Conv-NILM-Net, a causal and multi-appliance model for energy source separation

Simo Alami, Jérémie Decock, Rim Kaddah, Jesse Read

Conv-NILM-Net, a causal and multi-appliance model for energy source separation*

Mohamed Alami C.[1,2][0000−0002−9091−7095], Jérémie Decock[2], Rim kaddah[3], and Jesse Read[1][0000−0002−1013−6724]

[1] Ecole Polytechnique, Palaiseau, France
[2] Accenta, Paris, France
[3] IRT SystemX, Palaiseau, France

Abstract. Non-Intrusive Load Monitoring (NILM) seeks to save energy by estimating individual appliance power usage from a single aggregate measurement. Deep neural networks have become increasingly popular in attempting to solve NILM problems. However most used models are used for Load Identification rather than online Source Separation. Among source separation models, most use a single-task learning approach in which a neural network is trained exclusively for each appliance. This strategy is computationally expensive and ignores the fact that multiple appliances can be active simultaneously and dependencies between them. The rest of models are not causal, which is important for real-time application. Inspired by Convtas-Net, a model for speech separation, we propose Conv-NILM-net, a fully convolutional framework for end-to-end NILM. Conv-NILM-net is a causal model for multi appliance source separation. Our model is tested on two real datasets REDD and UK-DALE and clearly outperforms the state of the art while keeping a significantly smaller size than the competing models.

Keywords: NILM · Single Channel Source Separation · Deep Learning.

1 Introduction

In 2018, 26.1% of the total energy consumption in EU was attributed to households. This consumption mainly serves a heating purpose (78.4%). Moreover, most of the residential energy consumption is covered by natural gas (32.1%) and electricity (24.7%), while renewables account for just 19.5% [6]. However, as solar and wind generation rely on weather conditions, challenges due to intermittent generation have to be solved, and solutions for energy management such as demand response and photovoltaic (PV) battery management can play a key role in this regard. Machine Learning has proven to be a viable solution for smart home energy management [32]. These methods autonomously control heating and domestic hot water systems, which are the most relevant loads in a dwelling, helping consumers to reduce energy consumption and also improving their comfort.

* Supported by Accenta.ai

An efficient energy management system has to take into account users habits in order to anticipate their behaviour. However, comfort is hard to quantify as it remains purely subjective. We argue that in an energy management context, the users are the only ones that can offer a proper evaluation of their own comfort. Hence, a satisfactory hypothesis is to consider that their current behaviour and habits are the ones that optimise their comfort. Therefore, an efficient energy management system is one that can anticipate users habits, optimise consumption levels (for example by deciding which source to use, temperature settings etc.) while offering solutions that alter users known habits as little as possible.

Learning users' habits in a household is a hard problem mainly regarding data acquisition. The possible behaviours are diverse, if not unique, while monitoring inhabitants is not acceptable as it is a privacy infringement. From an energy provider perspective, the only available information is the household's total power consumption. A solution is therefore to decompose this consumption into the consumptions induced by each appliance in the household. The resulting disaggregated power time series can then be used as an input for a machine learning algorithm in order to learn consumption habits.

Energy disaggregation (also called non-intrusive load monitoring or NILM) is a computational technique for estimating the power demand of individual appliances from a single meter which measures the combined demand of multiple appliances. The NILM problem can be formulated as follows: Let $\bar{y}(t)$ the aggregated energy consumption measured at time t. With no loss of generality, $\bar{y}(t)$ can represent the active power (The power which is actually consumed or utilised in an AC Circuit in kW). Then $\bar{y}(t)$ can be expressed as in:

$$\bar{y}(t) = \sum_{i=1}^{C} y^{(i)}(t) + e(t) \tag{1}$$

Where C is the number of appliances, $y^{(i)}$ the consumption induced by appliance i and $e(t)$ some noise. The aim is to find $y^{(i)}$ given $\bar{y}(t)$.

There exist two approaches for NILM, namely load identification and source separation. In the first case, a first step called signature detection corresponds to the activation of a given appliance then a classification algorithm classifies the appliance category. The idea behind load identification is to build appropriate features called load signatures that allow to easily distinguish the referenced appliance from others within the installation. In the latter case, separation is directly obtained while retrieving the source signal.

In order to manage a building efficiently, for example using a reinforcement learning agent, it is necessary to use a model that performs source separation while being causal. In signal processing, a causal model is a model that performs the needed task (here source separation) without looking beyond time t rather than having to look in the future as presented in figure 2. The model can then be used as a backend for prediction, which is necessary for energy management.

We propose Conv-NILM-net, a fully convolutional and causal neural network for end-to-end energy source separation. Conv-NILM-net is inspired from Conv-TasNet [20], a convolutional model for speech separation. The model does not require more quantities than active power and disaggregates the signal for multiple appliances at once. We evaluate it on REDD and UK-DALE datasets, compared to recent models and achieves state of the art performance. Figure 1 presents an overview of the model and table 1 summarises the notations used throughout this paper.

