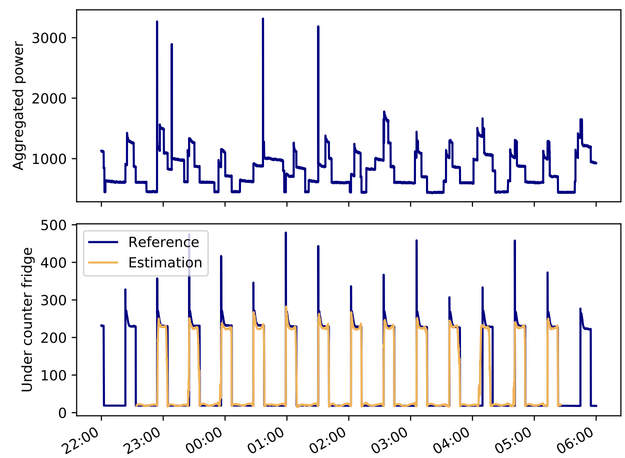

Fig. 3. Example of estimated power consumption for the under counter fridge obtained with the neural network. The same remark as for Figure 2 applies.

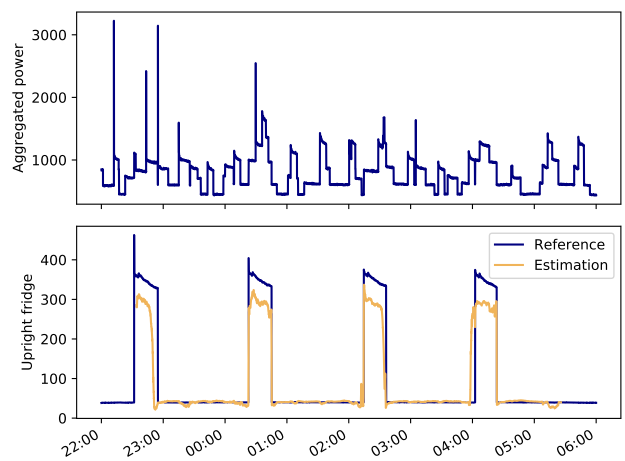

Fig. 4. Example of estimated power consumption for the upright fridge obtained with the neural network. The same remark as for Figure 2 applies.

output of the network to remove the noise. Then, the k-means algorithm is applied to obtain two clusters for the predicted values when the refrigeration unit is active and inactive. Taking the mean of the two cluster centers yields a threshold to detect cycles. Once the cycles are extracted from the predicted signal, they are compared to the reference as for unsupervised event detection. Since the 10-seconds window between detected and reference cycles was sometimes too restrictive for the neural network approach, we also computed the precision and recall for a 5-minutes windows. Furthermore, the performance of the neural network was also assessed in terms of mean absolute error and relative error in total energy (RETE) [12], i.e.,

$$RETE = \frac{|E - \hat{E}|}{\max(E, \hat{E})},$$

where E and $\hat{E}$ are the reference and predicted energy consumptions.

III. Results

First, typical power consumption from the three considered refrigeration units are illustrated in Figures 2, 3, and 4 for three nights from the test set. Each figure shows the aggregated power signal in the top plot and the reference power consumption in the bottom plot for one night. Furthermore, the predicted power consumption signals obtained with the neural networks for the three refrigeration units are shown alongside the references. The beginning and the end values are missing for the predictions as a complete window is required to predict the center point.

All three refrigeration units exhibited periodic cycles between active and inactive states. However, the cycle lengths and power consumption levels in the active state were different. In particular, the display cabinet had the shortest cylce length (Figure 2) and the upright fridge the longest (Figure 4). The predictions appear to be close to the references for the display cabinet and the under counter fridge. By contrast, the prediction for the upright fridge does not seem as accurate, particularly in terms of power consumption. Nonetheless, the cycles are still identified. The qualitative difference between the neural networks for the first two refrigeration units and the neural network for the last one might be linked to the long cycle length of the upright fridge. Indeed, the selected window size might be insufficient with respect to a cycle length of around 2 hours. Regardless, it was still possible to reliably extract cycles from the predictions obtained with the neural networks.

The detection accuracy of the approach based on unsupervised event detection for full refrigeration cycles was 94%. The mean precision and recall values by refrigeration unit are shown in Table II for both approaches. The unsupervised approach outperforms the neural network approach for the three refrigeration units with the default tolerance window of 10 seconds. In particular, the neural network for the upright

TABLE II
PRECISION AND RECALL OF EVENT DETECTION.

