This is a peer-reviewed, accepted author manuscript of the following research article: Tanoni, G, Sobot, T, Principi, E, Stankovic, V, Stankovic, L & Squartini, S 2024, 'A weakly supervised active learning framework for non-intrusive load monitoring', Integrated Computer-Aided Engineering.

A Weakly Supervised Active Learning Framework for Non-Intrusive Load Monitoring

Giulia Tanoni [a], Tamara Sobot [b], Emanuele Principi [a], Vladimir Stankovic [b], Lina Stankovic [b], Stefano Squartini [a,*]

[a] *Department of Information Engineering, Università Politecnica delle Marche, Ancona, IT*

E-mail: g.tanoni@pm.univpm.it, e.principi@univpm.it, s.squartini@univpm.it

[b] *Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, United Kingdom*
E-mail: tamara.todic@strath.ac.uk, vladimir.stankovic@strath.ac.uk, lina.stankovic@strath.ac.uk

Abstract. Energy efficiency is at a critical point now with rising energy prices and decarbonisation of the residential sector to meet the global NetZero agenda. Non-Intrusive Load Monitoring is a software-based technique to monitor individual appliances inside a building from a single aggregate meter reading and recent approaches are based on supervised deep learning. Such approaches are affected by practical constraints related to labelled data collection, particularly when a pre-trained model is deployed in an unknown target environment and needs to be adapted to the new data domain. In this case, transfer learning is usually adopted and the end-user is directly involved in the labelling process. Unlike previous literature, we propose a combined weakly supervised and active learning approach to reduce the quantity of data to be labelled and the end user effort in providing the labels. We demonstrate the efficacy of our method comparing it to a transfer learning approach based on weak supervision. Our method reduces the quantity of weakly annotated data required by up to 82.6 - 98.5% in four target domains while improving the appliance classification performance.

Keywords: Non-Intrusive Load Monitoring, Deep Learning, Weak Supervision, Active Learning, Transfer Learning

1. Introduction

Energy efficiency has gained great traction in recent years [1–3], to facilitate the transition to NET-zero economy. Energy awareness plays a key role in improving energy efficiency [4–9], and active user participation can potentially increase a household's energy flexibility, leading to energy savings of up to 30% [8]. Evidence in [6] suggests that energy awareness encourages end-users to purchase energy-efficient products. This perspective may motivate users to actively participate in energy conservation and invest in devices that provide future energy and monetary savings. In fact, a study [5] conducted on two groups of low-income consumers revealed that 46.5% are interested in saving energy both for environmental and financial reasons, compared to the rest who are interested only for financial benefit or only for environmental reasons. A recent review [9] emphasised that providing effective feedback about consumption is another way to engage users actively in the long term. Moreover, the findings of the study highlight the need to develop strategies and technologies that are more user-centred. Energy awareness can be improved by monitoring energy [3] and particularly via Load Monitoring that provides detailed information about consumption. Specifically, Non-Intrusive Load Monitoring (NILM) is a purely algorithmic approach to estimate individual appliance power consumption that contribute to the measured aggregate signal, via

*Corresponding author. E-mail: s.squartini@unipvm.it.

smart metering for example. Over the past 40 years, NILM has been demonstrated as an effective software-based method for obtaining detailed energy consumption information, avoiding the installation of several meters to monitor individual appliances. NILM algorithms can be unsupervised and supervised, with the latter being more popular due to their excellent performance. Signal processing [10] and Machine Learning (ML) [11–13] methods have been initially proposed for NILM. Deep neural networks gained wide attention in the community in many fields [14–19]. Following the work of Kelly et al. [11] that proposed three deep learning-based approaches, deep neural networks (DNNs) have been widely applied in NILM achieving the state-of-the-art performance [20–27]. Many of these techniques have not demonstrated their performance in unseen environments [28] due to significant differences between *source* and *target* signal domains and the related feature spaces [29]. In NILM, differences in appliance load signatures and unknown loads inside the aggregate signal [30, 31] mainly affect the performance when a pre-trained model is deployed in a target environment.

To overcome domain differences, transfer learning [32, 33] has been demonstrated to be an effective strategy in increasing generalisation capability: recent methods operate by pre-training a neural network on a large dataset and then fine-tuning it on data acquired from the target environment [28, 34–36]. However, these approaches need an additional acquisition and labelling phase to be applied. Generally, data acquisition and annotation are costly and time-consuming procedures, also requiring expertise in the specific application field. In fact, for NILM, acquiring new signals in the target domain requires the installation of electrical sensors for each monitored appliance or the users' involvement in manually annotating appliances' states by recording and reporting the on-time and off-time related to the usage of one or more appliances. This type of label is used in supervised learning approaches. In the authors' opinion, these annotations can be most effectively gathered via a mobile app on the users' smartphone, which allows them to provide feedback about their appliance usage. If the focus is on monitoring the state of appliances, as in this work, there is no need for any hardware installation.

To reduce the requirement for labelled data, approaches based on semi-supervised learning have been proposed recently [36–38]. A different approach to reducing the labelling effort has been proposed in [39, 40], where a weakly supervised method is demonstrated to be more effective than the semi-supervised one [38]. Weak supervision allows a lightened data annotation since labels are required in a coarser form [25]. In terms of the aforementioned manual annotations, under this approach, users would only need to indicate whether an appliance was used or not within a certain time window. Also, for the transfer learning procedure, in [41] weak supervision was demonstrated to be effective compared to a supervised strategy, especially in the practical scenario of acquiring labels from the user feedback. Considering the multi-label appliance classification task, a *weak* label is provided for an entire temporal segment of the aggregate signal indicating whether an appliance is ON or OFF within that segment. Differently, *strong* labels used in supervised learning methods are annotations at the sample level, i.e., they indicate whether an appliance is ON or OFF for each sample, thus representing more fine-grained information. In Fig. 1, the concepts of weak and strong labels are graphically explained. The srong labelling approach is more prone to errors and requires intrusive sub-metering or expert knowledge about appliance load signatures for manual labelling. On the other hand, weak labels can be obtained more easily directly from the users in the target environment, by simply asking them if an appliance was active or not in a certain time period during the day as opposed to labels for each sample.

Although weak labels reduce the labelling effort, the number of time periods that need to be labelled for fine-tuning could be still large. Active Learning (AL) approaches [42, 43] are used in literature to optimise data selection for artificial intelligence algorithms by choosing the most informative data, and that way reduce the number of data segments needed to be labelled and added to the training dataset, but without compromising the algorithm performance [44]. AL approaches have been widely used for deep learning algorithms recently [45]. Specifically for NILM, a supervised AL-based framework was proposed [46] to find the trade-off between accuracy and number of queries to enlarge the training set in an unseen domain, and to improve the transferability of NILM models. Although improving the performance, this approach was based on a small original training set with strong labels, requiring sample-by-sample annotations.

We suggest that integrating a weakly supervised learning strategy into the AL framework with transfer learning avoids the need for expert labelling of target domain data, and annotation effort is reduced both in

terms of the number of signal segments and the amount of information requested from users.

In this work, we propose a weak AL NILM approach to reduce the number of signals that need to be labelled by users. By asking users to assign only weak labels to the most uncertain segments of the aggregate signal and sampling the fine-tuning set, we further reduce the user annotation effort while obtaining improved performance compared to our previous work [41, 46] upon which we build. The proposed method is completely based on weak supervision, from the network pre-training to the adaptation in the target environment through to the AL procedure. We model the multi-label classification task as a Multiple-Instance Learning (MIL) [47] problem, and generate windows of aggregate samples as in [39] to which we refer as bags. We compare the proposed method with [41] and demonstrate that sampling the fine-tuning set via AL leads to better performance. Additionally, we compare our method with a NILM benchmark semi-supervised approach [48] demonstrating the effectiveness of weak labels over unlabelled data.

In the experiments, two widely used benchmark datasets, UK-DALE [49] and REFIT [50], were used to evaluate the performance of the proposed method. They were used respectively as the *source* and *target* domain datasets to pre-train, fine-tune, and test the neural network. The results show that the Weak AL approach improves the performance compared to an non-annotated fine-tuning set, demonstrating that significant benefits can be obtained with coarser information on a small number of signals.

The paper is organised as follows. Section 2 reviews recent approaches for multi-label classification and AL in NILM. Section 3 illustrates the contributions of this paper. Section 4 presents the problem formulation and the proposed method. Section 5 describes the experimental settings in detail. Section 6 presents and discusses the obtained results. Finally, Section 7 concludes the paper and discusses future work.

2. Background

2.1. NILM as Multi-label Appliance Classification

The recent trend in low-frequency NILM literature, as illustrated by the methods discussed below, focuses on the disaggregation of more commonly available smart meter time-series measurements of low-frequency aggregate active power. Furthermore, most NILM research proposed for multi-label appliance classification is based on ML and approaches the problem using a supervised learning strategy.

Reference [13] proposed Random k-Label set (RAkEL) and Multi-Label K-Nearest Neighbours (ML-KNN) using both time- and wavelet-domain features to train the ML models. Multi-label Restricted Boltzmann Machine (ML-RBM) was proposed by Verma and colleagues [20] due to its effectiveness in learning high-level features and correlations. To achieve higher accuracy with continuously varying appliances and overcome low-frequency sampling-related problems, deep dictionary learning was adopted in [21]. A Sparse Representation Classification approach was proposed in [22], reducing the number of logging data collected for training. Temporal pooling was implemented in [23] to concatenate different time resolution information. A Gated Recurrent Units (GRUs) based approach was proposed in [24], where features from the aggregate signal and spikes are extracted using convolutional layers. A convolutional-recurrent and random-forest (RF) based architecture that addresses label correlation and class-unbalancing was proposed in [51].

An encoder-decoder architecture based on a Long Short-Term Memory network (LSTM) was adopted in [26]. A CNN followed by three different fully connected sub-networks was implemented for multi-label state and event type classification in [27]. Deep Blind Compressed Sensing was proposed in [52], exploiting compressed information to reduce transmission rate to detect devices' states.

To reduce the quantity of annotations required to train the ML algorithms, semi-supervised learning strategies have also been proposed. A semi-supervised approach is proposed in [37] with the Virtual Adversarial Learning strategy while [38] proposed a semi-supervised learning procedure based on teacher-student architecture and a Temporal Convolutional Network. Alternatively [39] proposed an approach based on a Convolutional Recurrent Neural Network (CRNN) trained with weakly labelled data, lightening the labelling effort by using a coarser type of labels to train the network.

