Dynamic Aggregate Signal Reduction for Augmented Non-Intrusive Load Monitoring

Giulia Tanoni*, Enrik Xhani*, Muhammad Affan Khan*†, Stefano Squartini* and Emanuele Principi*
*Department of Information Engineering, Università Politecnica delle Marche, Italy
†Department of Information Engineering, Electrical Engineering, and Applied Mathematics, University of Salerno, Italy
Email: g.tanoni@staff.univpm.it

***Abstract*—Non-Intrusive Load Monitoring (NILM) typically relies on one-to-many or one-to-one algorithms to disaggregate energy usage of multiple appliances. Considering the wide variety of devices installed in residential and industrial buildings, we asked why not leverage the disaggregated power signals of one or more appliances to facilitate the disaggregation of devices with more complex activation profiles. In this way, both the complexity of the aggregate signal and the number of simultaneous operating appliances can be reduced. Thus, we propose a dynamic aggregate signal reduction framework for augmented NILM that takes advantage of predictions from multiple one-to-one networks during their training process to simplify the aggregate signal for other networks. This strategy enhances the reconstruction of activation profiles and reduces false positives and negatives by systematically removing appliance activations at each training step. The eligibility criterion for subtraction is based on the Signal-to-Noise Ratio estimated on each training instance. Real-world NILM applications often lack sufficient data to handle disaggregation effectively and our proposed subtraction mechanism addresses this issue by providing an augmented view of the same power signal that dynamically evolves at each epoch, depending on the activations of subtracted devices. To validate the methodology, two public residential datasets are used, with experiments designed to evaluate the reliability of the networks on unseen power signals and among several training scenarios. The framework demonstrates significant improvements, achieving up to 67.3% better performance in Mean Absolute Error and 57.4% in Normalized Error in assigned Power compared to the state-of-the-art.**

***Index Terms*—Load disaggregation, Dynamic subtraction, Data augmentation, Power Monitoring, Deep Learning**

I. Introduction

De-carbonisation of energy is crucial for mitigating the impact of climate change. This objective can be achieved primarily by increasing the utilisation of renewable energy sources and reducing overall energy consumption [1]. Final users play a key role, especially in residential buildings.

In this context, Non-Intrusive Load Monitoring (NILM), a set of techniques to extract consumption patterns of individual appliances from a single aggregate measurement, is well established, with adoption in both research and commercial applications. During the past three decades, numerous approaches have been proposed for NILM, with deep learning methods recently emerging as the most prevalent due to their effectiveness and robust generalisation capabilities [2]–[6]. For detailed reviews, see [7]–[9]. However, transformer-based [10] and hybrid architectures typically require larger datasets compared to convolutional or recurrent models. Given the intrusive nature of collecting appliance-level ground-truth data and associated privacy concerns, the availability of appliance-level power data is often insufficient to meet these requirements. Several strategies have been proposed, such as extracting common and specific features among appliances [2], or training only one network for multiple appliances (one-to-many) to share directly the features [11]–[13]. The one-to-many approach is typically adopted to reduce computational and memory costs, whereas, from a purely performance-oriented perspective, separate networks for each appliance (one-to-one) are often preferable. Moreover, environments with a high density of installed devices present further challenges, such as increased risk of appliance misidentification. Our first intuition is that, when monitoring multiple appliances simultaneously, it is possible to positively exploit the predicted activations (e.g., those that are easier to disaggregate) to facilitate the disaggregation of the more challenging ones e.g. by subtracting the predictions to the aggregate signal. Our second intuition is to dynamically integrate this subtraction mechanism into the training loop. In particular, we treat the resulting simplified aggregate signal as a form of data augmentation, which diversifies the input signal segments observed during training. This is especially valuable when data are scarce.

Only one recent work [14] has proposed a subtraction mechanism for NILM, but we identified several notable research gaps. In [14], subtraction is based on prior assumptions about each appliance's influence on the aggregate signal, evaluated using activation frequency and mean power consumption. This strategy is too static to accommodate the variability in appliance usage throughout the day or between weekdays and weekends. This fixed appliance ranking for sequential subtraction risks overfitting and may not generalize well to new domains (e.g., from one house to another). Prior experiments are limited to a single house (used both for training and test), an unrealistic scenario for practical deployment and services; the subtraction mechanism itself could introduce overfitting that should be investigated.

