IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON ARTIFICIAL INTELLIGENCE TECHNOLOGIES FOR ELECTRIC POWER SYSTEMS

Received February 7, 2020, accepted February 26, 2020, date of publication March 5, 2020, date of current version March 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2978513

PB-NILM: Pinball Guided Deep Non-Intrusive Load Monitoring

EDUARDO GOMES[1], (Student Member, IEEE), AND LUCAS PEREIRA[2], (Member, IEEE)
[1]ITI, LARSyS, M-ITI, 9020-105 Funchal, Portugal
[2]ITI, LARSyS, Técnico Lisboa, 1049-001 Lisboa, Portugal

Corresponding author: Lucas Pereira (lucas.pereira@tecnico.ulisboa.pt)

This work was supported in part by the funding under Grant M1420-01-0145-FEDER-000002, and in part by FCT under the LARSyS - FCT Plurianual funding 2020 - 2023, and Grant CEECIND/01179/2017.

ABSTRACT The work in this paper proposes the application of the pinball quantile loss function to guide a deep neural network for Non-Intrusive Load Monitoring. The proposed architecture leverages concepts such as Convolution Neural Networks and Recurrent Neural Networks. For evaluation purposes, this paper also presents a set of complementary performance metrics for energy estimation. Finally, this paper also reports on the results of a comprehensive benchmark between the proposed network and three alternative deep neural networks, when guided by the pinball and Mean Squared Error loss functions. The obtained results confirm the disaggregation superiority of the proposed system, while also showing that the performances obtained using the pinball loss function are consistently superior to the ones obtained using the Mean Squared Error loss.

INDEX TERMS Non-intrusive load monitoring, NILM, recurrent neural networks, convolutional neural networks, pinball quantile loss, mean squared error loss, benchmark.

I. INTRODUCTION

For many years, Non-Intrusive Load Monitoring (NILM) [1] has remained a challenge for researchers in the area. The problem is complex due to the sheer number of variables to take into consideration. Simply put, NILM has as objective determining which appliance is active and how much it is consuming at each time instance. However, the challenge lies in the limitation of available data for algorithm training, dependencies of unknown factors such as the number of appliances in each household, unique appliance characteristics, and different consumption patterns. NILM can be approached as a classification or regression problem. It can be considered as a classification problem when the objective is the detection and classification of an appliance among a complex signal. It is a regression problem when the aim is to estimate the consumption of the individual devices directly from the aggregated energy intake.

Many methods have been proposed throughout the years to solve this problem, ranging from ''classic'' machine-learning algorithms (e.g., Support Vector Machines, Lazy Learners, and Artificial Neural Networks), to advanced statistical learning methods like Hidden-Markov Models and Bayesian Statistics. A review of these methods is out of the scope of this paper. Instead, the interested reader can refer to the many literature reviews on these methods [2]–[4].

Recently, deep learning (DL) algorithms have been consistently establishing new state-of-the-art performances in many fields [5]. These include advances in language models such as [6], a language model capable of text synthesis, and summarizing, among other uses. Audio generation has also been benefited in [7] and [8], as well as image segmentation for medical applications [9]. NILM is no exception to this trend. As such, recent times have seen a burst in DL proposals to solve this problem, e.g., [10]–[15].

However, to achieve competitive results, DL methods require an abundance of training data. This is still a significant problem for NILM considering the lack of high-quality datasets, both in terms of duration and quality of the labels [16], [17]. Likewise, such approaches also benefit significantly from a high number of trainable parameters, requiring computational power that is not cheap nor readily available in most cases.

The contributions of this paper to the ongoing body of work in NILM are threefold. First, it proposes PB-NILM, a deep neural network composed of different types of layers such

The associate editor coordinating the review of this manuscript and approving it for publication was Canbing Li.

as convolutional, recurrent, and fully connected layers, and guided by the pinball (PB) quantile loss function [18], [19]. Second, it proposes a set of performance evaluation metrics, including the *Correctly Estimated Power* (*CEP*). Third, it provides a comprehensive evaluation and benchmark of PB-NILM against the same architecture when guided by the Mean Squared Error (MSE) loss function, and of three alternative deep learning architectures also guided by the PB and the MSE loss functions.

The remaining of this paper is organized as follows. Section II presents the main advances in the machine learning field, with varied applications, proceeding to the quintessential components of deep learning, and an overview of the pinball quantile loss function. Building on these works, Section III presents PB-NILM, a deep neural network that is guided by the PB quantile loss function and establishes a benchmark with the standard MSE loss function. The evaluation methodology is presented in Section IV. This section also introduces a new set of performance evaluation metrics and details the network's training and testing methodology. Section V presents and briefly discusses the performance evaluation results. Further discussions are then presented in Section VI. Finally, in Section VII, the paper concludes with an overview of the presented work, its limitations, and potential future work directions.

II. RELATED WORKS

Recent approaches to a variety of machine-learning problems try to leverage the power of DL, developing new architectures that are composed by what can be considered the fundamental building blocks of neural networks - convolutional [20], [21], recurrent [22], [23], and fully-connected layers.

These are often used in tandem to achieve competitive results in a particular task, with each layer providing a certain benefit to the architecture. Some of the widely used deep networks include the *WaveNet* [8] for audio generation and synthesis, *ResNet* [24], *DenseNet* [25] and *Xception* [26] for image classification and segmentation.

The *WaveNet* was developed to generate audio excerpts, by making use of dilated convolutions that preserve the size of the input data. This also allows for a much wider receptive field, thus improving the feature extraction process. In [13], the authors introduce *WaveNILM*, an architecture based in the *WaveNet* to leverage the dilation properties to achieve better disaggregation results. The latter was trained on the AMPds dataset [27], using 90% of the data for training, and the remaining 10% for testing. The performance of the method was reported using the *estimated accuracy* metric, which reports the total disaggregation over all timesteps. The proposed method was analyzed using the aggregated signal, achieving performances of over 85% in all the reported test cases.

Wu and Wang [14] propose the concatenation of convolutional layers to classify different appliances by making use of spectrograms based features. The authors explore the application of the *DenseNet* and *Xception* architectures on the UK-DALE [28] and REDD [29] datasets for appliance classification. A common characteristic between these architectures lies in the almost exclusive use of convolutional layers. In terms of classification results, the authors report F_1-*Score* values for the kettle, fridge, dishwasher, and microwave of over 0.90. Only the washing machine yields a lower value of 0.80. For load estimation, the *MAE* values are presented, with the highest error being just under six watts for the fridge.

While standard convolutional layers are useful for feature extraction and classification, they do not preserve spatial information. For example, it may be possible to detect a face in an image, but it does not reveal where it was detected. As such, and in the context of time series, it is relevant to mention Recurrent Neural Networks (RNNs), as these are designed to carry information through time.

Long-Short Term Memory (LSTM) networks [22] and Gated Recurrent Units (GRU) [23] are among the several existing types of RNN architectures. As NILM itself is a time-series problem, RNNs found themselves being used by NILM researches, having yielded competitive results for both event classification, and energy estimation.

In [10], the authors propose a solution based on RNNs, more specifically LSTMs, to perform classification of appliance consumption. The datasets used were the UK-DALE and REDD. In this work, Kim *et al.* present results that are at least double the performance of previous state-of-the-art approaches, with accuracy values ranging from 76% to 96% on houses 1 to 5 of the REDD dataset.

The work in [12] proposed a Convolutional Neural Network (CNN) based solution for the load classification and estimation of individual appliances. Kong *et al.* also make use of the UK-DALE dataset and highlight the issue of missing data. The results are reported using the *F1-Score* for classification and *Energy Accuracy* for load estimation. In terms of performance, the authors present superior results against other solutions, with values of *F1-Score* of over 0.85 and *Estimation Accuracy* of over 0.88, except for the washer-dryer of house 5, with 0.735.

In [15], the authors explore the classification of appliances with random forest classifiers, fully-connected networks, and CNNs using the PLAID dataset [30]. The observations on the performance of several methods are analyzed at different sampling rates. Peak performance was attained by a CNN based architecture, with the F_1-Score of 0.7619 at 1.2kHz.