It is worth highlighting that the approaches reviewed above still face domain adaptation issues when moving from one well-known data domain to another. Transfer learning methods are required to mitigate the domain shift. In [36], a semi-supervised Knowledge Distillation approach has been proposed to improve the domain adaptation to classify the activation states and recently in [41], a weakly supervised transfer learning

approach has been proposed to reduce the labelling effort exploiting coarser labels, assigned to an entire window of the aggregate signal, modelled as a bag of aggregate samples. Although a better performance was obtained, the approach still relied on a large number of windows from the aggregate signal.

2.2. AL for NILM

AL [44] is a concept introduced to reduce the labelling effort needed to train ML algorithms, selecting only a subset of data to be labelled while keeping an acceptable level of performance. Unlabelled data samples belonging to the query pool are usually ranked according to informativeness or distance criteria, or a combination of both. Then, based on the ranking, labels are requested only for a small portion of data, i.e., for the data samples that will contribute to the model training the most. AL has been popular in many areas recently, such as natural language processing [53] and medical image processing [54]. A recent survey of [45] gives an overview of AL approaches applied to deep learning algorithms.

AL for NILM has not been extensively investigated yet - there have only been a few attempts for event-based methods using high-frequency load measurements, based on: k-Nearest Neighbours (k-NN) in [55], Support Vector Machines (SVM) in [56], Random Forest with semi-supervised and AL combined in [57], and a DNN, using high-frequency measurements and event detection in [58], and only one approach using low-frequency measurements and supervised model-based NILM in [46]. However, in [46], only strong labels are used, which can be hard to obtain from end users in a real-world scenario.

3. Contributions

Weak supervision and AL-based strategies are effective in labelling effort reduction, but it is worth highlighting that:

- as reported in [41], the weakly supervised approach for NILM requires a dataset annotated with weak labels to train the network. This learning strategy could have a concrete consequence in a real-world data collection scenario where the end-user is involved in the labelling process.
- the AL proposed for NILM [46] depends on a sample-by-sample labelling strategy that is challenging for a non-expert end-user.

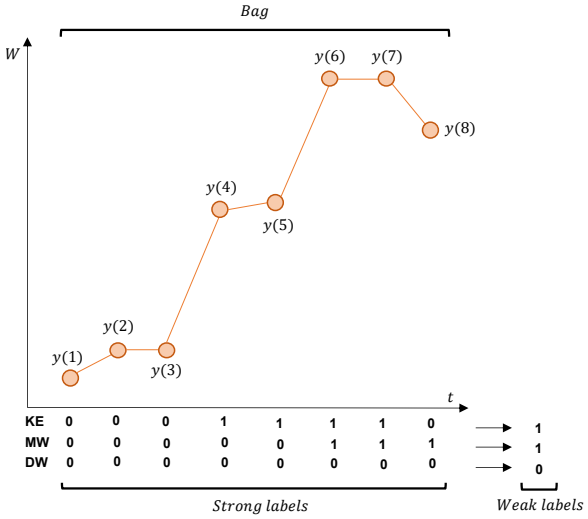

Fig. 1. Representation of strong and weak labels for a segment of the aggregate power signal. Strong labels give information about the state of activation for each instance (thus each sample of the signal $y(t)$) while the weak label gives an overall and coarse label indicating if, inside that aggregate window (i.e., a bag), a specific appliance is active. KE stands for Kettle, MW for Microwave, and DW for Dishwasher.

We address the above two challenges and fill the gaps of the existing literature by introducing a weak AL framework for low-frequency, model-based NILM solutions. In this way, we exploit the advantages of both weak supervision and AL strategies, by querying the user to assign weak (bag-level) labels to specific aggregate windows selected by the AL loop. In summary, the contributions of this work are:

- Algorithm 1, a multiple instance learning-based approach that embeds both weak supervision and AL to reduce the quantity of data to be weakly labelled, compared to the state of the art [28, 39], by selecting only the ones on which the network indicates poor confidence;
- Development of a feasible AL framework in a real-world scenario where the end-user does not need to annotate power profiles sample-by-sample, differently from [28, 39]. In this way, the effort is reduced, and annotations are less affected by errors.
- Adapted acquisition function to multi-label classification with weak labels (Algorithm 2) considering different behaviours and confidence levels for different appliances.

– Determining the optimal point, where additional samples will only negligibly, or not all improve performance via fine-tuning.
– Demonstration of the efficiency of the proposed method on two commonly used public datasets and four common household appliances (kettle, microwave, washing machine and dishwasher) to facilitate benchmarking.

Moreover, we demonstrate how the proposed approach of integrating the weakly supervised learning strategy into the AL framework improves network performance compared to our previous work [41] with reduced labelling effort.

4. Proposed Method

Each load inside a building contributes to the total power consumption $y(t)$ at time instant t based on the following relationship:

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

with $\epsilon(t)$ being the measurement noise, $s_n(t) \in \{0,1\}$ and $x_n(t)$, denoting, respectively, the state and the power associated to the n-th load at time instant t. In this work, we propose a method for multi-label appliance classification that refers to the estimation of $s_n(t)$ for all the $K \leqslant N$ appliances of interest, using one network. Thus, for each time instant, we have the activation state for each monitored appliance, as shown in Fig. 2, where for one particular input window, the activations for two detected appliances (Kettle and Dishwasher) are shown in that window.

The appliances' states are estimated using a CRNN, and the task is modelled as a MIL problem [47] to exploit weak labels. Based on the concept of *instances* and *bags*, MIL performs a weak supervision strategy in which the ground-truth is provided only at the bag level.

In our method, instances refer to the raw samples of the aggregate signal $y(t)$, and the corresponding *strong* labels are represented by one-hot vectors $\mathbf{s}(t) = [s_1(t), s_2(t), \ldots, s_K(t)]^T \in \{0,1\}^{K\times 1}$ composed of the appliances states. Bags refer to segments of $y(t)$ of length L, where the i-th bag is represented by the following vector:

$$\mathbf{y}_i = [y(iL), \ldots, y(iL+L-1)]^T \in \mathbb{R}^L. \tag{2}$$

The related *weak* label is encoded as a one-hot vector $\mathbf{w}_i$ having the same dimensions as $\mathbf{s}(t)$. Furthermore, $\mathbf{S}_i = [\mathbf{s}(iL), \mathbf{s}(iL+1), \ldots, \mathbf{s}(iL+L-1)] \in \{0,1\}^{K\times L}$ represents the set of strong labels of all the appliances related to bag i. Bags and instances, as well as weak labels and strong labels are represented in Fig. 1.

With the above definitions, it is now possible to define formally the multi-label appliance classification task based on weak labels. Specifically, by exploiting only the aggregate power signal $\mathbf{y}_i$, the aim is to learn a function $\mathbf{f}_{\boldsymbol{\theta}} : \mathbb{R}^L \rightarrow \{0,1\}^{K\times L}$ that provides an estimate $\hat{\mathbf{S}}_i$ of $\mathbf{S}_i$. In this work, the function $\mathbf{f}_{\boldsymbol{\theta}}(\cdot)$ is represented by a CRNN, with trainable parameters $\boldsymbol{\theta}$, described in the following section.

4.1. Neural Network Architecture

The proposed method is based on a CRNN that was originally proposed in [59] and then adapted for the NILM problem in [39, 41], demonstrating good results that exceed benchmarks. It comprises a convolutional and a recurrent subpart, as shown in Fig. 2. The H convolutional blocks are composed of a convolutional layer with $F \times H$ filters and kernel size K_e, a batch normalisation layer, a ReLu activation layer, and a drop out layer with rate p. The recurrent subpart consists of a bidirectional layer of U GRU units. In the following, the bag index i will be omitted for the sake of simplicity. The output $\hat{\mathbf{S}}$ is provided by a fully connected layer with sigmoid activation function. This layer will be denoted as *instance layer* in the following. The final layer of the network is the *bag layer* with sigmoid activation function that provides the estimate $\hat{\mathbf{w}}$. The relationship between the instance layer output $\hat{\mathbf{S}}$ and the bag layer output $\hat{\mathbf{w}}$ is calculated by applying the sigmoid function $\sigma(\cdot)$ to the output of a pooling function $\mathbf{p}_o : \mathbb{R}^{K\times L} \rightarrow \mathbb{R}^K$:

$$\hat{\mathbf{w}} = \sigma\left(\mathbf{p}_o(\hat{\mathbf{S}})\right). \tag{3}$$

As in [60] that applied MIL to the sound event detection task, we adopt the linear softmax pooling function defined as follows:

$$\hat{w}_k = \sigma\left(\frac{\sum_t \hat{s}_k^2(t)}{\sum_t \hat{s}_k(t)}\right), \tag{4}$$

where $\hat{w}_k$ is the k-th element of $\hat{\mathbf{w}}$, i.e., the weak label of the k-th appliance. After the bag layer, the instance-level predictions can be multiplied to the bag-level output. This procedure is performed during training and

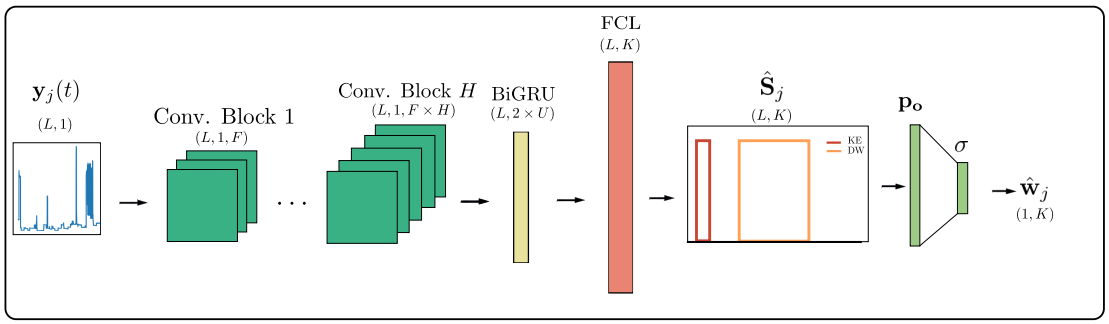

Fig. 2. CRNN architecture. FCL = Fully Connected Layer, BiGRU = Bidirectional Gated Recurrent Units.

referred to as Clip Smoothing (CS) [61]. It is based on the consistency between weak and strong labels to better deal with false activations. The final predictions are obtained applying a threshold to the instance level output.

