Few Labels are all you need: A Weakly Supervised Framework for Appliance Localization in Smart-Meter Series

Adrien Petralia
EDF R&D - Université Paris Cité
adrien.petralia@gmail.com

Paul Boniol
Inria, ENS, PSL, CNRS
paul.boniol@inria.fr

Philippe Charpentier
EDF R&D
philippe.charpentier@edf.fr

Themis Palpanas
Université Paris Cité - IUF
themis@mi.parisdescartes.fr

***Abstract*—Improving smart grid system management is crucial in the fight against climate change, and enabling consumers to play an active role in this effort is a significant challenge for electricity suppliers. In this regard, millions of smart meters have been deployed worldwide in the last decade, recording the main electricity power consumed in individual households. This data produces valuable information that can help them reduce their electricity footprint; nevertheless, the collected signal aggregates the consumption of the different appliances running simultaneously in the house, making it difficult to apprehend. Non-Intrusive Load Monitoring (NILM) refers to the challenge of estimating the power consumption, pattern, or on/off state activation of individual appliances using the main smart meter signal. Recent methods proposed to tackle this task are based on a fully supervised deep-learning approach that requires both the aggregate signal and the ground truth of individual appliance power. However, such labels are expensive to collect and extremely scarce in practice, as they require conducting intrusive surveys in households to monitor each appliance. In this paper, we introduce CamAL, a weakly supervised approach for appliance pattern localization that only requires information on the presence of an appliance in a household to be trained. CamAL merges an ensemble of deep-learning classifiers combined with an explainable classification method to be able to localize appliance patterns. Our experimental evaluation, conducted on 4 real-world datasets, demonstrates that CamAL significantly outperforms existing weakly supervised baselines and that current SotA fully supervised NILM approaches require significantly more labels to reach CamAL performances. The source of our experiments is available at: https://github.com/adrienpetralia/CamAL.**

***Index Terms*—Non Intrusive Load Monitoring, Smart Meters Data, Appliance Detection, Time Series Classification, XAI**

I. Introduction

Managing electricity consumption at the individual level has become a critical challenge in achieving more efficient smart grid management and contributing to the global effort to reduce energy usage. In response, energy suppliers such as EDF (Electricité De France) have begun offering various services to help customers better understand and manage their electricity consumption. Energy suppliers are installing meters that record the total aggregate electricity power consumed in the household at regular intervals.Although this information provides valuable data that suppliers already use for diverse applications such as forecasting energy demand, there is a need to develop solutions to extract detailed information on household consumption. Indeed, the collected signal results from the addition of all the appliances operating simultaneously, and suppliers face the challenge of extracting detailed information from these data, such as *if* and *when* a specific appliance has been used looking only at the aggregated consumption time series. Although extracting this valuable information is challenging due to the complexity of the aggregated data, it is crucial to help consumers manage their electricity consumption and provide them with more insight into their usage.

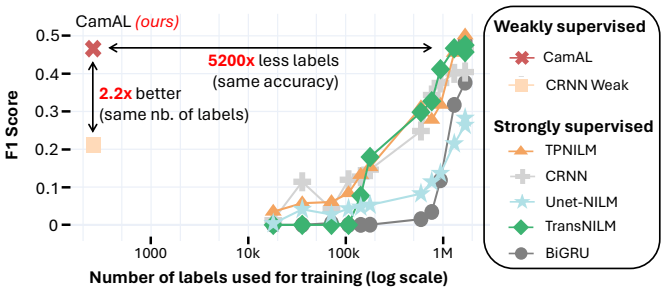


Fig. 1. Localization accuracy versus number of training labels for CamAL compared to six baseline methods on the dishwasher case from the IDEAL dataset. CamAL and the weakly supervised baseline are trained using only one label per house, indicating appliance ownership.

Non-Intrusive Load Monitoring (NILM) refers to the challenge of estimating the power consumption, usage patterns, or on/off state activation of individual appliances using only aggregate power readings from a household. Early NILM solutions approached this task as an optimization problem, relying on Combinatorial Optimization (CO) to estimate the proportion of total power consumed by active appliances at each time step [1]. Over the past decade, NILM research has surged, fueled by the release of publicly available smart meter datasets [2]–[4]. These datasets provide both aggregate power readings and appliance-level power consumption, often referred to as *strong labels*, which indicate the exact activation state and consumption power of individual appliances for each timestamp. Consequently, NILM approaches predominantly rely on fully supervised, primarily deep-learning models that require extensive strong-label data for training (cf. *strongly supervised* approaches in Figure 1).

While these datasets have enabled the development and benchmarking of new algorithms, they are limited—typically encompassing data from only about a dozen households—and

do not represent the diversity of appliances owned by all consumers. As a result, electricity suppliers need to invest in collecting their own data by instrumenting a large number of households. However, conducting such a survey is expensive in terms of time, and CO_2 emissions, as it requires sending technicians to individual households to install sensors that measure the consumption of each appliance. At the same time, it is also very expensive in terms of money: collecting ground truth appliance-level data from a mere few dozen households costs several hundreds of thousands of euros.

In practice, electricity suppliers only have at their disposal the information of an appliance's activation (or not) within a time frame, meaning one label for an entire series, so-called *weak labels*. Particularly, in challenging but more realistic scenarios, they only know the presence of the appliances in the household, meaning one label for an entire long series, without any guarantee on *when* the appliance is effectively used. Unfortunately, recent proposed NILM approaches cannot be trained and operate with such scarce labels: trying to train a NILM solution with only one label for an entire series (e.g., by replicating the label for all time steps) implies that it can no longer be used to localize an appliance; indeed, NILM solutions provide a probability of detection for each timestamp to be able to localize it. To address this issue, a recent study introduced the appliance localization problem using weak labels [5]. The authors proposed a deep-learning approach that formulates the challenge as a Multiple Instance Learning (MIL) problem. In this framework, the model is trained using only one label per series or by combining both strong and weak labels when available. However, their results on real-world challenging datasets showed that accuracy is notably low when using only weak labels; they had to combine both strong and weak labels to achieve better performance. Moreover, their approach was tested only on datasets that provide individual power consumption data for each appliance. It has not been evaluated in realistic scenarios where only the possession of the appliance in the household is known, without any data on when the appliance is actually used.

Recent studies have been conducted to detect the presence of appliances in consumption series using weakly supervised approaches [6]–[8]. In these studies, the appliance detection problem is cast as a time series classification problem, in which a classifier is trained using only one label per electrical time series, i.e., we only know whether the appliance has been switched ON in a given period. Although these methods show promising results in detecting *if* an appliance has been used, they cannot determine *when* the device has been switched on. At the same time, a few recent works have shown that classification-based explainability methods can be used to understand a classifier's decision by identifying the part of a time series that contributed to the label prediction [9]–[13]. These approaches have been tested on time series for explainable classification and anomaly detection tasks [14], [15] with promising results but have never been used for appliance localization. For the NILM problem, such methods could enable the localization of appliances while being trained using labels that only indicate *if* a device is turned on during a large time frame, significantly reducing the number of needed labels that current NILM approaches require. In addition, several studies [16], [17] demonstrate the importance of leveraging weak labels to efficiently solve diverse real-world problems while using a few amounts of strong labels.

While ***few labels are all we have***, we demonstrate in this paper that ***few labels are all we need***. This paper investigates for the first time the combination of explainable classification approaches to tackle the appliance pattern localization problem. Overall, our framework, called CamAL, contains the following steps: (i) We train a set of convolutional neural network classifiers on smart meter consumption series, using labels that indicate whether a specific appliance was turned on within a large given time frame. (ii) After training, our method performs localization by extracting and aggregating the class activation maps (CAM) from the classifiers if the appliance is detected. (iii) Finally, we post-process the aggregated CAM to refine the prediction and output the most probable timestamps corresponding to the appliance usage.

We empirically compare CamAL to current state-of-the-art NILM methods for appliance localization across multiple real-world datasets, showing that CamAL often achieves comparable or superior performance while requiring up to three orders of magnitude fewer labels (see Figure 1). More specifically, CamAL significantly outperforms existing weakly supervised approaches and strongly supervised NILM methods trained with the same number of labels—achieving up to a twofold improvement (see Figure 1)—and scales more efficiently to large datasets than any of these methods.

Overall, we demonstrate that our approach is more appropriate than classical NILM methods for use cases without access to per-timestamp labels, which corresponds to the vast majority of realistic and industrial applications. Consequently, CamAL is the first real non-invasive load monitoring method as it does not require practitioners to physically enter each household to install per-appliance sensors to train a solution. In addition, we demonstrate that CamAL scales to the real-world (possession only) datasets currently available to suppliers thanks to its weakly supervised paradigm, making it much faster to train than current NILM solutions. Even though the inference time of CamAL is not significantly faster than existing NILM methods, it drastically surpasses the weakly supervised one and lets practitioner saves a significant amount of time, money, cost of storage, and CO_2 emissions necessary to create the labels to train a strongly supervised method.

