On the Feasibility of Generic Deep Disaggregation for Single-Load Extraction

Karim Said Barsim and Bin Yang
{karim.barsim,bin.yang}@iss.uni-stuttgart.de
Institute of Signal Processing and System Theory, University of Stuttgart

***Abstract*— Recently, and with the growing development of big energy datasets, data-driven learning techniques began to represent a potential solution to the energy disaggregation problem outperforming engineered and hand-crafted models. However, most proposed deep disaggregation models are load-dependent in the sense that either expert knowledge or a hyper-parameter optimization stage is required prior to training and deployment (normally for each load category) even upon acquisition and cleansing of aggregate and sub-metered data. In this paper, we present a feasibility study on the development of a generic disaggregation model based on data-driven learning. Specifically, we present a generic deep disaggregation model capable of achieving state-of-art performance in load monitoring for a variety of load categories. The developed model is evaluated on the publicly available UK-DALE dataset with a moderately low sampling frequency and various domestic loads.**

***Index Terms*— Energy/Load disaggregation, Non-Intrusive Load Monitoring (NILM), Convolutional Neural Networks (CNN), UNet, SegNet, UK-DALE**

I. Introduction

Energy disaggregation (or Non-Intrusive Load Monitoring NILM) is the process of inferring individual load profiles at the end-use level from a single or a limited number of sensing points. Promising applications of disaggregated data have motivated a growing research community to reach a widely acceptable and scalable solution. Energy disaggregation proved to be a challenging source separation problem in which a considerably large number of parameters are to be estimated from a limited set of measurements with little constraints.

In the last decade, energy disaggregation has witnessed an unprecedented wide-spreading research which is easily observed from the wide variety of learning techniques applied to this problem alongside with the growing number of energy datasets developed specifically for this research field. More recently, and analogous to the current breakthrough in data-driven learning, deep neural networks have re-gained their interest in addressing the energy disaggregation problem, especially alongside with the recently developed large energy datasets required for training such complex models [1–8]. The progress in this trend, however, is relatively slow when compared to the development in either field separately. This is sometimes attributed to the high risk of over-fitting in neural network models [9], insufficiency or low diversity of publicly available energy datasets [4], or limited insights and understanding of the learning behavior of these models [5].

In this paper, we first present a feasibility study on the development of a generic data-driven model suitable for end-use load monitoring. The proposed disaggregation model exploits a fully convolutional neural network architecture and is generic in the sense that none of the model hyper-parameters is dependent on the load category. We assess the feasibility of such a model through empirical evaluation of the monitoring performance across various load categories in a publicly available energy dataset.

II. Related work

In this section, we briefly describe some of the most recent works on energy disaggregation and load monitoring that adopted data-driven learning techniques.

Mauch and Yang [2] exploited a generic two-layer bidirectional Recurrent Neural Network (RNN) architecture featuring Long Short Term Memory (LSTM) [10] units in extracting single load profiles. They tested their models on the Refrence Energy Disaggregation Dataset (REDD) [11] in a de-noised scheme [37]. Additionally, they validated the generalization of their architecture to previously unseen loads in new buildings. In a later work, Mauch and Yang [3] used a combination of discriminative and generative models in a two-stage eventless extraction of load profiles. Kelly and Knottenbelt [4] evaluated and compared three neural network architectures on domestic loads from the UK-Domestic Appliance Level Energy (UK-DALE) [7]. The first is a bidirectional RNN architecture with LSTM units similar to the one in [2], the second follows the architecture of a de-noising Auto-Encoder (dAE) [12], and the last is a regression-based disaggregator whose objective is to estimate the main key points of an activation cycle of the target load within a given window.

Similarly, He and Chai [6] applied two architectures, namely a convolutional dAE and an RNN, to the same problem. In their architectures, they also applied parallel convolutional layers with different kernel sizes analogous to the Inception module in GoogLeNet [13]. Zhang et al. [5] simplified the objective of the dAE architecture in [4] to predict a single time instance of the target load profile for a given window of the aggregate signal. Likewise, Nascimento [14] applied three neural network architectures, namely basic convolutional dAE, an RNN, and a ResNet-based model [15] to the same problem but on three target loads in the REDD dataset. He introduced several improvements such as redefining the loss function, exploiting batch normalization [16], and applying residual connections [15].