Murray *et al.*, in [11], attempt to apply transfer learning to the NILM problem. They propose two different networks based on CNNs and RNNs and evaluate them on three datasets (REFIT, UK-DALE, and REDD). To assess the transferability of the proposed networks, different sets of data were used during training and testing. The reported results vary from as little as 0.21 to 1 for the F_1-*score* for state estimation. As for energy estimation, the results range from 44% and 82% in terms of *estimation accuracy*.

Nevertheless, even with proper deep network topology, it can still be challenging to learn useful patterns and correctly perform designated tasks. One of the main challenges

of NILM is the unbalanced nature of the problem. More concretely, some appliances have frequent use and thus reveal a more tractable load over time. Others are not utilized very often, resulting in very few activations that algorithms can use to learn their working patterns [12].

Considering the different appliance load distributions, it is difficult to find a method to guide the learning of the various appliances properly. This challenge is not exclusive to the NILM problem, being very common in the forecasting of household load demand [31], [32]. For example, in [32] the authors used the pinball quantile loss function to train an LSTM for load forecasting with evident performance gains against the standard mean squared error (MSE) loss function. The reported improvements are in the order of 2.19% to 7.52% for residential consumers and 3.79% to 25.80% for small & medium enterprise consumers.

III. PB-NILM: PINBALL GUIDED NILM

Against this body of related work, this paper presents an in-depth study on the applicability of the pinball loss function to the NILM problem.

To this end, a custom deep neural network architecture is trained guided by the PB loss function (PB-NILM). The PB-NILM network is then put against the same architecture, but guided by the MSE loss function. For additional benchmarks, the PB-NILM is also put against three other deep net architectures for NILM, all of which are individually trained with the PB and MSE loss functions.

It is hypothesized that a standard loss function like MSE will guide the training towards the mean or median of the distribution, which is not the best option in the case of NILM, as previously mentioned. Consequently, the disaggregation results would not be consistent across appliances. In contrast, using the PB loss function, it is possible to guide the learning according to the underlying data by setting custom quantile value. In other words, it is possible to set different penalties (or losses) for under- and over-estimation errors based on the appliances to disaggregate.

Software wise, *Python* 3.6.8 was used, along with *Keras* [33] running on the *Tensorflow* backend [34]. We have also installed, as a *Tensorflow* requirement, the cuDNN library for GPU-accelerated calculations. Hardware wise, the computer consisted of an Intel i7-8700k CPU, an NVIDIA 1080TI graphics card and 64GB of RAM.

A. PROPOSED NETWORK ARCHITECTURE

The proposed deep neural network takes as input a time-series of aggregated active power measurements $(P_{t_0}, P_{t_1}, \ldots, P_{t_n})$. It outputs the power of an individual appliance, the difference between the aggregated load and the disaggregated appliance, and the total predicted power, all at time (t_{n+1}).

For this, the network is structured with two main branches (Appliance and Difference) and a minor branch (Total). Each branch is composed of a one-dimensional convolution layer, followed by a GRU and a Fully-Connected layer. A Batch Normalization layer follows each kernel operation [35].

TABLE 1. Architectures and respective implementations of the benchmark deeep neural networks.

Architecture	Implementation
PB-NILM (simplified)	proposed
seq2point [37]	NILMTK
WindowGRU [38]	NILMTK

Batch Normalization is used this way to control the outputs at the end of each layer that performs a kernel operation. It also facilitates the use of larger gradients, helping to speed up the network. This is, in turn, followed by a ReLU activation.

Dropout [36] was also employed to help prevent overfitting. The dropout action forces the network to exclude learned connections and allows it not to focus on predominant connections, hence exploring other 'pathways' towards the final output.

Figure 1 represents the proposed network and the flow of the data. The network was structured to have three outputs in such a way that the outputs are ''controlled'' among themselves. With *Tensorflow*, it was possible to set up in a way that the *Total* branch consists of the sum of the *Appliance* and *Difference* branches, in turn affecting their learning. The *Appliance* and *Difference* branches are not directly connected.

B. BENCHMARK ARCHITECTURES

The PB-NILM is benchmarked against three different architectures. These, are summarized in Table 1.

The first benchmark algorithm is a simplified version of the proposed PB-NILM architecture. More concretely, in this version, only the appliance disaggregation branch is considered. The other two architectures were taken from the *NILMTK contrib* repository [39]. These were selected based on the similarity of purpose (*seq2point*) and architecture (*WindowGRU*).

The seq2point network architecture involves using a chain of five (5) convolutional layers, followed by two (2) fully connected layers to provide an output. This network also makes use of the dropout technique [36].

The WindowGRU network architecture is similar to the proposed one, as it also contains a convolutional layer. Yet, it is then followed by two (2) bidirectional GRU layers. Similarly to the seq2point network, the WindowGRU is then followed by two (2) fully connected layers to provide an output. This network architecture uses dropout [36].

Note that to make the WindowGRU architecture compatible with the ''cuDNN environment'', the GRU layers were replaced with the CuDNNGRU variant. Furthermore, the implementation of the bi-directional layers follows the cuDNN library requirements. Finally, each network was adapted to support the PB loss function. This involved using five dense layers in parallel, each representing a quantile value.

C. LOSS FUNCTIONS

The proposed and benchmark networks were implemented using two distinct loss functions, MSE, and PB.

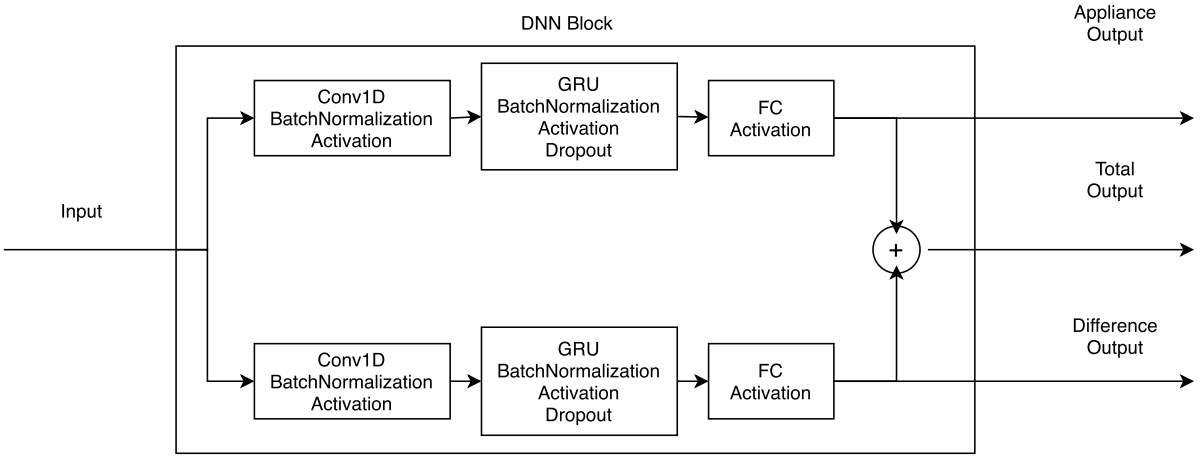

FIGURE 1. Block diagram showing the proposed network architecture.

1) MEAN SQUARED ERROR

The MSE is a widely used metric that can also serve as a loss function for training models dealing with regression problems. This function is defined by Equation 1, where Y_i stands for the ground truth value, $\hat{Y}_i$ stands for the predicted value, and n denotes the number of steps.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{1}$$

Due to the formulation of this function, it heavily penalizes larger errors. As such, it might not be suited to analyze appliances whose typical consumption patterns consist of sparse activation with high consumption such as toasters or a kettle. However, it is also easy to interpret and outputs a value in the units of the original data. In other words, it is possible to quantify the variance in the original measure (e.g., Watts for active power, or Amperes for current).

2) PINBALL

This function tries to take into account the distribution of the underlying data, an appliance in this case, and provide a more accurate prediction for potentially more unusual patterns. Retaking the kettle example, this function should be more capable of representing the small number of consumption events present in the entire dataset. The pinball function defined by Equation 2, where τ represents the desired quantile (a value between 0.01 and 0.99), and n represents the number of points taken into consideration, which in this case translates into the batch size.

$$pinball = \frac{\max(\tau \times (Y_i - \hat{Y}_i), (\tau - 1) \times (Y_i - \hat{Y}_i))}{n} \tag{2}$$

D. PB LOSS IMPLEMENTATION

The PB quantile loss function requires tuning for each appliance through the τ value, i.e., the quantile to estimate. In this work, it was opted to train a single network with the five τ values (0.05, 0.25, 0.5, 0.75, and 0.95). This approach reduces the training and testing efforts to only one network, and also ensures that the body of data remains the same, thus enabling a fair comparison between all the τ values.