4.2. Learning Strategy

Consider a pre-training dataset $\boldsymbol{D}^{(pt)}$ given by

$$\boldsymbol{D}^{(pt)} = \boldsymbol{D}^{(pt)}_{strong-weak} \cup \boldsymbol{D}^{(pt)}_{weak}, \tag{5}$$

where

$$\boldsymbol{D}^{(pt)}_{strong-weak} = \{(\mathbf{y}_1, \mathbf{w}_1, \mathbf{S}_1), \ldots, (\mathbf{y}_M, \mathbf{w}_M, \mathbf{S}_M)\}$$

is a dataset composed of M bags annotated with strong and weak labels, and

$$\boldsymbol{D}^{(pt)}_{weak} = \{(\mathbf{y}_{M+1}, \mathbf{w}_{M+1}), \ldots, (\mathbf{y}_{M+B}, \mathbf{w}_{M+B})\}$$

is a dataset composed of B bags annotated with weak labels only, from the source domain. Another set $\boldsymbol{D}_U = \{\mathbf{y}_{M+B+1}, \ldots, \mathbf{y}_{M+B+C}\}$ of electricity load measurements, called query pool, composed of C unlabelled bags is collected in the target environment, representing the pool for the AL process.

Based on the neural network architecture, two loss terms are defined $\mathcal{L}_s$ and $\mathcal{L}_w$, respectively, related to the instance and bag output. Both losses are the Binary Cross-Entropy functions for the related output calculated as follows:

$$\mathcal{L}_s = -\frac{1}{K}\frac{1}{L}\sum_{k=1}^{K}\sum_{t=1}^{L}[s_k(t)\log(\hat{s}_k(t)) + \\ + (1-s_k(t))\log(1-\hat{s}_k(t))], \tag{6}$$

and:

$$\mathcal{L}_w = -\frac{1}{K}\sum_{k=1}^{K}\left[w_k\log(\hat{w}_k) + (1-w_k)\log(1-\hat{w}_k)\right], \tag{7}$$

where the bag index i has been omitted for simplicity of notation. Learning is initially performed by pre-training the neural network on a large public dataset $\boldsymbol{D}^{(pt)}$. A significant advantage of the proposed method is that it allows to use strong or weak labels in the pre-training phase depending on the composition of $\boldsymbol{D}^{(pt)}$. The model is pre-trained both on strongly and weakly annotated data if $\boldsymbol{D}^{(pt)}_{strong-weak} \neq \varnothing$, or only on weakly annotated data if $\boldsymbol{D}^{(pt)}_{strong-weak} = \varnothing$. In the first case, the training loss is $\mathcal{L}_{pt} = \mathcal{L}_s + \lambda\mathcal{L}_w$, where λ balances the contribution of the two losses, while in the second case it is $\mathcal{L}_{pt} = \mathcal{L}_w$. During fine-tuning, the weights of all the convolutional blocks are not updated (i.e., they are frozen) to avoid performance degradation [28]. Instead, fine-tuning is performed only on the recurrent subpart and on the instance layer using the dataset $\boldsymbol{Q}_{tot,j}$. This dataset contains a set of bags, annotated only with weak labels, obtained by labelling a subset $\boldsymbol{Q}_{U,j}$ of $\boldsymbol{D}_U$ at each iteration j. Additional details on $\boldsymbol{Q}_{tot,j}$ and $\boldsymbol{Q}_{U,j}$ are provided in Section 4.3. The fine-tuning loss $\mathcal{L}_{ft}$ is equal to $\mathcal{L}_w$ since we suppose to collect only weak labels from the target environment (i.e., $\boldsymbol{Q}_{tot,j}$ is annotated only with weak labels).

4.3. Weakly Supervised AL Framework

The proposed Weakly Supervised AL framework, schematically illustrated in Fig. 3, comprises the CRNN model pre-trained using $\boldsymbol{D}^{(pt)}$, the query pool $\boldsymbol{D}_U$ for which only weak labels can be obtained on demand, and an acquisition function $q(\cdot)$ used to rank bags from

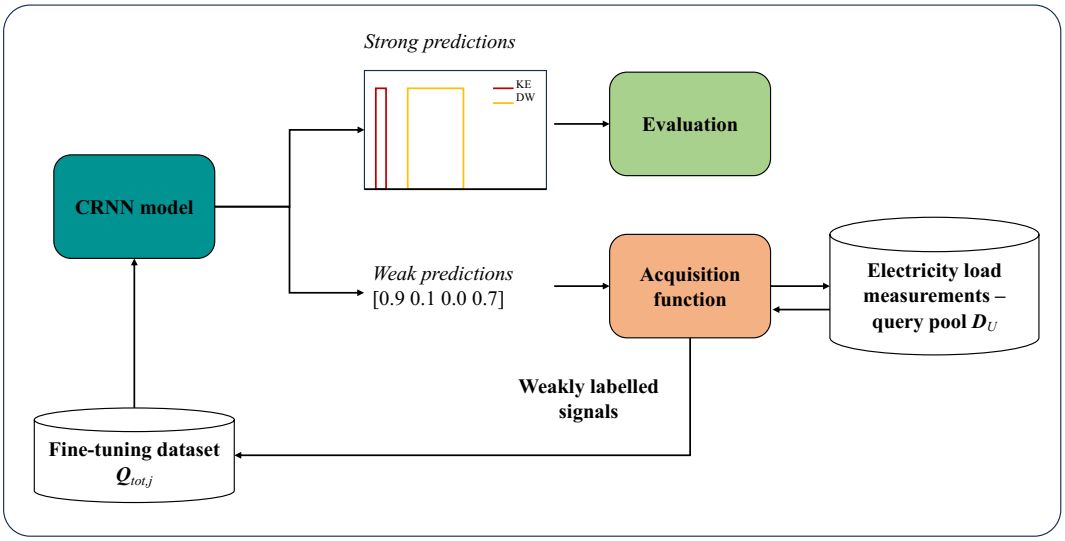

Fig. 3. Weakly Supervised AL Scheme. Each block corresponds to an element of the framework. The Convolutional Recurrent Neural Network (CRNN) model generates both strong and weak predictions. During the AL process, strong predictions are used to evaluate the current model, while weak predictions serve as input for the acquisition function. The acquisition function selects the windows to be labelled based on the uncertainty of the network predictions. The most uncertain windows are chosen, suggested to the user for annotation, and then incorporated into the fine-tuning set for the subsequent fine-tuning phase. A detailed description of the entire framework can be found in Section 4.3.

$\boldsymbol{D}_U$ and choose the most informative ones to be included in the fine-tuning of the model.

The AL process is iterative, and we indicate the iterations with j and with $\mathbf{f}_{\theta_j}$ the CRNN model trained at iteration j. The pre-trained model $\mathbf{f}_{\theta_0}$ first makes predictions on the whole query pool $\boldsymbol{D}_U$ and provides its predictions to the acquisition function. The acquisition function then chooses a subset $\boldsymbol{Q}_{U,j} \subset \boldsymbol{D}_U$, with $j = 1, ..., J$ indexing the current query of most informative aggregate bags, accounting for model uncertainty when making predictions; the more uncertain the model is about a bag, the more the bag contributes towards the model prediction if included in fine-tuning.

Then, labels are queried for the chosen subset of bags as a result of the acquisition function. Let $\boldsymbol{Q}_{weak,j}$ be the weakly annotated set during the j-th query, composed of P bags. At the end of the loop, the model is fine-tuned using bags belonging to

$$\boldsymbol{Q}_{tot,j} = \boldsymbol{Q}_{tot,j-1} \cup \boldsymbol{Q}_{weak,j},\ j = 1, ..., J, \tag{8}$$

queried up to the j-th query. Note that $\boldsymbol{Q}_{tot,0}$ is an empty set. The knowledge of the new, improved model $\mathbf{f}_{\theta_j}$, $j > 0$ is used to further select samples for labelling. This procedure runs iteratively until all bags from the query pool are exhausted.

A pseudo-code of the weak AL procedure proposed in this paper is given in Algorithm 1.

At the end of the process, only the model that satisfies the desired requirements (i.e., a balance good performance and small number of data) is employed to classify the appliances, without considering the previous intermediate models' predictions. In fact, the models generated after each fine-tuning phase are utilised to select the next batch of data for the subsequent fine-tuning phase. After this, the model can be discarded as it will not be used in the subsequent iterations.

The task of AL with weak labels for multi-appliance NILM model is challenging by itself, because only weak labels are available from the target domain, and also because with this method we aim to monitor multiple appliances contemporarily with the same network. The latter can be problematic because it is hard to improve the performance for all the devices simultaneously - picking bags to improve one appliance type does not mean improving the others as well - on the contrary, it can happen that improving performance for one appliance leads to decreased performance for the others. This behaviour affects the AL process, espe-

Algorithm 1 Pseudo-code for the Weakly Supervised AL procedure.

$j \leftarrow 1$
$\mathbf{f}_{\theta_0}$: pre-trained CRNN model
$q(\cdot)$: acquisition function
$\boldsymbol{D}_U$: query pool, unlabelled
P: batch size
$\boldsymbol{Q}_{tot,j} \leftarrow \varnothing$
while $\mid \boldsymbol{D}_U \mid > 0$ **do**
  $\boldsymbol{Q}_{U,j} \leftarrow q(\mathbf{f}_{\theta_{j-1}}, P, \boldsymbol{D}_U)$
  $\boldsymbol{D}_U \leftarrow \boldsymbol{D}_U \setminus \boldsymbol{Q}_{U,j}$
  $\boldsymbol{Q}_{weak,j} \leftarrow$ weakly labelled $\boldsymbol{Q}_{U,j}$
  $\boldsymbol{Q}_{tot,j} \leftarrow \boldsymbol{Q}_{tot,j-1} \cup \boldsymbol{Q}_{weak,j}$
  $\mathbf{f}_{\theta_j} \leftarrow \mathbf{f}_{\theta_0}$ fine-tuned with $\boldsymbol{Q}_{tot,j}$
  $j \leftarrow j+1$
end while

cially if there is an appliance with significantly lower performance compared to other loads present in the house - then chosen bags are more likely to improve the most problematic appliance, and not all of them simultaneously. We describe the strategy to address these issues next.

4.3.1. Acquisition function

Acquisition function $q(\cdot)$ is used to rank bags in $\mathbf{D}_U$ with respect to their informativeness, choosing the best subset $\mathbf{Q}_U$ to include in model fine-tuning.

The acquisition function used in this paper is uncertainty-based, which demonstrated in [46] to be the best performing among several compared acquisition functions. In iteration j, $j > 0$, bags with the highest uncertainty levels, $\boldsymbol{Q}_{U,j} \subset \boldsymbol{D}_U$ are chosen to be labelled, denoted as $\boldsymbol{Q}_{weak,j}$, and included in fine-tuning dataset $\boldsymbol{Q}_{tot,j}$.