Additionally, Lange et al. [17] adopted a deep neural network with constrained binary and linear activation units in the last two layers. Their first objective was to retrieve subcomponents of the input signal that sum up linearly to the aggregate active and reactive powers. Finally, they estimate the *on-off* activation vector of each load. Their approach, however, was applied on very high frequency current and voltage measurements (12 kHz) from the Building-Level fully labeled dataset for Electricity Disaggregation (BLUED) [18].

In many of these previous works [4–6, 14], each disaggregator is a neural network whose disaggregation window length (and consequently the width of subsequent layers) depends on the load being monitored. The disaggregation window of each load is manually adjusted in a per-load basis to fully capture a single activation cycle of the load. Moreover, the disaggregation performance widely differs amongst variant

load categories and a model that achieves remarkably well on one load might drastically fail for other load types.

III. Load Monitoring

In this work, we focus essentially on *single-load extraction of activation profiles*, of which we give a detailed description in the following.

A. Activation profiles: definition and estimation

In the simplest case, a load is modeled as a two-state machine which is assumed to be in the *on*-state whenever the load is consuming energy from the main power source, and in the *off*-state otherwise. Accordingly, load monitoring becomes a binary classification task. Note that in contrast to previous works, the consumption profile of a load during its *on*-state need not be a defined [19] nor a piecewise-defined function in time.

The desired signal (i.e. *ground truth*) of a load m in a window of N time instances is the binary-valued signal $\underline{\omega}^{(m)} \in \{0,1\}^N$ whose element $\omega^{(m)}(n)$ is set (i.e. to indicate an *on*-state) whenever the load is operating in one of its activation states at time instance n and unset otherwise. In this work, we refer to this signal as the *activation profile*. Applications that benefit from activation profiles include mainly activity monitoring and occupancy detection in which time-of-use information dominates the value of energy consumed.

We define the true activation profile of a load $\underline{\omega}^{(m)}$ via a threshold-based approach applied to the sub-metered real power signals and similar to the one used in [4] as follows. The sub-metered real-power $\underline{x}^{(m)}$ of a load m is compared against predefined thresholds to detect the operation intervals of the load. In order to avoid anomalies and false activations or deactivations, the load is assumed to be in an activation state (i.e. *on*) if its power draw $x^{(m)}(n)$ exceeds a given threshold $\mathcal{P}_{\text{on}}^{(m)}$ for a minimum period of time $\mathcal{N}_{\text{on}}^{(m)}$. Similarly, if the power draw drops below a predefined threshold $\mathcal{P}_{\text{off}}^{(m)}$ for a given period $\mathcal{N}_{\text{off}}^{(m)}$, the load is assumed to be disconnected. Otherwise, the load keeps its last observed state. Thus, the estimated activation profile is defined as

$$\omega^{(m)}(n) = \begin{cases} 1, & \text{if } x^{(m)}(k) \geqslant \mathcal{P}_{\text{on}}^{(m)}, \text{ for } n \leqslant k < n + \mathcal{N}_{\text{on}}^{(m)} \\ 0, & \text{if } x^{(m)}(k) \leqslant \mathcal{P}_{\text{off}}^{(m)}, \text{ for } n \leqslant k < n + \mathcal{N}_{\text{off}}^{(m)} \\ \omega^{(m)}(n-1), & \text{otherwise} \end{cases}$$

with the initial state assumed to be *off* (i.e. $\omega^{(m)}(0) = 0$) for all loads. Note that $\mathcal{P}_{\text{on}}^{(m)}$, $\mathcal{N}_{\text{on}}^{(m)}$, $\mathcal{P}_{\text{off}}^{(m)}$, and $\mathcal{N}_{\text{off}}^{(m)}$ are the only load-dependent parameters in this work, and they are used merely in estimating the ground truth signals. Values of these parameters are similar or close to those adopted in [4] and are listed in Table I for the sake of completeness.

TABLE I: Load-dependent parameters for estimating the activation profiles.