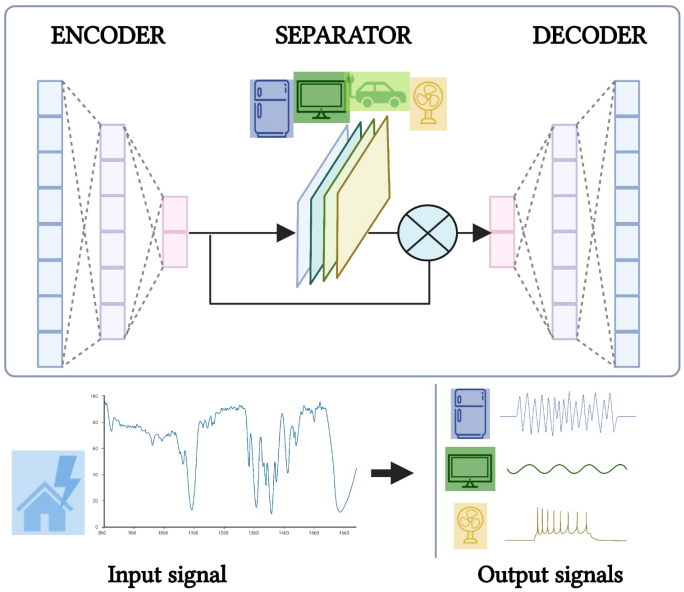

Fig. 1: Overview of Conv-NILM-net. It is composed of 2 blocs, the encoder/decoder and the separator. The encoder first projects the signal into a latent space, the separator disaggregates it into C corresponding to each appliance by learning C masks applied to input signal, then the decoder projects the C signals to the input space

2 Related Work

Most approaches in the literature are load identification approaches that predict the state of an appliance (on/off) and predict the average consumption of the given appliance during a certain period of time. Four appliance models are usually considered:

Table 1: Summary of notation.

$\bar{y}(t)$	Aggregated energy consumption at time t, $\bar{y}(t) \in \mathbb{R}^T$
C	Number of appliances
T	Length of a signal
$y^{(i)}(t)$	True energy consumption of appliance i at time t, $y^{(i)}(t) \in \mathbb{R}^T$
$\hat{y}^{(i)}(t)$	Predicted energy consumption of appliance i at time t, $y^{(i)}(t) \in \mathbb{R}^T$
Z	latent space representation of the mixture signal, $Z \in \mathbb{R}^{N \times K}$
m_i	learned mask for appliance i, $m_i \in \mathbb{R}^N$
s_i	filtered signal from the encoder for appliance i, $s_i \in \mathbb{R}^N$
N	Number of filters in encoder-decoder
L	Length of the filters
B	Number of channels in bottleneck and the residual paths' 1×1-conv blocks
H	Number of channels in convolutional blocks
P	Kernel size in convolutional blocks
X	Number of convolutional blocks in each repeat
R	Number of repeats

– Type I On/off devices: most appliances in households, such as bulbs and toasters;

– Type II Finite-State-Machines (FSM): the appliances in this category present states, typically in a periodical fashion. Examples are washer/dryers, refrigerators, and so on;

– Type III Continuously Varying Devices: the power of these appliances varies over time, but not in a periodic fashion. Examples are dimmers and tools.

– Type IV Permanent Consumer Devices: these are devices with constant power but that operate 24 h, such as alarms and external power supplies.

Current NILM methods work well for two-state appliances, but it is still difficult to identify some multi-state appliances, and even more challenging with continuous-state appliances. One of the most noticeable approaches called FHMM models each appliance as a hidden markov model (HMM) [13]. The HMM of each appliance is modelled independently, each one contributing to the aggregated power. AFAMAP [15] extends FHMM by predicting combinations of appliances working states. In AFAMAP, the posterior is constrained into one state change per time step. In [26], the authors propose a hierarchical FHMM in order to stop imposing independence between between appliances. The algorithm takes the active power as input and performs a clustering of the correlated signals then trains an HMM on the identified clusters called super devices. During the disaggregation step, the prediction is done using AFAMAP on the super devices then the clustering is reversed to find the the original appliance.

A critical step is the construction of load signatures or features that help to uniquely identify all types of home appliances with different operation modes. Event-based techniques have been employed to identify turn-on and turn-off

events using a variety of features like the active and reactive power [31, 2, 8], current and voltage harmonics [28, 27], transient behaviour particularly during the activation and/or deactivation [17, 3], current waveform characteristics [30]. Although the existing harmonic-based NILM methods achieved high load identification accuracy, their applicability is limited. The main drawback of this approach is that it requires harmonic current signatures with respect to all possible combinations of devices. Consequently, the complexity of this method increases exponentially with the number of electrical devices.

