POLITECNICO DI TORINO
Repository ISTITUZIONALE

Training Nonintrusive Load Monitoring Algorithms Without Supervision From Submeters

Original
Training Nonintrusive Load Monitoring Algorithms Without Supervision From Submeters / Castangia, M., Girmay, A.A., Camarda, C., Patti, E.. - In: IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS. - ISSN 1551-3203. - 20:4(2024), pp. 5440-5448. [10.1109/TII.2023.3334279]

Availability:
This version is available at: 11583/2984354 since: 2024-05-28T09:51:29Z

Publisher:
IEEE

Published
DOI:10.1109/TII.2023.3334279

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

13 September 2026

Training non-intrusive load monitoring algorithms without supervision from sub-meters

Marco Castangia, Awet Abraha Girmay, Christian Camarda and Edoardo Patti

***Abstract*—Non-intrusive load monitoring allows to estimate the energy consumption of major household appliances by just analyzing the aggregated power consumption collected at the main meter of the house. Recent disaggregation algorithms based on deep learning techniques showed superior performance with respect to previous methods. However, they require large amount of sub-meter data to be trained. In this work, we present a new solution for training non-intrusive load monitoring algorithms without any supervision from sub-meters. To achieve this goal, we divided the disaggregation algorithm into two stages named *appliance detection* and *state-based disaggregation*. In the first stage, we aim at identifying the start and stop times of the individual appliance operations within the whole-house power signal. In the second stage, we reconstruct the power signature of the target device by exploiting appliance-specific power states learned in the house. We tested our methodology on fridges, washing machines and dishwashers of a public dataset, showing double-digit improvements with respect to previous methods trained with sub-meter data. Most importantly, the proposed solution allows to collect a large number of appliance power signatures with minor costs, thus helping to achieve the generalization capabilities required by a real-world disaggregation system.**

***Index Terms*—non-intrusive load monitoring, nilm, smart meter, energy disaggregation, deep learning, neural network**

I. Introduction

The global climate crisis induced worldwide governments to cooperate with the aim of finding more effective solutions to drastically reduce their CO_2 emissions [1]. Whereas we can witness a positive trend in the adoption of renewable energy sources, we can also see a constant growth in the global energy demand as a consequence of the increasing energy needs (e.g. electric vehicles, heating, and air conditioning) [2]. Therefore, more efforts are definitely needed for the reduction and optimization of energy consumption.

Non-Intrusive Load Monitoring (NILM) is a software-based solution for estimating the energy consumption of major electrical devices from the analysis of whole-house power consumption [3]. The advantage of NILM with respect to intrusive approaches (e.g. smart plugs and smart appliances) relies on the scalability of the monitoring apparatus and the low costs for the maintenance and deployment of the equipment [4]. The information extracted by NILM is useful for both consumers and utilities. In particular, consumers can use this information to make better decisions on their appliances' utilization, with the goal of reducing their overall energy consumption. Indeed, several studies showed that a detailed energy breakdown enables higher energy savings of up to 12% of the total house's energy demand [5]. Utilities are interested in obtaining appliance-level power consumption for implementing better Demand Side Management (DSM) techniques which take into account the specific end-users' habits and needs [6].

Unfortunately, after more than thirty years from its first prototype, non-intrusive load monitoring remains mostly confined to the research labs [7]. The main cause that prevents the large application of NILM in the real-world is the need of sub-meter data for training appliance models. Indeed, appliance-level data are very costly to be obtained and prevents the collection of larger training sets that would finally enable the generalization of appliance models to unseen houses [8]. Therefore, we believe that NILM requires a completely different training paradigm to be adopted on a larger scale and finally bring the promised benefits to both end-users and utilities.

In this paper, we present a new NILM solution that can learn general appliance models without the need of sub-meter data for their training. To do this, we divided the disaggregation process into two steps: *appliance detection* and *state-based disaggregation*. During the first phase, we detect the individual appliance's usages in the aggregated load by means of a deep learning model previously trained to identify a large set of annotated operations. In the second phase, given the start and stop times of the operation, we reconstruct the power signature of the device by exploiting a set of appliance-specific power states learned in the house. The novelty of this work consists of avoiding the use of sub-meters to significantly reduce the costs for collecting new appliance operations. The possibility of collecting larger training sets will definitely increase the capability of deep learning models to generalize to unseen houses. In addition, we presented a new disaggregation algorithm for approximating the power signatures of the monitored devices by means of their power states, which can be easily extended or adapted to other types of electrical devices. Finally, by monitoring the whole-house power consumption we can simultaneously collect the power signatures of multiple appliances, thus extending our method to other devices without further costs for the monitoring equipment.

The rest of this paper is organized as follows. Section II provides an overview of popular NILM algorithms in the literature. Section III presents the datasets used for training and testing our models. Section IV explains the different processing steps composing our methodology. Section V shows the disaggregation accuracy achieved by our approach in comparison with literature solutions. Finally, Section VI summarizes the contributions of this work, proposing future

M. Castangia and E. Patti are with Politecnico di Torino, Turin, IT. E-mails: name.surname@polito.it.
A. Abraha Girmay and C. Camarda are with Midori S.R.L., Turin, IT. E-mail: name@midorisrl.eu

research directions.

