Active Learning with Cross-Dataset Validation in Event-Based Non-Intrusive Load Monitoring

Florian Liebgott and Bin Yang
Institute of Signal Processing and System Theory, University of Stuttgart
Email: {florian.liebgott, bin.yang}@iss.uni-stuttgart.de

***Abstract*—Supervised event-based NILM systems usually require a large set of labeled training data to achieve high classification accuracies. To minimize the cost of labeling a sufficient amount of events, active learning can be employed. By using only a small set of labeled samples for initial training followed by selecting only the most informative samples to be labeled, the total number of labeled training samples can be reduced significantly. The performance of an active learning system strongly depends on the choice of the initial training set and the used query strategy. We thus investigated the impact of different methods to select the dataset for initial training as well as various query strategies on the resulting classification accuracy in an event-based NILM framework. For evaluation we used two datasets, BLUED and ISS kitchen, on which we were able to achieve high classification accuracies with significantly less training samples compared to conventional training without active learning.**

I. Introduction

Non-Intrusive Load Monitoring (NILM) is the problem of estimating the individual load curves of appliances from an aggregated measurement [1]. A widely used approach [2], [3] is event-based disaggregation, in which the load curves are constructed based on the state changes of the appliances, the so-called events.

In a supervised event-based NILM system, a classifier is used to determine, to which appliance a detected event corresponds. To achieve a high classification accuracy, a lot of labeled data is needed to train the classifier. As the labeling of the events is a time-consuming and expensive task, it is desired to minimize the labeling cost by using as little training data as possible. Usually, a set of training data contains a lot of redundancy and not all samples are equally valuable for the training.

Active learning is a technique to reduce the labeling cost by labeling only those samples, that are most valuable for the training of the classifier. Key factors for the success of active learning are the query strategy, which is used to select the most informative samples, and the choice of the initial training set.

To the best of our knowledge, until now there has been almost no research concerning active learning for NILM systems. The only published work in [4] demonstrated that active learning can be used successfully in an event-based NILM system. Using a query strategy to select the unlabeled sample farthest from all labeled samples, they achieved promising results on a subset of the Building-Level fUlly-labeled dataset for Electricity Disaggregation (BLUED) [5] by using a k-nearest neighbor classifier. However, the query strategy intuitively seems to be unsuitable for datasets with classes that are distributed very unevenly in the feature space. We could not reproduce the results, especially since the evaluation seems to have been carried out on a not specified subset of BLUED. There is, however, no study about active learning in NILM concerning the choice of query strategies or methods to select the initial training data.

In this work, we compare different query strategies and methods to select the initial samples for active learning in event-based NILM and evaluate them on both BLUED and our own dataset.

II. Event-Based NILM Framework

We implemented an event-based NILM framework to estimate load profiles and energy usage of individual appliances from the measurement of voltage and aggregate current. The basic idea behind event-based NILM is the assumption, that an appliance has several states and that each of these states has a specific steady power level. The simplest case is an appliance with only two states: an off state, during which no power is consumed, and an on state with a constant power consumption. For these appliances, the only possible events are switching on and switching off. If an appliance has more states, every possible transition from one state to another has to be considered as an event. If all possible states and state transitions of an appliance are known, the load curve of the appliance can be constructed based on events detected in the aggregate power signal.

Appliances which exhibit variable load profiles instead of fixed states clearly violate the underlying assumption of event-based NILM and can thus not be modeled satisfactorily by an event-based NILM system.

An overview of our event-based NILM framework is depicted in Figure 1. After the calculation of active and reactive power from the measured voltage and aggregate current, we detect events in the power signals. An event is a sudden power change caused by the change of an appliance's state. For each event, different features are extracted and used for the event classification. The final step is the estimation of a load profile of and the total energy consumed by each appliance.

In our NILM framework, we use a multiclass soft-margin Support Vector Machine (SVM) with a radial basis function kernel to classify events. The feature vector $\underline{x}$ contains the difference in active and reactive power, ΔP and ΔQ, after and before the event as well as the difference ΔP_{peak} between

ISBN 978-0-9928626-7-1

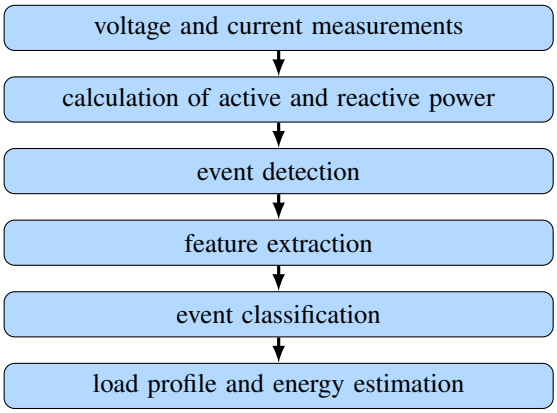

Fig. 1. Overview of our NILM framework

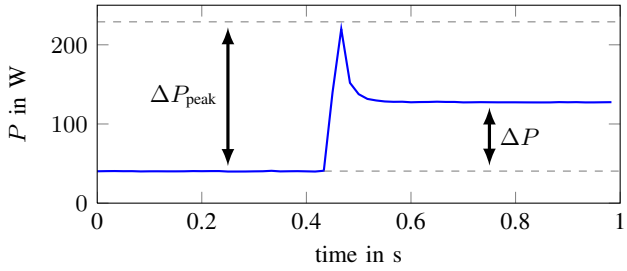

Fig. 2. Active power features of an event

the maximum and the minimum value of the active power during the event. Figure 2 depicts the active power features extracted from an event.