Overall, our contributions are the following:

- We first formalize the problem of appliance detection and localization, and we discuss the proposed methods available in the literature to solve these problems (**Section II**).
- We propose CamAL (*Class Activation based Appliance Localization*) a novel weakly supervised method for appliance pattern localization based on time series classification and explainable IA that requires only the appliance's possession label for training (**Section IV**).
- We experimentally compare CamAL with state-of-the-art

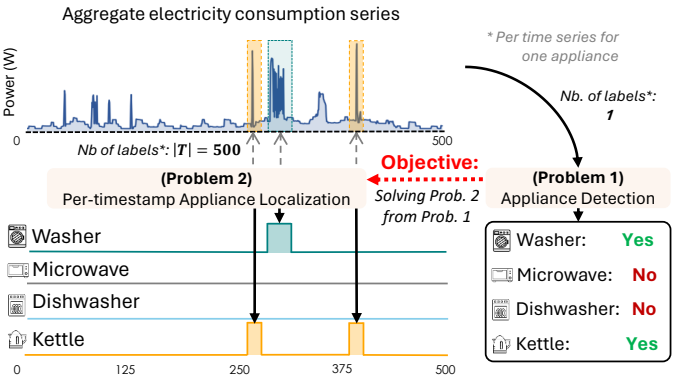


Fig. 2. Illustration of the appliance detection (1 label needed) and per-timestamp appliance localization ($|T|$ labels needed) problems. Our objective in this paper is to solve Problem 2 from Problem 1.

NILM approaches for appliance localization, and we empirically demonstrate that the NILM problem can be solved with significantly fewer labels (**Section V**).

- We demonstrate through a public dataset and a real industrial use case that CamAL can operate with one label only (i.e., the possession or not of the appliance), making our model the first accurate and scalable real Non-Intrusive Load monitoring system (**Section V-H**).
- We conclude by showing that our generated weak labels can be used to compensate for the scarcity or strong labels to maintain a high accuracy of strongly supervised NILM approaches (**Section V-I**).

II. Background and Related Work

A smart meter signal is a univariate time series $\boldsymbol{x} = \left(\mathbf{x}_{t_1}, \mathbf{x}_{t_2}, \ldots, \mathbf{x}_{t_T}\right)$ of T timestamped power consumption readings. The meter reading is defined as the time difference $\Delta_t = t_i - t_{i-1}$ between two consecutive timestamps t_i. Each element $\boldsymbol{x}_t$ (in Watts or Watt-hours) represents either the actual power at time t or the average power over the interval Δ_{t_i}.

The aggregate power consumption is defined as the sum of N appliance power signals $a_1(t), a_2(t), \ldots, a_N(t)$ that run simultaneously plus some noise $\epsilon(t)$, accounting for measurement errors. Formally, it is defined as:

$$x(t) = \sum_{j=1}^{N} a_j(t) + \epsilon(t) \tag{1}$$

where $x(t)$ is the total power consumption measured by the main meter at timestep t; N is the total number of appliances connected to the smart meter; and $\epsilon(t)$ is defined as the noise or the measurement error at timestep t.

Practitioners are interested in solving two problems: (i) discovering *if* an appliance has been activating (Appliance detection problem), and (ii) identifying *when* an appliance has been used (Per-Timestamp Appliance Localization Problem). The two problems are formalized as follows:

Problem 1 (Appliance Detection (cf. Figure 2))**.** Given an aggregate consumption smart meter series $\boldsymbol{x} = \left(\mathbf{x}_{t_1}, \mathbf{x}_{t_2}, \ldots, \mathbf{x}_{t_T}\right) \in \mathbb{R}_T^+$, an appliance a, we want to know if a has been used in $\boldsymbol{x}$ (i.e., was in an "ON" state, regardless of the time and number of activations).

Problem 2 (Per-timestamp Appliance Localizaton (cf. Figure 2))**.** The total active power consumed in a household is denoted by $x(t)$, the active power of the j-th appliance by $a_j(t)$, and its state by $s_j(t) \in \{0, 1\}$. Then we have:

$$x(t) = \sum_{j=1}^{N} s_j(t) a_j(t) + \varepsilon(t), \tag{2}$$

where $\varepsilon(t)$ represents the measurement noise, and

$$s_j(t) = \begin{cases} 0, & \text{if appliance } j \text{ is OFF at time index } t, \\ 1, & \text{if appliance } j \text{ is ON at time index } t. \end{cases} \tag{3}$$

We want to compute the consumption (or activation) of appliance j, $a_j(t)$, from $x(t)$.

In order to solve Problem 2, we can rewrite Equation 1 as:

$$x(t) = s(t)a(t) + v(t), \tag{4}$$

where the first term is the power of the appliance of interest, and $v(t)$ is a cumulative noise term corresponding to the sum of all the other appliances running simultaneously.

In cases where the objective is the direct estimation of the individual active power signal $a(t)$, NILM is treated as a regression problem and has been approached either as a denoising task or as a blind source separation task [18]. Conversely, when the objective is to estimate the appliance state $s(t)$, NILM represents a classification problem [18]. In both cases, the algorithm utilizes only the knowledge of the aggregate signal $x(t)$. This work focuses on appliance status detection, aiming to estimate the state variables $s(t)$ of the appliance of interest.

Note that the proposed methods to solve Problem 2 require one label per timestamp and per appliance. On the contrary, methods aiming to solve Problem 1, i.e. time series classifiers, require only labels indicating if an appliance has been used within a time frame. Such labels are significantly easier to collect with non-intrusive solutions, such as asking people to answer questionnaires. In the following sections, we discuss the existing methods proposed in the literature for solving the two different problems.

A. Non-Intrusive Load Monitoring

In the literature, Problem 2 and the corresponding proposed methods are usually referred to as energy disaggregation [1] (a.k.a. NILM). NILM relies on identifying the power consumption (or on/off state activation) of individual appliances using only the total aggregated load curve [7], [18]–[20].

1) Sequence-to-Sequence Approaches: Most methods proposed to solve Problems 2 are sequence-to-sequence approaches. Early NILM solutions involved Combinatorial Optimization (CO) to estimate the proportion of total power consumption used by distinct active appliances at each time step [1]. Later, semi-supervised and unsupervised machine learning algorithms, such as factorial hidden Markov Models (FHMM), were investigated [21]. These solutions mainly

used expert domain knowledge, and the accuracy reported is low compared to supervised ones [18]. NILM gained popularity in the late 2010s, following the release of smart meter datasets [2]–[4] allowing training and benchmarking supervised methods that demonstrate significantly better performance than semi-supervised and unsupervised ones [18]. Kelly et al. [22] were the first to investigate deep learning (DL) approaches to tackle the NILM problem and assessed the superiority of three different DL architectures against FHMM and CO. Since then, numerous studies have proposed different DL methods for solving the NILM based on various kinds of architectures. A plain Fully Convolutional Network (FCN) was proposed [23], followed by a Dilated Attention ResNet [24], and, inspired by solutions for image segmentation [25], the Temporal Pooling [26] and UNet [27] architectures were adapted to detect appliance status activation. More recently, hybrid architectures were investigated, such as BiGRU [28] that mixes Convolution layers and Bidirectional Gated Recurrent Units and inspired by Transformer-based approaches such as BERT [29], BERT4NILM [30] was proposed, followed by TransNILM [31], an extension of [26] specifically designed to solve the appliance localization problem.

Nevertheless, all the aforementioned sequence-to-sequence methods require the individual appliance load curves to be trained. Gathering such labels requires installing sensors measuring each device in many different houses. Generating large enough datasets to train accurate sequence-to-sequence methods is costly, invasive, and time-consuming.

2) Weakly Labeled Approaches: A recent study [5] introduced a weak supervision paradigm for NILM, casting it as a Multiple-Instance Learning (MIL) problem. The authors proposed a Convolutional Recurrent Neural Network (CRNN) designed to utilize both weak labels (sequence-level annotations indicating whether an appliance was active within a segment) and strong labels (fine-grained, frame-level annotations). The method required a combination of strong and weak labels to achieve acceptable accuracy. Furthermore, it was only tested on datasets with appliance-level power data, making it unsuitable for scenarios where only appliance possession information is available without usage details. These constraints limit its practicality and scalability in real-world applications.

B. Appliance Detection

Instead of tackling Problem 2 directly (usually not applicable in practice due to the lack of labels), we can target Problem 1, which can be seen as an intermediary step before Problem 2. As mentioned earlier, Problem 1 consists of detecting which devices have been used within a large time frame (instead of a prediction per timestamp). Problem 1 can be treated as a supervised binary classification problem and solved using a trained times series classifier [7]. In contrast to sequence-to-sequence methods, such classifiers require labels indicating if an appliance has been used within a time frame. Such labels are significantly easier to collect with non-intrusive solutions such as surveys (questionnaires) sent to the customers. Recent studies and benchmarks [6]–[8] evaluated the state-of-the-art time series classification methods on this problem and revealed that deep-learning methods are the most accurate and scalable to solve the appliance detection problem. Nevertheless, the output of such classifiers only indicates if a device is turned on during a time frame but cannot indicate when exactly the appliance has been used (i.e., solving Problem 2 directly).

C. Explainable Artificial Intelligence

Explainable artificial intelligence (XAI) is a growing field of interest. Using deep neural networks, common interpretable techniques employ approaches based on heatmap visualizations derived from the model architecture. These visualization techniques can be categorized into three distinct groups: gradient-based attention visualizations [12], visualization based on Class Activation Maps (CAMs) [11], and perturbation-based input manipulation [9], [10], [32]. Notably, CAM and gradient-based approaches are widely applied in computer vision for object localization [12], [33], [34]. These methods have also been explored in the time series domain, particularly for time series classification, where explainability in this context aims to identify discriminative features that explain why a time series belongs to a specific class [10], [14], as well as for anomaly detection [13], [15]. Techniques such as LIME [9], WindowSHAP [10], and class-activation-based methods such as CAM [11] and Grad-CAM [12] have been adapted to this task. However, none of these methods have been applied to address the Non-Intrusive Load Monitoring (NILM) problem. By applying an explanation method to a classifier trained to detect whether an appliance has been turned on, it is possible to localize the event by highlighting the most significant timestamps contributing to the classification decision. Therefore, using CAM applied on top of a trained classifier for appliance detection (solving Problem 1) can be used to localize the appliance pattern in a smart meter consumption series (solving Problem 2). Formally:

Definition II.1 (Class Activation Map)**.** For a given time series, and in the context of a trained deep-learning classifier that includes a Global Average Pooling (GAP) layer between the final convolutional layer and the last fully connected layer followed by a softmax activation; let denote $f^k(t)$ the $k-th$ feature map at timestep t of the last convolutional layer. Therefore, for a class of interest c, the CAM explanation, written CAM_c, is defined as the weighted sum: $\text{CAM}_c = \sum_k w_c^k f^k$ with w_c^k representing the weights associated with class c of the fully connected layer that functions as a classifier.