To address these limitations, we propose a novel approach for resolving device activation overlap, dynamically simplifying and lightening the aggregate input signal by leveraging disaggregation predictions from each network during training. Every signal segment processed by the network displays unique characteristics and appliance interactions; thus, this dynamicity should be incorporated directly into the training

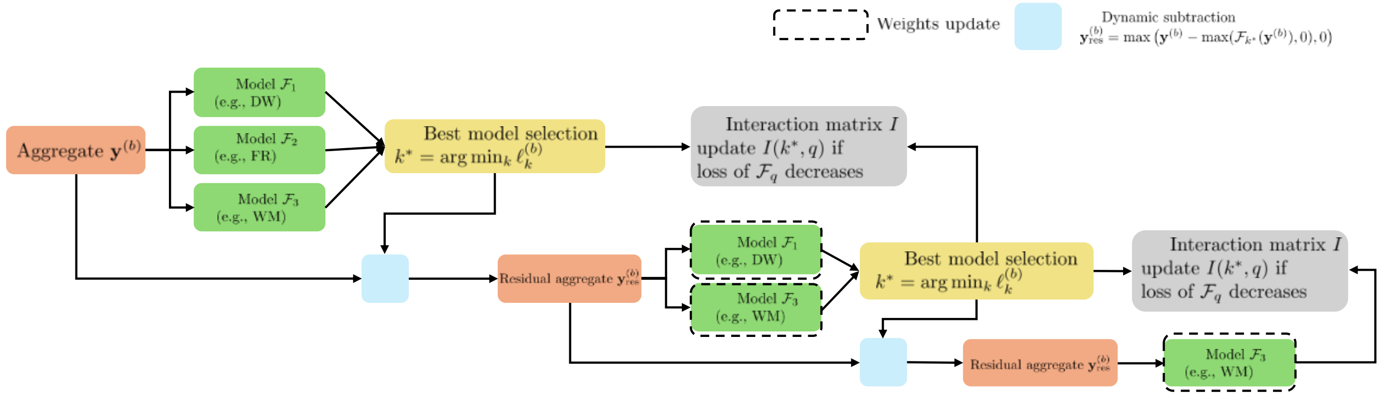


Fig. 1: Schematic view of the dynamic subtraction mechanism. For each window, the best-performing model is selected, its prediction is subtracted from the aggregate to obtain a residual input, which is then used as a new aggregate window for the remaining models. When these models benefit in terms of loss reduction, the interaction matrix I is updated.

phase, and the subtraction should depend on the current input and target signals themselves.

Our approach employs the Signal-to-Noise Ratio (SNR) to dynamically determine whether a specific training instance is eligible for subtraction, acting as data augmentation that improve the generalization across different houses. We demonstrate the essential role of dynamic subtraction through analysis of appliance interactions during training and the resultant disaggregation performance. The appliance's activation subtracted from the aggregate is predicted by the best-performing network and the residual signal is provided as input to the remaining networks. The subtraction strictly depends on the aggregate signal characteristics and the profile reconstruction quality: this ensures coherence with data from a given house and avoids generating unreal aggregate signals.

The outline of the paper is the following. Section II presents the mathematical formulation of the NILM problem, and Section III describes the methodology that we proposed to solve the task. In Section IV details about experimental settings are reported along with benchmarks used to compare our method and evaluation metrics. In Section V the results are discussed and Section VI concludes the paper with future directions.

II. Problem Formulation

The NILM problem can be formulated as follows. The aggregate signal $y(t)$ is the sum of the power absorbed by the N loads in the building:

$$y(t) = \sum_{n=1}^{N} x_n(t) + v(t) \tag{1}$$

where $x_n(t)$ is the power consumption of the n-th appliance and $v(t)$ denotes the noise term, including both measurement noise and the contribution from any ignored loads.

Focusing on a subset of $K \leq N$ appliances of interest:

$$y(t) = \sum_{k=1}^{K} x_k(t) + v(t) \tag{2}$$

where $x_k(t)$ is the true power profile of the k-th target appliance, and $v(t)$ now includes the remaining $N - K$ appliances and noise. Our goal is to simultaneously estimate the time profiles $x_k(t)$ for every $k = 1, \ldots, K$.