Deep Learning approaches have consistently outperformed HMM-based methods. Indeed, the number of features associated with the complexity induced by all the possible devices combinations make deep learning a natural candidate for NILM. In recent years, learning-based approaches were proposed to classify and directly estimate the power consumption of type-1 and type-2 appliances from an aggregated signal. Although FHMM-based NILM approaches are extensively used for power disaggregation, their performance is limited to the accurate approximation of appliance actual power consumption especially for type-2 (multi-state) and type-4 (always-on) appliances. Moreover, HMM-based methods have been reported to suffer from scalability and generalisation, which limits its real-world application. In contrast to classical event-based and state-based approaches, deep neural networks are capable of dealing with time complexity issues, scalability issues, and can learn very complex appliance signatures if trained with sufficient data.

Most recent NILM works employing deep neural networks used 1/6-Hz or 1/3-Hz sampled active power measurement as an input feature to train various deep neural networks. such as long short-term memory (LSTM) networks [21, 24], denoised autoencoder [11, 1] and Convolutional Neural Networks (CNN) [33, 29]. [11] proposed 3 different neural networks. A convolutional layer followed by an LSTM to estimate the disaggregated signal from the global one. They also used a denoising convolutional autoencoder to produce clean signals. The last neural network estimates the beginning and the end time of each appliance activation along with the mean consumption of each. [11] performs better than FHMM however their model was unable to identify multi-state appliances. To solve the multi-state appliance identification issue, [21] proposed a two-layer bidirectional LSTM based DNN model. Similarly, [29] proposed a two-step approach to identify multi-state appliances. They used a deep CNN model to identify the type of appliances and then used a k-means clustering algorithm to calculate the number of states of appliances.

Deep Learning also allowed source separation rather than load identification. This approach is more difficult but offers precise estimation of the consumption of each appliance in real time which includes continuous state appliances. [33, 9] proposed sequence-to-point learning-based CNN architecture with only active power as an input feature. In [4] gated linear unit convolutional layers [5] are used to extract information from the sequences of aggregate electricity consumption. In [23], the authors used a deep recurrent neural network using multi-feature in-

put space and post-processing. First, the mutual information method was used to select electrical parameters that had the most influence on the power consumption of each target appliance. Second, selected parameters were used to train the LSTM for each target appliance. Finally, a post-processing technique was used at the disaggregation stage to eliminate irrelevant predicted sequences, enhancing the classification and estimation accuracy of the algorithm.

In [8], the authors present WaveNILM which is a causal 1-D convolutional neural network inspired by WaveNet [22] for NILM on low-frequency data. They used various components of the complex power signal for NILM, current, active power, reactive power, and apparent power. WaveNILM, requires no significant alterations whether using one or many input signals. However, most of the existing DNNs models for NILM use a single-task learning approach in which a neural network is trained exclusively for each appliance. That is also the case for WaveNILM. This strategy is computationally expensive and ignores the fact that multiple appliances can be active simultaneously and dependencies between them. In [7] the authors introduce UNet-NILM for multi-task appliances' state detection and power estimation, applying a multi-label learning strategy and multi-target quantile regression. The UNet-NILM is a one-dimensional CNN based on the U-Net architecture initially proposed for image segmentation [25]. However, this model is not causal like WaveNILM.

Conv-NILM-net achieves the best of both worlds as it can handle source separation for any type of appliance, for multiple appliances simultaneously, it only needs the active power (although it is possible to other types of current in the same time).

3 Conv-NILM-Net

The model aims at separating C individual power sources $y^{(i)} \in \mathbb{R}^T$, where $i \in \{1, 2, \ldots, C\}$ from a mixture of signals representing the total consumption $\bar{y}(t) = \sum_{i=1}^{C} y^{(i)}(t) + e(t)$ and T is the length of the waveform. Therefore it take as input a single channel time series corresponding to the total consumption and outputs C time series corresponding to the consumption of each individual appliance. In this section, we present and detail our proposed architecture. We will describe the overall structure before focusing on the separation module.

3.1 Overall structure

Conv-NILM-Net is an adaptation of ConvTas-net [20]. Conv-Tasnet was originally only designed for speech separation and limited to two speakers. We propose an adaptation to energy load source separation with theoretically no limitation to the number of appliances. Our fully convolutional model is trainable end-to end and uses the aggregated active power as only input making the training easily deployable (no additional costly features needed).

Conv-NILM-net architecture consists of two parts: an encoder/decoder, and a separator. The encoder generates a multidimensional representation of the mixture signal; the separator learns masks applied to this representation to decompose the mixture signal, then the decoder translates the obtained signals from the encoded representation to the classic active power. The masks are found using a temporal convolutional network (TCN) consisting of stacked 1-D dilated convolutional blocks, which allows the network to model the long-term dependencies of the signal while maintaining a small model size.