In this work, we focus on the classification stage and therefore skip the event detection. Instead of using an event detector, we use the ground truth event information included in the dataset. Hence, we can assume a perfect event detection and do not have to consider the impact of errors in the event detection stage.

III. Active Learning

The objective of an active learning system is to reduce the labeling cost by selecting only the most informative samples and presenting them to a so-called oracle, usually a human expert.

In a pool-based active learning system, the samples are taken from a pool or set $\mathcal{U}_0$ of unlabeled data. Usually, active learning starts with the selection of a small initial set $\mathcal{L}_0$ from $\mathcal{U}_0$ to be labeled by the oracle, reducing the set of unlabeled data to $\mathcal{U}_1 = \mathcal{U}_0 \setminus \mathcal{L}_0$. The labeled samples are then added to the empty set of labeled training data $\mathcal{D}_0$, forming the extended training set $\mathcal{D}_1 = \mathcal{D}_0 \bigcup \mathcal{L}_0$, which is used for the initial training of the classifier.

After the initial training, the set $\mathcal{L}_i$ of the N_L most informative samples of the pool of unlabeled data $\mathcal{U}_i$ is selected based on an informativeness measure and presented to the oracle who labels these instances. The classifier is then trained again using the extended training set $\mathcal{D}_{i+1} = \mathcal{D}_i \bigcup \mathcal{L}_i$ and the set of unlabeled data is reduced correspondingly to $\mathcal{U}_{i+1} = \mathcal{U}_i \setminus \mathcal{L}_i$. This process is repeated, until a predefined stopping criterion is met, usually a desired classification accuracy or a maximum number of iterations.

The success of an active learning system depends on several degrees of freedom, among others the size and composition of the initial training set, the number of samples to be labeled in each iteration, the stopping criterion and the query strategy.

A. Query Strategies

Each query strategy uses an informativeness measure $\phi(\underline{x})$ that models how valuable the sample $\underline{x}$ is for the training of the classifier. The N_L samples with the highest informativeness values are then selected to form the set

$$\mathcal{L}_i = \bigcup_{n=1}^{N_L} \left\{ \underline{x}_n | \max_{\underline{x} \in \mathcal{U}_i} \phi(\underline{x}) \right\} \tag{1}$$

of the samples which are to be labeled. In our study, we considered the following query strategies:

1) Random sampling: The samples to be labeled are randomly chosen from the set of unlabeled data. Obviously, this is the reference strategy, which should be outperformed by a more sophisticated approach.

2) Uncertainty sampling: Probability-based query strategies such as uncertainty sampling [6] use the class probability $P(\hat{y}|\underline{x})$, that a sample $\underline{x}$ belongs to class $\hat{y}$, to find the most informative samples. Uncertainty sampling selects those samples, for which the maximum class probability is minimal. The informativeness measure is thus given by

$$\phi^{\mathrm{U}}(\underline{x}) = 1 - P(\hat{y}|\underline{x}) \quad \text{with} \quad \hat{y} = \arg\max_y P(y|\underline{x}). \tag{2}$$

3) Margin sampling: With uncertainty sampling, only the probability of the most probable class is taken into account. It gives an estimate about how confident the classifier is that this is the correct class. But it does not take into account the distribution of the probabilities of the other classes.

To overcome this shortcoming, margin sampling [7] uses the difference between the probability of the first and second most probable class $\hat{y}_f$ and $\hat{y}_s$ to find the most valuable samples. As a small difference between the highest probabilities indicates a low confidence of the classifier, the informativeness measure

$$\phi^{\mathrm{M}}(\underline{x}) = -|P(\hat{y}_f|\underline{x}) - P(\hat{y}_s|\underline{x})| \tag{3}$$

achieves its maximum value of 0, when the probabilities for the first and second most probable class are identical.

4) Entropy-based sampling: In information theory, Shannon entropy [8] is commonly used to measure the unpredictability of an information source. If we consider the classifier as such an information source, its unpredictability and therefore its entropy is maximum, if all possible classes have the same probability, and minimum, if one class has a probability of 1. If we thus choose the informativeness measure

$$\phi^{\mathrm{E}}(\underline{x}) = -\sum_i P(y_i|\underline{x}) \log P(y_i|\underline{x}) \tag{4}$$

as the Shannon entropy, the samples with the most equal class probabilities are selected as the most informative ones.