Load	$\mathcal{P}_{\text{on}} = \mathcal{P}_{\text{off}}$ [W]	$\mathcal{N}_{\text{on}}$ [min.]	$\mathcal{N}_{\text{off}}$ [min.]
Fridge (FR)	5	1	1
Lights (LC)	10	1	1
Dishwasher (DW)	10	30	5
Washing machine (WM)	20	30	5
Solar pump (SP)	20	1	1
TV	5	3	3
Boiler (BL)	25	5	5
Kettle (KT)	1000	1/3	1/6
Microwave (MC)	50	1/6	1/6
Toaster (TS)	300	1/6	1/20

B. Single load extraction

In single-load extraction, each disaggregator targets exclusively a single load in the monitored circuit and normally ignores dependencies amongst loads. While exploiting loads' dependencies is expected to improve the performance of a disaggregation system in a given building [9, 20, 21], it is also likely to reduce the generalization capability of such a system to new, previously unseen buildings. This is because such dependencies originate not only from the physical architecture of the power line network and the assumed signal model but also from the usage behavior of end-consumers which varies widely from one building to another, especially within the residential sector [22].

The load monitoring problem is modeled as K-separate binary classification tasks. Given a window of K samples of the aggregate real power signal $\underline{x}(n) = [x(n+k)]_{k=0}^{k=K-1}$ starting at the time instance n, the model $\underline{g}^{(m)}(\underline{x}(n), \boldsymbol{\theta}) \in [0,1]^K$ estimates the posterior probabilities of the activation profile for the analogous K time instances of m^{th} load where $\boldsymbol{\theta}$ is the model parameters (e.g. weights and biases in a neural network)

$$p\left(\underline{\omega}^{(m)}(n) = \mathbf{1} \mid \underline{x}(n)\right) = \underline{g}^{(m)}\left(\underline{x}(n); \boldsymbol{\theta}\right) \tag{1}$$

where the disaggregator's output is bound to the valid range of a probability function via a logistic sigmoid activation in the output layer of the network

$$\underline{g}^{(m)}\left(\underline{x}(n); \boldsymbol{\theta}\right) = \underline{\sigma}\left(\underline{\tilde{g}}^{(m)}\left(\underline{x}(n); \boldsymbol{\theta}\right)\right) \tag{2}$$

where $\underline{\tilde{g}}$ represents the sub-network from the input layer to activation signals of the output layer and $\underline{\sigma}$ is the logistic sigmoid function Eq. 5 applied element-wise to $\underline{\tilde{g}}$.

In the training phase, we refer to the pair $(\underline{x}(n), \underline{\omega}(n))$ as a single training segment with K data samples. Training segments are extracted from the whole time series signals $(\underline{x}, \underline{\omega})$ using *non-overlappling* windows which results in a training set whose inputs segments are

$$\boldsymbol{\mathfrak{X}} = (\, \underline{x}(0),\, \underline{x}(K),\, \underline{x}(2K),\, \ldots,\, \underline{x}((N_K - 1) \cdot K)) \tag{3}$$

and the corresponding activation segments

$$\boldsymbol{\Omega} = (\, \underline{\omega}(0),\, \underline{\omega}(K),\, \underline{\omega}(2K),\, \ldots,\, \underline{\omega}((N_K - 1) \cdot K)) \tag{4}$$

where $N_K = \lfloor N/K \rfloor$ is number of training segments (with the $\cdot^{(m)}$ notation omitted for brevity). Assuming all segments (and the K outputs of each segment) are conditionally independent given the input vector $\underline{x}(n)$ and identically distributed (i.i.d), then the likelihood function becomes

$$p\left(\boldsymbol{\Omega} \middle| \, \boldsymbol{\mathfrak{X}}, \boldsymbol{\theta}\right) = \prod_{n=0}^{N_K-1} \prod_{k=0}^{K-1} p\left(\omega(k + n \cdot K) \mid \underline{x}(n \cdot K)\right)$$

with ω being a Bernoulli distributed random variable

$$p\left(\boldsymbol{\Omega} \middle| \, \boldsymbol{\mathfrak{X}}, \boldsymbol{\theta}\right) = \prod_{n=0}^{N_K-1} \prod_{k=0}^{K-1} g_k(\underline{x}(n))^{\omega_k} \cdot (1 - g_k(\underline{x}(n)))^{1-\omega_k}$$

where $g_k(\underline{x}(n))$ is the k^{th} output of the disaggregator. The negative log-likelihood $-\mathcal{LL}$ then becomes

$$-\mathcal{LL} = -\sum_{n=0}^{N_K-1} \sum_{k=0}^{K-1} \omega_k \cdot \ln g_k(\underline{x}(n)) + (1 - \omega_k) \ln(1 - g_k(\underline{x}(n)))$$

which is known as *binary cross-entropy* and it is the adopted loss function in all our experiments. The choice of a logistic sigmoid activation function in the output layer together with the binary cross-entropy as the objective function is a standard combination in binary classification problems [23].