With respect to the proposed architecture, it outputs the following values: the appliance consumption, the remaining consumption (i.e., total minus appliance consumption, and the sum of these two. Therefore, the loss function was implemented to accommodate these changes. Equation 3 presents the PB loss formulation in PB-NILM.

$$loss = \frac{\sum_{i=1}^{n} loss(A_i)}{n} + \frac{\sum_{i=1}^{n} loss(D_i)}{n} + \frac{\sum_{i=1}^{n} loss(T_i)}{n} \tag{3}$$

where $loss(A_i)$ is the loss for appliance A, $loss(D_i)$ is the loss for the difference between the aggregated consumption and appliance A, and $loss(T_i)$ is the loss of the total power. Finally, i represents an individual quantile, and n is the number of quantiles considered (five in this work).

As for the remaining architectures, since they are all single-branch, the loss consists only of the first term in Equation 3.

IV. PERFORMANCE EVALUATION METHODOLOGY

This section presents the overall evaluation methodology utilized in this work. More precisely, it describes the training and testing dataset, the respective training and testing procedures, and a set of metrics developed explicitly for NILM performance evaluation.

A. DATASET

All the presented deep neural networks were evaluated using House 2 of the REFIT dataset [40]. It consists of twenty-one (21) months of active power measurements for the whole house and nine individual appliances. Due to some inconsistencies in the dataset, some data pre-processing was required. More concretely: 1) the data were resampled to 8 seconds with interpolation to the nearest neighbor for missing data. And 2) The aggregated consumption was set to be the maximum between the original aggregated data and the sum of the

TABLE 2. List of appliances in the used dataset. % > 0 is the percentage of the data greater than zero watts for each appliance over the entire dataset.

Appliance	% > 0	Appliance	% > 0
1) Fridge-Freezer	0.991	2) Washing Machine	0.093
3) Dishwasher	0.062	4) Television Site	0.131
5) Microwave	0.006	6) Toaster	0.002
7) Hi-Fi	0.087	8) Kettle	0.009
9) Overhead Fan	0.005		

appliances to guarantee that the amount of power resulting from the sum of the individual loads is never higher than the aggregated consumption.

Table 2 lists the individual appliances, and the % of data points greater than zero Watts after data pre-processing. As can be observed, there are considerable differences in usage between devices. For example, while the Fridge-Freezer is active most of the time, the usage periods of appliances such as the Kettle are seldom. Appliances 6, 7 and 9 were not considered for this paper as the performance across all architectures for these devices was extremely poor. This most likely results from a combination of very low usage and low power consumption.

B. TRAINING AND TESTING

For training and testing, the data is divided into two contiguous blocks, following a 70%-30% division, respectively. The number of timesteps for the inputs of the network is set to 40. This number represents about five minutes of data in the selected dataset. *EarlyStopping* was employed to help prevent the overfitting of the models. *ReduceLROnPlateu* was also applied to have the model take smaller steps during the training, thus potentially resulting in the discovery of new minimum values for the loss function.

Related literature suggests that the performance of NILM algorithms is consistently higher when disaggregating the sum of known appliance-level signals instead of the real aggregated consumption (i.e., as measured at the mains) [41], [42]. Consequently, it was decided to train and evaluate the resulting models considering the two possibilities: 1) against the sum of the known individual loads (also referred to as artificial aggregate), and 2) against the real aggregate data.

The former reflects a scenario in which the input of the network does not have any noise. I.e., the ground-truth fully explains the aggregated data. As for the latter, it represents the real-world scenario, in which the ground-truth data only partially explains the whole-house consumption.

C. PERFORMANCE METRICS

Many performance metrics have been proposed to evaluate the performance of NILM algorithms [16], [43]. For example, the works mentioned in the previous section used metrics such as *F1-Score*, *Mean Absolute Error* (MAE), *Root Mean-Squared Error* (RMSE), and *Estimated Accuracy* (EA).

In this work, we propose the *Correctly Estimated Power* (CEP) metric. This metric translates into the amount of correctly assigned power during the active periods of energy consumption. Higher values indicate a better match of assigned power. To further analyze the results of an algorithm, other supplementary metrics are used and determine various other effects, such as under and overestimation. The *CEP* is defined in Equation 4:

$$CEP = \begin{cases} \dfrac{C + C_{ue} + C_{oe}}{GT}, & \text{GT} > 0 \\ 1.00, & GT = 0 \end{cases} \tag{4}$$

where C is the correctly identified power, i.e., when the ground-truth and estimated power is the same. Note that in the absence of power, *CEP* defaults to 1 by design, since there should be no energy to explain. When this occurs, we encourage further examination of the O_z metric, defined ahead.

$$C = \sum (Y == \hat{Y}) \times \hat{Y} \tag{5}$$

C_{ue} is the correctly identified power when the estimated power ($\hat{y}$) is smaller than the ground truth (gt). I.e., there is under-estimation.

$$C_{ue} = \sum (Y > \hat{Y}) \times Y \tag{6}$$

C_{oe} is the correctly identified power when the estimated power is greater than the ground truth (i.e., over-estimation).

$$C_{oe} = \sum (Y < \hat{Y}) \times \hat{Y} \tag{7}$$

GT is the total power in the ground truth data.

$$GT = \sum Y \tag{8}$$

The *CEP* metric is supplemented by the *OE* and *UE* metrics, where *OE* is the ratio of overestimated power, and *UE* is the ratio of underestimated power. These metrics are calculated using the values of overestimation, defined as O, and underestimation, defined as U.

$$O = \sum (\hat{Y} - Y) \tag{9}$$

$$U = \sum (Y - \hat{Y}) \tag{10}$$

$$OE = \begin{cases} \dfrac{O}{GT}, & GT > 0 \\ \textit{not applicable}, & \text{otherwise} \end{cases} \tag{11}$$

$$UE = \frac{U}{GT} \tag{12}$$

Note that *OE* is only defined when there is consumption in the ground-truth. To accommodate the situations where there is no consumption in the ground-truth, the O_z metric was defined as follows:

$$O_z = \begin{cases} \dfrac{O}{GT}, & \text{Y} = 0 \ \ \& \ \ GT > 0 \\ \dfrac{\sum \hat{Y}}{n}, & GT = 0 \end{cases} \tag{13}$$

O_z takes the value of the *OE* when the ground-truth is equal to zero. In every other case, O_z translates into the average of power that was predicted over the period when the ground-truth was zero. We propose this metric as decoupled from the

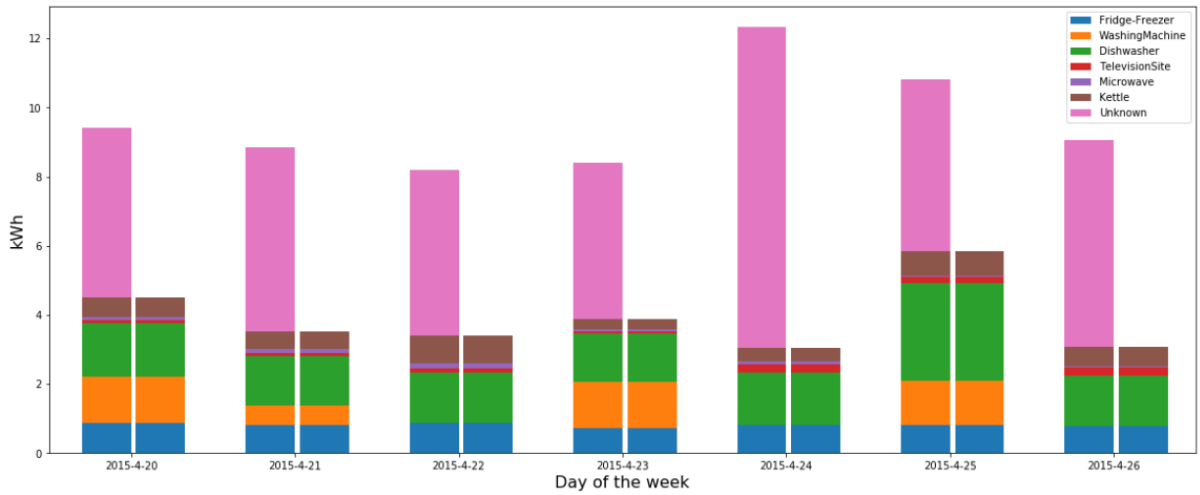

FIGURE 2. Energy over the week 2015-04-20 to 2015-04-26 on the two scenarios.

rest as to better analyze the results when there is no energy to estimate, i.e., when an appliance is not consuming. This metric is not limited in values since O_z can exceed 1.