II. Related works

G. Hart in [9] proposed the first implementation of a NILM system in which different appliances were classified based on the power change observed during the transitions between their operational states. However, NILM demonstrated the need for more information to distinguish between the different devices since they can easily present similar power states. Thus, researchers started to investigate more sophisticated models based on Additive Factorial Hidden Markov Models (AFHMM) to model the appliance behaviours [10]. Most importantly, AFHMMs represented for a long time a practical solution for implementing an unsupervised NILM system, but their actual implementation is not particularly accurate with respect to modern approaches, since they heavily rely on prior device information (e.g. power states). Other researchers used matrix factorization and sparse coding to estimate the contribution of the various devices to the aggregated power signal [11], but very few studies tried to improve these methods because of their prohibitive computational costs at higher sampling resolutions. Another promising NILM solution makes use of graph signal processing (GSP) to group together power transitions belonging to the same device based on historical observations [12]. In addition to the aforementioned approaches, there are several solutions that used time-series pattern recognition algorithms to identify the power signatures of the various devices within the aggregated signal, including dynamic time warping and motif mining [13].

As for now, deep learning models represent the most accurate solution to NILM in terms of disaggregation accuracy. Kelly et al. in [14] introduced the very first application of deep neural networks for the task of load disaggregation, showing superior performance with respect to previous methods based on combinatorial optimization and Factorial Hidden Markov Models (FHMM). The following works investigated different architectures and approaches to improve the disaggregation performance of deep neural networks. For example, Zhang et al. in [15] proposed to predict a single power measurement for each sliding window of aggregated power (sequence-to-point learning) instead of predicting the full length of the input sequence (sequence-to-sequence learning). Other studies opted for an intermediate solution that maps the aggregated load to shorter sequences (sequence-to-subsequence learning) in order to find a trade-off between computational costs and accuracy [16]. Then, major developments were made in the architectural layers of deep learning models with the aim of further increasing their accuracy [17], [18], [19]. In particular, Piccialli and Sudoso in [20] presented a new neural architecture employing the attention mechanism in combination with both convolutional and recurrent layers, showing superior performance with respect to other neural networks. Overall, researchers put significant efforts in the search for the optimal neural architecture, but a lot of work remains to be done to generalize the outstanding performance of these models to new households.

Klemenjak et al. in [21] highlighted the importance of transferability of appliance models for achieving larger applications of NILM in the real world. Previous works tried to solve this problem by implementing various transfer learning techniques to adapt the model parameters to unseen households [22], [23], [24]. However, we strongly believe that transfer learning does not solve the problem of model generalization because we still need to collect a small set of sub-meter data to fine-tune the model parameters in the new house. Therefore, better methods are needed to transfer prior knowledge on appliance behaviors to new houses without requiring the collection of additional sub-meter data in the house.

The recent works in energy disaggregation confirmed the superiority of deep learning techniques with respect to other approaches. However, their application on a large scale scenario remains difficult because of their low generalization capabilities to unseen houses. As for now, the most effective way to improve model generalization remains the collection of additional training data. However, gathering large amounts of ground truth annotations for NILM is nearly impossible with the current settings, which require the deployment of a conspicuous number of monitoring devices. In this work, we devised a new solution to solve this problem and potentially achieve the desired generalization of deep learning models. The novelty of our approach consists of moving the task of deep learning models from regression to classification. In fact, deep learning models can be trained to just classify the operational state of the device instead of predicting its instantaneous power consumption. The collection of annotations for the classification task can be carried out without the use of sub-meters. In fact, we just need to manually annotate the start and stop times of the individual appliance operations within the aggregated load of the house. Thanks to this method we are also able to collect a greater number of labels with minor costs because we only need to install a single smart meter to collect the aggregated load of the house. Once the start and stop times of the appliance operation have been determined, we can reconstruct its power signature by means of its major power states without adopting a regression approach. For this purpose, we also introduced a new state-based disaggregation algorithm to estimate the power states of the appliance and reconstruct its power signature as a sequence of power levels.

III. Dataset

In this section, we introduce the datasets used to train and test our methodology. In particular, we used a proprietary dataset for training our appliance models and a public dataset for evaluating their disaggregation performance, also in comparison to previous methods. The use of two different datasets for training and test is useful for demonstrating the generalization capabilities of our approach.

A. Proprietary training set

To train our appliance detection models we used a proprietary training set consisting of the aggregated loads of several households sampled with a resolution of one second. In more detail, the training set was collected in Italy in the period from November 2022 to May 2023 (7 months), and contains the main loads of 100 residential buildings

with different family compositions. The dataset also includes several manual annotations specifying the start and stop times of the individual appliance operations observed in the total load of the households. The collection process of these manual annotations is thoroughly described in Section IV.

In this work, we decided to limit the demonstration of our methodology to the fridge, the washing machine and the dishwasher. The reason is that the annotations for these appliances were easier to be obtained with respect to other devices since they are present in almost every household. In addition, the energy estimation of dishwashers and washing machines presents interesting applications in the implementation of demand response programs, since their activation can be dynamically shifted depending on the particular grid requirements [25]. Nevertheless, the application of the methodology presented in this paper is not limited to these devices and can be easily used to detect other devices showing a characteristic power signature. Table I contains relevant information regarding the training set of the fridge, the dishwasher and the washing machine. In particular, we reported the number of significantly different power signatures that we collected for each device. In addition, we included the total number of collected labels (i.e. manual annotations) for each appliance and the average number of labels obtained in each house. Notice that only a subset of the 100 monitored households was finally inserted in the training set. Indeed, when the same power signature is observed in multiple houses, we only add one of them to the training set.