Using encoder filters of legnth L, the model first segments the input total consumption into K overlapping frames $\bar{y}_k \in \mathbb{R}^L$, $k = 1, 2, \ldots, K$ each of length L with stride S. $\bar{y}_k$ is transformed into a N-dimensional representation, $Z \in \mathbb{R}^{N \times K}$:

$$Z = \mathcal{F}(w \cdot \bar{Y}) \tag{2}$$

where $Y \in \mathbb{R}^{L \times K}$ and $w \in \mathbb{R}^{N \times K}$ the N learnable basis filters of length L each. Z represents the latent space representation of the mixture series while $\mathcal{F}$ is a non-linear function. To ensure that the representation is non-negative, Conv-tasnet [20] uses the rectified linear unit (ReLU). However, this choice leads to a vanishing gradient behaviour, driving the norm of the gradients towards 0 thus making the model collapse as it eventually outputs null signals. Therefore, we replace ReLU with Leaky ReLU and only use ReLU for the last layer of the separation masks to enforce positive outputs.

The separator predicts a representation for each source by learning a mask in this latent space. It is performed by estimating C masks $m_i \in \mathbb{R}^N$. The representation of each source $s_i \in \mathbb{R}^N$, is then calculated by applying the corresponding mask m_i to the mixture representation, using element-wise multiplication:

$$s_i = Z \odot m_i \tag{3}$$

In Conv-tasnet as well as in [19], the masks were constrained such that $\sum_{i=1}^{C} m_i = 1$. This was applied based on the assumption that the encoder-decoder architecture can perfectly reconstruct the input mixture. indeed, in their model, $e(t) = 0$, $\forall t$. This assumption cannot be made in a NILM context, it is therefore relaxed.

The input signal of each source is then reconstructed by the decoder:

$$\hat{y}^{(i)} = s_i \cdot V \tag{4}$$

where $V \in \mathbb{R}^{N \times L}$ are the decoder filters, each of length L.

3.2 Separation module

The separator is mainly based on a temporal convolutional network (TCN) [16] and is detailed in figure 3. Temporal convolutions require the use of dilated convolutions which aim to increase the receptive field. Indeed, pooling or strided

convolutions are usually implemented for this purpose, however they reduce the resolution. Dilated convolutions allow exponential expansions of the receptive field without loss of resolution, while achieving same computation and memory costs. These are simply implemented by defining a spacing between the values in a kernel as illustrated in figure 2.

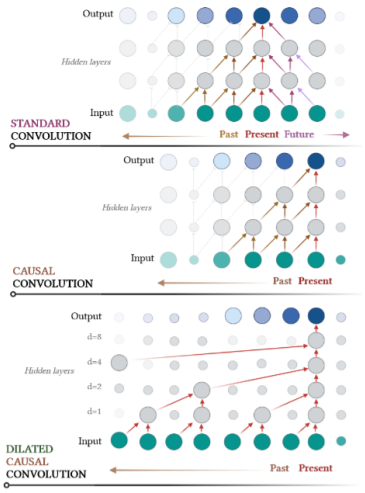

Fig. 2: Causal and dilated convolutions.

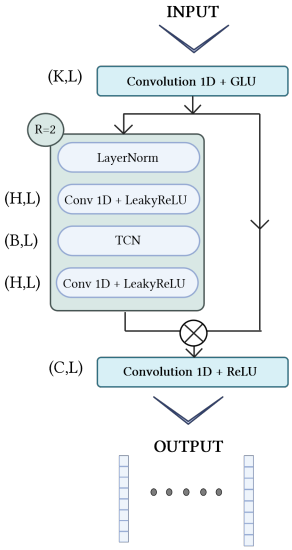

Fig. 3: Detailed representation of the separator.

In [21], the authors used LSTM [18] for NILM. This architecture can handle long sequences but suffers from the vanishing gradient issue while being computationally costly. We argue that a more efficient approach is to make use of 1D-convolutions. As illustrated in figure 2, convolutions for time series require future values (compared to the point of reference). During inference, these values are not accessible making the use of this model unpractical for prediction. This can be solved by giving a causal formulation to convolutions where the present value only depends of past ones. Moreover, the implementation is easy as it only requires an asymetric padding.

However relevant values become sparse. As in [20], Conv-NILM-net uses dilated layers with exponentially increasing dilation rates. The dilation factors increase exponentially to ensure a sufficiently large temporal context window to take advantage of the long-range dependencies in the signal. Therefore the dilation factors increase exponentially to ensure a sufficiently large temporal context

window to take advantage of the long-range dependencies in the signal. Therefore TCN consists of 1-D convolutional blocks with increasing dilation factors. Given kernels of length L and l layers, the receptive field of Conv-NILM-net is of size $RF = 2^l(L-1)$.

The output of the TCN is passed to a 1×1 conv block for mask estimation. This block also serves for dimensionality reduction and together with a nonlinear activation function estimates C mask vectors for the C target sources. The last layer of the last bloc uses a ReLU activation function to ensure the non-negativity of the outputs.

Contrary to speech separation, where simultaneous speeches are independent from one another, it is not the case in NILM context where appliance activations can be highly dependent. An elegant solution proposed in [10, 4], can be to use gated linear units (GLU) [5] to replace LeakyReLU activation functions. GLU allow the model to decide itself the relative importance of the kernels by using two parallel convolutions with the first followed by a sigmoid which result is then multiplied with the second convolution. The output of the sigmoid acts like a mask that activates depending on the input of the second convolution.