Weak level prediction of the model for a given bag is a vector containing probabilities of each appliance being in an active state inside that bag, which can be used to estimate uncertainty levels of the model. If a probability for a particular appliance is higher than decision threshold β then the model predicts that the appliance was active during the bag time period. The closer the prediction $\hat{w}_k$ of the model for an appliance k to β is, the more uncertain the model is about activation of this appliance, and the closer $\hat{w}_k$ to 1 or 0, the more certain the model is. We formally define an estimate of model uncertainty as:

$$\delta_k[i] = \begin{cases} \hat{w}_k[i] & \hat{w}_k[i] < \beta \\ 1 - \hat{w}_k[i] & \hat{w}_k[i] \geqslant \beta \end{cases} \tag{9}$$

with $\delta_k[i]$ being the estimated uncertainty of the model for bag i for single appliance k, and $\hat{w}_k[i]$ is the model output, i.e., the model's estimated probability that k-th appliance was active in the bag i.

Since the problem considered in this paper is multi-label classification, with multiple appliances considered at the same time, two ways of estimating the overall model uncertainty $\delta[i]$ for bag i are:

– by taking maximum uncertainty level across appliances present in the house:

$$\delta[i] = \max_k \delta_k[i] \tag{10}$$

– by averaging uncertainty level over all appliances present in the house:

$$\delta[i] = \frac{1}{K} \sum_{k=1}^{K} \delta_k[i]. \tag{11}$$

Then, the set of bags $\boldsymbol{Q}_{U,j}$ with the highest uncertainty $\delta[i]$ is included in the fine-tuning set. The resulting acquisition function, $q(\cdot)$, is as described in Algorithm 2.

Algorithm 2 Acquisition function

$\mathbf{f}_j$: CRNN model
$\boldsymbol{D}_U$: query pool, unlabelled
P: batch size
function $q(\mathbf{f}_j, P, \boldsymbol{D}_U)$
  for i in $\{1, ..., \mid \boldsymbol{D}_U \mid\}$ **do**
    $\hat{\mathbf{w}}[i] \leftarrow \mathbf{f}_j(\boldsymbol{D}_U[i])$
    calculate uncertainty $\delta[i]$
  end for
  $ind = argsort([\delta[1]...\delta[\mid \boldsymbol{D}_U \mid]], descend.)[:P]$
  return $\boldsymbol{D}_U[ind]$
end function

A toy example of how the acquisition function described above works, for both cases of maximising and averaging uncertainties of individual appliances is given in Table 1. Table 1 shows the selected bags (a batch of $P = 4$) in grey for maximum uncertainty across all appliances in the 4-th column and for maximum average uncertainty over all appliances in the 5-th column.

The code used to implement the approach is available on Github[1].

[1] https://github.com/GiuTan/WeaklySupervisedActiveLearning-for-NILM

Table 1

Uncertainty-based acquisition function example: Uncertainty levels for each appliance are calculated as per (9), and, maximum or mean uncertainty values are calculated based on (10) and (11), respectively. In this example, a batch of $P = 4$ most uncertain bags is chosen.

Bag index	Weak level prediction $\hat{w}_k[i]$				Uncertainty $\delta_k[i]$				Maximum	Mean
i	KE	MW	WM	DW	KE	MW	WM	DW	uncertainty	uncertainty
0	0.1	0.6	0.4	0.8	0.1	0.4	0.4	0.2	0.4	0.275
1	0.2	0.85	0.33	0.68	0.2	0.15	0.33	0.32	0.33	0.25
2	0.99	0.2	0.87	0.3	0.01	0.2	0.13	0.3	0.3	0.16
3	0.56	0.38	0.25	0.92	0.44	0.38	0.25	0.08	0.44	0.2875
4	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
5	0.67	0.43	0.01	0	0.33	0.43	0.01	0	0.43	0.1925
6	0.36	0.15	0.64	0.75	0.36	0.15	0.36	0.25	0.36	0.28
7	0.83	0.72	0.59	0.41	0.17	0.28	0.41	0.41	0.41	0.3175
8	0	0.5	0	0	0	0.5	0	0	0.5	0.125
9	0.04	0.99	0.88	0.02	0.04	0.01	0.12	0.02	0.12	0.0475

Table 2

Train and Validation sets characteristics of UK-DALE. The number of labels is reported in thousands. SL: Strong Labels. WL: Weak Labels.

	Train			Validation		
Appliance	Houses	Nr. of SL	Nr. of WL	Houses	Nr. of SL	Nr. of WL
KE	1, 3, 5	996.6	43.1	1, 3, 5	196.3	6.9
MW	1, 5	849.7	42.9	1, 5	157.2	7.0
WM	1, 5	837.7	32.1	1, 5	881.4	1.2
DW	1, 5	554.5	31.9	1, 5	790.1	0.9
Nr. of bags	99.993			10.428		

Table 3

Fine-Tuning and Test sets characteristics for REFIT. Number of labels is reported in thousands. WL: Weak Labels.

Appliance	Nr. of WL			
	House 2	House 4	House 5	House 19
KE	2.9	12	9.5	13.6
MW	-	12	-	13.6
WM	2.9	-	0.5	-
DW	2.9	-	0.5	-
Nr. of bags	2.9	12	9.5	13.6

5. Experimental Setting

5.1. Dataset

UK-DALE [49] and REFIT [50] datasets are used to evaluate the performance of the proposed method with typical appliances present in most households - Kettle (KE), Microwave (MW), Washing Machine (WM), and Dishwasher (DW). We decided not to include the fridge among the monitored appliances. This decision was made since a fridge is typically always in operation, which would mean the user would consistently assign the ON label. Although we did not monitor the fridge, it is present in the aggregate dataset.

UK-DALE contains data from 5 houses, with the aggregate power sampled at 1 Hz and appliance power sampled at 1/6 Hz, while REFIT contains measurements from 20 houses sampled at 1/8 Hz. To be aligned with UK-DALE, aggregate and appliance signals were up-sampled uniformly to 1/6 Hz. Selecting the same periods of data and following the procedure detailed in [39], both datasets have been used to create two sets of bags, one with UK-DALE data from Houses 1, 3 and 5 and one with data from four REFIT Houses 2, 4, 5 and 19. This choice has been made to include 4 houses that have different aggregate consumption characteristics, and have at least two appliances present in each house for evaluation, as shown in Table 3. Note that we balanced the occurrence of appliance activations and the number of strong labels associated with each appliance in both sets of bags. Table 2 and Table 3 report the details about training, validation, and test sets for the two sets of bags created, respectively, from UK-DALE and REFIT. The set used to validate the performance during AL process is the test set. Data was standardised subtracting the mean and dividing by the standard deviation. We estimated these values on the pre-training set.

5.2. Experiments setup

The experimental setup has been designed to evaluate several possible real-world scenarios that differ in annotation availability, based on the formulation in Section 4.2. In this way, we can evaluate the benefits from the AL procedure in more pre-training conditions. The performance has always been evaluated on 70% of the REFIT "Test and Fine-tuning" set reported in Table 3.

Referring to (5), we defined two pre-training dataset compositions:

- Scenario 1: only weakly labelled data is available: in this case, $\boldsymbol{D}^{(pt)}_{strong-weak} = \varnothing$ and $\boldsymbol{D}^{(pt)}_{weak} \neq \varnothing$ is composed of bags from the UK-DALE dataset.
- Scenario 2: both strongly and weakly labelled data from the same domain are available: in this case, $\boldsymbol{D}^{(pt)}_{strong-weak} \neq \varnothing$ and $\boldsymbol{D}^{(pt)}_{weak} \neq \varnothing$, and they are both composed of bags from the UK-DALE dataset.

Regardless of the pre-training condition, the validation set is represented by UK-DALE as reported in Table 2.

The bags that populate the query pool $\mathbf{D}_U$ for AL and that are used for the fine-tuning are up to 30% of the bags from each house of the REFIT "Test and Fine-Tuning set", reported in Table 3.

For each pre-training condition, the Hyperband algorithm [62] from Keras tuner has been used to select the hyperparameters values that achieve the highest performance on the validation set. During the AL process, we do not perform any optimisation of hyperparameters. This is because the structure of the fine-tuned network is the same as that of the pre-trained network. The pre-trained network has already been optimised during the pre-training phase, performed in our previous work [41]. Adam [63] is used as optimiser and the learning rate was fixed to 0.002 and F to 32. In our experiments we use $L = 2550$ (that is a window of 4.15 hours) samples for the bag dimension and $P = 64$ is the batch size.

When the source dataset is only weakly labelled, fine-tuning the bidirectional and instance layers has proven the best performing method on the validation set. When strongly labelled data are also available, only the instance layer has been fine-tuned.

The threshold for the quantisation of instance level predictions has been determined by optimal thresholding strategy on the test set for each pre-training condition.

5.3. Benchmark method

In [41] a weakly supervised transfer learning approach has been proposed. Both the pre-training and the fine-tuning exploits only weak labels, or both weak and strong labels. In the fine-tuning phase, a set of weakly annotated signals has been supplied to the network to adapt the pre-trained model on the target environment domain. The best models obtained from the proposed method have been compared to "No Fine-Tuning" model [41], thus prior to fine-tuning, and "Weak Transfer Learning" model [41] obtained using the complete set of query pool data weakly annotated.

Additionally, we benchmark our method against a semi-supervised method based on knowledge distillation, proposed in [48], that is pre-trained using only strong labels, but in the fine-tuning phase only unlabelled data is fed to the model, as we consider that labels from the target environment are not readily available. Because of absence of labels from the target environment, and the way that the model works, bags with the lowest uncertainty were chosen instead of the highest during the AL process for this benchmark.