V. RESULTS

The performance evaluation results are reported in two steps. First, the performance of the proposed network is presented individually; then, extensive benchmarks with the three alternative architectures are provided. The performance is reported for a week in the testing set - from 2015-04-20 to 2015-04-26. An overview of the consumption during this week is given in Figure 2.

For a more direct comparison with the *MSE*, only the results for the best pinball are presented. To this end, the best pinball was obtained by calculating the distance between the vector $\vec{s}$ given by $\vec{s} = (CEP, OE, O_z, UE)$ and the vector $\vec{p}$ with the perfect scores, given by $\vec{p} = (1, 0, 0, 0)$ [44], [45]. The pinball with the smallest distance was considered the best. In case of ties, the highest pinball value was selected. The distance is given by Equation 14.

$$d(\vec{s}, \vec{p}) = \sqrt{\sum_{i=1}^{n} (s_i - p_i)^2} = \sqrt{(1 - CEP)^2 + OE^2 + {O_z}^2 + UE^2} \quad (14)$$

A. PB-NILM ARCHITECTURE

For the proposed network evaluation, the metrics *CEP*, *OE*, O_z and *UE* are presented for each scenario (artificial and real aggregated consumption) and loss function (PB and MSE). The metrics were calculated per appliances for each day in the testing set.

1) MSE: SUM OF INDIVIDUAL LOADS

The results for the MSE in the sum of individual loads are presented in Table 3.

The Fridge is active over the entire week and achieves *CEP* values of 0.93 and above. In terms of overestimation, it performs adequately both in where ground-truth was present and when it was not. The worst results are in the days 22 and 25, where overestimation occurred in the order of 11% and 15%, respectively.

The Washing Machine reports mixed values for *CEP*. The days 20, 21, 23, and 25 present acceptable results, while days 22, 24, and 26 show an uncommon behavior. Particularly on days 22 and 26, where there is no power on the ground-truth values, but estimation took place anyway. On these days, as shown by the O_z metric, the amount reported is an average of the wrongly estimated power. Finally, on the day 24, the values obtained were the result of the poor quality of the data - a consumption of two watts over the entire day - and resulting in the O_z being the percentage of the estimated against the ground-truth and outputting a result very different from the rest.

The Dishwasher performed both well and consistently, with *CEP* values of 0.89 and above, while the overestimation was always between 0.01-0.02. In terms of O_z, the values obtained are also considered good. They are at a maximum of 0.07 and typically in the range of 0.01-0.04.

The Television Site is the overall most problematic appliance - good values of *CEP*, *OE*, at the cost of very high scores of O_z. This means that the network attributed power to the appliance at the wrong time. This is particularly visible on days 20 and 23, where almost all of the power is being assigned to the moments where no consumption occurred.

The Microwave performs poorly, with *CEP* values almost exclusively under 0.6. Combined with the high *OE* and O_z make this appliance, together with the Television Site, the worst-performing loads.

The Kettle is a very consistent appliance, with high *CEP* values, a small amount of overestimation, and almost no O_z. This means that the power estimated by the network is mostly correct, both in the time and energy domains.

TABLE 3. Results of disaggregation using the MSE loss on the *Sum of Loads* scenario.

	Fridge-Freezer				Washing Machine				Dishwasher			
Day	CEP	OE	O_z	UE	CEP	OE	O_z	UE	CEP	OE	O_z	UE
2015-04-20	0.93	0.07	0.00	0.07	0.90	0.09	0.01	0.10	0.90	0.01	0.04	0.10
2015-04-21	0.95	0.07	0.00	0.05	0.78	0.05	0.04	0.22	0.90	0.02	0.07	0.10
2015-04-22	0.97	0.11	0.00	0.03	1.00	0.00	0.97	0.00	0.94	0.02	0.02	0.06
2015-04-23	0.94	0.09	0.00	0.06	0.87	0.06	0.03	0.13	0.89	0.01	0.04	0.11
2015-04-24	0.94	0.06	0.00	0.06	0.00	0.00	13371.5	1.00	0.89	0.01	0.01	0.11
2015-04-25	0.93	0.15	0.00	0.07	0.89	0.07	0.02	0.11	0.91	0.02	0.02	0.09
2015-04-26	0.94	0.05	0.00	0.06	1.00	0.00	0.44	0.00	0.93	0.01	0.01	0.07
	Television Site				Microwave				Kettle			
Day	CEP	OE	O_z	UE	CEP	OE	O_z	UE	CEP	OE	O_z	UE
2015-04-20	0.57	0.01	0.91	0.43	0.54	0.14	0.10	0.46	0.86	0.06	0.00	0.14
2015-04-21	0.73	0.00	0.32	0.27	0.63	0.16	0.10	0.37	0.86	0.05	0.01	0.14
2015-04-22	0.85	0.05	0.20	0.15	0.58	0.19	0.22	0.42	0.86	0.05	0.03	0.14
2015-04-23	0.78	0.01	0.90	0.22	0.46	0.21	0.20	0.54	0.82	0.07	0.03	0.18
2015-04-24	0.87	0.00	0.11	0.13	0.52	0.16	0.04	0.48	0.88	0.07	0.00	0.12
2015-04-25	0.67	0.01	0.26	0.33	0.38	0.36	0.33	0.62	0.84	0.05	0.01	0.16
2015-04-26	0.89	0.00	0.08	0.11	0.52	0.16	0.19	0.48	0.84	0.07	0.01	0.16

TABLE 4. Results of disaggregation using the Pinball Quantile loss on the *Sum of Loads* scenario.

	Fridge-Freezer					Washing Machine					Dishwasher				
Day	Qt	CEP	OE	O_z	UE	Qt	CEP	OE	O_z	UE	Qt	CEP	OE	O_z	UE
2015-04-20	0.50	0.96	0.04	0.00	0.04	0.50	0.94	0.06	0.00	0.06	0.75	0.98	0.02	0.02	0.02
2015-04-21	0.50	0.97	0.03	0.00	0.03	0.75	0.95	0.08	0.00	0.05	0.75	0.98	0.02	0.04	0.02
2015-04-22	0.25	0.97	0.03	0.00	0.03	0.25	1.00	0.00	0.00	0.00	0.75	0.98	0.03	0.00	0.02
2015-04-23	0.50	0.97	0.04	0.00	0.03	0.75	0.94	0.09	0.01	0.06	0.75	0.98	0.02	0.02	0.02
2015-04-24	0.50	0.97	0.04	0.00	0.03	0.25	0.00	0.00	0.00	1.00	0.75	0.98	0.02	0.00	0.02
2015-04-25	0.50	0.96	0.07	0.00	0.04	0.75	0.94	0.10	0.00	0.06	0.75	0.97	0.02	0.01	0.03
2015-04-26	0.50	0.97	0.02	0.00	0.03	0.25	1.00	0.00	0.00	0.00	0.75	0.98	0.02	0.00	0.02
	Television Site					Microwave					Kettle				
Day	Qt	CEP	OE	O_z	UE	Qt	CEP	OE	O_z	UE	Qt	CEP	OE	O_z	UE
2015-04-20	0.75	0.84	0.04	0.69	0.16	0.75	0.74	0.29	0.07	0.26	0.75	0.92	0.10	0.00	0.08
2015-04-21	0.75	0.87	0.03	0.04	0.13	0.75	0.77	0.30	0.07	0.23	0.75	0.93	0.08	0.00	0.07
2015-04-22	0.50	0.95	0.09	0.05	0.05	0.75	0.76	0.35	0.16	0.24	0.75	0.92	0.08	0.02	0.08
2015-04-23	0.50	0.94	0.01	0.01	0.06	0.75	0.74	0.34	0.16	0.26	0.75	0.89	0.10	0.03	0.11
2015-04-24	0.75	0.98	0.02	0.04	0.02	0.75	0.61	0.25	0.01	0.39	0.75	0.92	0.11	0.00	0.08
2015-04-25	0.75	0.85	0.02	0.04	0.15	0.75	0.55	0.59	0.23	0.45	0.75	0.92	0.08	0.00	0.08
2015-04-26	0.75	0.99	0.01	0.01	0.01	0.75	0.75	0.33	0.16	0.25	0.75	0.90	0.10	0.01	0.10

Overall, when using the MSE loss function on the sum of the individual loads, acceptable results are achieved. The poor results for the Television Site and Microwave could result from a combination of the shallow representation of the appliance on the dataset and the low consumption associated with it. In contrast, while the Kettle is also underrepresented, it shows higher consumption peaks (see Table 2).