TABLE I: Description of the training set.

Appliance	Power signatures	Labels	Avg. labels per house
Fridge	14	2861	204.35
Dishwasher	40	395	9.85
Washing machine	45	346	7.69

B. Public test set

The UK Domestic Appliance-Level Electricity (UK-DALE) dataset is a public repository containing the power consumption of five households in the UK both at the aggregate and appliance level [26]. The aggregated load is available with a sampling frequency of 1 Hz, while the sub-meter loads are available with a resolution of 6 seconds. We decided to test our methodology on this dataset for three main reasons. The first reason is that it provides the same sampling frequency of our training set, i.e. one second, which is uncommon among the other public datasets [27]. The second reason regards the presence of two fridges, two washing machines and two dishwashers with the respective sub-meter power consumption, which are useful for evaluating the actual accuracy of our algorithm. Finally, this dataset is widely recognized among previous works as a benchmark for evaluating the performance of disaggregation algorithms. In particular, we used the last week of House 1 and House 2 of this dataset as our test set. The other buildings were discarded because only House 1 and House 2 contain the sub-meter power consumption of both washing machines and dishwashers in two separate circuits.

IV. Methodology

The main steps of our methodology are reported in the pipeline of Figure 1. We divided our processing stages into a training phase and a test phase. In the training phase, we start by collecting an initial training set of aggregated loads ideally obtained from a heterogenous set of users that own the appliances we are aiming to model. Then, we annotate the start and stop times of the appliance operations every time they appear in the aggregated load as long as we deem them useful for improving model performance (*Annotation of appliance operations*). Once we collected a sufficiently large training set, we can prepare the input sequences (*Preprocessing*) and train our appliance models to detect the operations of the target devices (*Training of appliance model*). At test time, the appliance models are deployed in new houses and their performance is monitored. Either false negatives or frequent false positives can trigger a retrain of the appliance models, which can be improved with additional data from the test set. In fact, aggregated loads from the test set can be added to the training set and, once annotated, can be used to further improve the appliance detection performance. As soon as we recognize the operation of a target device in a new house and the end-user confirms the correctness of our detection, we save the major power states of the appliance power signature. In this way, the next time we detect an activation of the device in the new house, we can confidently provide the energy breakdown for that appliance thanks to the precise knowledge of its power states (*State-based disaggregation*). In the following, we describe in more detail the different processing steps of our pipeline.

A. Annotation of appliance operations

To create a robust appliance detection model we first need to collect a relatively large set of operations for the target device. An appliance can present very different power signatures depending on the specific operational cycle selected by the user and its manufacturer. For this reason, a good training set should contain diverse power signatures of the same device to allow generalizing to unseen houses. As a matter of fact, the generalization capability of the appliance detection model directly depends on the diversity of the collected power signatures for the target device. Therefore, when collecting a training set for a new device we must favor the diversity of operational cycles over the number of usages collected for the same device. For certain devices (e.g. fridges), it may even be useful collecting annotations in different climatic conditions, even though most of the electrical devices are not directly affected by the weather since they work by predefined programs. To collect the largest possible number of diverse power signatures, we deployed multiple smart meters over a large number of households for a shorter period of time. Notice that collecting power signatures directly from the main load of the house has the enormous advantage of simultaneously capturing multiple devices from the same meter, which can be used in the future to train additional appliance models.

Once we collected aggregated data from a sufficiently large set of households, we can start the labeling process which

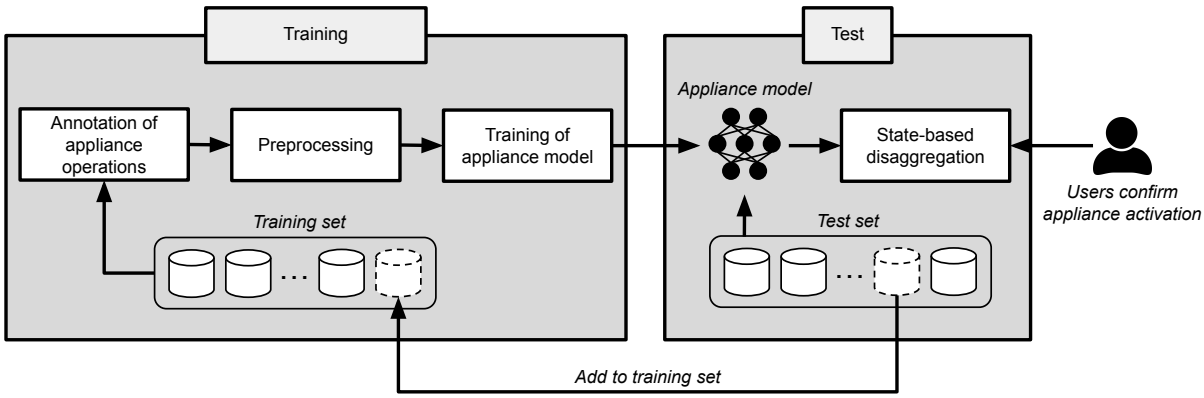

Fig. 1: The pipeline of our NILM system.

consists of annotating the start and stop times of the individual appliance operations within the whole-house aggregated load. The annotation process can be conducted by exploiting prior knowledge of the typical device behavior and by looking for repeating patterns in the aggregated load. In addition to that, we can set up a specialized application by means of a mobile app for collecting annotations directly from the user, who can push notifications every time he or she uses the appliance. The degree of manual involvement is carefully kept under control thanks to the use of dedicated software tools that automate most of the data annotation tasks and significantly reduce human efforts. In particular, we leverage multiple graphing tools to limit the search space of aggregated loads, thus focusing only on the specific power signatures we want to collect. Finally, once we annotated a sufficient number of diverse operation cycles, we can train a first detection model for the desired device.