5.4. Evaluation metrics

Defining True Positive ($\mathrm{TP}^{(k)}$) as the number of correctly classified active samples for appliance k, False Positive ($\mathrm{FP}^{(k)}$) as the number of inactive samples incorrectly classified as active and False Negative ($\mathrm{FN}^{(k)}$) as the number of active samples incorrectly classified as inactive, we used the $\mathrm{F}_1^{(k)}$-score, commonly used in NILM classification literature, expressed as

$$\mathrm{F}_1^{(k)} = \frac{\mathrm{TP}^{(k)}}{(\mathrm{TP}^{(k)} + 1/2\ (\mathrm{FN}^{(k)} + \mathrm{FP}^{(k)}))}$$

for k-th appliance. We report also the micro average F_1-micro, that considers the quantity of samples for each appliance in the test set and it is expressed by:

$$\mathrm{F}_1\text{-micro} = \frac{\sum_{k=1}^{K} \mathrm{TP}^{(k)}}{\sum_{k=1}^{K} (\mathrm{TP}^{(k)} + 1/2\ (\mathrm{FN}^{(k)} + \mathrm{FP}^{(k)}))}$$

Optimal point of AL iteration process is determined as a point at iteration j with F_1-score $F_{1,j}$ that has the minimum distance d_j from an "ideal" point - no data labelled, and perfect performance of F_1 = 1, as in [46]. The distance is calculated according to Equation

(12), where $|\boldsymbol{Q}_{tot,j}|$ denotes the total number of bags queried up to iteration j, and $|\boldsymbol{Q}_{tot,J}|$ denotes the maximum number of bags that can be queried.

$$d_j = \sqrt{\left(\frac{|\,\boldsymbol{Q}_{tot,j}\,|}{|\,\boldsymbol{Q}_{tot,J}\,|}\right)^2 + (1 - F_{1,j})^2}. \quad (12)$$

6. Experimental results

This section presents the results obtained from the two experimental scenarios, as well as from the semi-supervised benchmark method. F_1-scores are shown per appliance for each house. Models pre-trained on UK-DALE were transferred to REFIT houses 2, 4, 5 and 19 - Dataset column indicates the fine-tuning test set. The optimal points and maximum performances obtained during the AL process are given together with the percentage of query pool data labelled and added to the fine-tuning dataset to achieve that performance. Note that not all houses contain all the appliances - results are shown only for monitored appliances installed in the selected buildings.

6.1. Semi-supervised benchmark results

Experimental results for the semi-supervised benchmark approach [48] are presented in Table 4. In this case, strongly labelled data were used during the pre-training phase, and unlabelled data were utilised throughout the AL process. This scenario is challenging because with the semi-supervised strategy the model is fine-tuned with unseen data from the target environment without any labels provided. According to Table 4, the performance in House 2 does not improve after fine-tuning with all available data (100% of unlabelled bags used). There is a very limited improvement with AL for kettle only, but the performance level of the fine-tuning case with 100% of unlabelled bags used can be achieved using a smaller amount of data (6.7% - 13.3%). In House 4, performance improves when all available bags from target environment are used, and the amount of data can be reduced to at least 38% of all data. In house 5, the situation is similar as in house 2 - no improvement after fine-tuning with all available unlabelled bags, and only small improvement for kettle with large portion of unlabelled bags used with AL. There is a similar situation in house 19 - no improvement after fine-tuning with all available unlabelled data, but small improvement for microwave with AL. The results from this benchmarking scenario suggest that while some improvement can be achieved using only unlabelled data to fine-tune the model to the new environment, it is not sufficient, and adding some labelled data is desirable. Therefore, results for weakly supervised AL scenarios are presented next.

6.2. Weakly Supervised AL Performance

Experimental results for the scenario where only weakly labelled data is available in the pre-training phase - pre-training scenario 1, and weak labels are used throughout the AL process, are presented in Table 5. This scenario is very challenging, because the model never sees strong labels, neither during pre-training nor during fine-tuning phase.

In House 2, with weak transfer learning (100% bags labelled), performance increases compared to the one before fine-tuning (0% bags labelled) for dishwasher, but drops for kettle and washing machine due to overfitting. However, for kettle, with AL when maximising uncertainty over appliances, performance increase is achieved at optimal AL point with 13.3% bags labelled, and when averaging uncertainty over appliances, performance increases with labelling 20% of bags, reducing labelling effort by 86.7% and 80% respectively. For washing machine, labelling 6.7% of bags retains performance whether uncertainty is maximised or averaged over appliances. For dishwasher, performance is increased at optimal point with only 13.3% of bags labelled with maximising, and with 6.7% when averaging uncertainty over appliances. Micro F_1-score is retained in all AL cases.

This situation is a consequence of different appliance signature characteristics - a kettle activation, as a short duration appliance, is more likely to be present in bags with other activations from other devices, and hence needs more queries to augment its learning to see sufficient kettle activations with different aggregates. Washing machine is likely to be confused with dishwasher and, hence, in the absence of strong labels its performance cannot be improved, especially for the low-power state. For dishwasher, there are more high power samples in one activation and, therefore, with more training samples in the weak labels, it is possible to improve.

In House 4, weak transfer learning (100% bags labelled) increases performance for both kettle and microwave, as well as the micro F_1-score. With weak AL, for kettle, at optimal point, performance increase is achieved with 1.7% and 8.8% bags labelled when

maximising and averaging uncertainty over appliances, respectively, reducing labelling effort by 98.3% and 92.2%. For microwave, at optimal point performance is increased with 1.7% and 10.5% bags labelled when maximising and averaging uncertainty over appliances, respectively. Micro F_1-score increased at optimal points with only 1.7% and 10.5% bags labelled when maximising and averaging uncertainty over appliances, respectively.

Considering best F_1-score, kettle needs 52% additional samples for fine-tuning when considering mean uncertainty across appliances but only 1.7% more when considering maximum uncertainty. This is due to the fact that House 4 is much noisier in terms of unknown appliances present in the aggregate signal - it has noise to aggregate ratio (NAR [64]) of 0.91, with noise calculated as in [41], compared to the NAR value of house 2 which is 0.79. Microwave needs more additional samples due to its short activation time and high probability of activation in presence of other appliances, hence, the model requires more weakly labelled bags to improve.

In house 5, performance is poor before fine-tuning for washing machine and dishwasher. However, overall performance, as well as per-appliance performance, does improve (or remains the same for the dishwasher) with weak transfer learning (100% bags labelled), and also with weak AL with reduced amount of labelled data. With weak AL, the amount of data that needs labelling increases from 2.2 to 10.7 % when maximising uncertainty across appliances, and from 2.2 to 26.1 % when averaging, at optimal points. At best F_1-score, washing machine and dishwasher need significantly larger portion of labelled data, due to poor performance in the beginning. Consequently, micro F_1-score also peaks at higher percentage of data labelled. House 5 is noisier than House 2 as indicated by a NAR value of 0.84, but lower than House 4, hence it exhibits a good performance for kettle, but washing machine and dishwasher have more complex patterns which are different from device to device, so it is hard to improve them significantly with weak labels only for this house.

In House 19, performance improves with AL exceeding the performance of weak fine-tuning (100 % bags labelled), requiring only 1.5 - 3.2 % of bags to be weakly labelled when maximising, and 2.7 to 8.1 % when averaging uncertainty across appliances, at optimal points. NAR value of House 19 is the highest among all test houses - 0.93, but starting performance before any fine-tuning is good, which indicates that this domain has more similarities with training data than previous testing domains.

Table 6 shows results where strong and weak labels are used in the pre-training phase - pre-training scenario 2, and weak labels are used in the AL phase. This scenario is more favourable compared to the previous one, because even though only weak labels are available during fine-tuning phase, strong labels are available in the pre-training phase.

Compared to Scenario 1, as expected, performance for all appliances in all houses is improved over the baseline [41] with significantly less additional fine-tuning data. This behaviour can be attributed to the inclusion of strong labels during the pre-training phase, which increased the network's knowledge, thereby necessitating a lesser quantity of data to achieve comparable or improved results.

Next we discuss levels of uncertainty observed at the start of the AL process. In Scenario 1, weak labels only are present in the pre-training phase, and the model tends to be either overconfident or very unconfident (as shown by the uncertainty histogram in Fig. 4 (top) - most of bags have low uncertainty values - and lower uncertainty means higher confidence), and the performance before fine-tuning is not as good as with strong labels present (Scenario 2). On the other hand, when strong labels are present in the pre-training phase (Scenario 2), performance before fine-tuning is better, but there are not as many low uncertainty (high confidence) bags as in Scenario 1 (as shown in Figure 4). The model has been shown strong labels, hence better performance, but is also more uncertain (i.e., histogram is more flat) due to learning from strong labels with overlapping activations of multiple appliances contained in a bag. It is also worth noting that more high uncertainty bags are observed for kettle than for microwave. Uncertainty levels among bags that are queried for REFIT house 4 in each experimental scenario are shown in Figure 5: Scenario 1 with mean uncertainty across appliances – upper left; Scenario 1 with maximum uncertainty across appliances – upper right; Scenario 2 with mean uncertainty across appliances - lower left; and Scenario 2 with maximum uncertainty across appliances – lower right. The figures show uncertainty level of microwave (orange) stacked to uncertainty level of kettle (blue) for each bag queried in the beginning of the AL process, before any fine tuning. In case of using maximum uncertainty across appliances as overall uncertainty measure, the model tends to pick bags in which uncertainty is high for kettle, but not necessarily for microwave –

Table 4

Benchmark - semi supervised method [48]. Model is pre-trained using strong labels, but fine-tuned using only unlabelled data from target environment. Results of the proposed approach are shown in the following format: metric (% of activation samples added to fine-tuning dataset).

	Method	Labels	Dataset	KE	MW	WM	DW	F_1-micro
H2	No Fine-Tuning [48]	Strong	UK-DALE	0.55	-	0.41	0.58	0.50
	Unsupervised Transfer Learning [48]	-	REFIT	0.55	-	0.41	0.58	0.50
	AL (max uncertainty) - optimal point	-	REFIT	0.55 (13.3%)	-	0.41 (6.7%)	0.58 (6.7%)	0.50 (6.7%)
	AL (max uncertainty) - best F1	-	REFIT	0.56 (80%)	-	0.41 (6.7%)	0.58 (6.7%)	0.50 (6.7%)
	AL (mean uncertainty) - optimal point	-	REFIT	0.54 (13.3%)	-	0.41 (6.7%)	0.58 (6.7%)	0.50 (6.7%)
	AL (mean uncertainty) - best F1	-	REFIT	0.56 (73.3%)	-	0.41 (6.7%)	0.58(6.7%)	0.50 (6.7%)
H4	No Fine-Tuning [48]	Strong	UK-DALE	0.42	0.38	-	-	0.39
	Unsupervised Transfer Learning [48]	-	REFIT	0.44	0.44	-	-	0.44
	AL (max uncertainty) - optimal point	-	REFIT	0.44 (13.8%)	0.41 (10.3%)	-	-	0.42 (13.8%)
	AL (max uncertainty) - best F1	-	REFIT	0.45 (20.7%)	0.44 (38%)	-	-	0.44 (38%)
	AL (mean uncertainty) - optimal point	-	REFIT	0.45 (1.7%)	0.41 (12.1%)	-	-	0.41 (12.1%)
	AL (mean uncertainty) - best F1	-	REFIT	0.45 (1.7%)	0.44 (98.2%)	-	-	0.44 (98.2%)
H5	No Fine-Tuning [48]	Strong	UK-DALE	0.86	-	0.02	0.04	0.05
	Unsupervised Transfer Learning [48]	-	REFIT	0.86	-	0.02	0.04	0.05
	AL (max uncertainty) - optimal point	-	REFIT	0.86 (4.3 %)	-	0.02 (2.2 %)	0.04 (2.2 %)	0.05 (2.2 %)
	AL (max uncertainty) - best F1	-	REFIT	0.87 (60.9 %)	-	0.02 (2.2 %)	0.04 (2.2 %)	0.05 (2.2 %)
	AL (mean uncertainty) - optimal point	-	REFIT	0.86 (4.3 %)	-	0.02 (2.2 %)	0.04 (2.2 %)	0.05 (2.2 %)
	AL (mean uncertainty) - best F1	-	REFIT	0.87 (97.8 %)	-	0.02 (2.2 %)	0.04 (2.2 %)	0.05 (2.2 %)
H19	No Fine-Tuning [48]	Strong	UK-DALE	0.82	0.61	-	-	0.69
	Unsupervised Transfer Learning [48]	-	REFIT	0.82	0.61	-	-	0.69
	AL (max uncertainty) - optimal point	-	REFIT	0.82 (3.1 %)	0.63 (1.5 %)	-	-	0.70 (1.5 %)
	AL (max uncertainty) - best F1	-	REFIT	0.82 (3.1 %)	0.64 (89.2 %)	-	-	0.70 (1.5 %)
	AL (mean uncertainty) - optimal point	-	REFIT	0.82 (3.1 %)	0.62 (1.5 %)	-	-	0.69 (1.5 %)
	AL (mean uncertainty) - best F1	-	REFIT	0.83 (43.1 %)	0.63 (60 %)	-	-	0.70 (60 %)