We partition $y(t)$ and each $x_k(t)$ into non-overlapping windows of length L. The j-th window becomes:

$$\mathbf{y}_j = [y(jL),\, y(jL+1),\, \ldots,\, y(jL+L-1)]^T \in \mathbb{R}^L$$

$$\mathbf{x}_j^{(k)} = [x_k(jL),\, x_k(jL+1),\, \ldots,\, x_k(jL+L-1)]^T \in \mathbb{R}^L$$

For each window j, define the matrix of target appliances:

$$\mathbf{X}_j = \begin{bmatrix} \mathbf{x}_j^{(1)} & \mathbf{x}_j^{(2)} & \cdots & \mathbf{x}_j^{(K)} \end{bmatrix} \in \mathbb{R}^{L \times K}$$

The objective is to estimate, for each window, all targets $\{\mathbf{x}_j^{(k)}\}_{k=1}^{K}$ given $\mathbf{y}_j$, by learning functions $\mathcal{F}_k : \mathbb{R}^L \to \mathbb{R}^L$ for each k:

$$\hat{\mathbf{x}}_j^{(k)} = \mathcal{F}_k(\mathbf{y}_j)$$

III. Dynamic Aggregate Signal Reduction with Augmentation

Let a mini-batch be $\mathcal{B} = \left\{ \left(\mathbf{y}^{(b)}, \mathbf{x}^{(b,1)}, \ldots, \mathbf{x}^{(b,K)} \right) \right\}_{b=1}^{B}$, where $\mathbf{y}^{(b)}$ is the windowed aggregate input, and $\mathbf{x}^{(b,k)}$ is the windowed signal for appliance k in batch index b.

For simplicity, we explain the method for $K = 3$, but it generalizes directly.

a) Supervised phase.: Each model $\mathcal{F}_k$ (for $k = 1, 2, 3$) is first updated independently over the batch by minimizing the MSE loss:

$$\mathcal{L}_{\text{sup}}^{(k)} = \frac{1}{B} \sum_{b=1}^{B} \left\| \mathcal{F}_k(\mathbf{y}^{(b)}) - \mathbf{x}^{(b,k)} \right\|^2$$

b) Batch Segments Selection.: For each sample b, compute the SNR between the sum of powers of the K target appliances and the background:

$$\text{SNR}^{(b)} = \frac{\sum_{k=1}^{K} \|\mathbf{x}^{(b,k)}\|_1}{\max\left(\|\mathbf{y}^{(b)}\|_1 - \sum_{k=1}^{K} \|\mathbf{x}^{(b,k)}\|_1,\ \epsilon \right)}$$

where $\|\cdot\|_1$ denotes the ℓ_1 norm, and $\epsilon > 0$ avoids division by zero.

Define a dynamic threshold T as a quantile q of the batch SNR distribution:

$$T = \mathrm{Quantile}_q\left(\{\mathrm{SNR}^{(b)}\}_{b=1}^{B}\right)$$

c) Dynamic Aggregate Reduction.: For each b such that $\mathrm{SNR}^{(b)} < T$:

1) Compute MSE losses for all models:

$$\ell_k^{(b)} = \left\|\mathcal{F}_k(\mathbf{y}^{(b)}) - \mathbf{x}^{(b,k)}\right\|^2$$

2) Identify the **best model** $k^* = \arg\min_k \ell_k^{(b)}$ and compute the residual aggregate by subtracting the (non-negative) prediction:

$$\mathbf{y}_{\mathrm{res}}^{(b)} = \max\left(\mathbf{y}^{(b)} - \max(\mathcal{F}_{k^*}(\mathbf{y}^{(b)}), 0), 0\right)$$

3) For each $k \neq k^*$, recompute the loss using $\mathbf{y}_{\mathrm{res}}^{(b)}$ as input, and update the weights.
4) Maintain an *interaction matrix* $I \in \mathbb{N}^{K\times K}$, where $I(k^*, q)$ counts occasions where model k^* aided model q in loss reduction.
5) Repeat steps 2–4 for remaining models.

d) Validation phase.: During validation, models are evaluated in an order determined by the interaction matrix I (most influential first); each receives as input the aggregate with contributions from previous models subtracted.

If a model k stops improving after a predefined number of epochs its weights are frozen but the model still participates in the subtraction loop both during training and validation of the remaining models. This k-indexed notation guarantees full alignment between appliance, model, target, and all derivations.