MSE, L_1, or even SI-SNR [20] losses are often used for NILM or source separation problems. The MSE takes the average squared error on all time steps for all disaggregated signals (ie appliances). We found that taking the mean on appliances is detrimental to the learning process as the error is distributed over all appliances. Therefore, the signals get mixed and artefacts of most consuming appliances appear on the remaining ones as illustrated in figure 4. We therefore choose to sum the error over all appliance rather than averaging it. The window mean squared error is calculated as:

$$\text{WMSE} = \frac{1}{T}\sum_{t=0}^{T}\sum_{i=1}^{C}\left(\hat{y}^{(i)}(t) - \bar{y}^{(i)}(t)\right)^2 \tag{5}$$

4 Experimental Methodology, Results, and Discussion

4.1 Datasets and parameters

Our experiments are done on two real-world datasets, Reference Energy Disaggregation Dataset (REDD) [14] and UK-DALE [12]. REDD records the power for 6 houses with sampling frequency of 1Hz for mains meter and 1/3Hz for appliance-channel meters. We choose to disaggregate the five top appliances for each building. UK-DALE data set published in 2015, records the power demand from five UK houses. In each house we record both the whole-house mains power demand every six seconds as well as power demand from individual appliances every six seconds.

For REDD, we converted the disaggregated data to 1Hz using linear extrapolation and kept 1/6 Hz frequency for UK-DALE. Usually the data are normalised

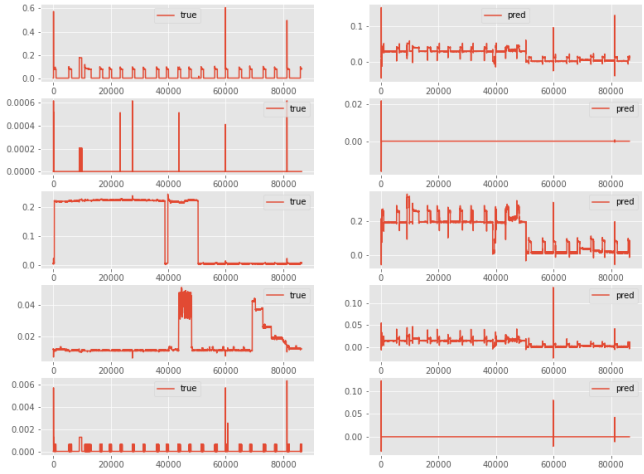

Fig. 4: Outputs of Conv-NILM-net of top 5 appliances of REDD building 1 when trained using classic MSE. We observe the presence of the artefacts although the MSE is minimized.

and the series of each appliance are scaled individually. We argue that appliance scaling is not practical as it is not possible to apply the scaling factors to the global signal available during inference. Therefore we chose to only use a min-max scaler for all appliances combined directly on the mixture power signal. In some contributions like seq2seq and seq2point [9, 33], a sliding window of the aggregate power is used as the input sequence, and the midpoint of the window corresponding to the target device reading is used as the output. This preprocessing smooths the power loads and makes the target values to retrieve easier. All results presented in section 4.3 for these implementations were obtained using smoothing. The results presented for Conv-NILM-NET were obtained without smoothing, making the difference in performance even more noticeable.

For UK-DALE dataset, we compare our results to UNET-NILM [7] and seq2point [9]. The constructed artificial aggregate consumption is obtained by taking the summation of selected appliances plus additional one appliance (Television in this setting). For UNET-NILM, the authors used a quantile filter to smooth the signal. This is not required for Conv-NILM-Net.

In our implementation for Conv-NILM-Net, we used for, each dataset, one day as input. This means that with 1Hz frequency, the input to Conv-NILM-net was 86400 points for REDD and 14400 for UK-DALE (1/6 Hz). The used parameters for Conv-NILM net are: $N = 32$; $L = 48$; $B = 2$; $H = P = X = 3$; $R = 2$. The meaning of each notation is made available in table 1 where we kept the same notation as in [20]. The model was trained for 2000 epochs using 10-fold cross-validation and a batch size of 5. We used Adam optimiser with an initial learning rate $lr = 0.01$, betas $= (0.9, 0.999)$, eps $= 0.01$.