In practice, model training is a never ending process as we likely need to periodically retrain our models whenever we encounter an unknown operating cycle that has not been included in the training set. Indeed, applications of new equipment and change of ratings inevitably causes the appliances to continuously evolve their power signatures. However, if we do not see any detection for a device after a certain period of time, we inspect the aggregated load of that house and start collecting additional annotations to enrich the existing training set of that device, thus continuously improving the model performance at every incremental update (see Figure 1). Notice that appliance models are shared among all the houses in the test set. Therefore, any update in the appliance models has an immediate effect in all the monitored houses. In the long run, we expect that those updates will become increasingly less frequent as we collect more operations and gradually cover all the possible power signatures of the target device.

B. Preprocessing

The training set of the appliance detection model consists of a set of sub-sequences $X^{(i)}$ of length T extracted from the aggregated loads of the monitored houses. The sub-sequences are generated by extracting consecutive windows of aggregated load of length T from the whole day of power consumption. In order to reduce the number of sub-sequences generated, we decided to shift the input window by 15 minutes instead of just moving to the next timestamp. For each sub-sequence $X^{(i)}$ we have a ground truth sub-sequence $y^{(i)}$ of the same length encoding the operational state of the device in that input window. The ground truth sub-sequence $y^{(i)}$ is generated from the data annotations of the training set. In more detail, the operational state $y_t^{(i)}$ is equal to 1 if the device is active at timestamp t and 0 otherwise, where t is comprised in the interval $[0, T]$.

TABLE II: Hyper-parameters for preprocessing.

	Fridge	Dishwasher	Washing machine
Sequence length (T)	8192	8192	8192
Sampling frequency (S)	1 second	2 seconds	1 second
Scaling factor (ρ)	3000 W	3000 W	3000 W

The hyper-parameters used for the sub-sequence generation are reported in Table II. Notice that we used a different sampling frequency for the dishwasher with respect to the fridge and washing machine. In fact, we found that the dishwasher requires a larger input window to be recognized. In practice, the length T of the input sequence remains the same for all devices, but the actual length of the input window depends on the sampling frequency S adopted for the device. Therefore, in the case of the dishwasher we used an input window of $T \times S = 16384$ seconds (about 4.5 hours), while for the fridge and the washing machine we used an input window of $T \times S = 8192$ seconds (about 2.2 hours).

In order to help our models, we decided to normalize each input sequence $X^{(i)}$ by subtracting its own mean $\mu^{(i)}$ and dividing the result by a scaling factor ρ. Interestingly, we found that subtracting the mean of the sequence instead of the training set mean slightly improves the accuracy of our models, because it eliminates the effects of vertical translations due to overlaps with other sources of power absorption.

$$X_{scaled}^{(i)} = \frac{X^{(i)} - \mu^{(i)}}{\rho} \tag{1}$$

C. Training of appliance models

To identify the start and stop times of the appliance operations, we implemented a sequence-to-sequence model that

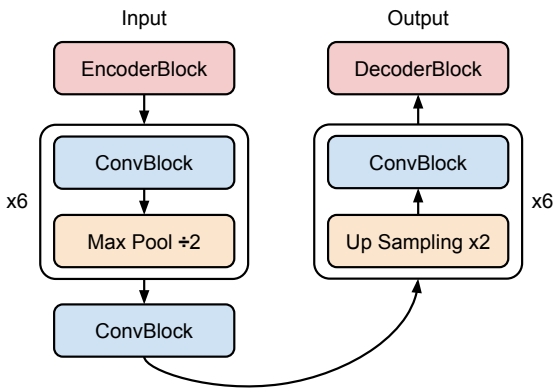

Fig. 2: The architecture of the appliance detection model.

maps the aggregated load $X^{(i)}$ to a sequence of output probabilities $\hat{y}^{(i)}$. Each output probability $\hat{y}_t^{(i)}$ indicates whether the appliance is active or not at timestamp t. The start times correspond to those timestamps where we have a state change from inactive to active. Similarly, the stop times correspond to the timestamps where we have a state change from active to inactive. Since we are trying to solve a classification problem, we trained our models to minimize the binary cross entropy loss between the predicted output probabilities $\hat{y}^{(i)}$ and the ground truth sequences $y^{(i)}$ encoding the actual operational state of the device.