III. Research Questions

As a consequence, and as illustrated in red in Figure 2, the objective of this paper is to use an explainability approach (such as CAM) applied on top of classifiers (trained for Problem 1) to solve Problem 2. This would mean that the NILM problem can be handled without conducting expensive surveys to gather labels to train accurate sequence-to-sequence

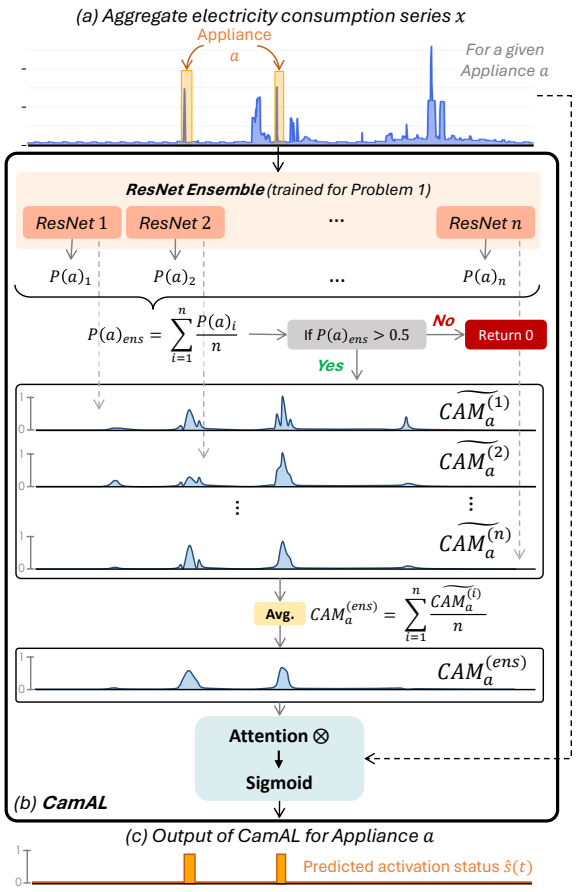


Fig. 3. CamAL framework overview.

solutions (i.e., requiring practitioners to install dedicated sensors per appliance in a large number of households).

From a business and industrial point of view, answering this question is crucial as using this approach would save a tremendous amount of time and money while reducing significantly CO_2 emissions. We divide the latter into different research questions that we will address in this paper:

- **RQ1**: *Are weak labels enough to reach the performances of NILM methods trained on strong labels?*
- **RQ2**: *Are Appliance Detection accuracy and Appliance Localization accuracy correlated?*
- **RQ3**: *What are the optimal design choices to perform weakly supervised NILM?*
- **RQ4**: *Is the information of appliance possession in this household (one label only) enough to train CamAL?*
- **RQ5**: *Can we use CamAL predictions to train strongly supervised NILM approaches?*

IV. CamAL: A Weakly Supervised Approach for Appliance Pattern Localization

We now describe CamAL, our proposed approach that enables the detection and localization of appliance patterns in aggregated consumption series. CamAL can be decomposed into two parts (see Figure 3): (1) an ensemble of deep-learning classifiers that performs the detection and (2) an explainability-based module that localizes the appliance (when detected). The ensemble of deep learning classifiers is based on different Convolutional ResNet architectures with varying kernel sizes. In simple terms, the explainability-based module can be described as extracting the CAM of all the different classifiers of the ensemble and using it as an attention mask to highlight the parts of the input sequence that contribute the most to the decision. In this section, we first describe the ResNets ensemble used for appliance detection and then delve into the details of our appliance localization-based module.

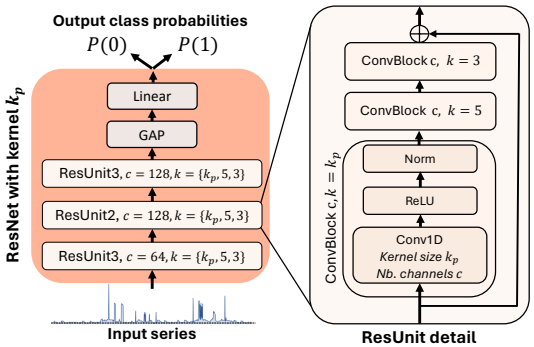


Fig. 4. Detail of the ResNet architecture (for a specific kernel k_p).

A. Step 1: An Ensembling Approach for Appliance Detection

Detecting whether an appliance has been used during a specified period can be framed as a time series classification (TSC) problem, as discussed in previous literature [7]. For this task, a binary classifier is trained to recognize the use of an appliance, assigning a single label (0 or 1) to the entire time series. Previous research [7], [8] has shown that deep learning methods are particularly effective for this challenge. These approaches include convolution-based methods, such as non-deep learning ones such as Rocket and its variant [35], [36] and deep learning ones, such as ResNet, InceptionTime [37], as well as convolutional-transformer methods such as TransApp [8]. Nevertheless, non-deep learning convolutional-based methods like Rocket are not interpretable and, therefore, unsuitable for solving our problem. Conversely, architectures such as InceptionTime and TransApp were designed as general, purpose models to achieve good classification performance regardless of the pattern length, in our case, different appliance usage patterns. In addition, these models are deeper and less efficient than simple ResNet architectures.

Therefore, to develop an ensemble of classifiers that is both accurate and efficient, CamAL employs convolutional residual networks (ResNets) tuned for specific appliance patterns to detect the presence of an appliance in a given series. These networks are recognized for their accuracy, scalability, and well-studied decision-making processes on time series data, making them a robust backbone for our solution.

1) CamAL ResNets Ensemble: The Residual Network (ResNet) architecture was introduced to address the gradient vanishing problem encountered in large CNNs [38]. A ResNet architecture has been proposed for time series classification in [14] and has shown great performance on different benchmark [39]. The architecture is composed of 3 stacked residual

Algorithm 1 CamAL ResNet Ensemble Training for an Appliance a

Require: training dataset $\mathcal{D}_{\text{train}}$, validation dataset $\mathcal{D}_{\text{validation}}$, number of ensemble networks n (by default, $n = 5$)

Split $\mathcal{D}_{\text{train}}$ into $\mathcal{D}_{\text{train-sub}}$ (80%) and $\mathcal{D}_{\text{val-sub}}$ (20%) to monitor training and prevent overfitting.
for each kernel size $k_p \in K_p$ **do**
  for trial $t \in \{1, 2, 3\}$ **do**
    Train a ResNet with kernel size k_p on $\mathcal{D}_{\text{train-sub}}$.
    Evaluate the model on $\mathcal{D}_{\text{validation}}$ and store the validation loss.
  end for
end for
Collect all trained models and their validation losses.
Select the n models with the lowest validation loss on $\mathcal{D}_{\text{validation}}$.
return Ensemble of n models trained to detect a.

blocks connected by residual connections: this means that the input of a residual block is taken and added to its output. Each residual block comprises 3 convolutional blocks as described in the ConvNet architecture (same kernel size $\{8, 5, 3\}$, but each layer in a block uses the same number of filters). The three residual blocks came with respectively $\{64, 128, 128\}$ filters, and, at this end, a global average pooling is performed along the temporal dimension followed by a linear layer and a softmax activation function to perform classification.

We leverage the proposed baseline [14] to an ensemble of n networks differing in kernel sizes within the convolutional layers. By default, we set $n = 5$. More specifically, the ensemble is based on an ensemble of networks trained with different kernel sizes k_p (with $k_p \in K_p = \{5, 7, 9, 15, 25\}$). The ResNet architecture used in our ensemble is shown in Figure 4 according to kernel size k_p. This design choice is based on the intuition that varying the size of the kernel changes the receptive fields of the convolutional neural network (CNN), offering different levels of explainability. We use the procedure described in Algorithm 1 to train the ResNet ensemble, which aims to train multiple ResNets with the same kernel k_p and then select the networks that best detect the appliance a regarding a validation dataset. We motivate the choice of the number of ResNets n used in our ensemble as well as the introduction of different kernels k_p in Section V-G.

B. Step 2: Appliance Pattern Localization

Identifying the discriminative features that influence a classifier's decision-making process has been extensively studied. Using deep-learning architecture for classification tasks, different methods have been proposed to highlight (i.e., localize) the parts of an input instance that contribute the most to the final decision of the classifier [11]–[13]. Based on this previous work, we developed a specific CAM-based method to localize appliance patterns in a given consumption series. Our approach involves extracting the CAMs from all the ResNets in the trained ensemble, computing their average, and applying the resulting map as an attention mask to the input series. This process highlights the regions in the time series that are most indicative of the appliance's operation while considering the shape of the aggregate signal to better localize the exact appliance activation time. The detailed steps of our method are outlined below and depicted in Figure 3:

*1) **ResNet Ensemble Prediction**:* An aggregated input sequence $\boldsymbol{x}$ is fed into the ensemble of ResNet models. Each model predicts the probability of detection that appliance a is present in $\boldsymbol{x}$. The ensemble prediction probability is computed by averaging the individual model probabilities: $\text{Prob}_{\text{ens}} = \frac{1}{n}\sum_{i=1}^{n} \text{Prob}_i$, where n is the number of models in the ensemble, and Prob_i is the prediction of model i.