A. Neural Network Structure

We focus our efforts on developing the proposed training methodology rather than on the design of novel neural network architectures, aiming to demonstrate the feasibility and robustness of the dynamic subtraction mechanism during training. To begin with, and to avoid the influence of other factors, we adopt a well-established architecture that has demonstrated competitive disaggregation performance compared to more recent models (see, e.g., [15], [16]), even in challenging conditions. For this reason, we choose to demonstrate our approach using the Convolutional Recurrent Neural Network (CRNN) first introduced in [17] and the optimized for the NILM task in [18]. The CRNN architecture is divided in two parts, with the first consisting of 3 blocks each featuring a 2D convolutional layer with 32 filters and a kernel size of 5, a batch normalization layer, an activation layer with the Rectified Linear Unit function, and a dropout mechanism for regularization. The final convolutional block connects to the recurrent part of the network, which includes a bidirectional layer with 64 Gated Recurrent Units. Completing the structure, we added is a couple of fully-connected layers (with 512, and 1 neuron respectively) with a linear activation function responsible for producing the active power prediction.

IV. Experiments

A. Dataset

The experiments were conducted using active power signals from the REFIT and AMPds [19], [20]. The original signals of REFIT, sampled at 8 s intervals, were resampled to 60 s to emulate lower-frequency scenarios, reflecting realistic conditions in which both data availability and temporal granularity are limited. AMPds signals are already at 1 minute period.

REFIT was chosen to validate our approach because its aggregate signals, particularly for several houses, contain a significant proportion of unmonitored loads, which can strongly impact disaggregation performance (see [21]). Houses 3, 5, 7, and 15 were used separately for training and validation, while houses 2 and 9 were reserved for testing. In this way, we can evaluate different training scenarios with diverse and scarce data availability that allows to assess the how much our method is generalizable and effective on the testing houses. AMPds, despite it contains signals only from one house, was selected for the variety of appliances monitored. This allows to evaluate the influence of various devices in the subtraction mechanism. For testing, we select the last 20% of the available data.
The signals have been pre-processed to normalize $y(t)$ and $x_k(t)$ within $[0, 1]$ using values estimated on the training set.

B. Hyperparameters

For the proposed method, several values of the quantile parameter that defines the SNR threshold are investigated: 0.01, 0.1, 0.3, 0.5, 0.7, 0.9, and 1. This allows us to assess the effect of SNR-based segment selection on model performance, ranging from scenarios where only the noisiest segments are selected to those where all segments are included. Windows where all appliance activations are zero are excluded from training, as including them could negatively affect the process. In such cases, the SNR is zero, but these segments are not meaningful for the disaggregation task because there is no activation to disaggregate. The length of the signal window processed by the network is set to 1200 samples for the REFIT and 600 samples for AMPds. The training is stopped if the performance on the validation set does not improve for 10 consecutive epochs. The batch size is 32.

C. Benchmarks

We demonstrate the validity of our approach through a comparison with two benchmarks: the baseline CRNN [18] and the Inception method [14]. The baseline CRNN consists of three independent CRNNs, each trained to perform disaggregation for a specific appliance, with the architecture constructed as described in III-A. Additionally, we implement the Inception method, which has been recently proposed in the literature and also employs a signal subtraction mechanism. Unlike our approach, however, the Inception method statically defines the order of appliance subtraction and do not allow intermediate weights updates. With these benchmarks, our goal is twofold: first, to show that our method outperforms standard supervised training that lacks dynamic subtraction and data augmentation; second, to demonstrate that the dynamic subtraction strategy introduced by our method is particularly effective in scenarios with limited data availability. Beyond this, we also aim to validate our method's robustness in real-world conditions by training on data from different houses than those used for testing, differently from [14]. For a fair comparison, all methods are evaluated using the same experimental configuration.

D. Experimental scenarios

The monitored appliances, selected according to signal availability in the two datasets, are fridge (FR), washing machine (WM), and dishwasher (DW) for REFIT, and also the clothes dryer (CD) for AMPds. The fridge was chosen due to its continuous operation, which can significantly influence the disaggregation of other appliances. The washing machine and dishwasher were included because their partially overlapping activation patterns make them prone to false positive identifications; thus, their inclusion allows us to test the ability of our framework to reduce misclassification errors. The following experiments are conducted for all the methods:

- REFIT: 4 training scenarios (house 3, 5, 7 and 15) to disaggragate fridge, washing machine and dishwasher and 1 testing scenario (house 2 and 9);
- AMPds: 2 training scenarios (house 1) to disaggregate in one case fridge, washing machine and diswasher, and in another case fridge, clothes dryer and dishwasher. The test is conducted on a separate portion of data of the same house as expressed in Section IV-A.

This experimental protocol enables us to evaluate the effectiveness of our approach across different training houses, which exhibit varying activation patterns and appliance usages, and on testing houses, to determine whether the model can generalize to unseen data. Lastly, we assess the performance of the method under different appliances combinations.