2) PINBALL: SUM OF INDIVIDUAL LOADS

The results for the pinball in the sum of individual loads are presented in Table 4.

As can be observed, the overall performance of the trained models is positive. The results for the Fridge-Freezer remain consistently good, with *CEP* values of 0.96-0.97. This appliance also yields very few overestimation both in terms of OE, and O_z.

The Washing Machine performs very well using the pinball quantile loss function, with an exception in day 24. Still, this happens due to noise in the ground-truth data, which shows only two (2) Watts of consumption during the day. Furthermore, the absence of O_z is a critical advantage over the MSE loss, since the ground-truth values are very likely the result of a data acquisition error.

TABLE 5. Results of disaggregation using the MSE loss on the *Aggregate* scenario.

	Fridge-Freezer				Washing Machine				Dishwasher			
Day	CEP	OE	O_z	UE	CEP	OE	O_z	UE	CEP	OE	O_z	UE
2015-04-20	0.62	0.24	0.00	0.38	0.71	0.10	0.03	0.29	0.73	0.01	0.13	0.27
2015-04-21	0.66	0.20	0.00	0.34	0.55	0.04	0.12	0.22	0.78	0.01	0.14	0.22
2015-04-22	0.66	0.28	0.00	0.34	1.00	0.00	5.76	0.00	0.73	0.01	0.02	0.27
2015-04-23	0.60	0.28	0.00	0.40	0.75	0.06	0.05	0.25	0.67	0.01	0.11	0.33
2015-04-24	0.64	0.28	0.00	0.36	0.00	0.00	132450.5	1.00	0.67	0.01	1.19	0.33
2015-04-25	0.59	0.40	0.00	0.41	0.58	0.08	0.07	0.42	0.72	0.01	0.14	0.28
2015-04-26	0.61	0.28	0.00	0.39	1.00	0.00	3.95	0.00	0.64	0.01	0.16	0.36
	Television Site				Microwave				Kettle			
Day	CEP	OE	O_z	UE	CEP	OE	O_z	UE	CEP	OE	O_z	UE
2015-04-20	0.25	0.01	0.80	0.75	0.41	0.12	0.50	0.59	0.76	0.06	0.07	0.24
2015-04-21	0.17	0.00	0.70	0.83	0.51	0.14	0.30	0.49	0.79	0.04	0.07	0.21
2015-04-22	0.14	0.00	0.55	0.86	0.41	0.16	1.03	0.59	0.80	0.05	0.06	0.20
2015-04-23	0.27	0.00	1.61	0.73	0.44	0.14	0.72	0.56	0.73	0.08	0.19	0.27
2015-04-24	0.22	0.00	0.18	0.78	0.25	0.11	0.69	0.75	0.79	0.08	0.12	0.21
2015-04-25	0.17	0.00	0.36	0.83	0.26	0.20	1.20	0.74	0.70	0.05	0.18	0.30
2015-04-26	0.12	0.00	0.18	0.88	0.42	0.15	0.65	0.58	0.75	0.05	0.09	0.25

The Dishwasher presents very high values of *CEP*, together with low *OE* and O_z. With the pinball quantile loss, this is the highest-performing appliance. Overall, this appliance is fairly consistent over the entire week.

With the pinball quantile loss, the Television Site performs remarkably better when compared with its MSE counterpart, with overall *CEP* values much higher and lower *OE* and O_z.

The Kettle remains a very good appliance, with consistent values of *CEP* throughout the week. There are small increases in the *OE* values against the MSE loss. However, the overall O_z is lower than in the MSE loss.

3) MSE: AGGREGATE CONSUMPTION

The results for the MSE in the aggregate consumption are presented in Table 5.

Against the aggregate consumption, the *CEP* values are overall worse than on the Sum of total loads scenario. The tendency on the aggregate scenario is to report lower *CEP*, as well as higher *OE* and O_z.

The Fridge reflects the overall characteristics of the scenario, with lower values of *CEP*, although consistent - ranging from 0.59 to 0.66. The results for *OE* show a significant increase, especially on day 25, with a value of 0.40.

The Washing Machine follows the same trend, with values of *CEP* showing a decrease relative to the sum of loads scenario, with a slight increase of *OE* and a very significant rise in O_z. Notably, on day 24, this appliance had only two watts to be predicted, resulting in an abnormally large value of O_z.

The Dishwasher was less impacted by the scenario settings and presented values of *CEP* in the range of 0.64 to 0.78. This comes with little overestimation, although higher O_z is reported.

The Television Site, one of the most problematic in the previous scenario, continues to underperform. This is shown by a very low value of *CEP*, coupled with very high values of O_z.

The Microwave, like Television Site, reflects drastic changes in the scenario. It reports lower *CEP* and higher *OE* and O_z, especially on the days 22 and 25.

While the Kettle decreased the performance with respect to the *CEP* metric, the results it still acceptable and show a minimal increase of *OE*. However, O_z is now considerably higher. Overall this is the best performing appliance using the MSE loss.

4) PINBALL: AGGREGATE CONSUMPTION

Against the aggregate consumption, the pinball quantile loss function remains a solid alternative to the standard MSE loss.

The results for the Fridge-Freezer remain reasonably high, with *CEP* values of 0.75 to 0.88. The overestimation, however, increased as well. The values of *OE* roughly follow the same increase as in the MSE case.

The Washing Machine is an average appliance, with a wide range of *CEP* values. One key advantage against the MSE loss, however, is that on day 24, there was no consumption predicted - more in line with the ground-truth value of 2 Watts.

The Dishwasher yields high values of *CEP* in this scenario. Yet, this comes with a caveat - higher *OE* and O_z as well.

The Television Site continues to display poor values of *CEP*. The notable exception is on day 24 when the quantile 0.95 was selected. This results in a very high value for *CEP*. Yet, this increase arrives at the cost of an increase of *OE*, and O_z.

TABLE 6. Results of disaggregation using the Pinball Quantile loss on the *Aggregate* scenario.