*2) **Appliance Detection**:* If the ensemble probability exceeds a threshold (e.g., $\text{Prob}_{\text{ens}} > 0.5$), the appliance is considered detected in the current window. Otherwise, the appliance is undetected, and the activation status (i.e., localization) is set to 0 for each timestamp.

*3) **CAM Extraction**:* If the appliance a is detected, we extract each ResNet's CAM for class 1. As introduced before, for univariate time series, the CAM for class c at timestamp t is defined as: $\text{CAM}_{c=1}^{(i)}(t) = \sum_k w_{c=1}^k \cdot f^k(t)$, where w_c^k are the weights associated with the k-th filter for class c, and $f^k(t)$ is the activation of the k-th feature map at time t for the CAM that correspond to the i-th ResNet in the ensemble.

*4) **CAM Processing**:* Each $\text{CAM}^{(i)}$ is normalized to the range $[0, 1]$ by dividing it by its maximum value. Then, the average of each extracted CAM of the ensemble is computed as follows: $\text{CAM}^{(ens)}(t) = \frac{1}{n}\sum_{i=1}^{n} \widetilde{\text{CAM}}^{(i)}(t)$.

*5) **Attention Mechanism**:* $\text{CAM}^{(ens)}$ serves as an attention mask, highlighting the ensemble decision for each timestamp. We apply this mask to the input sequence through pointwise multiplication and pass the results through a sigmoid activation function to map the values in $[0, 1]$: $s(t) = \textit{Sigmoid}(\text{CAM}^{(ens)}(t) \circ \boldsymbol{x}(t))$.

*6) **Appliance Status**:* The obtained signal is then rounded to obtain binary labels (1 if $s(t) \geq 0.5$), indicating the appliance's status and resulting in a binary time series $\hat{s}(t)$.

C. From Binary Labels to Consumption per Appliance

To estimate the individual power consumption $\hat{p}_a(t)$ of an appliance a using the predicted status signal $\hat{s}(t)$ from CamAL, we employ a straightforward method. First, we multiply the binary status signal $\hat{s}(t)$—where $\hat{s}(t) = 1$ by the mean power consumption P_a of the appliance (this parameter can be inferred from the dataset or provide by expert) as:
$\hat{p}_a^{\text{initial}}(t) = \hat{s}(t) \cdot P_a$.

Then, to ensure that the estimated individual power consumption does not exceed the total aggregate power consumption at any given time t, we apply a clipping operation that aims to adjust the estimated power so that it is always less than or equal to the observed aggregate consumption $x(t)$: $\hat{p}_a(t) = \min\left(\hat{p}_a^{\text{initial}}(t),\, x(t)\right)$.

V. Experimental Evaluation

All experiments are performed on a server with 2 Intel Xeon Platinum 8260 CPUs, 384GB RAM, and 4 NVidia Tesla V100 GPUs with 32GB RAM. The code (Python 3.12) is publicly available [40], as well as a corresponding demo [41].

TABLE I
DATASETS DETAILS, PREPROCESSING PARAMETERS AND APPLIANCE

Dataset	Nb. houses	Max. ffill	Appliance	ON Power	Avg. Power
UKDALE	5	3 min	Dishwasher Microwave Kettle	300 W 200 W 500 W	800 W 1000 W 2000 W
REFIT	20	3 min	Dishwasher Washing Machine Microwave Kettle	300 W 300 W 200 W 500 W	800 W 500 W 1000 W 2000 W
IDEAL	39 (+216 w/o submeter)	30 min	Dishwasher Washing Machine Shower	300 W 300 W 1000 W	800 W 500 W 8000 W
EDF EV	24	1h30	Electric Vehicle	1000 W	4000 W
EDF Weak	558 (w/o submeter)	1h30	Electric Vehicle (Possession only)	/	/

A. Datasets

We use 5 datasets in our study. The first four—**UKDALE** [3], **REFIT** [4], and **IDEAL** [42] (all publicly available), and the private **EDF EV** dataset, provide both aggregate household power and individual appliance measurements. The fifth, EDF **EDF Weak**, is a private survey-based dataset containing only aggregate consumption data and electric vehicle ownership information. Further details are provided below.

1) Public Datasets: UKDALE [3] and REFIT [4] are two well-known datasets used in many research papers to assess the performance of NILM approaches [7], [23], [30], [31]. The two datasets contain high-frequency sampled data collected from small groups of houses in the UK and focus on small appliances. The IDEAL [42] dataset comprises data from 255 households in the United Kingdom. For all the houses, the aggregate main power consumption of the house was recorded, and each participant filled out a questionnaire to provide some information about their household, including the type and number of appliances owned. In addition, more detailed data are available for a subset of 39 households, including the individual electricity power consumption for different monitored appliances.

2) EDF Datasets: We use two private EDF datasets:
[EDF EV]: Data from 24 French households (July 2022–February 2024) with an average recording duration of 397 days (range: 175–587 days). Each house has 30-minute aggregate power readings and corresponding EV charger load curves.
[EDF Weak]: A survey-based dataset of 558 French households (September 2020–December 2022). Only total household power consumption was recorded, while EV ownership information was obtained via questionnaires.

B. Data Processing

According to the parameters reported in Table I, we resample and readjust recorded values to round timestamps by averaging the power consumed during the interval Δ_t and forward-filling the missing values.

To meet a challenging real-world scenario in this study, we evaluate the model's performance using unseen data from different houses within the same dataset [18]. This means that distinct houses were used for training and evaluation to ensure robust performance assessment. In addition, we note that we select for each dataset the appliances that consumed the most power and are suitable for localization (in contrast to always ON devices such as the Fridge).

For the UKDALE dataset containing only 5 houses, we use houses 1, 3, and 4 for training while randomly selecting houses 2 and 5 as validation or test sets. For all the other datasets, the houses used for train, valid, and test are randomly chosen. More precisely, the test set contains 2, 6, and 4 houses, and the validation set contains 2, 2, and 4 houses for REFIT, IDEAL, and EDF EV, respectively.

As two comparative baselines require the use of a non-overlapping window length $w = 510$ as input [26], [31], we slice the consumption data into non-overlapping subsequences of length $w = 510$ for training and evaluating all models. Subsequences containing any remaining missing values after our preprocessing are discarded. We note that we scaled the data by dividing the aggregate input consumption series by 1000 to ensure training stability. The ground true status is calculated according to the "ON" status threshold reported in Table I. In addition, before evaluating the models, we apply the process described in Section IV-C to all the models according to the appliance average power (i.e., P_a) reported in Table I.

C. Selected Baselines

We compare our solution against different sequence-to-sequence strongly supervised baselines. Our selection of competitors is based on their performance in previous studies [28], [31]). We include two CNN-based architectures, **Unet-NILM** [25], a UNet convolutional-based architecture, and **TPNILM** [26], a temporal pooling-based architecture. In addition, we include a recently proposed recurrent-based architecture, **BiGRU** [28], that combines Convolution and Recurrent layers and **TransNILM** [31], a SotA Tranformer-based architecture based on temporal pooling.

Finally, we include the CRNN architecture proposed in [5] that we decline in two versions. In the rest of our paper, we refer to **CRNN** for the supervised version of the architecture trained using both strong and weak labels for each subsequence. Conversely, we refer to **CRNN Weak** for the weakly supervised version of the architecture trained only weak labels.

For training, the strongly supervised baseline received one label per timestamp, while the two weakly supervised ones received one label per subsequence. For all models, we used the default parameters provided by the authors and trained them using the Binary Cross Entropy Loss. In addition, we note that each baseline, including CamAL, is trained in one model per appliance setting.

[Theoretical Model Complexity] We derive the theoretical complexity and the number of trainable parameters of selected baselines, including CamAL (see Table II). We use L for the time series length; C, K are the number of channels and the kernel size used in a convolutional layers kernel, respectively; I, H are the input hidden dimensions and the number of recurrent units used in a recurrent layer, respectively; D is the inner dimension used in a Transformer layer.

TABLE II
THEORETICAL COMPLEXITY AND NUMBER OF TRAINABLE PARAMETERS FOR THE DIFFERENT BASELINES.

Model	Theoretical Complexity	# Trainable Param.
CamAL	$O\big(n_{\text{ResNet}} \cdot L \cdot C^2 \cdot K\big)$	$n_{\text{ResNet}} \times 570$K
CRNN (Weak/Strong)	$O\big(L \cdot C^2 \cdot K \cdot (I \cdot H + H^2)\big)$	1049K
BiGRU	$O\big(L \cdot C^2 \cdot K \cdot (I \cdot H + H^2)\big)$	244K
Unet-NILM	$O\big(L \cdot C^2 \cdot K\big)$	3197K
TPNILM	$O\big(L \cdot C^2 \cdot K\big)$	328K
TransNILM	$O\big(L^2 \cdot D \cdot L \cdot C^2 \cdot K \cdot (I \cdot H + H^2)\big)$	12418K

D. Evaluation Metrics

In this work, we primarily focus on assessing the performance of the different baselines regarding their ability to detect *when* an appliance was used and the underlying power estimated. We note that we also study the ability of CamAL to detect *if* an appliance has been used in a given series using standard classification metrics. In addition, as we proposed an ensemble approach, we evaluate the training time of our solution compared to the other baselines. More precisely, we use the following measures:

[Appliance Localization and Energy Estimation]: The F1 Score, defined as the harmonic mean of Precision (Pr) and Recall (Rc), is used to evaluate the model's predictive performance by balancing correct detections against false positives. To measure the quality of energy estimation, we used the standard Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). In addition, we used the Matching Ratio (MR), based on the overlapping rate of true and estimated prediction, and considered as the best indicator performance for energy disaggregation [43]: $MR = \frac{\sum_{t=1}^{T} min(\hat{y}_t, y_t)}{\sum_{t=1}^{T} max(\hat{y}_t, y_t)}$, where T represents the total number of intervals, y_t is the true and $\hat{y}_t$ is the predicted power usage of an appliance.