E. Evaluation Metrics

We evaluate model performance using the Mean Absolute Error (MAE) and Normalized Error in assigned Power (NEP). For each appliance/model k, given predictions $\hat{x}_k^{(t)}$ and true values $x_k^{(t)}$ over the validation set of T samples, the metric is $\mathrm{MAE}_k = \frac{1}{T}\sum_{t=1}^{T}|\hat{x}_k^{(t)} - x_k^{(t)}|$. This metric is widely used but strongly influenced by the presence of zeros thus we introduced NEP, the MAE normalized by the total appliance's energy consumption. It allows to evaluate the importance of the error based on the appliance operating characteristics:

$$\mathrm{NEP}_k = T \cdot \frac{MAE_k}{\sum_{t=1}^{T} x_k^{(t)}}. \quad (3)$$

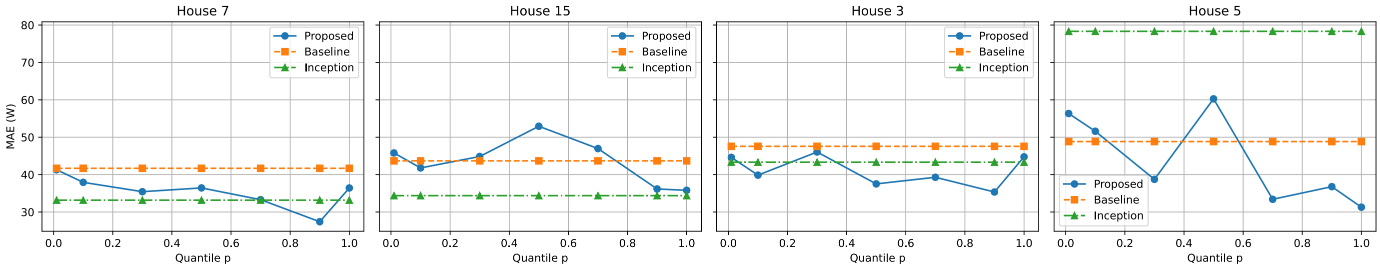
TABLE I: Comparison in terms of overall MAE and NEP between Baseline, Proposed and Inception [14] methods. For the Proposed approach the reported results are the best obtained among the quantiles.

Train. House	Appliance	Baseline		Proposed		Inception [14]	
		REFIT					
		MAE	NEP	MAE	NEP	MAE	NEP
15	DW	48.26	0.99	**47.87**	**0.81**	48.52	1.07
	FR	43.97	0.85	40.61	**0.64**	**36.56**	1.00
	WM	38.81	0.99	18.92	**0.67**	**18.08**	1.19
	AVG.	43.68	0.94	35.80	**0.71**	**34.39**	1.09
7	DW	41.25	0.43	**26.34**	**0.31**	46.13	1.00
	FR	35.84	**0.78**	**35.60**	0.83	36.55	1.00
	WM	47.99	1.06	20.27	**0.71**	**16.83**	1.10
	AVG.	41.69	0.76	**27.40**	**0.62**	33.17	1.03
5	DW	65.73	0.63	**29.33**	**0.37**	94.09	2.32
	FR	38.53	0.70	**32.16**	**0.59**	35.18	0.91
	WM	42.26	1.02	**32.38**	**0.92**	105.69	9.81
	AVG.	48.84	0.78	**31.29**	**0.63**	78.32	4.35
3	DW	64.71	0.96	**43.45**	**0.70**	69.01	1.83
	FR	45.68	0.77	42.05	**0.64**	**36.99**	1.05
	WM	32.32	1.01	**20.56**	**0.66**	23.99	1.73
	AVG.	47.57	0.91	**35.35**	**0.67**	43.33	1.54
Train. House	Appliance	AMPds					
		MAE	NEP	MAE	NEP	MAE	NEP
1	DW	22.57	0.54	**7.37**	**0.23**	17.36	1.00
	FR	43.96	0.56	**26.55**	**0.32**	45.96	1.01
	WM	7.44	0.94	5.21	**0.52**	**4.69**	1.00
	AVG.	24.65	0.68	**13.04**	**0.35**	22.58	1.00
1	DW	23.13	0.57	**12.25**	**0.26**	21.24	1.77
	FR	43.69	0.56	**27.59**	**0.32**	51.38	2.95
	CD	19.69	**0.04**	14.95	0.10	**10.31**	0.30
	AVG.	28.83	0.39	**18.26**	**0.22**	27.79	1.67