Figure 2 shows the architecture of the sequence-to-sequence model that we used to predict the activation probabilities of the target appliances. The encoder block (EncoderBlock in Figure 2) applies a first convolution to drastically reduce the length of the input sequences and facilitate the analysis of the next layers. To do this, the encoder block applies a convolution with the same stride and kernel size, which are both equal to 8 data points. In practice, the encoder block creates new embeddings from consecutive chunks of the input sequence, where each chunk has a size equal to the kernel size. Then, the encoder block applies batch normalization and the Rectified Linear Unit (ReLU) activation function to the output of the first convolution. After that, we used six convolutions interleaved with max pooling layers to further compress the information extracted by our network. Each convolutional block (ConvBlock in Figure 2) actually applies two convolutions with a kernel size of 3, each one followed by batch normalization and ReLU activation function. After the last convolution, we insert as many convolutional layers interleaved with upsampling layers, in order to map the latent representations back to the original length of the input sequence. Finally, the decoder block (DecoderBlock in Figure 2) applies a transposed convolution that generates the final output probabilities of our model, one for each timestamp of the input sequence. Every convolutional layer used in our network generates feature maps with 64 dimensions. Table III reports the hyper-parameters that we used for the training process. The model was implemented in Python 3 with the help of the PyTorch framework [28].

TABLE III: Training hyper-parameters of the appliance detection model.

Hyper-parameter	Value
Optimizer	Adam
Loss	binary cross entropy
Learning rate	0.001
Epochs	150
Batch size	256
Stopping criteria	early stopping with patience equal to 10

D. State-based disaggregation

The state-based disaggregation constitutes the final stage of our methodology and has the purpose of reconstructing the power signature of the target device by means of its principal power states. Prior to signal reconstruction, we need to know the exact power levels characterizing the different operational modes of the device. These power states can be directly determined from the aggregated load during the very first recognition of the appliance, given that we do not have significant overlaps with other devices. In this work, the end-user is expected to avoid such overlaps in the first recognition of the device to facilitate the estimation of the appliance power states. In more detail, given a window of clean non-overlapping aggregated load containing the target device, the estimation of the power states works as follows. First of all, we subtract the minimum power value of the activation window to remove the contribution of baseline consumption. After that, we divide the power measurements into two clusters: power values greater than 1500W are put into the *high states* cluster, while other values are put into the *low states* cluster. The *high power* state corresponds to the mean value of the *high states* cluster, while the *low power* state is equal to the mean value of the *low states* cluster. Table IV reports the power states estimated for the fridges (FR), dishwashers (DW) and washing machines (WM) of UK-DALE. In the case of the washing machine, the low state roughly corresponds to the power consumption of the spin cycles, while the high state corresponds to the power consumption of the water heating phase. For the dishwasher, we only used the high state to describe the power demand of the water heating stages. Finally, for the fridges we only used the low power state.

TABLE IV: Estimated power states of the target appliances in UK-DALE.

	House 1			House 2		
	FR	DW	WM	FR	DW	WM
Low State	92 W	-	125 W	87 W	-	107 W
High State	-	2342 W	1843 W	-	2079 W	1891 W

Once the major operational states of the device have been determined, we can reconstruct its power signature at the next activation. The signal reconstruction requires a slightly different procedure for each device. For the fridge, we simply fill the entire activation window with its low power state. For the dishwasher, we subtract the minimum value from the activation window and assign the high power state to each timestamp where the high state fits under the aggregated load curve. In the case of the washing machine, the disaggregation

procedure requires a couple of additional steps. We start by subtracting the minimum value from the aggregated load. Then we assign the low power state to the disaggregated load. Finally, we sum the high power state to those values in the first half of the cycle where the high state fits under the aggregated load (we assumed that water heating occurs only in the first half of the washing machine operation). In all cases, the final result is a series of power states that resemble the original power signature of the monitored device.

V. Results

This section describes the disaggregation results obtained with the proposed NILM solution. We first present the evaluation metrics that we used to validate the disaggregation performance of our algorithm. Then, we show the actual results obtained on the test set with the selected evaluation metrics. Finally, we compare our results with those of previous neural architectures trained with sub-meter data.

A. Evaluation metrics

Disaggregation algorithms are commonly evaluated in terms of both classification and regression metrics [29]. On the one hand, the assessment of classification performance aims at quantifying the accuracy of the machine learning models in detecting the individual appliance's operations within the whole-house aggregated load. On the other hand, regression metrics are used to evaluate the average difference between the estimated and the actual sub-metered power consumption.

The labels for the classification task were derived by following the methodology used in [20]: if the power consumption of the target device is greater than 15 W, we assign a positive label, otherwise we assign a negative label. Below, we reported the formulas for deriving the F1 score, which is the most commonly adopted classification metric in previous works. Notice that we used acronyms for indicating True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN).