[Appliance Detection]: The F1 Score is widely used to benchmark binary classification problems with imbalanced data. Nevertheless, this measure is often applied only to the minority class. However, in appliance detection scenarios, the minority class may vary, depending on the frequency of use of the appliance and the subsequence window length used for generating the dataset. Therefore, to evaluate overall performance and account for variability, we used the Balanced Accuracy, that provides an indicator regardless of the minority class: Balanced Accuracy $= \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right)$.

[Scalability]: We measured the training time (total and per epoch) as well as the inference time (throughput) to assess the scalability of the different approaches.In particular, to evaluate the ability of our *ensemble* solution to scale to large datasets of consumption series compared to other baselines.

Based on the experimental setup described above, we address the different Research Questions (RQs) enumerated in Section III in the following sections.

E. *RQ1: Weakly vs Strongly Supervised Approaches*

In this Section, we answer RQ1 by comparing the performance of CamAL to other baselines and by evaluating the performance regarding the number of labels needed for training each method. We evaluate each baseline by varying the number of instances (i.e., the number of subsequences) provided during the training. For the UKDALE dataset, which consists of only very few houses, we simply divide the dataset by percentage regardless of the number of houses. For all the other datasets, we gradually add subsequence data from houses for training. As a reminder, for the strongly supervised baselines, one subsequence corresponds to 510 labels (i.e., one label per timestamp). In contrast, CamAL and the other weakly supervised baseline received one label per subsequence.

TABLE III
WEAKLY SUPERVISED APPROACHES RESULTS (AVERAGED OVER 5 RUNS).

		CamAL				**CRNN (Weak)**			
Datasets	Case	F1	MAE	RMSE	MR	F1	MAE	RMSE	MR
REFIT	Dishwasher	**0.54**	**44.8**	**242.3**	**0.2**	0.0	50.5	295.6	0.0
	Kettle	**0.7**	**10.8**	**125.6**	**0.48**	0.3	353.6	664.2	0.1
	Microwave	**0.16**	**4.3**	**64.6**	**0.09**	0.0	5.6	72.3	0.0
	Washer	**0.14**	19.6	**172.7**	**0.03**	0.0	**19.5**	176.3	0.0
UKDALE	Dishwasher	**0.46**	40.4	273.8	**0.03**	0.0	**35.5**	**252.4**	0.0
	Kettle	**0.76**	**20.9**	**191.7**	**0.3**	0.3	61.9	267.6	0.1
	Microwave	**0.13**	**6.9**	81.3	0.0	0.1	19.8	**77.1**	**0.03**
IDEAL	Dishwasher	**0.32**	**10.6**	**116.7**	**0.11**	0.0	10.8	125.9	0.0
	Shower	**0.89**	**9.7**	**131.8**	**0.8**	0.7	39.8	489.4	0.5
	Washer	**0.04**	14.7	**151.1**	**0.01**	0.0	**14.6**	151.8	0.0
EDF EV	EV	**0.74**	**230.0**	**850.2**	**0.46**	0.3	2371.0	2818.9	0.1
Avg.		**0.38**	**38.5**	**227.2**	**0.23**	0.16	273.6	522.4	0.07

1) Results: Figure 5 reports the results for all the appliance localization of the 4 datasets for all the baselines in terms of accuracy (F1 Score) regarding the number of labels used for training each method. The results show that for each case, the NILM baselines require significantly more labels to achieve the same accuracy as CamAL, from 20 times more for the Microwave case on the UKDALE dataset to 500 times more for the Washer case on the IDEAL dataset. On average, we found that the NILM baselines require $144.27\times$ more labels to be able to achieve the same performance as CamAL. However, we also note that in all the scenarios, the fully supervised baselines outperform our solution when using all the possible labels available at the cost of large differences in labels used for training. Nevertheless, in 5 cases out of 11, CamAL almost equals the performances of strongly supervised approaches. In addition, we note that CamAL significantly outperforms CRNN Weak regardless of the number of labels used for training for almost all datasets and appliances.

Table III reports the detailed results of our method compared to the other weakly supervised one (CRNN Weak) using all the instances (label) available for all the cases and datasets. The results demonstrate that CamAL significantly outperforms the other weakly supervised methods on all the datasets and appliance localization scores. More specifically, we note an improvement of the average score on all datasets and cases of more than 135% in terms of F1 Score and 247% in MR.

The training time of the different baselines averaged across all cases, and datasets are reported in Figure 7(a) (left), while the training time according to the number of labels used for training for the IDEAL dataset (averaged for all cases) is shown in Figure 7(a) (right). The results show that

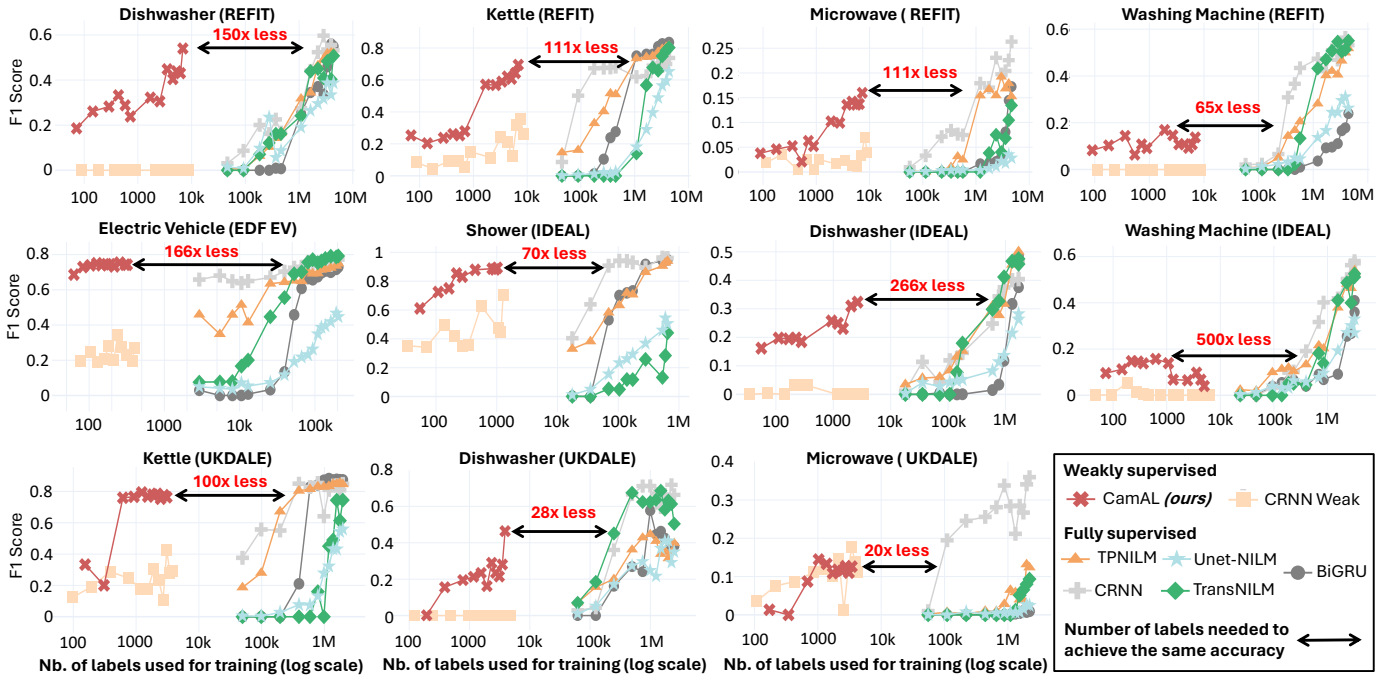


Fig. 5. Overall results comparison according to the number of labels for training for each method.

CamAL is among the two fastest solutions and is much faster than the other weakly supervised baseline, despite being an ensembling method. These results corroborate the theoretical analysis carried out in Section V-C.

F. *RQ2: Classification vs Localization*

In this Section, we answer RQ2, which aims to understand the correlation between the detection (i.e., Problem 1 framed as a classification problem) performance of CamAL and its appliance localization pattern ability (i.e., Problem 2).

Each point of the scatter plot on Figure 6(b) corresponds to the performances obtained by CamAL for each appliance and dataset. The y-axis shows the localization score (F1 Score) according to the classification score (Balanced Accuracy) shown on the x-axis. The results demonstrate a specific correlation between these two scores, highlighted by the 3rd-order regression line plotted on the graph. More specifically, we note that reaching a good accuracy (more than 0.9) implies getting a good localization of the appliance pattern (more than 0.7 in terms of F1 Score). However, the reciprocity is not true; for example, a relatively good localization performance is reached with a lower classification accuracy on the EDF EV dataset, while the same detection performance does not provide good localization performance in other cases on other datasets. Nevertheless, the detection accuracy can be used as a proxy to assess the localization accuracy (especially in cases when the labels are not available).

