Chapter 6

Binary versus Multiclass Deep Learning Modelling in Energy Disaggregation

Pascal A. Schirmer and Iosif Mporas

Abstract This paper compares two different deep-learning architectures for the use in energy disaggregation and Non-Intrusive Load Monitoring. Non-Intrusive Load Monitoring breaks down the aggregated energy consumption into individual appliance consumptions, thus detecting device operation. In detail, the “One versus All” approach, where one deep neural network per appliance is trained, and the “Multi-Output” approach, where the number of output nodes is equal to the number of appliances, are compared to each other. Evaluation is done on a state-of-the-art baseline system using standard performance measures and a set of publicly available datasets out of the REDD database.

Keywords Energy disaggregation · Non-Intrusive load monitoring · Deep-Learning · Deep neural networks

6.1 Introduction

Due to global warming average temperatures are rising and several techniques for energy reduction have been proposed in order to reduce the total energy consumption. However, to make use of those techniques accurate and fine-grained monitoring of electrical energy consumption is needed [1], since the energy consumption of most households is monitored via monthly aggregated measurements and thus cannot provide real-time feedback. Moreover, according to [2] the largest improvements in terms of energy savings can be made when monitoring energy consumption on device level. The term Non-Intrusive Load Monitoring (NILM) is used to describe the estimation of the power consumption of individual appliances, based on a single measurement on the inlet of a household or building [3]. In contrast to NILM, the term

P. A. Schirmer (✉) · I. Mporas
School of Engineering and Computer Science, University of Hertfordshire, Hatfield AL10 9AB,
UK
e-mail: p.schirmer@herts.ac.uk

I. Mporas
e-mail: i.mporas@herts.ac.uk

Intrusive Load Monitoring (ILM) is used when multiple sensors are used, usually one per device. ILM compared to NILM has the drawback of higher cost through wiring and data acquisition making it unsuitable for monitoring households where appliances can change. Conversely, NILM has the goal of finding the inverse of the aggregation function through a disaggregation algorithm using as input only the aggregated power consumption which makes it a highly underdetermined problem and thus impossible to solve analytically [4].

Several NILM methodologies based on deep neural networks have been proposed in the literature, e.g. Convolutional Neural Networks (CNNs) [5], Recurrent Neural Networks (RNNs) [6] and Long Short Time Memory (LSTM) [7]. Additionally, combinations of machine learning algorithms for fusion of information [8] and modelling of temporal dynamics [9] have also been proposed, especially for low sampling frequencies [10]. Particularly, these models operate either according to the “One versus All” approach, where one Deep Neural Network (DNN) per appliance is trained or the “Multi-Output” approach, where the number of output nodes is equal to the number of appliances. As it is not clear which architecture leads to better performances a comparison of these two architectures is needed.

The remainder of this paper is organized as follows: In Sect. 6.2 the two NILM systems based on DNNs are presented. In Sect. 6.3 the experimental setup is described and in Sect. 6.4 the evaluation results are presented. Finally, the paper is concluded in Sect. 6.5.

6.2 Proposed Architecture

NILM energy disaggregation can be formulated as the task of determining the power consumption on device level based on the measurements of one sensor, within a time window (frame or epoch). Specifically, for a set of $M - 1$ known devices each consuming power p_m , with $1 \leq m \leq M - 1$, the aggregated power p_{agg} measured by the sensor will be [11]:

$$p_{agg} = f(p_1, p_2, \dots, p_{M-1}, g) = \sum_{m=1}^{M-1} p_m + g = \sum_{m=1}^M p_m \quad (6.1)$$

where $g = p_M$ is a ‘ghost’ power consumption usually consumed by one or more unknown devices. In NILM the goal is to find estimations $\hat{p}_m, \hat{g}$ of the power consumption of each device m using an estimation method f^{-1} with minimal estimation error [11], i.e.

$$\begin{aligned} \hat{P} &= \{\hat{p}_1, \hat{p}_2, \dots, \hat{p}_{M-1}, \hat{g}\} = f^{-1}(g(p_{agg})) \\ \text{s.t. } \operatorname{argmin}_{f^{-1}} &\left\{ (p_{agg} - \sum_1^M \hat{p}_m)^2 \right\} \end{aligned} \quad (6.2)$$

Fig. 6.1 Block diagram of the NILM architecture, including the smart meter (data source), pre-processing, feature extraction and appliance detection

where p_{agg} is the aggregated power consumption, p_m the power consumption of m -th device with $p_g = p_M$ being the ‘ghost’ power consumption, $\hat{P} = \{\hat{P}_m, \hat{P}_g\}$ the estimates of the per device power consumptions, f^{-1} an estimation method and $g()$ a function transforming a time window of the aggregated power consumption into a multidimensional feature vector $F \in \mathbb{R}^N$. The block diagram of the NILM architecture adopted in the present evaluation is illustrated in Fig. 6.1 and consists of three stages, namely the pre-processing, feature extraction and appliance detection.