according to histograms in Figure 4, kettle has more high uncertainty bags in general. On the other hand, if using mean uncertainty across appliances as overall uncertainty measure, bags are picked so that both appliances have high uncertainty. Therefore, as described in Section 4.3, querying based on mean uncertainty is more reliable and gives better overall improvement of the model.

From both Tables 5 and 6, we observe that with our proposed optimal point (Eq 12), performance improvement (House 2: 1.2%, House 4: 14%, House 5: 2.9%, House 19: 14%), for both acquisition functions, is almost the same as best F1 performance, with significantly less additional fine-tuning data.

AL curve with optimal points marked obtained in house 4 with mean uncertainty over appliances is shown in Fig. 6. In the beginning of the AL process, useful bags are chosen in the first couple of iterations, after which performance becomes steady for kettle, and improves further for microwave.

From the presented results, it is evident that sometimes adding less data is better than adding more, because not all data samples are useful, and not all data samples do improve the pre-trained model. Therefore, AL approaches can be used to select only high-uncertainty data and label and add only them to fine-tuning dataset. An important note is that *weak labels only* can be used throughout the AL process, and model performance can still improve. This is very encouraging, especially bearing in mind that weak labels

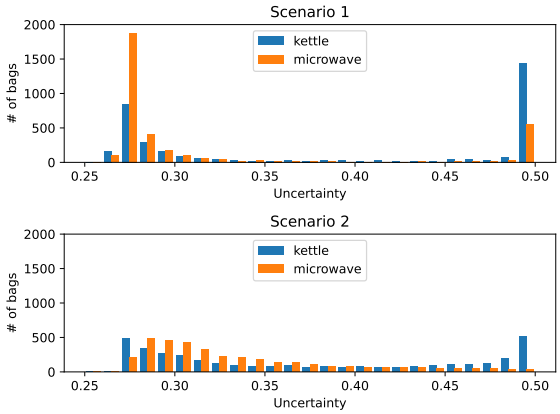

Fig. 4. Observed uncertainty levels in Scenario 1 (top) and Scenario 2 (bottom) for the whole query pool of house 4 bags.

Table 5

Results - pre-training Scenario 1. Results of the proposed approach are shown in the following format: metric (% of activation samples added to fine-tuning dataset).

	Method	Labels	Dataset	KE	MW	WM	DW	F_1-micro
H2	No Fine-Tuning [41]	Weak	UK-DALE	0.73	-	0.62	0.70	0.67
	Weak Transfer Learning [41]	Weak	REFIT	0.59	-	0.42	0.73	0.58
	Proposed (max uncertainty) - optimal point	Weak	REFIT	0.74 (13.3%)	-	0.62 (6.7%)	0.71 (13.3%)	0.67 (6.7%)
	Proposed (max uncertainty) - best F_1	Weak	REFIT	0.79 (73.3%)	-	0.62 (6.7%)	0.74 (33.3%)	0.67 (6.7%)
	Proposed (mean uncertainty) - optimal point	Weak	REFIT	0.80 (20%)	-	0.62 (6.7%)	0.71 (6.7%)	0.67 (6.7%)
	Proposed (mean uncertainty) - best F_1	Weak	REFIT	0.80 (20%)	-	0.62 (6.7%)	0.73 (20%)	0.67 (6.7%)
H4	No Fine-Tuning [41]	Weak	UK-DALE	0.54	0.53	-	-	0.53
	Weak Transfer Learning [41]	Weak	REFIT	0.59	0.65	-	-	0.63
	Proposed (max uncertainty) - optimal point	Weak	REFIT	0.61(1.7%)	0.64 (1.7%)	-	-	0.63 (1.7%)
	Proposed (max uncertainty) - best F_1	Weak	REFIT	0.61 (1.7%)	0.72 (67.2%)	-	-	0.65 (67.2%)
	Proposed (mean uncertainty) - optimal point	Weak	REFIT	0.58 (8.8%)	0.63 (10.5%)	-	-	0.61 (10.5%)
	Proposed (mean uncertainty) - best F_1	Weak	REFIT	0.60 (52.6%)	0.70 (66.7%)	-	-	0.65 (66.7%)
H5	No Fine-Tuning [41]	Weak	UK-DALE	0.78	-	0.24	0.28	0.51
	Weak Transfer Learning [41]	Weak	REFIT	0.79	-	0.32	0.28	0.55
	Proposed (max uncertainty) - optimal point	Weak	REFIT	0.80 (2.2%)	-	0.30 (6.5 %)	0.27 (10.7 %)	0.56 (10.7 %)
	Proposed (max uncertainty) - best F_1	Weak	REFIT	0.80 (2.2 %)	-	0.36 (95.6 %)	0.28 (50 %)	0.57 (54.3 %)
	Proposed (mean uncertainty) - optimal point	Weak	REFIT	0.80 (2.2 %)	-	0.34 (26.1 %)	0.28 (4.3 %)	0.56 (6.5 %)
	Proposed (mean uncertainty) - best F_1	Weak	REFIT	0.80 (2.2 %)	-	0.34 (26.1 %)	0.29 (52.2 %)	0.56 (6.5 %)
H19	No Fine-Tuning [41]	Weak	UK-DALE	0.66	0.68	-	-	0.67
	Weak Transfer Learning [41]	Weak	REFIT	0.75	0.69	-	-	0.71
	Proposed (max uncertainty) - optimal point	Weak	REFIT	0.80 (3.1 %)	0.70 (1.5 %)	-	-	0.73 (1.5 %)
	Proposed (max uncertainty) - best F_1	Weak	REFIT	0.81 (64.6 %)	0.71 (29.2 %)	-	-	0.73 (1.5 %)
	Proposed (mean uncertainty) - optimal point	Weak	REFIT	0.78 (2.7 %)	0.70 (8.1 %)	-	-	0.73 (2.7 %)
	Proposed (mean uncertainty) - best F_1	Weak	REFIT	0.79 (13.5 %)	0.71 (27 %)	-	-	0.74 (13.5 %)

Table 6

Results - pre-training Scenario 2. Results of the proposed approach are shown in the following format: metric (% of activation samples added to fine-tuning dataset).

	Method	Labels	Dataset	KE	MW	WM	DW	F_1-micro
H2	No Fine-Tuning [41]	Strong & Weak	UK-DALE	0.78	-	0.78	0.84	0.82
	Weak Transfer Learning [41]	Weak	REFIT	0.83	-	0.82	0.83	0.82
	Proposed (max uncertainty) - optimal point	Weak	REFIT	0.82 (6.7%)	-	0.80 (6.7%)	0.83 (6.7%)	0.82 (6.7%)
	Proposed (max uncertainty) - best F_1	Weak	REFIT	0.83 (13.33%)	-	0.82 (46.7%)	0.84 (93.3%)	0.82 (6.7%)
	Proposed (mean uncertainty) - optimal point	Weak	REFIT	0.83 (6.7%)	-	0.80 (6.7%)	0.83 (6.7%)	0.82 (6.7%)
	Proposed (mean uncertainty) - best F_1	Weak	REFIT	0.84 (86.7%)	-	0.82 (26.7%)	0.84 (33.3%)	0.83 (66.7%)
H4	No Fine-Tuning [41]	Strong & Weak	UK-DALE	0.71	0.69	-	-	0.69
	Weak Transfer Learning [41]	Weak	REFIT	0.73	0.73	-	-	0.73
	Proposed (max uncertainty) - optimal point	Weak	REFIT	0.76 (6.9%)	0.84 (5.2%)	-	-	0.81 (5.2%)
	Proposed (max uncertainty) - best F_1	Weak	REFIT	0.77 (14%)	0.86 (73.7%)	-	-	0.81 (5.2%)
	Proposed (mean uncertainty) - optimal point	Weak	REFIT	0.78 (1.7%)	0.85 (1.7%)	-	-	0.83 (1.7%)
	Proposed (mean uncertainty) - best F_1	Weak	REFIT	0.78 (1.7%)	0.86 (28.1%)	-	-	0.83 (1.7%)
H5	No Fine-Tuning [41]	Strong & Weak	UK-DALE	0.94	-	0.20	0.43	0.60
	Weak Transfer Learning [41]	Weak	REFIT	0.95	-	0.41	0.55	0.70
	Proposed (max uncertainty) - optimal point	Weak	REFIT	0.96 (4.3%)	-	0.41 (26.1%)	0.54 (17.4%)	0.69 (17.4%)
	Proposed (max uncertainty) - best F_1	Weak	REFIT	0.96 (4.3 %)	-	0.42 (76.1%)	0.57 (60.9%)	0.72 (65.2%)
	Proposed (mean uncertainty) - optimal point	Weak	REFIT	0.96 (2.2 %)	-	0.36 (28.3 %)	0.51 (2.2 %)	0.67 (2.2%)
	Proposed (mean uncertainty) - best F_1	Weak	REFIT	0.96 (2.1 %)	-	0.40 (39.1 %)	0.58 (28.3 %)	0.71 (28.3 %)
H19	No Fine-Tuning [41]	Strong & Weak	UK-DALE	0.88	0.75	-	-	0.80
	Weak Transfer Learning [41]	Weak	REFIT	0.76	0.69	-	-	0.71
	Proposed (max uncertainty) - optimal point	Weak	REFIT	0.91 (7.7 %)	0.73 (1.5 %)	-	-	0.78 (1.5 %)
	Proposed (max uncertainty) - best F_1	Weak	REFIT	0.94 (72.3 %)	0.73 (1.5 %)	-	-	0.78 (1.5 %)
	Proposed (mean uncertainty) - optimal point	Weak	REFIT	0.89 (4.6 %)	0.76 (7.7 %)	-	-	0.80 (1.5 %)
	Proposed (mean uncertainty) - best F_1	Weak	REFIT	0.89 (4.6 %)	0.76 (7.7 %)	-	-	0.81 (7.7 %)