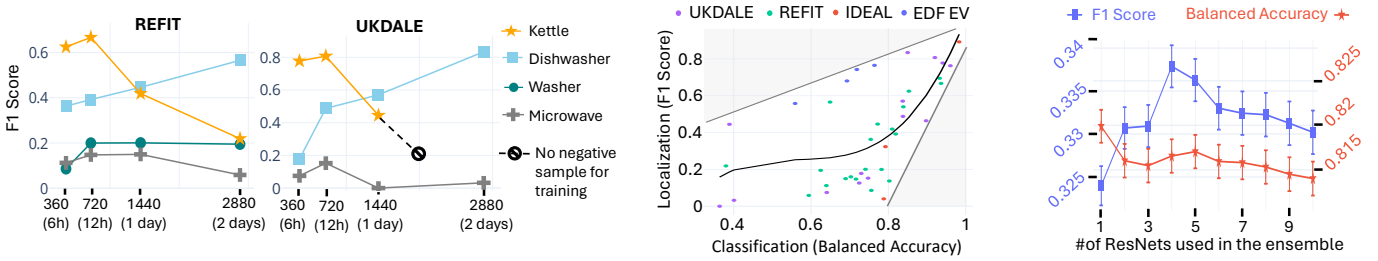
G. *RQ3: Ablation Studies*

In this Section, we perform different experiments to assess the influence of key parameters (answering RQ3). First, we study the influence of window length on CamAL performances. In other words, *how weak can the labels be?* Then, we conduct an ablation study to assess the proposed design choice of CamAL; we study the influence of (1) the number of ResNets used inside the CamAL ensemble, (2) the diversity of the size of kernels used in our networks, and (3) the importance of the attention-sigmoid module.

TABLE IV
INFLUENCE OF CAMAL DESIGN CHOICE ON PERFORMANCE (FOR ALL REFIT APPLIANCES AND AVERAGED OVER 10 RUNS).

Metric	CamAL	w/o Attention module	w/o Different kernel k_p
F1 ↑	**0.336**	0.165 *(-50.85%)*	0.317 *(-5.6%)*
Pr ↑	**0.511**	0.159 *(-68.85%)*	0.499 *(-2.21%)*
Rc ↑	0.291	**0.300** *(+3.08%)*	0.275 *(-5.68%)*
MAE ↓	**21.096**	26.843 *(-27.25%)*	21.336 *(-1.14%)*
MR ↑	**0.162**	0.114 *(-29.59%)*	0.156 *(-3.82%)*

1) How Weak Can the Labels Be?: Figure 6(a) shows the influence of the window length used for training CamAL on the localization performances reached for the different appliances using the UKDALE and REFIT datasets. We note that to match with the rest of our experimental studies, the testing set remains the same, and we use subsequences of length 510. The results demonstrate that for small appliances such as Kettle and Microwave, CamAL benefits from small windows (one label every 6 hours or 12 hours), while it is the opposite for big appliances, for which CamAL seems to perform better using longer windows. This can be explained by the fact that the Kettle and the Microwave are used mostly daily for these two datasets. Therefore, using 1 or 2 days led to obtaining a really unbalanced dataset and, therefore, little data available for training our method after balancing the class. This phenomenon is highlighted by the fact that it was not



(a) Impact of the training window length on CamAL's appliance localization performance.

(b) CamAL detection vs. localization scores averaged across datasets and cases .

(c) CamAL performances according to the # of ResNets (REFIT dataset).

Fig. 6. CamAL performance analysis: (a) effect of training window length, (b) detection vs. localization scores, and (c) influence of the number of ResNets.

possible to train CamAL using a window length of 2 days on the UKDALE dataset, as no negative samples were available. Consequently, the window length impacts the accuracy as it affects our ability to build a balanced training set. For large enough datasets with a balanced number of households with and without the corresponding appliances, the impact of the window length would be minimal.

2) What Is the Impact of CamAL's Design on Performance?: Figure 6(c) shows the results averaged over all the cases of the REFIT dataset in terms of localization (F1 Score) and classification score (Balanced Accuracy) by varying the number of ResNet n used in the ensemble from 1 to 15. We can notice that the classification score is stable regarding the number of ResNet used in the ensemble. In contrast, the localization score varies according to the number of classifiers used: it is minimal when using only one ResNet, reaches a peak around 4-5 ResNets, and then decreases. These results confirm that using an ensemble of ResNet instead of a single one leads to better localization performance. However, using too many classifiers can slightly hurt CamAL performances.

Table IV regroup the results of the ablation studies conducted on the REFIT dataset (averaged over all the cases) to study the influence of CamAL design. The first column shows the results for CamAL; the second shows the results for CamAL ablated from the Attention-Sigmoid module, and the last columns correspond to CamAL using an ensemble of ResNets that doesn't use different kernel size (we set $k_p = 7$ for all the ResNets, as originally proposed in [14]). First, we notice that using the Attention-Sigmoid module greatly improves the overall performances of CamAL by more than 50% in terms of F1 Score and nearly 70% in terms of Precision. This highlights that only using the Average of the CAM extracted from the ensemble doesn't suit our problem. In fact, using the raw average of the CAMs leads to obtaining a slightly better Recall while all the other metrics are negatively impacted, meaning that the number of false activations is too high. Secondly, the results obtained using a fixed kernel in each ResNet lead to a slight drop in the results, demonstrating the importance of using different receptive fields to obtain different activation maps.

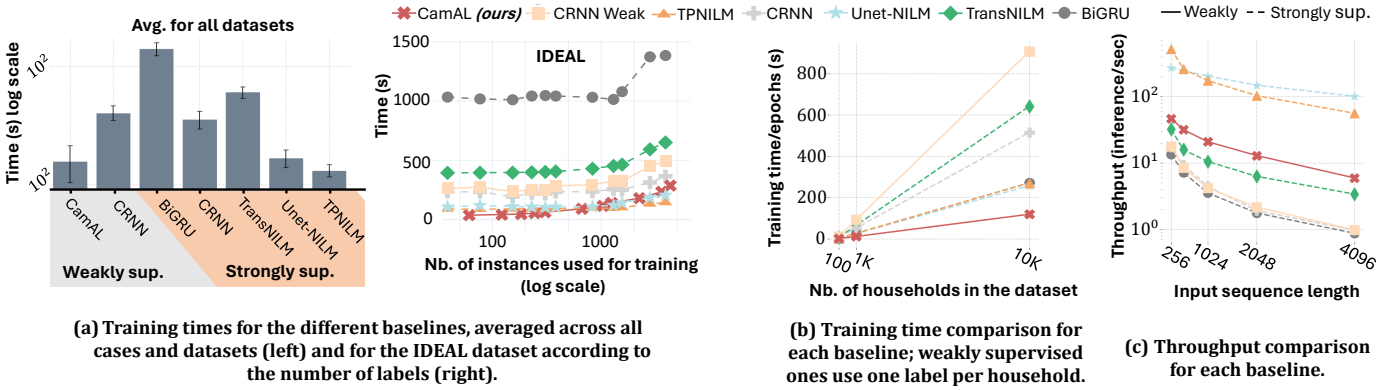
*H. **RQ4**: An Extreme (Yet Realistic) Scenario with Only One Weak Label per Household*

Although we demonstrate in the previous sections the benefits of CamAL regarding the low number of labels needed to achieve similar performance as NILM approaches, the training still relies on expensive to gather datasets (one label per time interval). However, one question that arises is: *Can we actually use only one label?* (i.e., one label indicating that the user has the appliance or not). This question is particularly important because, if successful, CamAL would require the practitioner to ask only once about the list of appliances that their consumers have in their household.

We study this extreme (but realistic) scenario by evaluating the accuracy of our approach, as well as the implications in terms of monetary cost and carbon footprint.

1) Performance Comparison of the Different Approaches: We performed experiments using the IDEAL dataset, the EDF EV, and the EDF Weak datasets (which are the only two datasets that provide enough possession-level information). For the IDEAL datasets, we used the possession information of the different appliances owned by the 255 households provided in the questionnaires to get the label for our training dataset. Subsequently, the per-timestamp labeled subgroup of 39 households was used to test the method's performance on ground truth data. For EDF datasets, we use the EDF Weak dataset as training, while the EDF EV dataset (which is labeled per timestamp) is for the test.

[Possession Only Pipeline] The two weakly labeled datasets used for training (the 255 households of the IDEAL dataset and the EDF Weak dataset) are composed of variable length electricity consumption series and *labels of possession* for different appliance a. We divide the datasets in a standard 70%/10%/20% random split for the training, validation, and test sets. We first balanced the training set through random undersampling to equalize class distribution. Following prior work [8], we then sliced household consumption into smaller subsequences to augment the training data, experimenting with various tumbling window sizes w: for IDEAL, we tested $w = \{1440, 2880, 5760, 10080, 20160, 30240, 40320, 50400\}$, corresponding to time windows ranging from one day to five weeks; for EDF Weak we used $w = \{256, 512, 1024\}$. Note that the label of the entire consumption series (i.e., *label of possession*) is assigned to all sliced subsequences



(a) Training times for the different baselines, averaged across all cases and datasets (left) and for the IDEAL dataset according to the number of labels (right).

(b) Training time comparison for each baseline; weakly supervised ones use one label per household.

(c) Throughput comparison for each baseline.