4.2 Metrics

We evaluate the performance of the framework using the mean absolute error (MAE). Estimated accuracy is also a common metric for evaluating disaggregated power.

$$\text{Est.Acc.} = 1 - \frac{\sum_{t=1}^{T}\sum_{i=1}^{C}|\hat{y}^{(i)}(t) - \bar{y}^{(i)}(t)|}{2\sum_{t=1}^{T}\sum_{i=1}^{C}\bar{y}^{(i)}(t)} \tag{6}$$

Where $\hat{y}^{(i)}(t)$ is the predicted power level of appliance i at time t, and $\bar{y}^{(i)}(t)$ is the ground truth. The above expression yields total estimated accuracy; if needed, the summation over i can be removed creating an appliance-specific estimation accuracy. We also report the Signal Aggregate Error (SAE):

$$\text{SAE} = \frac{|\hat{r} - r|}{r} \tag{7}$$

where $\hat{r}$ and r represent the predicted total energy consumption of an appliance and the ground truth one. SAE measures the total error in the energy within a period, which is accurate for reporting daily power consumption even if its consumption is inaccurate in every time point.

4.3 Results on REDD

Table 2 presents the results obtained on REDD dataset for five building. For each building with disaggregated the top five appliances and reported the MAE, estimated accuracy ans SAE. We tested 3 versions on Conv-NILM-net. We observe that the causal+GLU tend to perform better on average but its results are very close to the causal implementation while increasing the number of parameters dramatically. We therefore tend to prefer the causal version of our model.

Table 3 compares the performance of Conv-NILM-net with state of the art models on 3 appliances that appear on REDD dataset. These appliances were selected as they are the only one presented in [34]. We therefore were limited to these appliance to compare our framework. We observe that our models outperform the state of the art by a margin. It decreases the MAE by 45% for the fridge, 51% for the microwave and even by 80% for the dishwaser. The best performing model is the causal model. In the appendix we present some outputs of the model for buildings 1 to 4 from REDD. These were obtained when disaggregating the top 5 appliances detailed in the same order as in table 2.

4.4 Results on UK-DALE

Table 4 compares the MAE of our model on UK-DALE dataset with UNET-NILM and seq2point. Our model outperforms the state of the art on the selected

		Model								
		Conv-NILM-NET			Causal			Causal + GLU		
Building	Appliance	MAE	Est.Acc	SAE	MAE	Est.Acc	SAE	MAE	Est.Acc	SAE
Building 1	Fridge	0.049	0.900	0.058	0.006	0.981	0.053	0.004	0.987	0.051
	Washer dryer	0.005	0.937	0.074	0.002	0.993	0.088	0.002	0.990	0.103
	Light	0.063	0.970	0.030	0.007	0.980	0.021	0.008	0.984	0.033
	Sockets	0.015	0.874	0.092	0.005	0.984	0.131	0.006	0.992	0.191
	Dishwasher	0.006	0.916	0.102	0.002	0.993	0.095	0.003	0.997	0.070
	Total	0.027	0.919	0.071	0.005	0.986	0.078	**0.004**	**0.989**	**0.054**
Building 2	Fridge	0.038	0.912	0.052	0.032	0.939	0.068	0.041	0.956	0.054
	Washer dryer	0.018	0.914	0.109	0.031	0.940	0.098	0.034	0.967	0.087
	Light	0.012	0.986	0.024	0.002	0.981	0.104	0.008	0.987	0.099
	Sockets	0.002	0.993	0.059	0.001	0.991	0.032	0.006	0.990	0.058
	Dishwasher	0.0006	0.997	0.115	0.001	0.993	0.029	0.003	0.992	0.031
	Total	**0.014**	0.960	0.0718	**0.014**	0.969	**0.066**	0.018	**0.978**	**0.066**
Building 3	Fridge	0.006	0.820	0.089	0.072	0.822	0.078	0.009	0.856	0.068
	Washer dryer	0.007	0.997	0.061	0.003	0.993	0.065	0.004	0.993	0.087
	Light	0.009	0.863	0.123	0.037	0.854	0.098	0.006	0.900	0.098
	Sockets	0.009	0.961	0.091	0.040	0.941	0.080	0.007	0.942	0.126
	Dishwasher	0.007	0.960	0.085	0.032	0.940	0.096	0.005	0.989	0.078
	Total	**0.006**	0.920	0.90	0.037	0.910	**0.083**	**0.006**	**0.936**	0.091
Building 4	Fridge	0.003	0.961	0.054	0.007	0.982	0.021	0.050	0.930	0.043
	Washer dryer	0.002	0.947	0.105	0.016	0.933	0.087	0.031	0.954	0.056
	Light	0.002	0.936	0.076	0.019	0.940	0.055	0.002	0.901	0.080
	Sockets	0.001	0.981	0.098	0.008	0.980	0.016	0.006	0.937	0.024
	Dishwasher	0.0006	0.994	0.132	0.002	0.995	0.098	0.020	0.892	0.069
	Total	**0.002**	0.964	0.093	0.011	**0.966**	0.055	0.022	0.923	**0.054**
Building 5	Fridge	0.003	0.883	0.005	0.005	0.880	0.004	0.005	0.991	0.093
	Washer dryer	0.003	0.992	0.001	0.0009	0.993	0.001	0.003	0.983	0.012
	Light	0.0006	0.999	0.0001	0.0001	0.999	0.002	0.002	0.977	0.10
	Sockets	0.0006	0.966	0.002	0.002	0.97	0.005	0.007	0.931	0.037
	Dishwasher	0.0008	0.913	0.004	0.004	0.91	0.002	0.010	0.990	0.078
	Total	**0.002**	0.950	**0.002**	**0.002**	0.95	**0.002**	0.005	**0.974**	0.064

Table 2: Conv-NILM-Net scaled results on top five appliances REDD dataset. Best average results are highlighted in bold.