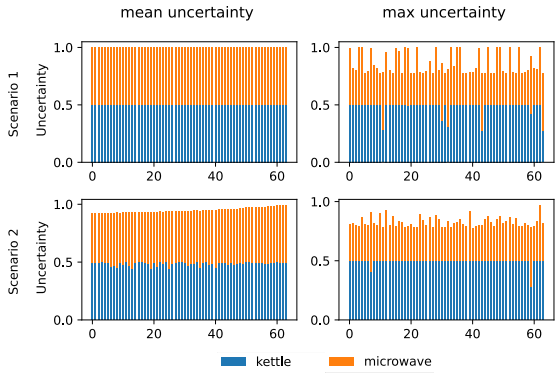

Fig. 5. Observed ratio of uncertainty between kettle and microwave in Scenarios 1 (top) and 2 (bottom), when using mean (left) and maximum (right) uncertainty across present appliances.

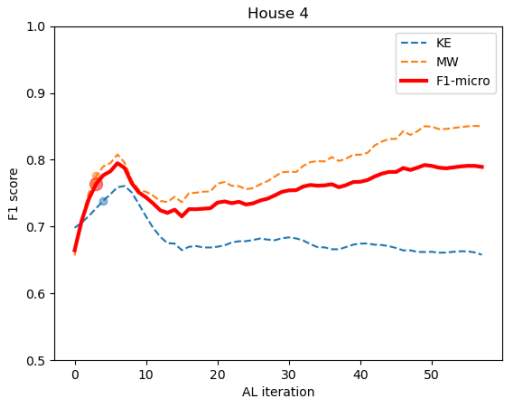

Fig. 6. AL curve obtained at REFIT house 4 in Scenario 2 when averaging uncertainty across present appliances. Original curve is smoothed using Savitsky-Golay filter of length 11 and order 3.

are easily obtained, and that they could be obtained even from lay users, who do not have any knowledge of NILM and appliance signatures - weak labels could be inferred by only asking users when did they run specific device.

6.3. Complexity

In this section, we provide a brief discussion on the complexity of the proposed approach. It is worth noting that this framework is primarily designed for data efficiency without compromising performance, but the method itself does not focus on reducing computational complexity.

In each AL iteration, there are two phases that require significant computational resources: acquisition and fine-tuning phase. In the acquisition phase, the model needs to examine all signal segments belonging to the query pool and rank them by uncertainty, which has the complexity of $O(n^2)$. The cost of this step reduces as the AL process progresses because the size of the query pool decreases. The fine-tuning phase then uses acquired signal segments to fine-tune the model. The cost of this increases as the AL process progresses because the fine-tuning set size increases as newly queried signal segments are added. The CRNN model used in this paper consumes 976.28 kB of memory and has 1,100,847,745 FLOPs.

7. Conclusions

Non-Intrusive Load Monitoring approaches need to be adapted to the new data domain, when deployed in a target unseen environment, to ensure acceptable performance. To this aim, data and labels collection phase is required. Usually this task is performed by the end users or service providers, where the labelling process is time-consuming. The works in literature that proposed approaches to help in reducing the user effort to provide labels, still face issues related to the feasibility of obtaining sample-by-sample annotations or to the large quantity of data to be annotated to obtain acceptable performance.

We proposed a weakly supervised AL framework in order to address the above gaps, exploiting weak labels and the AL loop to collect annotations for a reduced set of data. We also propose an approach whereby it is possible to determine the minimum number of samples needed to achieve optimal performance and prove experimentally that under multiple scenarios and appliances, across 4 test houses, including additional samples does not significantly improve performance. We also demonstrated that our approach exceeds the performance of a benchmark method while reducing the labelling effort by up to 82.6-98.5% in four target domains.

Future works will extend the method by considering criteria based on explainability [65, 66] to select the subset of data to be labelled by the users. Moreover, advanced neural network techniques [67–69] will be included to improve the effectiveness and efficiency of the method.

8. Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 955422.

References

[1] M. Sun, F.M. Nakoty, Q. Liu, X. Liu, Y. Yang and T. Shen, Non-Intrusive Load Monitoring System Framework and Load Disaggregation Algorithms: A Survey, in: *Proc. of ICAMechS*, 2019, pp. 284–288. doi:10.1109/ICAMechS.2019.8861646.

[2] H. Chen, P. Gangopadhyay, B. Singh and S. Shankar, Measuring preferences for energy efficiency in ACI and EU nations and uncovering their impacts on energy conservation, *Renewable and Sustainable Energy Reviews* **156** (2022), 111944. doi:https://doi.org/10.1016/j.rser.2021.111944.

[3] P. García Gómez, I. González-Rodríguez and C.R. Vela, Enhanced Memetic Search for Reducing Energy Consumption in Fuzzy Flexible Job Shops, *Integrated Computer-Aided Engineering* (2023), 151–167. doi:10.3233/ICA-230699.

[4] M.G. Hassan, R. Hirst, C. Siemieniuch and A.F. Zobaa, The impact of energy awareness on energy efficiency, *International Journal of Sustainable Engineering* **2**(4) (2009), 284–297. doi:10.1080/19397030903121968.

[5] I. Vassileva and J. Campillo, Increasing energy efficiency in low-income households through targeting awareness and behavioral change, *Renewable Energy* **67** (2014), 59–63, Renewable Energy for Sustainable Development and Decarbonisation. doi:https://doi.org/10.1016/j.renene.2013.11.046.

[6] N.A. Mamoun, M.I. Zuriekat, H.I.A. Jabali and N.A. Asfour, Determinants of purchasing intentions of energy-efficient products: The roles of energy awareness and perceived benefits, *International Journal of Energy Sector Management* **13** (2019), 128–148. doi:https://doi.org/10.1108/IJESM-05-2018-0009.

[7] M. Sanduleac, D. Stanescu, C. Stanescu and M. Florea, Energy awareness, an important goal for empowering the end customer, in: *Proc. of OPTIM and Proc. of ACEMP*, 2017, pp. 599–604. doi:10.1109/OPTIM.2017.7975034.

[8] M. Benachir and C. Moulay Larbi, Impact of household transitions on domestic energy consumption and its applicability to urban energy planning, *Frontiers of Engineering Management* (2017).

[9] L. Marchi and J. Gaspari, Energy Conservation at Home: A Critical Review on the Role of End-User Behavior, *Energies* **16**(22) (2023). doi:10.3390/en16227596.

[10] B. Zhao, K. He, L. Stankovic and V. Stankovic, Improving Event-Based Non-Intrusive Load Monitoring Using Graph Signal Processing, *IEEE Access* **6** (2018), 53944–53959. doi:10.1109/ACCESS.2018.2871343.

[11] J. Kelly and W. Knottenbelt, Neural NILM: Deep Neural Networks Applied to Energy Disaggregation, in: *Proc. 2nd ACM Int. Conf. on Embedded Syst. Energy-Efficient Built Environ.*, New York, USA, 2015, pp. 55–64. ISBN 9781450339810.

[12] C. Zhang, M. Zhong, Z. Wang, N. Goddard and C. Sutton, Sequence-to-Point Learning with Neural Networks for Non-Intrusive Load Monitoring, AAAI Press, 2018. ISBN 978-1-57735-800-8.

[13] S.M. Tabatabaei, S. Dick and W. Xu, Toward Non-Intrusive Load Monitoring via Multi-Label Classification, *IEEE Trans. Smart Grid* **8**(1) (2017), 26–40.

[14] H.S. Nogay and H. Adeli, Detection of Epileptic Seizure Using Pretrained Deep Convolutional Neural Network and Transfer Learning, *European neurology* (2020), 602–614. doi:10.1159/000512985.

[15] H. Selcuk Nogay and H. Adeli, Diagnostic of autism spectrum disorder based on structural brain MRI images using, grid search optimization, and convolutional neural networks, *Biomedical Signal Processing and Control* **79** (2023), 104234. doi:https://doi.org/10.1016/j.bspc.2022.104234.

[16] U. Jesús, D. Martín and A.J. María, An Improved Deep Learning Architecture for Multi-object Tracking Systems, *Integrated Computer-Aided Engineering* (2023).

[17] B.K. Oh, S.H. Yoo and H.S. Park, A measured data correlation-based strain estimation technique for building structures using convolutional neural network, *Integrated Computer-Aided Engineering* (2023).

[18] Y. Xue, H. Zhu and F. Neri, A Self-Adaptive Multi-Objective Feature Selection Approach for Classification Problems, *Integr. Comput.-Aided Eng.* **29**(1) (2022), 3–21–. doi:10.3233/ICA-210664.

[19] Y. Xue, Z. Yixia and F. Neri, A Method based on Evolutionary Algorithms and Channel Attention Mechanism to Enhance Cycle Generative Adversarial Network Performance for Image Translation, *Integrated Computer-Aided Engineering* (2023).

[20] S. Verma, S. Singh and A. Majumdar, Multi Label Restricted Boltzmann Machine for Non-intrusive Load Monitoring, in: *Proc. of ICASSP*, 2019, pp. 8345–8349.

[21] V. Singhal, J. Maggu and A. Majumdar, Simultaneous Detection of Multiple Appliances From Smart-Meter Measurements via Multi-Label Consistent Deep Dictionary Learning and Deep Transform Learning, *IEEE Transactions on Smart Grid* **10**(3) (2019), 2969–2978. doi:10.1109/TSG.2018.2815763.

[22] S. Singh and A. Majumdar, Non-Intrusive Load Monitoring via Multi-Label Sparse Representation-Based Classification, *IEEE Transactions on Smart Grid* **11**(2) (2020), 1799–1801. doi:10.1109/TSG.2019.2938090.