	Fridge-Freezer					Washing Machine					Dishwasher				
Day	Qt	CEP	OE	O_z	UE	Qt	CEP	OE	O_z	UE	Qt	CEP	OE	O_z	UE
2015-04-20	0.75	0.79	0.29	0.00	0.21	0.75	0.80	0.11	0.00	0.20	0.75	0.90	0.01	0.11	0.10
2015-04-21	0.75	0.87	0.23	0.00	0.13	0.75	0.65	0.09	0.16	0.35	0.75	0.88	0.02	0.26	0.12
2015-04-22	0.75	0.88	0.28	0.00	0.12	0.25	1.00	0.00	0.00	0.00	0.95	0.96	0.07	0.11	0.04
2015-04-23	0.75	0.76	0.32	0.00	0.24	0.75	0.90	0.11	0.06	0.10	0.75	0.86	0.02	0.02	0.14
2015-04-24	0.50	0.75	0.32	0.00	0.25	0.25	0.00	0.00	0.00	1.00	0.25	0.73	0.01	1.11	0.27
2015-04-25	0.75	0.78	0.47	0.00	0.22	0.75	0.70	0.11	0.03	0.30	0.75	0.87	0.01	0.14	0.13
2015-04-26	0.75	0.83	0.36	0.00	0.17	0.25	1.00	0.00	0.00	0.00	0.75	0.77	0.02	0.20	0.23
	Television Site					Microwave					Kettle				
Day	Qt	CEP	OE	O_z	UE	Qt	CEP	OE	O_z	UE	Qt	CEP	OE	O_z	UE
2015-04-20	0.75	0.19	0.01	0.08	0.81	0.75	0.54	0.23	0.11	0.46	0.75	0.92	0.09	0.01	0.08
2015-04-21	0.75	0.13	0.01	0.04	0.87	0.75	0.68	0.26	0.11	0.32	0.75	0.95	0.08	0.02	0.05
2015-04-22	0.75	0.05	0.00	0.13	0.95	0.75	0.55	0.27	0.69	0.45	0.75	0.94	0.10	0.03	0.06
2015-04-23	0.75	0.26	0.01	0.07	0.74	0.75	0.62	0.24	0.18	0.38	0.75	0.89	0.13	0.12	0.11
2015-04-24	0.95	0.91	0.14	1.09	0.09	0.75	0.34	0.19	0.30	0.66	0.75	0.94	0.14	0.03	0.06
2015-04-25	0.75	0.02	0.00	0.10	0.98	0.75	0.47	0.40	0.18	0.53	0.75	0.91	0.09	0.02	0.09
2015-04-26	0.95	0.72	0.09	0.91	0.28	0.75	0.55	0.22	0.36	0.45	0.75	0.88	0.07	0.02	0.12

TABLE 7. Median of best results for each appliance over all network architectures - sum of loads scenario.

	pb-NILM		pb-NILM (simplified)		seq2point		WindowGRU	
Appliance	MSE	PB	MSE	PB	MSE	PB	MSE	PB
Fridge-Freezer	0.11	0.06	0.12	**0.05**	0.11	0.11	0.13	0.08
Washing Machine	0.32	**0.11**	0.30	0.16	0.56	0.37	0.45	0.22
Dishwasher	0.14	**0.04**	0.18	0.08	0.16	0.10	0.12	0.32
Television Site	0.5	**0.13**	0.29	0.14	0.41	0.27	0.22	**0.13**
Microwave	0.7	**0.51**	0.67	1.12	0.83	1.41[a]	1.41[a]	1.41[a]
Kettle	0.21	**0.15**	0.23	0.16	0.28	**0.15**	0.26	0.17
Average	0.33 (0.2)	**0.17 (0.2)**	0.30 (0.2)	0.29 (0.4)	0.39 (0.3)	0.40 (0.5)	0.43 (0.5)	0.39 (0.5)

[a] 1.41 is the distance when a network is not able to learn a model for a specific appliance, and estimates 0 Watts in every time step instead.

The Microwave reports low values of *CEP*, as well as high values of *OE* and O_z. An interesting result was the selection of the 0.75 quantile over the entire week, suggesting this quantile may be very well suited for this appliance in particular.

The Kettle is the best performing appliance in this scenario utilizing the pinball quantile loss. It reports consistent values of *CEP* ($\geq$ 0.88), as well as minimal increases in the *OE*, and O_z metrics.

B. BENCHMARKS

Two different benchmarks are provided. The first compares the overall performance of the four disaggregation architectures, whereas the second compares the performance of the loss functions.

Regarding the former, the median of the best distances over the testing week is computed for each appliance. This is done for each scenario, for a total of 12 comparisons per architecture (i.e., six appliances and two loss functions). As for the latter, the distances between the best PB and the MSE were compared. This was done for each scenario, for a total of 42 comparisons (i.e., six appliances across seven days). In case of ties, i.e., the same distance between the PB and MSE, this information is labelled as such.

1) ARCHITECTURES: PROPOSED VS. OTHERS

The performances of the four networks architectures evaluated in this work are presented in Tables 7 and 8. For each appliance, the loss function with the shortest distance across the four architectures is shown on a grey background. The overall best loss function is highlighted in bold.

Overall, the results show the superior performance of the proposed architecture when using the PB loss function. In fact, out of 48 comparisons (6 appliances x 4 networks x 2 scenarios), it shows better performance on 45 occasions (93.75%). Furthermore, it is important to remark that on the other three circumstances, the winning architecture is also guided by the PB loss function.

TABLE 8. Median of best results for each appliance over all network architectures - aggregate scenario.

	pb-NILM		pb-NILM (simplified)		seq2point		WindowGRU	
Appliance	MSE	PB	MSE	PB	MSE	PB	MSE	PB
Fridge-Freezer	0.59	0.43	0.62	0.42	0.54	**0.40**	0.59	0.43
Washing Machine	0.65	**0.31**	0.71	0.74	0.9	0.76	0.85	0.34
Dishwasher	0.42	**0.22**	0.7	0.32	0.56	0.35	0.47	0.34
Television Site	1.33	**1.15**	1.24	1.24	1.28	1.38	1.24	1.41[b]
Microwave	1.08	0.77	1.01	**0.66**	1.07	1.41[a]	1.41[a]	1.41[a]
Kettle	0.35	**0.15**	0.38	0.18	0.45	0.16	0.39	1.41[a]
Average	0.74 (0.4)	**0.51 (0.4)**	0.78 (0.3)	0.59 (0.4)	0.80 (0.3)	0.74 (0.5)	0.83 (0.4)	0.89 (0.6)

[a] 1.41 is the distance when a network is not able to learn a model for a specific appliance, and estimates 0 Watts in every time step instead.

[b] In this appliance, the network was able to learn a model in the 0.95 quantile. The distance was 1.26 and 1.08 for days 24 and 26, respectively.

Examining the average distance of the appliances in each network, it can be observed that in both scenarios, the most effective networks are the proposed one and its single branch version, independently of the employed loss function.

Another relevant aspect of these two networks is that they are capable of learning all of the appliances. This does not happen on the other two alternatives. More precisely, the WindowGRU did not manage to learn the Microwave independently of the loss function and the disaggregation scenario. Furthermore, this architecture also did not manage to learn a representation for the Kettle in the aggregated scenario with the PB loss. As for the seq2point architecture, it did not manage to disaggregate the microwave when using the PB loss in both disaggregation scenarios.

2) LOSS FUNCTIONS: PB VS. MSE

To further understand the differences in performance between the PB and the MSE, Figure 3 shows the distribution of the winning across the four networks in the two studied scenarios. Note that each bar represents 41 comparisons, resulting from six appliances and seven days.

As it can be observed there is a prevalence of wins for the PB loss across all the networks. However, the difference is not so evident in the seq2point and WindowGRU architectures. This happens mostly due to the poor performance of the PB loss function concerning the Kettle and the Television on the WindowGRU, and the Microwave on the seq2point.

Examining the distribution of the losses in more detail, it is shown that on the sum of loads scenario, the MSE loss achieved zero wins in the proposed network, nine on the single branch version, 11 on the seq2point, and seven on the WindowGRU. On the aggregate scenario, the MSE achieved two wins on the proposed network, ten on the single branch version, ten on the seq2point, and 14 on the WindowGRU. There are also two ties in the seq2point, and seven on the WindowGRU, in both scenarios.

Regarding the distribution of the winning pinball loss quantiles, the most prominent one is the 0.75, with 46% of the wins in the sum of loads, and 55% on the real aggregate scenario. Another prominent quantile is 0.5. However, this is only true in the artificial aggregate scenario, where it wins 26% of the time.

On the other extreme are the smaller quantiles, 0.05 and 0.25, with less than 5% of wins independently of the scenario. Finally, it is also noticeable the presence of the 0.95 quantile across the four networks in the real-aggregated scenario (10% of wins), contrasting the artificial aggregate scenario where the 0.95 quantile only wins in one occasion in the seq2point architecture.

VI. DISCUSSION

Overall, the results presented in this paper show the superior performance of the proposed network, in particular of the pinball guided version that achieved nine wins out of 12 in the benchmarks.

Our results also show that despite much superior performance of the proposed architecture guided by the PB loss, both the MSE and PB loss functions perform adequately against the sum of loads. Yet, the same does not hold against the aggregated consumption signal that shows noticeable losses in performance.