Fig. 7. Comparisons of (a) average training time; (b) per epoch training time when varying the # of households; and (c) running inference time.

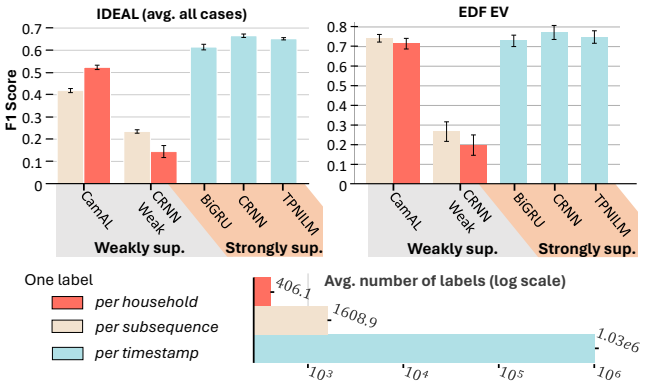
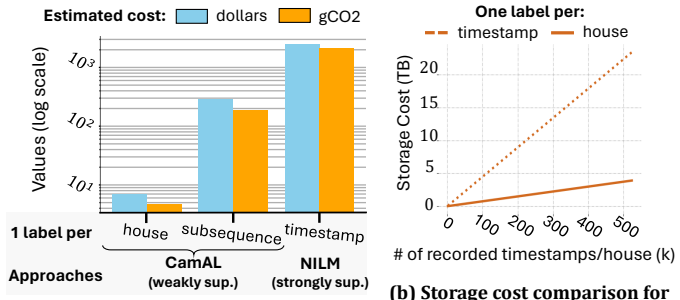


Fig. 8. Results comparison for the baselines trained using the different types of labels (one label per household, per subsequence, or per timestamp).



(a) Comparison of estimated costs per user (i.e., per household).

(b) Storage cost comparison for *weak* vs. *strong* labels for 1M households (1min frequency).

Fig. 9. Costs comparison: (a) dollars and gCO_2 per user; (b) Storage.

during the training process without any other information. We then test the baselines using the same setting as reported in Section V-C. Note that the reported localization score corresponds to the best score reached in terms of classification (Balanced Accuracy) for a given window length w.

[Results] The results are reported in Figure 8. For comparisons, we report the accuracy of strongly supervised methods and weakly supervised approaches trained with labels per subsequences obtained in the previous sections. On the IDEAL dataset, CamAL trained on household possession labels achieves better results than when trained using subsequences from the 39 submetered households. Moreover, as shown in Figure 1, CamAL uses more than 5200 times fewer labels than strongly supervised methods while achieving nearly the same accuracy in the Dishwasher scenario of the IDEAL dataset. Finally, experiments on the EDF datasets demonstrate that training with possession information yields results equivalent to CamAL trained with one label per subsequence and comparable to those obtained by strongly supervised methods.

Interestingly, the CRNN baseline performs worse on both datasets when trained with possession labels than when trained with labels per subsequence. Overall, these results demonstrate that training CamAL on the appliance detection task and using only the possession label can enable good localization results.

2) Cost Comparison of the Different Approaches: As mentioned earlier, building NILM datasets can be costly, which is the main motivation for proposing methods that can operate with weak labels. In this section, we compare the cost of collecting and storing real submeter appliance consumption data (typically NILM datasets) with surveys that ask consumers to complete a questionnaire (such as EDF Weak).

EDF must invest approximately \$1000/household in sensors and another \$1500/household per year in maintenance to collect different appliances' submeter signals. Gathering the possession information of each appliance owned in the household is done by sending a simple questionnaire that the customers fill out for a total cost of \$10/household. As an electricity supplier that wants to achieve net zero carbon in France, EDF is also concerned about the CO_2 emission of such a deployed solution. To monitor a household, the company has to send a technician to instrument the house with sensors for an average CO_2 emission cost of at least $2134g$ (assuming car CO_2 emission of $97g/km$ and an average commute distance in France of $22km$ [44], [45]). On the other hand, a study recently estimates the cost of visiting a website to be around $4.62gCO_2$ [46], which can be seen as a lower bound of a dedicated website built for consumers to answer a questionnaire on the appliances in their household.

Consider also that each timestamp of recorded electricity consumption data is stored in BIGINT values (8 bytes), while the possession information is stored in VARCHAR values (10 bytes). The costs of these two solutions are reported in Figure 9. Figure 9(a) shows that obtaining the label to be able to train the supervised method is by far the most expensive in

terms of both money ($) and emissions ($gCO2$). Conversely, asking consumers to answer surveys (daily or weekly) to obtain labels on subsequences reduces both costs by an order of magnitude, and asking for the possession information only (what CamAL uses), further reduces both costs by more than an order of magnitude. Moreover, Figure 9(b) shows that collecting strong labels for 1 million households for 5 appliances every minute results in ∼15TB/year, 6x more data than simply collecting weak labels (appliance possession information only). Overall, when compared to the strong label needs of current NILM solutions, gathering the weak labels to train CamAL reduces the monetary cost and carbon footprint by >2 orders of magnitude while also drastically reducing the storage cost, leading to a truly scalable solution.

3) Assessing CamAL's Scalability: To assess the baseline's real-world scalabilities to larger datasets, we performed experiments on synthetic data to measure the training time per epoch according to the number of households. More specifically, we generated a random consumption dataset (i.e., white noise), including both total aggregated consumption and per-timestamp appliance ground truth labels) at a 30-minute sampling rate (i.e., series of length 17520). Indeed, sequence-to-sequence NILM approaches need to operate on subsequences of an entire consumption series of a household to be trained to achieve suitable performances (e.g., windows of length 510) [47]. We trained all the baselines on a single GPU using a batch size of 64. For all the strongly supervised baselines, the entire sequences are first broken down into smaller subsequences of length 510; in contrast, the two weakly supervised approaches take the whole sequence directly as input. As shown in Figure 7(b), CamAL remains substantially more efficient than strongly supervised NILM baselines according to the number of households used for training, demonstrating the potential real-world aspect of our approach when applied to large-scale datasets.

Figure 7(c) shows the throughput (inference/sec) by varying the input subsequence length given as input, measured on a single CPU. First, we can see that CamAL is significantly more efficient than the other weakly supervised baseline (CRNN Weak). In addition, we note that CamAL is more efficient than three out of the 5 NILM baselines. The only two more efficient are convolutional-based baselines (TPNILM and Unet-NILM), but they require far more labels to be trained.

I. *RQ5: A Data Augmentation Perspective*

As a perspective of our proposed approach, we investigate in this final Section the use of our method for generating *soft labels* that can be used to enhance the performance of strongly supervised NILM approaches in case of lack of strong labels. Trained on the EDF Weak dataset for EV detection (Sec. V-H), CamAL outputs are used as soft labels for the EDF EV dataset. In the most extreme case, no ground truth data are used, only CamAL's predictions. Subsequently, we incrementally add ground truth labels from an increasing number of houses (up to 8) to assess performance improvements.

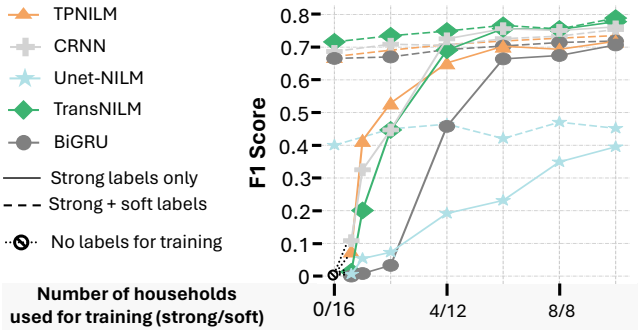


Fig. 10. Performance of strongly supervised baselines trained on CamAL soft labels (i.e., outputs) using the EDF EV dataset.

The results shown in Figure 10 demonstrate that the supervised baselines can be trained using only soft labels without a significant loss in accuracy. Additionally, when ground truth labels are scarce, all baselines achieve significantly better results by combining both strong and soft labels. For example, when using strong labels in at most one household, adding soft labels improves results between 34% (for TPNILM) and 1200% (for BiGRU). These results open new directions, including improving CamAL soft labels to obtain individual appliance power. While multiplying the localized binary signal by a single average power rating is a useful simplification, more advanced post-processing methods are needed to refine the estimated consumption further.

VI. Conclusions

We introduced CamAL, a weakly supervised approach for appliance pattern localization that only requires knowing the presence (or not) of the appliance in a household. By leveraging an ensemble of deep-learning classifiers combined with explainable classification methods, CamAL significantly reduces the need for strong per-timestamp labels and can be trained using only appliance possession information. Our experiments on 4 real-world datasets have shown that CamAL not only outperforms existing weakly supervised baselines but also reaches comparable performance to fully supervised NILM approaches while using considerably fewer labels. This makes CamAL the first truly non-invasive solution for load monitoring, aligning well with the needs of electricity suppliers and households seeking to avoid unnecessary installation costs and carbon emissions.

CamAL is already in use within EDF for internal consumption analyses and also available online as a demo [41], and plans for broader deployment are under consideration. Overall, CamAL opens a new direction in NILM research, proving that effective appliance localization can be achieved with minimal supervision using explainability-based approaches.

Acknowledgments

Supported by EDF R&D, ANRT French program, and EU Horizon projects AI4Europe (101070000), TwinODIS (101160009), ARMADA (101168951), DataGEMS (101188416), RECITALS (101168490).