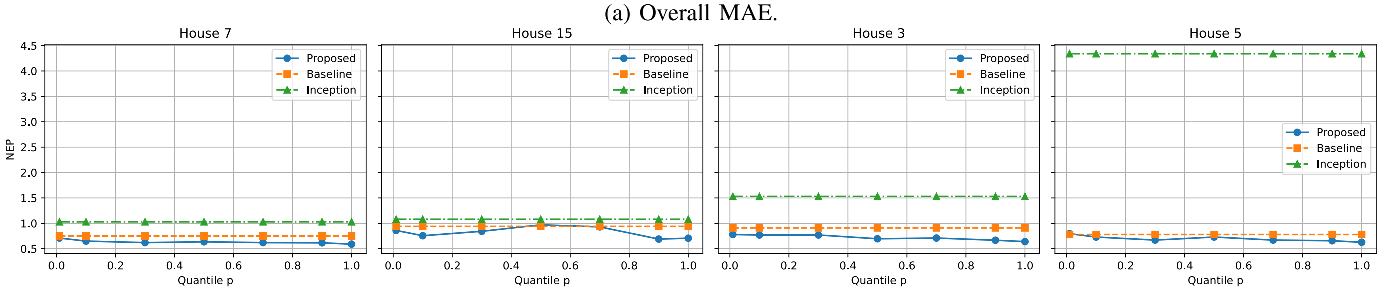
V. RESULTS

The proposed approach consistently proves to be the most effective methodology across all the houses we have considered, for both datasets, and even for different appliances combinations, as shown in Table I. Although in the House 15 training scenario the MAE is higher compared to the Inception method, this metric should always be interpreted alongside the NEP and the quality of the reconstructed activation profile. In fact, if we examine the reconstructed profiles for House 15 training scenario (see Fig. 4), we observe that the disaggregation for the washing machine is more accurate and more closely matches the target signal with the proposed method. Baseline underestimates the activations, while Inception completely ignores them. In general, the appliance that is most influenced is the DW, which exhibits the lowest error in each configuration (Table I) across houses and appliance combinations, as also observed for AMPds. The WM is the second appliance to benefit from improved profile reconstruction, as its consumption is typically underestimated.

Looking at the performance trend while varying the quantile for REFIT (Figure 2), it is interesting to note that our method shows superior performance at the extremes, i.e., when the quantile threshold is set close to zero (only the noisiest windows are considered in the subtraction) or close to one (all windows are included in the subtraction). Since the network is exposed to different versions of the aggregate signal during the intermediate training steps, considering either all windows or only a minimal subset can ensure a more balanced repetition of activations across the various appliances. By contrast, at

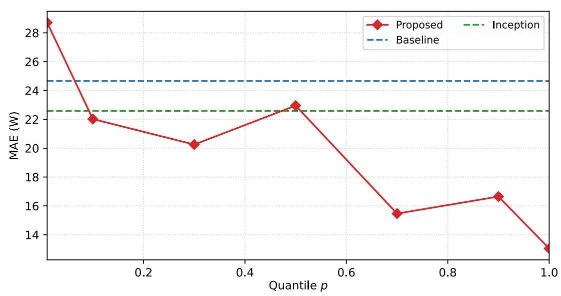


(a) Overall MAE.

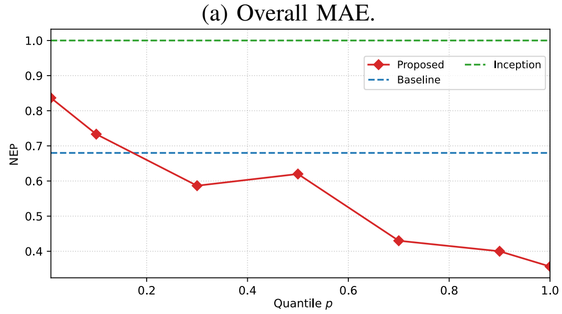


(b) Overall NEP.

Fig. 2: Comparison of Baseline, Inception and proposed method for different quantiles on REFIT dataset.



(a) Overall MAE.



(b) Overall NEP.

Fig. 3: Comparison of Baseline, Inception and Proposed method on average on DW, FR, and WM for different quantiles on the AMPds dataset.

intermediate percentile values, the distribution of segments may become less balanced, which can lead to a degradation in disaggregation performance due to overfitting. Despite this, the best results are obtained with higher quantile values, which means that the more windows are involved in the dynamic subtraction, the more the learning improves. This confirms the benefits of our methodology.