Model	**Fridge**	**Microwave**	**Dishwasher**
seq2point	28.104	28.199	20.048
seq2seq	30.63	33.272	19.449
GLU-Res [4]	21.97	25.202	33.37
CNN-DI[34]	26.801	19.455	17.665
Conv-NILM-NET	14.67	9.67	3.56
Conv-NILM-NET (causal)	14.21	8.51	3.29
Conv-NILM-NET (GLU, causal)	15.02	9.76	3.31

Table 3: MAE results for Buiding 1 of REDD dataset.

Appliance	Model					
	1D-CNN	UNET-NILM	Seq2point	Ours	Ours (causal)	Ours (causal + GLU)
Kettle	20.390	16.003	2.16	1.85	1.9	2.5
Freezer	18.583	15.124	8.136	5.32	5.01	6.1
Dish washer	9.884	6.764	3.49	2.42	2.01	2.55
Washing machine	15.758	11.506	4.063	2.3	2.15	2.39
Microwave	9.690	6.475	1.305	0.902	0.91	1.05
Total	14.86	11.174	3.831	2.56	2.40	2.918

Table 4: Experimental results (MAE) in the UK-DALE dataset.

appliances. The causal model performs the best again while the total average is decreased by 33% compared to seq2pont.

Table 5 compares the size of Conv-NILM-net with state of the art models in terms of number of parameters. We observe that the fully convolution architecture of our model along with its particular architecture (encoder/decoder + separator) allow to obtain state of the art results with a model of approximately 40K parameters. This also possible because, contrary to other models, we use a unique loss for only one task. For instance UNET-NILM uses two separate loss functions, one to detect activation and an other to regress the average consumption while Seq2point [9] uses bidirectional residual networks which are very deep. It is also valuable to notice that models like UNET-NILM are specialized on individual appliances, meaning that in one needs to disaggregate 5 appliances, it requires 5 models, multiplying the number of parameters.

Models	# parameters
seq2point	$29.2M$
seq2seq	$29.8M$
GLU-Res	$1.2M$
CNN-DI	$738K$
Conv-NILM-net	41088

Table 5: number of parameters

Finally, figure 5 presents some results on top five appliances of first 4 building of REDD dataset. For each building the appliances are presented in the same order as in table 2. The left panels corresponds to the disaggregated target signals and the right panels presents the predicted output from Conv-NILM-net.

5 Conclusion

In this work, we presented Conv-NILM-net, an adaptation of Convtas-net to non intrusive load monitoring. We tested our model on two real world dataset

(a) building 1 (b) Building 2

(c) Building 3 (d) Building 4

Fig. 5: Selected results on top five appliances of first 4 building of REDD dataset. For each building the appliances are presented in the same order as in table 2. The left panels correspond to the disaggregated target signals and the right panels presents the predicted output from Conv-NILM-net.

and showed that Conv-NILM-net outperforms the state of the art by a margin. We presented 2 alternate models, one being causal and other using Gated Linear Units (GLU). These models allowed accurate disaggregation of several appliances at once while being much more smaller than their existing counterparts. Finally, the causal model allows consumption prediction and is ideal as input to an energy management system or a reinforcement learning model. In future work, we will use causal conv-NILM-net as a prediction model and test it in a reinforcement learning context. We will also test the GLU augmented model to verify if this implementation effectively takes into account appliances inter-dependencies and helps achieve better consumption predictions.