REFERENCES

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

[2] J. Z. Kolter and M. J. Johnson, "Redd: A public data set for energy disaggregation research," in *Proceedings of the SustKDD workshop on Data Mining Applications in Sustainability*, 2011.

[3] J. Kelly and W. Knottenbelt, "The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes," *Scientific Data*, vol. 2, 03 2015.

[4] S. Firth, T. Kane, V. Dimitriou, T. Hassan, F. Fouchal, M. Coleman *et al.*, "REFIT Smart Home dataset," 2017.

[5] G. Tanoni, E. Principi, and S. Squartini, "Multilabel appliance classification with weakly labeled data for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, 2023.

[6] C. Deng, K. Wu, and B. Wang, "Residential appliance detection using attention-based deep convolutional neural network," *CSEE Journal of Power and Energy Systems*, vol. 8, no. 2, pp. 621–633, 2022.

[7] A. Petralia, P. Charpentier, P. Boniol, and T. Palpanas, "Appliance detection using very low-frequency smart meter time series," in *Proceedings of the 14th ACM International Conference on Future Energy Systems*, ser. e-Energy '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 214–225. [Online]. Available: https://doi.org/10.1145/3575813.3595198

[8] A. Petralia, P. Charpentier, and T. Palpanas, "Adf & transapp: A transformer-based framework for appliance detection using smart meter consumption series," *Proc. VLDB Endow.*, vol. 17, no. 3, p. 553–562, nov 2023. [Online]. Available: https://doi.org/10.14778/3632093.3632115

[9] T. Sivill and P. Flach, "Limesegment: Meaningful, realistic time series explanations," in *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, G. Camps-Valls, F. J. R. Ruiz, and I. Valera, Eds., vol. 151. PMLR, 28–30 Mar 2022, pp. 3418–3433.

[10] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

[11] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:6789015

[12] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *2017 ICCV*, 2017.

[13] P. Boniol, M. Meftah, E. Remy, and T. Palpanas, "Dcam: Dimension-wise class activation map for explaining multivariate data series classification," in *Proceedings of the 2022 International Conference on Management of Data*, ser. SIGMOD '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 1175–1189. [Online]. Available: https://doi.org/10.1145/3514221.3526183

[14] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," 2016. [Online]. Available: https://arxiv.org/abs/1611.06455

[15] P. Boniol, M. Meftah, E. Remy, B. Didier, and T. Palpanas, "dcnn/dcam: anomaly precursors discovery in multivariate time series with deep convolutional neural networks," *Data-Centric Engineering*, vol. 4, p. e30, 2023.

[16] A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Ré, "Snorkel: rapid training data creation with weak supervision," *Proc. VLDB Endow.*, vol. 11, no. 3, p. 269–282, Nov. 2017. [Online]. Available: https://doi.org/10.14778/3157794.3157797

[17] A. J. Ratner, S. H. Bach, H. R. Ehrenberg, and C. Ré, "Snorkel: Fast training set generation for information extraction," in *Proceedings of the 2017 ACM International Conference on Management of Data*, ser. SIGMOD '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 1683–1686. [Online]. Available: https://doi.org/10.1145/3035918.3056442

[18] H. Rafiq, P. Manandhar, E. Rodriguez-Ubinas, O. Ahmed Qureshi, and T. Palpanas, "A review of current methods and challenges of advanced deep learning-based non-intrusive load monitoring (nilm) in residential context," *Energy and Buildings*, vol. 305, p. 113890, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378778824000069

[19] P. Laviron, X. Dai, B. Huquet, and T. Palpanas, "Electricity demand activation extraction: From known to unknown signatures, using similarity search," in *e-Energy '21: The Twelfth ACM International Conference on Future Energy Systems, Virtual Event, Torino, Italy, 28 June - 2 July, 2021*, H. de Meer and M. Meo, Eds. ACM, 2021, pp. 148–159. [Online]. Available: https://doi.org/10.1145/3447555.3464865

[20] M. Kaselimi, E. Protopapadakis, A. Voulodimos, N. Doulamis, and A. Doulamis, "Towards trustworthy energy disaggregation: A review of challenges, methods, and perspectives for non-intrusive load monitoring," *Sensors*, vol. 22, p. 5872, 08 2022.

[21] H. Kim, M. Marwah, M. F. Arlitt, G. Lyon, and J. Han, "Unsupervised disaggregation of low frequency power measurements," in *SDM*, 2011. [Online]. Available: https://api.semanticscholar.org/CorpusID:18447017

[22] J. Kelly and W. Knottenbelt, "Neural NILM," in *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. ACM, nov 2015. [Online]. Available: https://doi.org/10.1145\%2F2821650.2821672

[23] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[24] M. Xia, W. Liu, Y. Xu, K. Wang, and X. Zhang, "Dilated residual attention network for load disaggregation," *Neural Computing and Applications*, vol. 31, no. 12, pp. 8931–8953, 12 2019.

[25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241.

[26] L. Massidda, M. Marrocu, and S. Manca, "Non-intrusive load disaggregation by convolutional neural network and multilabel classification," *Applied Sciences*, vol. 10, no. 4, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/4/1454

[27] A. Faustine, L. Pereira, H. Bousbiat, and S. Kulkarni, "Unet-nilm: A deep neural network for multi-tasks appliances state detection and power estimation in nilm," in *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, ser. NILM'20. New York, NY, USA: Association for Computing Machinery, 2020, p. 84–88. [Online]. Available: https://doi.org/10.1145/3427771.3427859

[28] D. Precioso Garcelán and D. Gomez-Ullate, "Thresholding methods in non-intrusive load monitoring," *The Journal of Supercomputing*, vol. 79, pp. 1–24, 04 2023.

[29] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *North American Chapter of the Association for Computational Linguistics*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:52967399

[30] Z. Yue, C. R. Witzig, D. Jorde, and H.-A. Jacobsen, "Bert4nilm: A bidirectional transformer model for non-intrusive load monitoring," in *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, ser. NILM'20. New York, NY, USA: Association for Computing Machinery, 2020, p. 89–93. [Online]. Available: https://doi.org/10.1145/3427771.3429390

[31] X. Cheng, M. Zhao, J. Zhang, J. Wang, X. Pan, and X. Liu, "Transnilm: A transformer-based deep learning model for non-intrusive load monitoring," in *Proceedings of the 2022 International Conference on High Performance Big Data and Intelligent Systems (HDIS)*, 2022, pp. 13–20.

[32] J. a. Bento, P. Saleiro, A. F. Cruz, M. A. Figueiredo, and P. Bizarro, "Timeshap: Explaining recurrent models through sequence perturbations," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, ser. KDD '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 2565–2573. [Online]. Available: https://doi.org/10.1145/3447548.3467166

[33] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018, pp. 839–847.

[34] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-cam: Score-weighted visual explanations for convo-

lutional neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 24–25.
[35] A. Dempster, F. Petitjean, and G. I. Webb, "Rocket: exceptionally fast and accurate time series classification using random convolutional kernels," *Data Mining and Knowledge Discovery*, vol. 34, no. 5, pp. 1454–1495, 2020.
[36] A. Dempster, D. F. Schmidt, and G. I. Webb, "MiniRocket: A very fast (almost) deterministic transform for time series classification," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. New York: ACM, 2021, pp. 248–257.
[37] H. I. Fawaz, B. Lucas, G. Forestier, C. Pelletier, D. F. Schmidt, J. Weber, G. I. Webb, L. Idoumghar, P.-A. Muller, and F. Petitjean, "InceptionTime: Finding AlexNet for time series classification," *Data Mining and Knowledge Discovery*, vol. 34, no. 6, pp. 1936–1962, sep 2020.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
[39] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: A review," *Data Min. Knowl. Discov.*, vol. 33, no. 4, p. 917–963, jul 2019.
[40] A. Petralia. Source code of camal experiments. [Online]. Available: https://github.com/adrienpetralia/CamAL
[41] A. Petralia, P. Boniol, P. Charpentier, and T. Palpanas, "Devicescope: An interactive app to detect and localize appliance patterns in electricity consumption time series," in *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, 2025.
[42] M. Pullinger, J. Kilgour, N. Goddard, N. Berliner, L. Webb, M. Dzikovska, H. Lovell, J. Mann, C. Sutton, J. Webb, and M. Zhong, "The ideal household energy dataset, electricity, gas, contextual sensor data and survey data for 255 uk homes," *Scientific Data*, 2021.
[43] E. Mayhorn, G. Sullivan, J. M. Petersen, R. Butner, and E. M. Johnson, "Load disaggregation technologies: Real world and laboratory performance," 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:115779193
[44] Agence de la transition écologique (ADEME), "Évolution du taux moyen d'émissions de CO2 en France," https://carlabelling.ademe.fr/chiffrescles/r/evolutionTauxCo2, accessed: 2024-11.
[45] Observatoire des Territoires, "Se déplacer au quotidien : enjeux spatiaux, enjeux sociaux," 2019, Accessed: 2024-11. [Online]. Available: https://www.observatoire-des-territoires.gouv.fr/sites/default/files/2023-05/fiche_analyse_mobilites_quotidiennes.pdf
[46] RESET.org, "What's the Carbon Footprint of Your Website?" 2024, accessed: 2024-11. [Online]. Available: https://en.reset.org/whats-carbon-footprint-your-website/
[47] A. Reinhardt and M. Bouchur, "On the impact of the sequence length on sequence-to-sequence and sequence-to-point learning for nilm," in *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, ser. NILM'20. New York, NY, USA: Association for Computing Machinery, 2020, p. 75–78. [Online]. Available: https://doi.org/10.1145/3427771.3427857