Refrigeration unit	Clustering		Neural network	
	Precision	Recall	Precision	Recall
Display cabinet	0.99	0.94	0.90 (0.99†)	0.90 (0.99†)
Under counter fridge	0.97	0.90	0.73 (0.97†)	0.74 (0.99†)
Upright fridge	0.77	0.92	0.14 (0.90†)	0.15 (0.93†)

† Values computed with a 5-minutes window instead of the 10-seconds window for matching reference and predicted cycles.

fridge only yielded a precision and recall around 15%. As the neural networks were not able to obtain accurate start and end times for the active states in some cases, the precision and recall were also computed with a longer tolerance window of 5 minutes. The resulting values are shown in parentheses in Table II for the neural networks. In this case, the results are much better as both precision and recall are above 97% for the display cabinet and the under counter fridge and above 90% for the upright fridge. In some applications such as predictive maintenance, the longer tolerance window might not be a problem as a 5-minutes error can be considered negligible with respect to a cycle length above 1 hour. Finally, the mean absolute errors of the neural networks were 11.15, 8.33, and 23.08 for the display cabinet, the under counter fridge, and the upright fridge and the RETE values were 0.059, 0.021, and 0.124.

IV. DISCUSSION

The two proposed approaches to detect and classify refrigeration units achieve good precision and recall on the dataset presented in this paper. Nonetheless, the approach based on neural network requires a longer tolerance window of 5 minutes. The start and end time of cycles are not estimated with sufficient accuracy to yield good precision and recall with a shorter window of 10 seconds. Although the reduced performance with shorter tolerance is an issue, it might not be a concern for specific applications. Indeed, an error of 5 minutes for a power cycle of 1 hour or more can be considered negligible. Furthermore, the dataset used in this study is relatively small with only 37 nights of 8 hours. The performance of neural networks has been shown to improve with additional training data in various tasks. We expect that a similar improvement will be observed for this task with more data. In particular, the neural network approach might be able to take into account interfering appliances active during daytime. It is also worth mentioning that we used the same input window size for all refrigeration units to have a general model. Tuning the window to a specific unit might also help to improve accuracy. In particular, the upright fridge presented above has very long cycles and the selected window size might be a suboptimal.

As mentioned above, the main limitation of the approach based on neural networks is the size of our dataset. But based on results obtained in other fields, we expect that more data will increase the detection accuracy of refrigeration units. We also need to train a network for each unit which can be time-consuming. Consequently, the approach based on unsupervised event detection is probably more appropriate for small datasets at the cost of power consumption prediction. Indeed, this approach can detect power cycles but not estimate the power consumed by a unit over time.

REFERENCES

[1] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.

[2] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford, and P. Armstrong, "Power signature analysis," *IEEE Power and Energy Magazine*, vol. 1, no. 2, pp. 56–63, 2003.

[3] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors*, vol. 12, no. 12, pp. 16 838–16 866, 2012.

[4] K. Suzuki, S. Inagaki, T. Suzuki, H. Nakamura, and K. Ito, "Nonintrusive appliance load monitoring based on integer programming," in *2008 SICE Annual Conference*, 2008, pp. 2742–2747.

[5] J. Z. Kolter and T. Jaakkola, "Approximate inference in additive factorial hmms with application to energy disaggregation," in *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, N. D. Lawrence and M. Girolami, Eds., vol. 22. La Palma, Canary Islands: PMLR, 21–23 Apr 2012, pp. 1472–1482.

[6] M. Zhong, N. Goddard, and C. Sutton, "Signal aggregate constraints in additive factorial HMMs, with application to energy disaggregation," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 3590–3598.

[7] ——, "Latent bayesian melding for integrating individual and population models," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 3618–3626.

[8] H. Lange and M. Bergés, "Efficient inference in dual-emission FHMM for energy disaggregation," in *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[9] S. Arberet and A. Hutter, "Non-intrusive load curve disaggregation using sparse decomposition with a translation-invariant boxcar dictionary," in *IEEE PES Innovative Smart Grid Technologies, Europe*, 2014, pp. 1–6.

[10] S. Singh, M. Gulati, and A. Majumdar, "Greedy deep disaggregating sparse coding," in *Proc. NILM2016 3rd International Workshop on Non-Intrusive Load Monitoring*, 2016.

[11] S. M. Tabatabaei, S. Dick, and W. Xu, "Toward non-intrusive load monitoring via multi-label classification," *IEEE Transactions on Smart Grid*, vol. 8, no. 1, pp. 26–40, 2017.

[12] J. Kelly and W. Knottenbelt, "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*, ser. BuildSys '15. New York, NY, USA: ACM, 2015, pp. 55–64.

[13] F. C. C. Garcia, C. M. C. Creayla, and E. Q. B. Macabebe, "Development of an intelligent system for smart home energy disaggregation using stacked denoising autoencoders," *Procedia Computer Science*, vol. 105, no. Supplement C, pp. 248–255, 2017.

[14] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for nonintrusive load monitoring," 2016.

[15] L. De Baets, J. Ruyssinck, C. Develder, T. Dhaene, and D. Deschrijver, "Appliance classification using VI trajectories and convolutional neural networks," *Energy and Buildings*, vol. 158, no. Supplement C, pp. 32–36, 2018.

[16] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 2, pp. 224–227, 1979.

[17] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.