References

1. Roberto Bonfigli, Andrea Felicetti, Emanuele Principi, Marco Fagiani, Stefano Squartini, and Francesco Piazza. Denoising autoencoders for non-intrusive load monitoring: Improvements and comparative evaluation. *Energy and Buildings*, 158:1461–1474, 2018.
2. Roberto Bonfigli, Emanuele Principi, Marco Fagiani, Marco Severini, Stefano Squartini, and Francesco Piazza. Non-intrusive load monitoring by using active and reactive power in additive factorial hidden markov models. *Applied Energy*, 208:1590–1607, 12 2017.
3. Hsueh-Hsien Chang. Load identification of non-intrusive load-monitoring system in smart home. *WSEAS Transactions on Systems*, 9:498–510, 05 2010.
4. Kunjin Chen, Qin Wang, Ziyu He, Kunlong Chen, Jun Hu, and Jinliang He. Convolutional sequence to sequence non-intrusive load monitoring, 2018.
5. Yann N. Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks, 2016.
6. EUROSTAT. Energy consumption in households. 2018.
7. Anthony Faustine, Lucas Pereira, Hafsa Bousbiat, and Shridhar Kulkarni. Unetnilm: A deep neural network for multi-tasks appliances state detection and power estimation in nilm. In *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, NILM'20, page 84–88, New York, NY, USA, 2020. Association for Computing Machinery.
8. Alon Harell, Stephen Makonin, and Ivan V. Bajić. Wavenilm: A causal neural network for power disaggregation from the complex power signal, 2019.
9. Ziyue Jia, Linfeng Yang, Zhenrong Zhang, Hui Liu, and Fannie Kong. Sequence to point learning based on bidirectional dilated residual network for non intrusive load monitoring, 2020.
10. Berkan Kadioglu, Michael Horgan, Xiaoyu Liu, Jordi Pons, Dan Darcy, and Vivek Kumar. An empirical study of conv-tasnet, 2020.
11. Jack Kelly and William Knottenbelt. Neural NILM. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. ACM, nov 2015.
12. Jack Kelly and William Knottenbelt. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Scientific Data*, 2(150007), 2015.
13. Hyungsul Kim, Manish Marwah, Martin Arlitt, Geoff Lyon, and Jiawei Han. *Unsupervised Disaggregation of Low Frequency Power Measurements*, pages 747–758.
14. J Kolter and Matthew Johnson. Redd: A public data set for energy disaggregation research. *Artif. Intell.*, 25, 01 2011.
15. J. Zico Kolter and Tommi Jaakkola. Approximate inference in additive factorial hmms with application to energy disaggregation. In Neil D. Lawrence and Mark Girolami, editors, *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, volume 22 of *Proceedings of Machine Learning Research*, pages 1472–1482, La Palma, Canary Islands, 21–23 Apr 2012. PMLR.
16. Colin Lea, Rene Vidal, Austin Reiter, and Gregory D. Hager. Temporal convolutional networks: A unified approach to action segmentation, 2016.
17. Jing Li and Honggeng Yang. The investigation of residential load identification technologies. In *2012 Asia-Pacific Power and Energy Engineering Conference*, pages 1–4, 2012.

18. Benjamin Lindemann, Timo Müller, Hannes Vietz, Nasser Jazdi, and Michael Weyrich. A survey on long short-term memory networks for time series prediction. *Procedia CIRP*, 99:650–655, 2021. 14th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 15-17 July 2020.
19. Yi Luo and Nima Mesgarani. Tasnet: time-domain audio separation network for real-time, single-channel speech separation, 2017.
20. Yi Luo and Nima Mesgarani. Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(8):1256–1266, aug 2019.
21. Lukas Mauch and Bin Yang. A new approach for supervised power disaggregation by using a deep recurrent lstm network. In *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 63–67, 2015.
22. Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio, 2016.
23. Hasan Rafiq, Xiaohan Shi, Hengxu Zhang, Huimin Li, and Manesh Kumar Ochani. A deep recurrent neural network for non-intrusive load monitoring based on multi-feature input space and post-processing. *Energies*, 13(9), 2020.
24. Hasan Rafiq, Hengxu Zhang, Huimin Li, and Manesh Kumar Ochani. 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)*, pages 234–239, 2018.
25. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.
26. Pascal A. Schirmer, Iosif Mporas, and Michael Paraskevas. Energy disaggregation using elastic matching algorithms. *Entropy*, 22(1), 2020.
27. Sunil Semwal, Gautam Shah, and Rai Sachindra Prasad. Identification residential appliance using nialm. 12 2014.
28. Huijuan Shao, Manish Marwah, and Naren Ramakrishnan. A temporal motif mining approach to unsupervised energy disaggregation. *Proceedings of the 27th AAAI Conference on Artificial Intelligence, AAAI 2013*, 01 2013.
29. Zhang Shuichang, Yin, Cong, and Du Yanhua. Multi-state household appliance identification based on convolutional neural networks and clustering. *Energies*, 13:792, 02 2020.
30. D. Srinivasan, W.S. Ng, and A.C. Liew. Neural-network-based signature recognition for harmonic source identification. *IEEE Transactions on Power Delivery*, 21(1):398–405, 2006.
31. Michele Valenti, Roberto Bonfigli, Emanuele Principi, Squartini, and Stefano. Exploiting the reactive power in deep neural models for non-intrusive load monitoring. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2018.
32. Liang Yu, Shuqi Qin, Meng Zhang, Chao Shen, Tao Jiang, and Xiaohong Guan. A review of deep reinforcement learning for smart building energy management. *IEEE Internet of Things Journal*, 8(15):12046–12063, aug 2021.
33. Chaoyun Zhang, Mingjun Zhong, Zongzuo Wang, Nigel Goddard, and Charles Sutton. Sequence-to-point learning with neural networks for nonintrusive load monitoring. 2016.
34. Yuanmeng Zhang, Guanghua Yang, and Shaodan Ma. Non-intrusive load monitoring based on convolutional neural network with differential input. *Procedia CIRP*, 83:670–674, 2019. 11th CIRP Conference on Industrial Product-Service Systems.