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{4}$$

For regression, the mean absolute error (MAE) is the most widely used metric to evaluate the disaggregation accuracy. Given the predicted power $\hat{y}_i$ and the measured power y_i of the target appliance at instant i, we can define MAE as the mean of the absolute deviation between $\hat{y}_i$ and y_i for the whole monitoring period of length M. Another common regression metric employed in the literature is the signal aggregate error (SAE), which measures the average error on the total power predicted over disjoint windows of length K, where $M = K \times N$. Following the guidelines of [18], we computed SAE over consecutive windows of one hour (K = 3600), where $r(\tau)$ is the sum of power consumption measured over the window τ and $\hat{r}(\tau)$ is the sum of the predicted power consumption over the same window.

$$MAE = \frac{1}{M} \sum_{i=1}^{M} |y_i - \hat{y}_i| \tag{5}$$

$$SAE = \frac{1}{N} \sum_{\tau=1}^{N} \frac{1}{K} |r(\tau) - \hat{r}(\tau)| \tag{6}$$

B. Disaggregation results

Table V shows the disaggregation results obtained on House 1 and House 2 of the UK-DALE dataset for the fridge (FR), the dishwasher (DW) and the washing machine (WM). Firstly, we can notice that we obtained quite different F1 scores for the fridge in House 1 (0.86) and House 2 (0.98). In particular, we are overly predicting the fridge operations of House 1 as indicated by the lower precision (0.80) with respect to the recall (0.92). Indeed, as depicted in Figure 3, the fridge of House 1 presents a slightly less regular pattern which makes it more difficult to model its duty cycle. We can also notice higher F1 scores for the washing machine in comparison to the dishwasher, with an average F1 of 0.93 for the washing machine and 0.87 for the dishwasher. Interestingly, the washing machine exhibits a very high precision score in both House 1 (1.00) and House 2 (0.99), highlighting that positive predictions for that device are correct most of the time. On the contrary, the dishwashers show lower precision scores with respect to their recall, which indicates that false positives are more likely in this case. This difference is probably due to the fact that the washing machine presents more distinctive features (e.g. spin cycles) than the dishwasher, which instead can be confused with combinations of other heating devices. However, we significantly reduced the impact of these phenomena by adopting smart data augmentation techniques that simulate these situations in the training phase.

Independently of the detection accuracy, the reconstruction errors are also influenced by the complexity of the individual power signatures. For example, in House 1 we obtained lower MAEs for the dishwasher (3.78 W) than the washing machine (11.20 W), despite having worse classification performance in the dishwasher case. In the same way, the dishwasher's SAE (3.23 W) is lower than the washing machine's SAE (3.62 W). The reason for these differences is that the washing machine generally presents a more convoluted pattern with respect to the dishwasher, which can be more easily approximated by its characteristic high power state (see Figure 5). Indeed, the spin cycles of the washing machine are quite hard to recover and are better approximated by their mean value (see Figure 4). Furthermore, other features such as short spikes are not captured by our state-based disaggregation algorithm (see Figure 3). More sophisticated appliance-specific methods could be devised to improve the fidelity of signal reconstruction. However, the benefits of such methods would be almost negligible for the purpose of energy breakdown, which remains the final goal of load monitoring.

TABLE V: Disaggregation results for fridge, dishwasher and washing machine of House 1 and House 2 of the UK-DALE dataset.

	House 1			House 2		
	FR	DW	WM	FR	DW	WM
MAE	14.93	3.78	11.20	3.54	3.55	4.12
SAE	11.86	3.23	3.62	1.66	3.11	2.64
Precision	0.80	0.76	1.00	0.97	0.86	0.99
Recall	0.92	0.97	0.87	0.98	0.93	0.90
F1	0.86	0.85	0.93	0.98	0.90	0.94

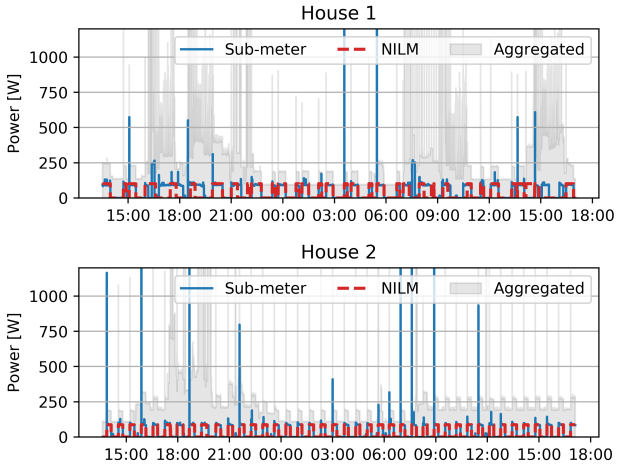

Fig. 3: Disaggregation of the fridge in House 1 and House 2 of UK-DALE.

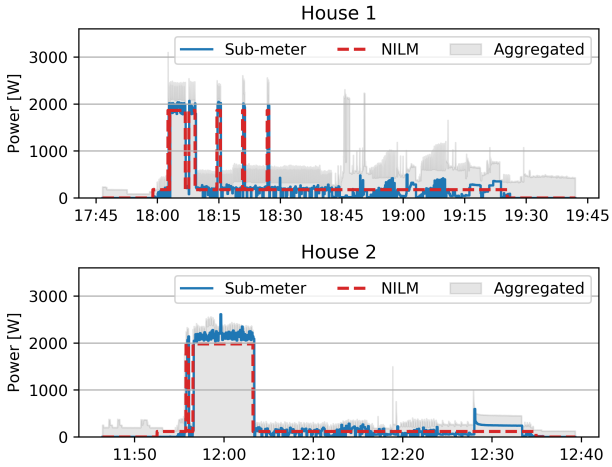

Fig. 4: Disaggregation of the washing machine in House 1 and House 2 of UK-DALE.