For AMPds, as can be seen in Figure 3, both MAE and NEP exhibit a decreasing trend as the quantile increases, meaning that all the windows included in the subtraction process are significant for enhancing the quality of learning. This can be justified by the fact that the networks require more data to properly learn to disaggregate complex appliance patterns. This consideration seems to be confirmed by the results for both REFIT and AMPds. The only exception occurs for the quantile equal to 1 when we disaggregate the clothes dryer (CD), where the performance on CD worsens and the average error increases but still being lower than the baseline; however, its NEP remains relatively stable.

Overall, the proposed methodology proves effective for four of the most common household appliances, even under different appliance combinations. Best improvements obtained are 67.3% for MAE and 57.4% for NEP for the DW for AMPds, 39.6% and 42.9% respectively for FR, 57.8% and 44.7% for WM. For the CD, MAE has an improvement of 24.1% while the NEP worsens despite is very low. Lastly, the interaction matrices recorded during training confirm that the most influenced appliances are those that show substantial improvements compared to the baseline (Fig. 5), as it happens for WM and DW when the training is performed on House 7, whereas the most influencing appliances exhibit stable performance or only slight improvements or degradations. In House 5 scenario, where all three appliances show significant improvement, the DW and FR models mutually influence each other, leading to a reduction in reconstruction errors.

VI. Conclusion

Environments with a high density of installed devices present further challenges for NILM algorithms, such as increased risk of appliance misidentification or underestimation. We proposed a novel approach for resolving device activation overlap, dynamically simplifying and lightening the aggregate

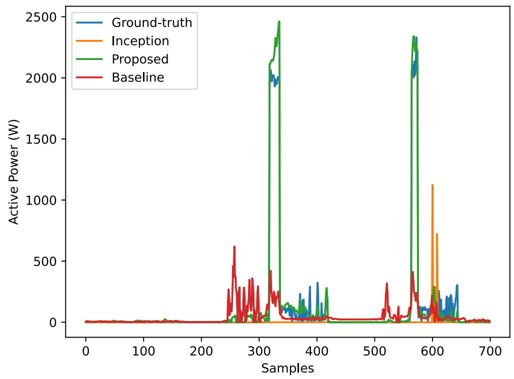
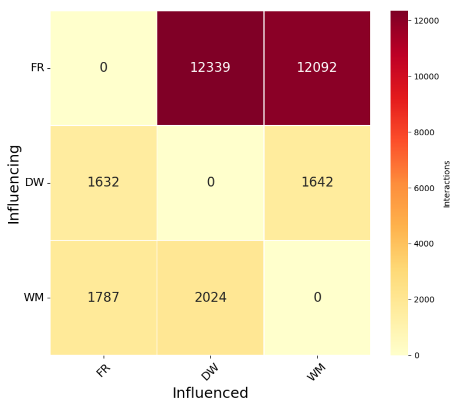
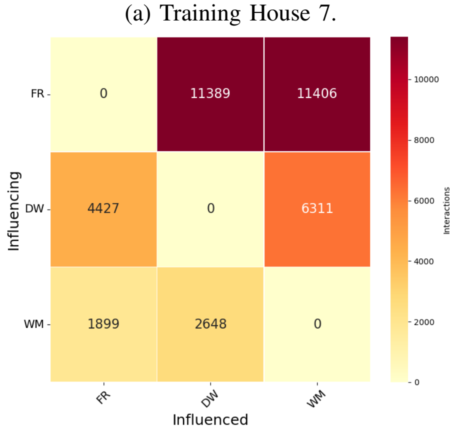


Fig. 4: Washing machine disaggregation comparing the approches trained on house 15.



(a) Training House 7.



(b) Training House 5.

Fig. 5: Interaction matrices I for two training scenarios.

input signal by leveraging disaggregation predictions from each network during training. This mechanism enables a sort of incorporated data augmentation that improves both performance and generalization across different houses. Based on the experiments conducted on two datasets, our approach demonstrated effective more than a baseline approach where each network is trained separately and a benchmark approach that proposed static subtraction. Future directions will include more networks (and thus more appliances) and will explore the potential of the approach in more complex environments such as industrial plants, with a wider number of devices installed.

References

[1] R. Agarwal, M. Garg, D. Tejaswini, V. Garg, P. Srivastava, J. Mathur, and R. Gupta, "A review of residential energy feedback studies," *Energy and Buildings*, vol. 290, p. 113071, 2023.