[23] L. Massidda, M. Marrocu and S. Manca, Non-intrusive load disaggregation by convolutional neural network and multilabel classification, *Applied Sciences* **10** (2020), 1454.

[24] H. Çimen, E.J. Palacios-Garcia, N. Çetinkaya, J.C. Vasquez and J.M. Guerrero, A Dual-input Multi-label Classification Approach for Non-Intrusive Load Monitoring via Deep Learning, in: *Proc. of ZINC*, 2020, pp. 259–263.

[25] Z.H. Zhou, A brief introduction to weakly supervised learning, *National Science Review* **5**(1) (2018), 44–53.

[26] S. Verma, S. Singh and A. Majumdar, Multi-label LSTM autoencoder for non-intrusive appliance load monitoring, *Electr. Power Syst. Res.* **199** (2021), 107414.

[27] L.d.S. Nolasco, A.E. Lazzaretti and B.M. Mulinari, DeepDFML-NILM: A New CNN-Based Architecture for Detection, Feature Extraction and Multi-Label Classification in NILM Signals, *IEEE Sensors J.* **22**(1) (2022), 501–509.

[28] M. D'Incecco, S. Squartini and M. Zhong, Transfer Learning for Non-Intrusive Load Monitoring, *IEEE Trans. Smart Grid* **11**(2) (2020), 1419–1429.

[29] S. Panigrahi, A. Nanda and T. Swarnkar, A Survey on Transfer Learning, *Smart Innov. Syst. Technol.* **194** (2021), 781–789. ISBN 9789811559709.

[30] S. Sahrane, M. Adnane and M. Haddadi, Multi-label load disaggregation in presence of non-targeted loads, *Electric Power Systems Research* **199**(June) (2021), 107435. doi:10.1016/j.epsr.2021.107435.

[31] 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* **4**(1) (2021). doi:10.1186/s42162-021-00137-9.

[32] L. Li, F. He, R. Fan, B. Fan and X. Yan, 3D reconstruction based on hierarchical reinforcement learning with transferability, *Integrated Computer-Aided Engineering* **30** (2023), 1–13. doi:10.3233/ICA-230710.

[33] C. Ieracitano, N. Mammone, A. Paviglianiti and F.C. Morabito, A Conditional Generative Adversarial Network and Transfer Learning-Oriented Anomaly Classification System for Electrospun Nanofibers, *International journal of neural systems* (2022), 2250054.

[34] L. Wang, S. Mao, B.M. Wilamowski and R.M. Nelms, Pre-trained models for non-intrusive appliance load monitoring, *IEEE Transactions on Green Communications and Networking* **6**(1) (2021), 56–68.

[35] J. Lin, J. Ma, J. Zhu and H. Liang, Deep Domain Adaptation for Non-Intrusive Load Monitoring Based on a Knowledge Transfer Learning Network, *IEEE Trans. Smart Grid* **13**(1) (2022), 280–292.

[36] C.-H. Hur, H.-E. Lee, Y.-J. Kim and S.-G. Kang, Semi-Supervised Domain Adaptation for Multi-Label Classification on Nonintrusive Load Monitoring, *Sensors* **22**(15) (2022), 5838. doi:10.3390/s22155838.

[37] N. Miao, S. Zhao, Q. Shi and R. Zhang, Non-Intrusive Load Disaggregation Using Semi-Supervised Learning Method, in: *Proc. of SPAC*, 2019, pp. 17–22. doi:10.1109/SPAC49953.2019.237865.

[38] Y. Yang, J. Zhong, W. Li, T.A. Gulliver and S. Li, Semisupervised Multilabel Deep Learning Based Nonintrusive Load Monitoring in Smart Grids, *IEEE Trans. Ind. Inf.* **16**(11) (2020), 6892–6902.

[39] G. Tanoni, E. Principi and S. Squartini, Multilabel Appliance Classification With Weakly Labeled Data for Non-Intrusive Load Monitoring, *IEEE Transactions on Smart Grid* **14**(1) (2023), 440–452. doi:10.1109/TSG.2022.3191908.

[40] L. Serafini, G. Tanoni, E. Principi, S. Spinsante and S. Squartini, A Multiple Instance Regression Approach to Electrical Load Disaggregation, in: *Proc. of EUSIPCO*, 2022, pp. 1666–1670. doi:10.23919/EUSIPCO55093.2022.9909747.

[41] G. Tanoni, E. Principi, L. Mandolini and S. Squartini, Weakly Supervised Transfer Learning for Multi-label Appliance Classification, in: *Applied Intelligence and Informatics*, Springer Nature Switzerland, Cham, 2022, pp. 360–375. ISBN 978-3-031-24801-6.

[42] M. Saneii, A. Kazemeini, S.E. Seilabi, M. Miralinaghi and S. Labi, A methodology for scheduling within-day roadway work zones using deep neural networks and active learning, *Computer-Aided Civil and Infrastructure Engineering* **38** (2022), 1101–1126.

[43] Y. Yuan, F.T.K. Au, D. Yang and J. Zhang, Active learning structural model updating of a multisensory system based on Kriging method and Bayesian inference, *Computer-Aided Civil and Infrastructure Engineering* **38**(3) (2023), 353–371. doi:https://doi.org/10.1111/mice.12822.

[44] B. Settles, Active learning literature survey (2009). https://api.semanticscholar.org/CorpusID:324600.

[45] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B.B. Gupta, X. Chen and X. Wang, A survey of deep active learning, *ACM computing surveys (CSUR)* **54**(9) (2021), 1–40.

[46] T. Todic, V. Stankovic and L. Stankovic, An active learning framework for the low-frequency Non-Intrusive Load Monitoring problem, *Applied Energy* **341** (2023), 121078. doi:https://doi.org/10.1016/j.apenergy.2023.121078.

[47] T.G. Dietterich, R.H. Lathrop and T. Lozano-Pérez, Solving the multiple instance problem with axis-parallel rectangles, *Artificial Intelligence* **89**(1) (1997), 31–71.

[48] Y. Yang, J. Zhong, W. Li, T.A. Gulliver and S. Li, Semisupervised multilabel deep learning based nonintrusive load monitoring in smart grids, *IEEE Transactions on Industrial Informatics* **16**(11) (2019), 6892–6902.

[49] J. Kelly and W. Knottenbelt, The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes, *Scientific Data* **2**(150007) (2015).

[50] D. Murray, L. Stankovic and V. Stankovic, An electrical load measurements dataset of United Kingdom households from a two-year longitudinal study, *Scientific Data* **4**(1) (2017), 160122.

[51] X. Zhou, S. Li, C. Liu, H. Zhu, N. Dong and T. Xiao, Non-Intrusive Load Monitoring Using a CNN-LSTM-RF Model Considering Label Correlation and Class-Imbalance, *IEEE Access* **9** (2021), 84306–84315.

[52] S. Singh and A. Majumdar, Multi-Label Deep Blind Compressed Sensing for Low-Frequency Non-Intrusive Load Monitoring, *IEEE Trans. Smart Grid* **13**(1) (2022), 4–7.

[53] Z. Zhang, E. Strubell and E. Hovy, A survey of active learning for natural language processing, *arXiv preprint arXiv:2210.10109* (2022).

[54] S. Budd, E.C. Robinson and B. Kainz, A survey on active learning and human-in-the-loop deep learning for medical image analysis, *Medical Image Analysis* **71** (2021), 102062. doi:https://doi.org/10.1016/j.media.2021.102062.

[55] X. Jin, Active Learning Framework for Non-Intrusive Load Monitoring, Technical Report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2016.

[56] F. Liebgott and B. Yang, Active learning with cross-dataset validation in event-based non-intrusive load monitoring, in: *2017 25th European Signal Processing Conference (EUSIPCO)*, IEEE, 2017, pp. 296–300.

[57] A.M. Fatouh, O.A. Nasr and M. Eissa, New semi-supervised and active learning combination technique for non-intrusive load monitoring, in: *Proc. of SEGE*, IEEE, 2018, pp. 181–185.

[58] L. Guo, S. Wang, H. Chen and Q. Shi, A load identification method based on active deep learning and discrete wavelet transform, *IEEE Access* **8** (2020), 113932–113942.

[59] K. Cho, B. van Merrienboer, Çaglar Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk and Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, in: *Proc. of EMNL*, 2014, pp. 1724–1734.

[60] Y. Wang, J. Li and F. Metze, A Comparison of Five Multiple Instance Learning Pooling Functions for Sound Event Detection with Weak Labeling, in: *Proc. of ICASSP*, 2019, pp. 31–35.

[61] H. Dinkel, X. Cai, Z. Yan, Y. Wang, J. Zhang and Y. Wang, The Smallrice Submission To The Dcase2021 Task 4 Challenge: A Lightweight Approach For Semi-Supervised Sound Event Detection With Unsupervised Data Augmentation, in: *Proc. of DCASE*, 2021.

[62] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh and A. Talwalkar, Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization, *J. Mach. Learn. Res.* **18**(1) (2017), 6765–6816–.

[63] D. Kingma and J. Ba, Adam: A Method for Stochastic Optimization, in: *Proc. of ICLR*, 2014.

[64] C. Klemenjak, S. Makonin and W. Elmenreich, Towards comparability in non-intrusive load monitoring: On data and performance evaluation, in: *Proc. of ISGT*, 2020, pp. 1–5.

[65] D. Batic, G. Tanoni, L. Stankovic, V. Stankovic and E. Principi, Improving knowledge distillation for non-intrusive load monitoring through explainability guided learning, *Proc. of the IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)* (2023), accepted.

[66] R. Machlev, L. Heistrene, M. Perl, K.Y. Levy, J. Belikov, S. Mannor and Y. Levron, Explainable Artificial Intelligence (XAI) techniques for energy and power systems: Review, challenges and opportunities, *Energy and AI* **9** (2022), 100169. doi:https://doi.org/10.1016/j.egyai.2022.100169.

[67] M.H. Rafiei and H. Adeli, A New Neural Dynamic Classification Algorithm, *IEEE Transactions on Neural Networks and Learning Systems* **28**(12) (2017), 3074–3083. doi:10.1109/TNNLS.2017.2682102.

[68] K.M.R. Alam, N. Siddique and H. Adeli, A dynamic ensemble learning algorithm for neural networks, *Neural Computing and Applications* (2020).

[69] G. Tanoni, L. Stankovic, V. Stankovic, S. Squartini and E. Principi, Knowledge Distillation for Scalable Nonintrusive Load Monitoring, *IEEE Transactions on Industrial Informatics* (2023), 1–12. doi:10.1109/TII.2023.3328436.