Finally, the following decision rule is used to estimate the final class labels

$$\hat{\omega}(n) = \begin{cases} 0 & \text{if } p(\omega(n)=0 \,|\, \underline{x}(n)) > p(\omega(n)=1 \,|\, \underline{x}(n)) \\ 1 & \text{otherwise} \end{cases}$$

We point out that the concept of load activation cycles, a complete cycle of operation *off*→ *on*→ *off*, is not considered. In other words, an activation cycle of a load can extend over several K-length segments (such as lighting circuits and dishwashers) or arise more than once within the same window segment (as in fridge and kettle activations). This is an important property since a disaggregator need not wait till the deactivation of a load (i.e. switch-*off* event) but rather can provide near real-time feedback from a partial segment of the activation, normally with some delay.

IV. Model Architecture

Figure 1 shows the architecture of the proposed fully convolutional neural network model. The model consists of 46 layers in five parts (an input layer, 40 encoding and decoding layers, 4 representation layers, and an output layer) reaching 41M trainable parameters. Each layer includes a sequence of elementary operations shown in the figure and briefly introduced in the sequel.

Dilated Convolutions `CONV(d, k)`: The core operation of each layer is the cross-correlation defined as

$$\texttt{CONV(d, k)}: f(x) \stackrel{\text{def}}{=} b(n) + \sum_{k=-\lfloor \texttt{k}/2 \rfloor}^{k=\lfloor \texttt{k}/2 \rfloor} x(n + \texttt{d} \cdot k) \cdot \kappa(k)$$

where `d` is the dilation rate [25], `k` is the kernel size, $\underline{b}$ is the bias vector, and $\underline{\kappa}$ is the layer's kernel.

Batch Normalization `BN` [16]: is a composition of two affine transformations applied to the output of each layer based on mini-batch statistics

$$\texttt{BN}: f(x) \stackrel{\text{def}}{=} \gamma\, \hat{x} + \beta = \gamma\, \frac{x - \mu_{\mathcal{B}}}{\sigma_{\mathcal{B}}} + \beta$$

where x is the original output of a unit, $\mu_{\mathcal{B}}$ and $\sigma^2_{\mathcal{B}}$ are the sample mean and variance of all outputs of this neuron over the mini-batch $\mathcal{B}$, and γ and β are two learnable parameters.

Leaky Rectified Linear Units `LReLU` [26]: is a non-linear activation function defined as

$$\texttt{LReLU}: f(x) \stackrel{\alpha \leq 1}{=} \max(\alpha x, x)$$

Activation noise (noise injection) `GN` [27]: is a regularization technique applied during the training phase only and consists of injecting small additive Gaussian noise (with variance σ^2) to the output of the layer to avoid over-fitting

$$\texttt{GN}: f(x) \stackrel{\text{def}}{=} x + z \sim \mathcal{N}(0, \sigma^2)$$

Sigmoidal activations `LogSg`: is a bounded activation function applied to the first hidden layer and the output layer of the model

$$\texttt{LogSg}: f(x) \stackrel{\text{def}}{=} (1 + \exp(-x))^{-1} \tag{5}$$

Down- and up-sampling: take place only across blocks where down-sampling is performed using MaxPooling while up-sampling is applied using forward-filling.

Parameter initialization and updates: model parameters are initialized from a zero-mean uniform distribution [28] and learned using a gradient-based stochastic optimization [29] with an update rule based on the ADAM Algorithm [30] with Nesterov momentum [31].