A noticeable appliance that suffers from the change in scenario is the Fridge-Freezer that sees increases in the distance to the perfect score from less than 0.09 to over 0.4 across the four networks when guided by the PB loss function (from 0.11 to 0.62 in the MSE). Another appliance that suffered greatly was the Dishwasher, with the median distance to perfect score going from as little as 0.04 up to 0.35 when guided by the PB (0.12 to 0.56 in the MSE).

Interestingly, the Kettle is the only appliance that has comparable performances in the two scenarios (note that the WindowGRU does not learn the Kettle in the PB guided version). A possible explanation for this result is much higher instantaneous consumption of this appliance when compared to the others, which makes it less susceptible to the increase in power in the real aggregate scenario.

Concerning loss functions, the results show that the PB loss function is very competitive independently of the deep network architecture. Overall, the PB loss is the winner in about 80% of the comparisons. Again here, there are some

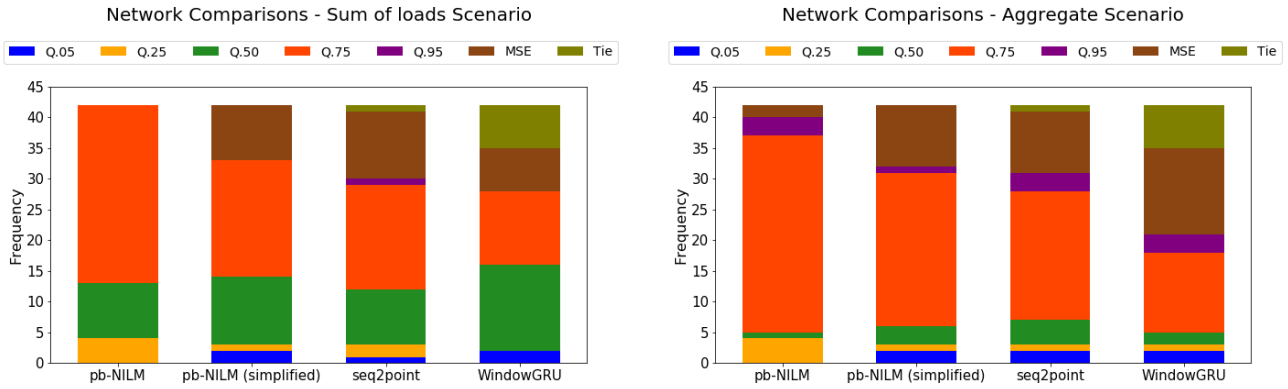

FIGURE 3. Loss functions with the best distance across network architectures and scenarios: left (sum of loads), right (real aggregate).

differences between the two scenarios. The PB wins 84% of the cases in the sum of loads and 79% in the real aggregate. Another noticeable difference is the fact that the 0.5 quantile is only predominant in the artificial aggregate scenario. Instead, in the real aggregate scenario, going for higher quantiles (in this case, 0.75 and 0.95) will yield superior performances.

On the other hand, smaller quantiles (0.05 and 0.25) have a meager number of wins independently of the scenario. These quantiles only win when the appliances are not consuming any energy or are consuming very little. For example, in the proposed architecture, the 0.25 quantile wins on the three days that the washing machine does not consume any power.

Ultimately, higher quantiles can be considered as a more liberal approach to power disaggregation, whereas low quantiles are more conservative in this regard. With the pinball loss formulation, if the predicted values are much higher than the ground-truth, bigger quantiles will penalize less the error (I.e., incentives overestimation). The opposite is also true. I.e., when the ground-truth values are higher than the predictions, lower quantile values will provide lesser penalties in regards to the loss (i.e., incentives underestimation). Thus, to consider all the options, it is fundamental to have quantiles across the entire range of values. In fact, our results seem to suggest that quantiles should be selected between 0.25 and 0.95 to have the best performance.

To conclude, it is also evident from our results that evaluating the performance of NILM solutions in the sum of loads will yield overly optimistic results. Furthermore, the performance in the ''denoised'' scenario will likely be very far from that obtained when testing in the real-world scenario, which will ultimately make benchmarks less meaningful. This result is in line with the findings from [42] that stress the need to avoid evaluations on the sum of loads instead of the real aggregated data.

VII. CONCLUSION AND FUTURE WORK DIRECTIONS

This paper proposes a deep neural network architecture, an alternative loss function to the NILM problem, and a set of metrics for NILM performance evaluation.

Overall, our results show that the proposed network, when guided by the PB loss function, yields disaggregation performance. Furthermore, the benchmark results also show that the PB loss function can increase the performance of two state-of-the-art algorithms implemented in the latest release of NILM Toolkit. Ultimately, it is clear the PB loss function has its room in NILM research, and therefore should be further explored by this community.

Furthermore, a great deal of the NILM problem lies in the uncertainty and constant variation associated with the data and the respective usage patterns. Thus, we argue that given the flexibility of the PB quantile loss function, that is, the ability to specify the τ value in regards to the desired pattern makes it so that it has the edge over other conventional approaches.

While the proposed network performs reasonably well in most cases, we acknowledge that it may be further optimized and tuned for increased performance. One immediate improvement would be concerning the temporal processing by tuning the timesteps for each appliance type (or category). In terms of changes to the architecture, another possibility would be to improve the feature extraction step, by making use of auto-encoders [46].

Concerning the loss functions, we report superior results using the pinball quantile loss on both scenarios. Yet, in the current work, only a limited number of quantile values were explored. Consequently, future work should investigate the possibility of finding the recommended quantiles for each appliance type or category. This, of course, should be performed across multiple datasets such that it is possible to generalize the findings.

This work also proposes a set of metrics for assessing the NILM performance concerning energy estimation. Altogether, these metrics contribute to the interpretability of the results since they can be combined to provide different insights into the results. Potential uses include highlighting faults in the *pipeline*, and finding opportunities to fine-tune the algorithms based on the application needs. For example, in a scenario where over-estimation is a problem, the algorithm should be modified such that the *OE* metric reports lower values even if the *CEP* is penalized.

Yet, one limitation of the *CEP* metric is that it gives the same importance to the three components. Thus, future work should also explore the definition of weighted versions of this metric.

Another limitation of *CEP* is the fact that it only reports the performance concerning the success rate. As such, proper conclusions must take into account other metrics such as *OE* and O_z. Future work regarding the evaluation of the metric suite lies in providing feedback in a more summarised fashion. Consequently, future work with this respect should look at expanding this metric such that it is capable of reporting by itself the most important trade-offs.

During this work, some issues with the data were identified, which may have negatively affected the performance of the proposed networks. These included temporal shifts and differences in power consumption between aggregated and ground-truth data. For example, it was identified that in some portions of the dataset, the Fridge-Freezer has a consumption of 80 Watts in the ground-truth and 70 Watts in the aggregated signal. Ultimately, this may help explain the consistent over-estimation of the fridge in the second evaluation scenario.

APPENDIX: SUPPLEMENTARY MATERIAL

For replication purposes, we are releasing the individual results obtained in the different models trained and tested in this work. The results of the benchmarks that were conducted across the four architectures are also provided. The data is made available through the Open Science Framework under https://osf.io/x6qz7/ (DOI: 10.17605/OSF.IO/X6QZ7).

REFERENCES

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

[2] N. F. Esa, M. P. Abdullah, and M. Y. Hassan, "A review disaggregation method in non-intrusive appliance load monitoring," *Renew. Sustain. Energy Rev.*, vol. 66, pp. 163–173, Dec. 2016.

[3] B. Najafi, S. Moaveninejad, and F. Rinaldi, "Data analytics for energy disaggregation: Methods and applications," in *Big Data Application in Power Systems*, R. Arghandeh and Y. Zhou, Eds. Amsterdam, The Netherlands: Elsevier, 2018, ch. 17, pp. 377–408. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128119686000176

[4] C. Nalmpantis and D. Vrakas, "Machine learning approaches for non-intrusive load monitoring: From qualitative to quantitative comparation," *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 217–243, Jan. 2018. [Online]. Available: https://link.springer.com/article/10.1007/s10462-018-9613-7

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015. [Online]. Available: https://www.nature.com/articles/nature14539

[6] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[7] J. Engel, K. K. Agrawal, S. Chen, I. Gulrajani, C. Donahue, and A. Roberts, "GANSynth: Adversarial neural audio synthesis," Feb. 2019, *arXiv:1902.08710*. [Online]. Available: http://arxiv.org/abs/1902.08710

[8] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," Sep. 2016, *arXiv:1609.03499*. [Online]. Available: http://arxiv.org/abs/1609.03499

[9] S. Chen, K. Ma, and Y. Zheng, "Med3d: Transfer learning for 3D medical image analysis," Apr. 2019, *arXiv:1904.00625*. [Online]. Available: https://arxiv.org/abs/1904.00625

[10] J. Kim, T.-T.-H. Le, and H. Kim, "Nonintrusive load monitoring based on advanced deep learning and novel signature," *Comput. Intell. Neurosci.*, vol. 2017, pp. 1–22, Oct. 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5651160/

[11] D. Murray, L. Stankovic, V. Stankovic, S. Lulic, and S. Sladojevic, "Transferability of neural network approaches for low-rate energy disaggregation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 8330–8334.