In detail, the aggregated power consumption signal calculated from a smart meter is initially pre-processed i.e. passed through a median filter [12] and then frame blocked in time frames. After pre-processing feature vectors, F of dimensionality N , one for each frame are calculated. In the appliance detection stage, the feature vectors are processed by a regression algorithm using a set of pre-trained appliance models to estimate the power consumption of each device. The output of the regression algorithm estimates the corresponding device consumption and a set of thresholds, T_m with $1 \leq m \leq M$ with $T_g = T_M$, for each device including the ghost device ($m = M$) is used to decide whether a device is switched on or off. In the present evaluation the estimation method is implemented using two different deep-learning architectures as shown in Fig. 6.2.

As can be seen in Fig. 6.2 the two architectures only differ in their number of output nodes with architecture (a) using a single output node and one DNN per device and architecture (b) using one output node per device and a single DNN for all devices.

Fig. 6.2 Two deep neural network architectures using **a** one node and **b** M nodes in the output layer

6.3 Experimental Setup

The NILM architecture presented in Sect. 6.2 was evaluated using several publicly available datasets and a deep neural network for regression.

a. Datasets

To evaluate performance five different datasets of the REDD [13] database were used. The REDD database was chosen as it contains power consumption measurements per device as well as the aggregated consumption. The REDD-5 dataset was excluded as its measurement duration is significantly shorter than the rest of the datasets in the REDD database [14]. The evaluated datasets and their characteristics are tabulated in Table 6.1 with the number of appliances denoted in the column #App. In the same column, the number of appliances in brackets is the number of appliances after excluding devices with power consumption below 25 W, which were added to the power of the ghost device, similarly to the experimental setup followed in [15]. The next three columns in Table 6.1 are listing the sampling period T_s , the duration T of the aggregated signal used and the appliance type for each evaluated dataset. The appliances type categorization is based on their operation as described in [11].

b. Pre-processing and Parameterization

During pre-processing the aggregated signal was processed by a median filter of 5 samples as proposed in [12] and then was frame blocked in frames of 10 samples with overlap between successive frames equal to 50% (i.e. 5 samples). Specifically, raw samples have been used at the input stage of the DNN, thus $F_{1,...,N}$ being the raw samples of each frame respectively. Furthermore, the number of hidden layers for each architecture was optimized using a bootstrap training dataset resulting into an architecture with 3 hidden layers and 32 sigmoid nodes for (a) and 2 hidden layers and 32 sigmoid nodes for (b).

Table 6.1 List of the evaluated datasets and their corresponding properties

Dataset	Properties			
	# App.	T_s	T	Appliance type
REDD-1	18 (17)	3 s	7d	One state/multi state/non-linear
REDD-2	9 (10)	3 s	11d	One state/multi state
REDD-3	20 (18)	3 s	14d	One state/multi state/non-linear
REDD-4	18 (16)	3 s	14d	One state/multi state/continuous/non-linear
REDD-6	15 (14)	3 s	12d	One state/multi state/continuous/non-linear

Table 6.2 Performance evaluation of 5 datasets for two deep learning architectures using estimation accuracy

Dataset	REDD-1 (%)	REDD-2 (%)	REDD-3 (%)	REDD-4 (%)	REDD-6 (%)	AVG (%)
One versus All	71.38	86.00	70.44	68.21	87.17	76.64
Multi-output	71.87	85.65	71.69	67.18	85.03	76.28

6.4 Experimental Results

The NILM architecture presented in Sect. 6.2 was evaluated according to the experimental setup described in Sect. 6.3. The performance was evaluated in terms of appliance power estimation accuracy (E_{ACC}), as proposed in [13] and defined in Eq. 6.3. The accuracy estimation is considering the estimated power $\hat{p}_m$ for each device m , where T is the number of frames and M is the number of disaggregated devices.

$$E_{ACC} = 1 - \frac{\sum_{t=1}^T \sum_{m=1}^M |\hat{p}_m^t - p_m^t|}{2 \sum_{t=1}^T \sum_{m=1}^M |p_m^t|} \quad (6.3)$$

To compare the two architectures the publicly available REDD database is used [13]. The results are tabulated in Table 6.2.