V. Performance Measures

Early works on energy disaggregation tended to adopt the simple accuracy index in evaluating the performance of a

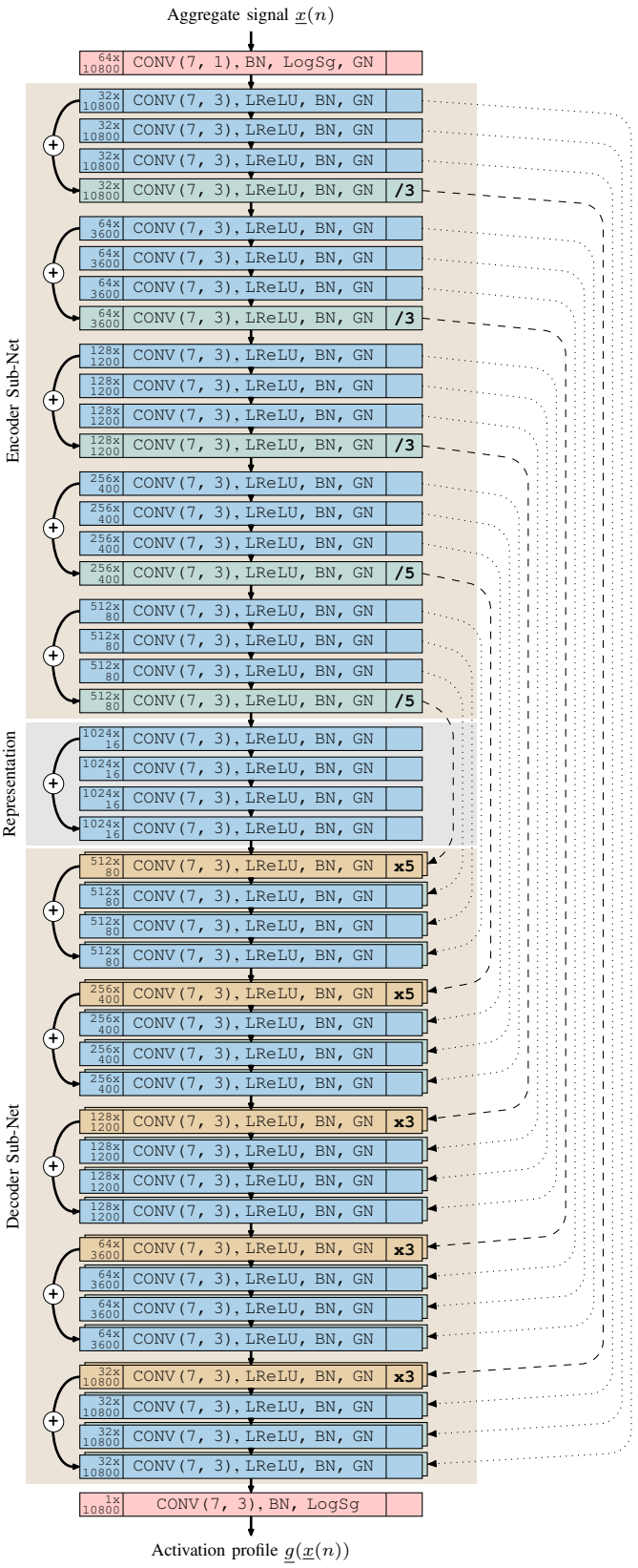

Fig. 1: Architecture of the proposed energy monitoring model. Dashed and dotted lines to the right represent *outer* and *inner* skip connections, respectively. Solid lines to the left represent residual connections [15]. Skip connections use channel aggregation while residual connections use element-wise addition. Green-shaded layers are followed by a pooling step, while the red-yellow shaded ones are preceded by an un-pooling operation.

disaggregation system [32–34]. Later works, however, realized the misleading interpretation of this measure (resulting from its bias towards the prevailing class) and proposed *precision*, *recall*, and f_1*-score* as alternative measures for assessing the disaggregation performance [20, 35–37].

We, however, believe that these measures represent a one-sided rigorous solution to the biasness of the accuracy index. In fact, these metrics are fused to the assumption of scarce load usage and fail to provide valuable interpretation of performance if this assumption is violated.