C. Comparison with state-of-the-art

Table VI shows a comparison between the proposed solution and previous neural networks trained with sub-meter data. Because of their consistency with our test settings and given their relevance in the literature, we considered the following five neural architectures for our comparison: Denoising Auto-Encoder (DAE) [30], Sequence-to-point (Seq2Point) [15], Subtask Gated Network (SGN) [18], Scale and Context-Aware Network (SCANet) [19] and Load Disaggregation with

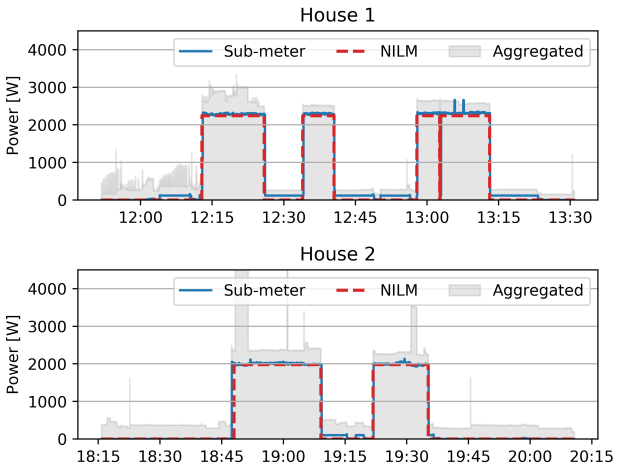

Fig. 5: Disaggregation of the dishwasher in House 1 and House 2 of UK-DALE.

Attention (LDwA) [20]. Please, notice that the results of these models were directly obtained from the experiments of Piccialli et al. [20] and refer to the performance obtained in House 2 of UK-DALE with the very same data we used for our tests, making the comparison fair.

According to the results reported in Table VI, the proposed solution outperforms all previous neural networks in terms of both classification accuracy and regression performance. In general, we found that increasing the accuracy of appliance detection (i.e. higher F1 scores) also leads to lower errors in the power estimation as a consequence of the reduced number of mispredicted operations. In addition, we also found that our two-steps solution produced a more significant improvement with respect to the architectural advancements introduced by previous methods. This fact suggests that the exceptional ability to collect a large training set is of primary importance for the overall progress of energy disaggregation, while the specific neural architecture chosen plays only a minor role in the reduction of errors. Furthermore, we cannot ignore the gain introduced by our state-based disaggregation with respect to the regression approaches trained on sub-meters. In fact, the use of power states revealed particularly effective for the reconstruction of power signatures and avoids the typical random fluctuations that are inevitably present in the output of neural networks. In summary, thanks to our methodology we were able to collect a larger set of appliance power signatures that demonstrated to be generalizable to other buildings and drastically enhanced the accuracy of existing deep learning models.

VI. Conclusion

In this paper, we presented a practical NILM algorithm that can estimate the power consumption of common household appliances without the use of sub-meters for model training. To achieve this goal, we divided the disaggregation process into two main steps, which we named *appliance detection* and *state-based disaggregation*. During the first phase, we use a pre-trained appliance model that leverages deep learning techniques to recognize the different appliance operations.

TABLE VI: Comparison between the proposed solution and previous neural networks trained with sub-meters in terms of mean absolute error (MAE), signal aggregate error (SAE) and F1 score.

		FR	DW	WM
DAE [30]	MAE	17.72	22.18	13.64
	SAE	8.74	18.24	10.67
	F1	0.76	0.55	0.25
Seq2Point [15]	MAE	17.48	15.96	10.87
	SAE	8.01	10.65	8.69
	F1	0.80	0.51	0.49
SGN [18]	MAE	16.27	10.91	9.74
	SAE	6.61	7.86	7.14
	F1	0.84	0.60	0.61
SCANet [19]	MAE	15.16	8.71	8.48
	SAE	6.54	4.86	5.77
	F1	0.86	0.63	0.63
LDwA [20]	MAE	13.24	6.57	7.26
	SAE	6.02	3.91	4.87
	F1	0.87	0.69	0.72
Proposed	MAE	**3.54**	**3.55**	**4.12**
	SAE	**1.66**	**3.11**	**2.64**
	F1	**0.98**	**0.90**	**0.94**

During the second stage, we reconstruct the appliance load by maintaining a set of house-specific parameters, i.e. the individual power states of the device. These parameters can be automatically learned or can be easily estimated with the help of the user, who can confirm the identification of the target device in the first recognition. The proposed solution has been compared with advanced deep learning models trained to directly predict the power consumption from the sub-meter. The comparison shows that our methodology can provide superior disaggregation performance, thus demonstrating that we can train reliable disaggregation algorithms without the use of sub-meters.

REFERENCES

[1] COP27, "2022 united nations climate change conference," November 2022. [Online]. Available: https://www.cop27.eg

[2] IEA, "World energy outlook - october 2020," October 2020. [Online]. Available: https://www.iea.org/reports/world-energy-outlook-2020/outlook-for-energy-demand

[3] P. A. Schirmer and I. Mporas, "Non-intrusive load monitoring: A review," *IEEE Transactions on Smart Grid*, 2022.

[4] A. Ruano, A. Hernandez, J. Ureña, M. Ruano, and J. Garcia, "Nilm techniques for intelligent home energy management and ambient assisted living: A review," *Energies*, vol. 12, no. 11, p. 2203, 2019.