As can be seen in Table 6.2 the performance of datasets with smaller number of appliances (e.g. REDD-2/6) is significantly higher than for the datasets with higher number of appliances (e.g. REDD-1/3/4). Furthermore, the “One versus All” approach slightly outperforms the “Multi-Output” approach performing 0.36% better on average. However, it has to be mentioned that the “One versus All” approach requires the training of M deep neural networks resulting into significantly higher training times.

6.5 Conclusion

In this paper two different deep learning architectures for non-intrusive load monitoring were compared. Specifically, the “One versus All” approach using one deep neural network per appliance was compared to the “Multi-Output” approach using one deep neural network with the same number of output nodes than the number of appliances. It was shown, that both architectures have similar performance with average accuracies of 76.6% for the “One versus All” approach and 76.3% for the “Multi-Output” approach respectively. However, in terms of training time it must be considered, that for the “One versus All” approach M deep neural networks must be trained, resulting in significant higher training times.

Acknowledgements This work was supported by the UA Doctoral Training Alliance (<https://www.unialliance.ac.uk/>) for Energy in the United Kingdom.

References

1. A. Chis, J. Rajasekharan, J. Lunden, V. Koivunen, Demand response for renewable energy integration and load balancing in smart grid communities, in *2016 24th European Signal Processing Conference (EUSIPCO)*, (2016), pp. 1423–1427
2. M.N. Meziane, K. Abed-Meraim, Modeling and estimation of transient current signals, in *2015 23rd European Signal Processing Conference (EUSIPCO)*, (2015), pp. 1960–1964
3. G.W. Hart, Nonintrusive appliance load monitoring. *Proc. IEEE* **80**(12), 1870–1891 (1992). <https://doi.org/10.1109/5.192069>
4. M. Gaur, A. Majumdar, Disaggregating transform learning for non-intrusive load monitoring. *IEEE Access* **6**, 46256–46265 (2018). <https://doi.org/10.1109/ACCESS.2018.2850707>
5. A. Harell, S. Makonin, I.V. Bajic, Wavenilm: a causal neural network for power disaggregation from the complex power signal, in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (2019), pp. 8335–8339
6. P.A. Schirmer, I. Mporas, Energy Disaggregation Using Fractional Calculus, in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (Barcelona, Spain, 2020), Apr. 2020–Aug. 2020, pp. 3257–3261
7. M. Kaselimi, N. Doulamis, A. Doulamis, A. Voulodimos, E. Protopapadakis, Bayesian-optimized Bidirectional LSTM regression model for Non-intrusive Load monitoring, in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (2019), pp. 2747–2751
8. P.A. Schirmer, I. Mporas, A. Sheikh-Akbari, Energy disaggregation using two-stage fusion of binary device detectors. *Energies* **13**(9), 2148 (2020). <https://doi.org/10.3390/en13092148>
9. P.A. Schirmer, I. Mporas, A. Sheikh-Akbari, Robust energy disaggregation using appliance-specific temporal contextual information. *EURASIP J. Adv. Signal Process.* **2020**(1), 394 (2020). <https://doi.org/10.1186/s13634-020-0664-y>
10. P.A. Schirmer, I. Mporas, energy disaggregation from low sampling frequency measurements using multi-layer zero crossing rate, in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (Barcelona, Spain, 2020) Apr. 2020–Aug. 2020, pp. 3777–3781
11. P.A. Schirmer, I. Mporas, Statistical and electrical features evaluation for electrical appliances energy disaggregation. *Sustainability* **11**(11), 3222 (2019). <https://doi.org/10.3390/su11113222>
12. C. Beckel, W. Kleiminger, R. Cicchetti, T. Staake, S. Santini, The ECO data set and the performance of non-intrusive load monitoring algorithms, in *BuildSys'14: Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, (2014), pp. 80–89
13. J.Z. Kolter, M.J. Johnson, eds., *REDD: A Public Data Set for Energy Disaggregation Research*, (2011)
14. V. Andrean, X.-H. Zhao, D.F. Teshome, T.-D. Huang, K.-L. Lian, A hybrid method of cascade-filtering and committee decision mechanism for non-intrusive load monitoring. *IEEE Access* **6**, 41212–41223 (2018). <https://doi.org/10.1109/ACCESS.2018.2856278>
15. S.M. Tabatabaei, S. Dick, W. Xu, Toward non-intrusive load monitoring via multi-label classification. *IEEE Trans. Smart Grid* **8**(1), 26–40 (2017). <https://doi.org/10.1109/TSG.2016.2584581>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.