Given the raw-count contingency table

Classes		Predictions $\hat{\omega}^+$	Predictions $\hat{\omega}^-$	
	ω^+	TP	FN	RP
	ω^-	FP	TN	RN
		PP	PN	N

TP: True Positives
TN: True Negatives
FP: False Positive
FN: False Negatives
RP: Real Positives
RN: Real Negatives
PP: Positive Predictions
PN: Negative Predictions
N: Num. of samples/events

the aforementioned measure are defined as

$$\text{accuracy} = (\texttt{TP} + \texttt{TN}) \,/\, (\texttt{TP} + \texttt{TN} + \texttt{FP} + \texttt{FN}) \tag{6}$$

$$\text{precision} = \texttt{TPA} = \texttt{TP} \,/\, (\texttt{TP} + \texttt{FP})$$

$$\text{recall} = \texttt{TPR} = \texttt{TP} \,/\, (\texttt{TP} + \texttt{FN})$$

$$\texttt{f}_1\texttt{-s} = (2 \times \texttt{TP}) \,/\, (2 \times \texttt{TP} + \texttt{FN} + \texttt{FP}) \tag{7}$$

In the case of scarce load usage, the probability of negative samples becomes relatively high and the accuracy index becomes a single-sided measure, namely the true negative rate. In this case, a trivial disaggregator (one that always predicts the prevailing class) ambiguously yields near optimal accuracy.

The information retrieval approach to alleviate this bias is to simply *ignore* the prevailing term in Eq. 6, namely TN, which results in either the Jaccard index or the f-measure $\texttt{f}_1\texttt{-s}$ Eq. 7. We find this to be an extreme and ill-argued solution, especially in assessing energy monitoring performance. First, the scarce load usage is not always valid and is usually violated in commercial buildings or some residential loads such as refrigerators, air conditioners, space heaters, or electric vehicles. Additionally, the class of always-on loads suffers from the exact opposite situation where the class imbalance is due to the prevailing positive class and a trivial system in this case yields misleading near-optimal score for both the accuracy and the f-measure.

Second, when the scarce usage assumption is valid (e.g. for various miscellaneous appliances such as kettles, irons, vacuum cleaners ... etc), the extent of class imbalance varies widely amongst loads as well as users. These variations are not reflected by any means in either of the information retrieval measures. For these reasons, we claimed that precision and recall are inflexible measures since they are fused to a one-sided assumption regardless of the real distribution of classes.

Powers [38] introduced *informedness* B, *markedness* M, and their geometric mean *Matthews Correlation Coefficient* MCC as alternative, unbiased evaluation measures

$$\texttt{B} = \texttt{TPR} + \texttt{TNR} - 1 \tag{8}$$

$$\texttt{M} = \texttt{TPA} + \texttt{TNA} - 1 \tag{9}$$

$$\texttt{MCC} = \sqrt{\texttt{B} \cdot \texttt{M}} \tag{10}$$

where TNA = TN / (TN + FN) is the inverse-precision and TNR = TN / (TN + FP) is the inverse recall. Similar to the information retrieval measures, these alternatives were proposed and adopted in similar application domains such as medical diagnostics [39, 40] and recommender system evaluations [41]. We believe that the requirements of performance evaluation in these applications are more similar to those in energy disaggregation.

TABLE II: Performance comparison of 11 loads from the first building in UK-DALE [7]. rn is the probability of the negative class in the evaluation fold and %-NM is the *percent-noisy measure* [37].

	rn	%-NM	TPA	TPR	B	M	$\texttt{f}_1$-s	MCC
FR	0.55	0.87	0.92	0.88	0.81	0.82	0.896	0.815
LC	0.69	0.92	0.52	0.67	0.39	0.35	0.589	0.373
DW	0.98	0.95	0.85	0.49	0.49	0.84	0.623	0.641
WM	0.94	0.91	0.97	0.99	0.99	0.96	0.979	0.978
SP	0.78	0.97	0.46	0.24	0.16	0.27	0.312	0.204
TV	0.90	0.97	0.74	0.69	0.67	0.71	0.716	0.686
BL	0.91	0.95	0.34	0.75	0.60	0.31	0.468	0.431
KT	0.99	0.95	0.87	0.87	0.87	0.87	0.870	0.869
MC	0.99	0.97	0.62	0.46	0.45	0.62	0.526	0.529
TS	0.99	0.98	0.67	0.72	0.72	0.67	0.698	0.697
KL	0.87	0.94	0.46	0.55	0.46	0.39	0.502	0.422

TABLE III: Performance comparison between the proposed model AE and the *rectangles* architecture in [4] Regr on same load instances (left) and unseen instances from new buildings (right). All values represent the f-measure.