[2] K. Li, J. Feng, Y. Yao, Y. Xing, Q. Xiao, and J. Wang, "Nilm model for multi-appliance power disaggregation based on the combination of common features and individual features," *Measurement*, vol. 263, p. 120104, 2026.

[3] H. Chen, J. Chen, Y. Chai, W. Guo, C. Jia, B. Yang, and Z. Xin, "Scene-aware non-intrusive load monitoring using large language models," *IEEE Transactions on Smart Grid*, vol. 17, no. 1, pp. 874–876, 2026.

[4] L. Jiang, M. Liu, J. S. Jin, Y. Zheng, and X. He, "Convtransnilm: A parallel transformer model with convolution for energy disaggregation," *Energy and Buildings*, vol. 347, p. 116361, 2025.

[5] X. Sun, J. Hu, W. Hu, D. Cao, Z. Chen, and F. Blaabjerg, "Non-intrusive load monitoring based on process-adaptive multi-target regression and transformer-enabled two-stream input network," *Applied Energy*, vol. 393, p. 126046, 2025.

[6] A. Huzzat, A. S. Khwaja, A. A. Alnoman, B. Adhikari, A. Anpalagan, and I. Woungang, "Gru-bert for nilm: A hybrid deep learning architecture for load disaggregation," *AI*, vol. 6, no. 9, 2025.

[7] Z. Yan, M. Nardello, D. Brunelli, and H. Wen, "Reproducible non-intrusive load monitoring: A survey of open-source methods and practical challenges," *Renewable and Sustainable Energy Reviews*, vol. 226, p. 116255, 2026.

[8] Z. Luo, Z. Li, T. Zhang, G. Wang, W. Deng, J. Li, and J. Liang, "A low-frequency residential nilm approach based on adaptive event detection," *Journal of Building Engineering*, vol. 106, p. 112543, 2025.

[9] M. H. Saleem, M. Taha, M. A. A. Rehmani *et al.*, "A comprehensive review of machine learning and deep learning models for non-intrusive load monitoring: performance, analyses, practical insights, and emerging trends," *Applied Intelligence*, vol. 55, p. 1020, 2025.

[10] H. Chen, J. Chen, Y. Chai, W. Guo, C. Jia, B. Yang, and Z. Xin, "Scene-aware non-intrusive load monitoring using large language models," *IEEE Transactions on Smart Grid*, vol. 17, no. 1, pp. 874–876, 2026.

[11] R. Laceda Taloma, P. Pisani, and F. Cuomo, "Mantra: a multi-appliance transformer for non-intrusive load monitoring," in *2024 IEEE SmartGridComm*, 2024, pp. 432–437.

[12] 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.

[13] D. Li, J. Li, X. Zeng, V. Stankovic, L. Stankovic, and Q. Shi, "Non-intrusive load monitoring for multi-objects in smart building," in *2021 International BalkanCom*, 2021, pp. 117–121.

[14] Y. Belguermi, P. Wira, and G. Hermann, "Neural network-aided nilm (nnan) disaggregation: Revealing appliance consumption patterns with iterative subtraction," *Machine Learning with Applications*, vol. 20, p. 100667, 2025.

[15] G. Tanoni, L. Stankovic, V. Stankovic, S. Squartini, and E. Principi, "Knowledge distillation for scalable nonintrusive load monitoring," *IEEE Trans. on Ind. Inf.*, vol. 20, no. 3, pp. 4710–4721, 2024.

[16] G. Tanoni, R. L. Taloma, E. Principi, D. Comminiello, and S. Squartini, "A deep cascade framework for non-intrusive power disaggregation in solar-powered households," in *2025 IEEE ISCAS*, 2025, pp. 1–5.

[17] 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 *Proceedings of the 2014 EMNLP*, Oct. 2014, pp. 1724–1734.

[18] G. Tanoni, E. Principi, and S. Squartini, "Multilabel appliance classification with weakly labeled data for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 14, no. 1, pp. 440–452, 2023.

[19] D. Murray, L. Stankovic, and V. Stankovic, "An electrical load measurements dataset of united kingdom households from a two-year longitudinal study," *Scientific data*, vol. 4, no. 1, pp. 1–12, 2017.

[20] S. Makonin, B. Ellert, I. V. Bajic, and F. Popowich, "Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014," *Scientific Data*, vol. 3, no. 160037, pp. 1–12, 2016.

[21] C. Klemenjak, S. Makonin, and W. Elmenreich, "Investigating the performance gap between testing on real and denoised aggregates in non-intrusive load monitoring," *Energy Informatics*, vol. 4, no. 3, 2021.