[12] W. Kong, Z. Y. Dong, B. Wang, J. Zhao, and J. Huang, "A practical solution for non-intrusive type II load monitoring based on deep learning and post-processing," *IEEE Trans. Smart Grid*, vol. 11, no. 1, pp. 148–160, Jan. 2020.

[13] A. Harell, S. Makonin, and I. V. Bajić, "Wavenilm: A causal neural network for power disaggregation from the complex power signal," Feb. 2019, *arXiv:1902.08736*. [Online]. Available: http://arxiv.org/abs/1902.08736

[14] Q. Wu and F. Wang, "Concatenate convolutional neural networks for non-intrusive load monitoring across complex background," *Energies*, vol. 12, no. 8, p. 1572, Apr. 2019.

[15] P. Davies, J. Dennis, J. Hansom, W. Martin, A. Stankevicius, and L. Ward, "Deep neural networks for appliance transient classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 8320–8324.

[16] L. Pereira and N. Nunes, "Performance evaluation in non-intrusive load monitoring: Datasets, metrics, and tools—A review," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 8, no. 6, p. e1265, May 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/widm.1265

[17] C. Klemenjak, A. Reinhardt, L. Pereira, M. Berges, S. Makonin, and W. Elmenreich, "Electricity consumption data sets: Pitfalls and opportunities," in *Proc. 6th ACM Int. Conf. Syst. Energy-Efficient Buildings, Cities, Transp. (BuildSys)*, 2019, pp. 159–162.

[18] I. Steinwart and A. Christmann, "Estimating conditional quantiles with the help of the pinball loss," *Bernoulli*, vol. 17, no. 1, pp. 211–225, Feb. 2011.

[19] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4580–4584.

[20] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol. Cybern.*, vol. 36, no. 4, pp. 193–202, Apr. 1980.

[21] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in *Shape, Contour and Grouping in Computer Vision*. London, U.K.: Springer-Verlag, 1999, p. 319.

[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[23] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Dec. 2015, *arXiv:1512.03385*. [Online]. Available: http://arxiv.org/abs/1512.03385

[25] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," Aug. 2016, *arXiv:1608.06993*. [Online]. Available: http://arxiv.org/abs/1608.06993

[26] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," Oct. 2016, *arXiv:1610.02357*. [Online]. Available: http://arxiv.org/abs/1610.02357

[27] S. Makonin, B. Ellert, I. V. Bajić, and F. Popowich, "Electricity, water, and natural gas consumption of a residential house in canada from 2012 to 2014," *Sci. Data*, vol. 3, no. 1, pp. 1–12, Jun. 2016.

[28] J. Kelly and W. Knottenbelt, "The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes," *Sci. Data*, vol. 2, no. 1, Mar. 2015, Art. no. 150007.

[29] J. Z. Kolter and M. J. Johnson, "REDD: A public data set for energy disaggregation research," in *Proc. Workshop Data Mining Appl. Sustainability (SIGKDD)*, San Diego, CA, USA, vol. 25, 2011, pp. 59–62.

[30] J. Gao, S. Giri, E. C. Kara, and M. Bergés, "PLAID: A public dataset of high-resoultion electrical appliance measurements for load identification research: Demo abstract," in *Proc. 1st ACM Conf. Embedded Syst. Energy-Efficient Buildings*, 2014, pp. 198–199.

[31] T. Hong, P. Pinson, S. Fan, H. Zareipour, A. Troccoli, and R. J. Hyndman, "Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond," *Int. J. Forecasting*, vol. 32, no. 3, pp. 896–913, Jul. 2016.
[32] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," *Appl. Energy*, vol. 235, pp. 10–20, Feb. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306261918316465
[33] F. Chollet *et al.* (2015). Keras: The Python deep learning library. GitHub. [Online]. Available: https://keras.io
[34] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Oper. Syst. Design Implement. (OSDI)*, 2016, pp. 265–283.
[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: http://arxiv.org/abs/1502.03167
[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.
[37] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," in *Proc. 32nd AAAI Conf. Artif. Intell.*, New Orleans, LA, USA, 2018, pp. 1–8. [Online]. Available: https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16623/15980
[38] O. Krystalakos, C. Nalmpantis, and D. Vrakas, "Sliding window approach for online energy disaggregation using artificial neural networks," in *Proc. 10th Hellenic Conf. Artif. Intell. (SETN)*, Patras, Greece: Association for Computing Machinery, Jul. 2018, Art. no. 7, doi: 10.1145/3200947.3201011.
[39] N. Batra, R. Kukunuri, A. Pandey, R. Malakar, R. Kumar, O. Krystalakos, M. Zhong, P. Meira, and O. Parson, "Towards reproducible state-of-the-art energy disaggregation," in *Proc. 6th ACM Int. Conf. Syst. Energy-Efficient Buildings, Cities, Transp. (BuildSys)*, New York, NY, USA, 2019, pp. 193–202. [Online]. Available: http://doi.acm.org/10.1145/3360322.3360844
[40] D. Murray, L. Stankovic, and V. Stankovic, "An electrical load measurements dataset of united kingdom households from a two-year longitudinal study," *Sci. Data*, vol. 4, no. 1, Jan. 2017, Art. no. 160122.
[41] S. Makonin and F. Popowich, "Nonintrusive load monitoring (NILM) performance evaluation," *Energy Efficiency*, vol. 8, no. 4, pp. 809–814, Oct. 2014. [Online]. Available: http://link.springer.com/article/10.1007/s12053-014-9306-2
[42] C. Klemenjak, S. Makonin, and W. Elmenreich, "Towards comparability in non-intrusive load monitoring: On data and performance evaluation," in *Proc. IEEE Power Energy Soc. Innov. Smart Grid Technol. Conf. (ISGT)*, Washington, DC, USA, 2020, pp. 1–5.
[43] L. Pereira and N. Nunes, "A comparison of performance metrics for event classification in non-intrusive load monitoring," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Oct. 2017, pp. 159–164.
[44] K. D. Anderson, M. E. Berges, A. Ocneanu, D. Benitez, and J. M. F. Moura, "Event detection for non intrusive load monitoring," in *Proc. 38th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2012, pp. 3312–3317.
[45] L. Pereira, "Hardware and software platforms to deploy and evaluate non-intrusive load monitoring systems," Ph.D. dissertation, Univ. Madeira, Funchal, Portugal, 2016.
[46] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, New York, NY, USA, 2008, pp. 1096–1103. [Online]. Available: http://doi.acm.org/10.1145/1390156.1390294

EDUARDO GOMES (Student Member, IEEE) received the master's degree in informatics engineering from the University of Madeira, in 2018.

He is currently a Research Assistant with ITI/LARSyS and a member of the FEELab, where he works under the supervision of Dr. Lucas Pereira. His research interests include the development and applicability of machine learning solutions to challenges, such as non-intrusive load monitoring.

LUCAS PEREIRA (Member, IEEE) received the Ph.D. degree in computer science from the University of Madeira, Portugal, in 2016.

Since then, he has been with ITI/LARSYS, where he leads the Further Energy and Environment research Laboratory (FEELab). Since 2019, he has been a Research Associate with Técnico Lisboa. He works towards bridging the gap between laboratory and real-world applicability of ICT for sustainable development, with a significant focus on smart-grids and smart built environments. His research interests lie in the intersections between computer science and data science, including sensing and data acquisition, human-computer interaction, and machine-learning.