Load	Same instances		Accross buildings	
	Regr. [4]	AE	Regr. [4]	AE
FR	0.810	**0.879**	0.820	**0.927**
DW	0.720	**0.796**	0.740	**0.804**
MC	0.620	**0.705**	0.210	**0.366**
WM	0.490	**0.960**	0.270	**0.410**
KT	0.710	**0.783**	0.700	**0.819**

VI. Experiments and Results

The developed model is evaluated on the freely available UK-DALE dataset [7], an energy dataset acquired from five residential buildings. In this work, the 1 Hz real power measurements represent the input signals to disaggregate while the reference ones are the 1/6 Hz measurements up-sampled (using fill-forward) to 1 Hz.

Table II shows the performance measures of the proposed model evaluated on 11 loads from the first building in the adopted dataset with a 3-hour monitoring window for all load categories. Data folds are real power measurements from January and February of 2015 for training and validation, respectively, while the remaining 10 months of the 2015 represents the evaluation fold. While we provide these results as benchmarking ones, assessment of feasibility is observed in the following experiment.

In Table III, we compare the monitoring performance of our model AE with the previous work in [4], specifically the regression-based model Regr. (referred to as *rectangles* architecture). We use the exact data folds adopted in [4] for training and evaluation and define two test cases. The first trains and evaluates on the same load instances but future periods of operation while the second evaluates on new load instances (from new buildings). In both cases, the proposed model outperformed previous works in all load categories.

VII. Conclusion and Future Work

In this paper, we assessed the feasibility of a generic deep disaggregation model for end-use load monitoring using a fully convolutional neural network evaluated on a variety of load categories. The proposed model (with a fixed architecture and set of hyper-parameters) outperforms previous work and showed relatively acceptable performance across different loads.

REFERENCES

[1] K. K. Kaman, M. Faramarzi, S. Ibrahim, and M. A. M. Yunus, "Artificial Neural Network for Non-Intrusive Electrical Energy Monitoring System". *Indonesian Journal of Electrical Engineering and Computer Science* 6 (1):124–131, Apr. 2017.

[2] L. Mauch and B. Yang, "A new approach for supervised power disaggregation by using a deep recurrent LSTM network". *In proceedings of the 3rd IEEE Global Conference on Signal and Information Processing (GlobalSIP)*:63–67, Dec. 2015.

[3] ——, "A novel DNN-HMM-based approach for extracting single loads from aggregate power signals". *In proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*:2384–2388, Mar. 2016.

[4] J. Kelly and W. J. Knottenbelt, "Neural NILM: Deep Neural Networks Applied to Energy Disaggregation". *CoRR* abs/1507.06594 , Aug. 2015.

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

[6] W. He and Y. Chai, "An Empirical Study on Energy Disaggregation via Deep Learning". *The 2016 2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE2016)*, Beijing, ChinaNov. 2016.

[7] J. Kelly and W. J. Knottenbelt, "The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes". *Scientific Data*, Feb. 2015.

[8] O. Parson, G. Fisher, A. Hersey, N. Batra, J. Kelly, A. Singh, W. Knottenbelt, and A. Rogers, ""Dataport and NILMTK: A Building Data Set Designed for Non-intrusive Load Monitoring"". *In proceedings of the 3rd IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Orlando, Florida, USADec. 2015.

[9] S. Makonin, , "Real-Time Embedded Low-Frequency Load Disaggregation". ."

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory". *Neural Comput.* 9 (8):1735–1780, Nov. 1997.

[11] J. Z. Kolter and M. J. Johnson, "REDD: A public data set for energy disaggregation research". *proceedings of the SustKDD Workshop on Data Mining Applications in Sustainability*, San Diego, CA, USAApr. 2011.

[12] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion". *Journal of Machine Learning Research* 11 :3371–3408, Dec. 2010.

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions". *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*:1–9, Jun. 2015.

[14] P. Nascimento, , "Applications of Deep Learning Technologies on NILM". , Brazil, , Apr. 2016.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition". , Dec. 2015.

[16] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". , Mar. 2015.

[17] H. Lange and M. Bergés, "BOLT: Energy Disaggregation by Online Binary Matrix Factorization of Current Waveforms". *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*ser. BuildSys '16:11–20New York, NY, USA, 2016.

[18] K. Anderson, A. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Berges, "BLUED: a fully labeled public dataset for Event-Based Non-Intrusive load monitoring research". *Proceedings of the 2nd KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, Beijing, ChinaAug. 2012.