[5] R. Gopinath, M. Kumar, C. P. C. Joshua, and K. Srinivas, "Energy management using non-intrusive load monitoring techniques–state-of-the-art and future research directions," *Sustainable Cities and Society*, vol. 62, p. 102411, 2020.

[6] F. Bandeiras, E. Pinheiro, M. Gomes, P. Coelho, and J. Fernandes, "Review of the cooperation and operation of microgrid clusters," *Renewable and Sustainable Energy Reviews*, vol. 133, p. 110311, 2020.

[7] M. D. Silva, Q. Liu, and O. F. Darteh, "A recent review of nilm framework: Development and challenges," in *2022 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. IEEE, 2022, pp. 1–7.

[8] Y. Himeur, A. Alsalemi, F. Bensaali, A. Amira, and A. Al-Kababji, "Recent trends of smart nonintrusive load monitoring in buildings: A review, open challenges, and future directions," *International Journal of Intelligent Systems*, vol. 37, no. 10, pp. 7124–7179, 2022.

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

[10] T. Ji, L. Liu, T. Wang, W. Lin, M. Li, and Q. Wu, "Non-intrusive load monitoring using additive factorial approximate maximum a posteriori based on iterative fuzzy c-means," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6667–6677, 2019.

[11] A. Majumdar, "Trainingless energy disaggregation without plug-level sensing," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–8, 2022.

[12] B. Zhao, K. He, L. Stankovic, and V. Stankovic, "Improving event-based non-intrusive load monitoring using graph signal processing," *IEEE Access*, vol. 6, pp. 53 944–53 959, 2018.

[13] B. Liu, J. Zheng, W. Luan, and Z. Liu, "Appliance power pattern mining via motif discovery in unsupervised nilm," in *2021 IEEE 5th Conference on Energy Internet and Energy System Integration (EI2)*. IEEE, 2021, pp. 3172–3177.

[14] 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*, 2015, pp. 55–64.

[15] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[16] Y. Pan, K. Liu, Z. Shen, X. Cai, and Z. Jia, "Sequence-to-subsequence learning with conditional gan for power disaggregation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 3202–3206.

[17] K. Wang, H. Zhong, N. Yu, and Q. Xia, "Nonintrusive load monitoring based on sequence-to-sequence model with attention mechanism," in *Zhongguo Dianji Gongcheng Xuebao/Proceedings of the Chinese Society of Electrical Engineering*, vol. 39, no. 1, 2019, pp. 75–83.

[18] C. Shin, S. Joo, J. Yim, H. Lee, T. Moon, and W. Rhee, "Subtask gated networks for non-intrusive load monitoring," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 1150–1157.

[19] K. Chen, Y. Zhang, Q. Wang, J. Hu, H. Fan, and J. He, "Scale- and context-aware convolutional non-intrusive load monitoring," *IEEE Transactions on Power Systems*, vol. 35, no. 3, pp. 2362–2373, 2019.

[20] V. Piccialli and A. M. Sudoso, "Improving non-intrusive load disaggregation through an attention-based deep neural network," *Energies*, vol. 14, no. 4, p. 847, 2021.

[21] C. Klemenjak, A. Faustine, S. Makonin, and W. Elmenreich, "On metrics to assess the transferability of machine learning models in non-intrusive load monitoring," *arXiv preprint arXiv:1912.06200*, 2019.

[22] Y. Liu, L. Zhong, J. Qiu, J. Lu, and W. Wang, "Unsupervised domain adaptation for nonintrusive load monitoring via adversarial and joint adaptation network," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 1, pp. 266–277, 2021.

[23] L. Wang, S. Mao, B. M. Wilamowski, and R. M. Nelms, "Pre-trained models for non-intrusive appliance load monitoring," *IEEE Transactions on Green Communications and Networking*, vol. 6, no. 1, pp. 56–68, 2021.

[24] D. Li, J. Li, X. Zeng, V. Stankovic, L. Stankovic, C. Xiao, and Q. Shi, "Transfer learning for multi-objective non-intrusive load monitoring in smart building," *Applied Energy*, vol. 329, p. 120223, 2023.

[25] A. Amin, O. Kem, P. Gallegos, P. Chervet, F. Ksontini, and M. Mourshed, "Demand response in buildings: Unlocking energy flexibility through district-level electro-thermal simulation," *Applied Energy*, vol. 305, p. 117836, 2022.

[26] J. Kelly and W. Knottenbelt, "The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes," *Scientific Data*, vol. 2, no. 150007, 2015.

[27] H. K. Iqbal, F. H. Malik, A. Muhammad, M. A. Qureshi, M. N. Abbasi, and A. R. Chishti, "A critical review of state-of-the-art non-intrusive load monitoring datasets," *Electric Power Systems Research*, vol. 192, p. 106921, 2021.

[28] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[29] G.-F. Angelis, C. Timplalexis, S. Krinidis, D. Ioannidis, and D. Tzovaras, "Nilm applications: Literature review of learning approaches, recent developments and challenges," *Energy and Buildings*, p. 111951, 2022.

[30] R. Bonfigli, A. Felicetti, E. Principi, M. Fagiani, S. Squartini, and F. Piazza, "Denoising autoencoders for non-intrusive load monitoring: improvements and comparative evaluation," *Energy and Buildings*, vol. 158, pp. 1461–1474, 2018.