[19] M. Zeifman and K. Roth, "Viterbi algorithm with sparse transitions (VAST) for nonintrusive load monitoring". *2011 IEEE Symposium on Computational Intelligence Applications In Smart Grid (CIASG)*:1–8, Apr. 2011.

[20] H. Kim, M. Marwah, M. F. Arlitt, G. Lyon, and J. Han, "Unsupervised Disaggregation of Low Frequency Power Measurements". *proceedings of the 11th International Conference on Data Mining*:747–758. Arizona: SIAM, 2010.

[21] J. Z. Kolter, S. Batra, and A. Y. Ng, "Energy Disaggregation via Discriminative Sparse Coding". *Proceedings of the 23rd International Conference on Neural Information Processing Systems*ser. NIPS'10:1153–1161USA, 2010.

[22] N. Batra, O. Parson, M. Berges, A. Singh, and A. Rogers, "A Comparison of Non-Intrusive Load Monitoring Methods for Commercial and Residential Buildings". *arXiv preprint, arXiv:1408.6595*

[23] C. M. Bishop*Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[24] P. Y. Simard, D. Steinkraus, and J. Platt, "Best practices for convolutional neural networks applied to visual document analysis". , Aug. 2003.

[25] F. Yu, V. Koltun, and T. Funkhouser, "Multi-scale Context Aggregation by Dilated Convolutions". *arXiv:1511.07122v3*, Apr. 2016.

[26] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models". *Proceedings of the 30th International Conference on Machine Learning*Atlanta, Georgia, USA, 2013.

[27] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines". *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*J. Fürnkranz and T. Joachims, Eds.:807–814, 2010.

[28] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks". *proceedings of the 13th International Conference on Artificial Intelligence and Statistics*ser. Proceedings of Machine Learning ResearchY. W. Teh and M. Titterington, Eds. 9:249–256, Chia Laguna Resort, Sardinia, Italy13–15 May 2010.

[29] L. Bottou, *Stochastic Gradient Descent Tricks*. Berlin, Heidelberg: Springer, 2012., pp. 421–436.

[30] D. P. Kingma and J. L. Ba, "Adam: A Method for Stochastic Optimization". , Jan. 2017.

[31] T. Dozat, , "Incorporating Nesterov Momentum into Adam". , Standford University, Tech. Rep. 54, , May 2015.

[32] H.-H. Chang, C.-L. Lin, and J.-K. Lee, "Load Identification in Nonintrusive Load Monitoring using Steady-state and Turn-on Transient Energy Algorithms". *In proceedings of the 14th International Conference on Computer Supported Cooperative Work in Design*:27–32, Apr. 2010.

[33] , "Belkin Energy Disaggregation Competition". , https://www.kaggle.com/c/belkin-energy-disaggregation-competition, accessed: 2017.06.16.

[34] S. Makonin, F. Popowich, L. Bartram, B. Gill, and I. V. Bajic, "AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research". *Electrical Power and Energy Conference (EPEC), 2013 IEEE*:1–6, Aug. 2013.

[35] B. Christian, W. Kleiminger, R. Cicchetti, T. Staake, and S. Santini, "The ECO Data Set and the Performance of Non-Intrusive Load Monitoring Algorithms". *proceedings of the 1st ACM International Conference on Embedded Systems for Energy-Efficient Buildings (BuildSys)*:80–89, Memphis, TN, USANov. 2014.

[36] E. Holmegaard and M. B. Kjaergaard, "NILM in an Industrial Setting: A Load Characterization and Algorithm Evaluation". *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*:1–8, May 2016.

[37] S. Makonin and F. Popowich, "Nonintrusive Load Monitoring (NILM) Performance Evaluation: A unified approach for accuracy reporting". *Energy Efficiency* 8 (4):809–8142015

[38] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness, and correlation". *Journal of Machine Learning Technologies* 2 :37–632011

[39] W. J. Youden, "Index for Rating Diagnostic Tests". *Cancer* Vol. 3 (1):32–351950

[40] C. E. Metz, "Basic principles of ROC analysis". *Seminars in Nuclear Medicine* 8 (4):283–2981978

[41] G. Schröder, M. Thiele, and W. Lehner, "Setting Goals and Choosing Metrics for Recommender System Evaluations". *UCERSTI2 Workshop at the 5th ACM Conference on Recommender Systems* 23Chicago, USA, 2011.