ISBN 978-0-9928626-7-1 © EURASIP 2017

5) Distance-based sampling: Distance-based sampling uses the distance $d(\underline{x})$ of a sample $\underline{x}$ to the decision boundary as a measure for uncertainty. Intuitively, the nearer a sample is to the decision boundary, the more valuable it is in the training of a classifier, especially when using an SVM. The informativeness measure is therefore given as

$$\phi^{\mathrm{D}}(\underline{x}) = -d(\underline{x}). \tag{5}$$

In our work, we use the Euclidean distance.

6) Information density: To reduce the impact of outliers, Settles et al. [9] proposed an approach called information density. The uncertainty measure $\phi(\underline{x})$ of any arbitrary query strategy is weighted with a density $\mathrm{sim}(\underline{x}, \underline{x}_m)$, which gives the similarity of a sample $\underline{x}$ to all other unlabeled samples $\underline{x}_m$. This results in the density weighted informativeness measure

$$\phi^{\mathrm{I}}(\underline{x}) = \phi(\underline{x}) \left(\frac{1}{|\mathcal{U}_i| - 1} \sum_{\underline{x}_m \in \mathcal{U}_i \setminus \underline{x}} \mathrm{sim}(\underline{x}, \underline{x}_m) \right)^{\beta}. \tag{6}$$

The parameter β allows to adjust the influence of the weighting term. A common choice for the similarity measure is cosine similarity.

7) Maximum mean distance: For comparison, we also evaluated the approach used by Yin [4], who uses the maximum mean distance of a sample $\underline{x}$ to all labeled samples $\underline{x}_l$ as informativeness measure, which is thus given by

$$\phi^{\mathrm{Y}}(\underline{x}) = \frac{1}{|\mathcal{D}_i|} \sum_{\underline{x}_l \in \mathcal{D}_i} d(\underline{x}, \underline{x}_l). \tag{7}$$

We use the Euclidean distance as distance measure $d(\underline{x}, \underline{x}_l)$ between the samples $\underline{x}$ and $\underline{x}_l$.

B. Initial Training Set

We studied three ways to create the initial training set:

1) Random selection: The easiest and least expensive way to select the samples for the initial training set is to randomly select the samples from the pool of unlabeled data. However, if the dataset is heavily unbalanced, it is possible that the initial training set does not contain samples from all classes.

2) Class-based selection: To ensure that every class is represented in the initial training set, a fixed number of samples is picked from each class. To get the initial samples, either additional measurements of each appliance are needed or samples have to be randomly selected from the pool of unlabeled data until for each class the desired number of samples are labeled.

3) Cluster-based selection: In order to avoid the additional cost for ensuring that all classes are considered in class-based selection, we propose to select the initial training set based on a clustering of the unlabeled data. For the clustering, we use the density-based DBSCAN algorithm [10]. As we observed that clusters in the feature space have a higher variance for higher feature values, we compensate this by transforming each feature x logarithmically to

$$x' = \begin{cases} \ln(x) & \text{if} \quad x > 0 \\ -\ln(-x) & \text{if} \quad x < 0 \end{cases} \tag{8}$$

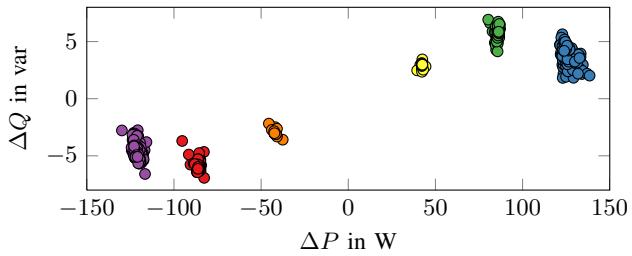

Fig. 3. Clustering of the events of a refrigerator resulting in six clusters.

before applying the clustering algorithm.

We chose the parameters for DBSCAN such that a sample is considered as a core sample when at least ten samples with a maximum distance of 0.3 in the transformed feature space are in its neighborhood. From these core samples, DSBSCAN constructs the clusters.

From each cluster found by DBSCAN, the sample closest to the centroid is selected for the initial training set, because it is the most representative sample for that cluster. Depending on the distribution of the data, DBSCAN may find fewer clusters than there are classes and consequently the initial training set may contain fewer samples than there are classes. Therefore, cluster-based selection can not ensure that each class is represented in the initial training set.

IV. Experimental Setup

The active learning system was tested on BLUED [5], which contains measurements of two phases of a single-family home in the USA for one week. All events are labeled with the identifier of the corresponding appliance, but there is no information about the type of state change. To identify the different state transitions of an appliance, we performed a clustering of all events of an appliance. Each cluster corresponds to a state transition of the appliance. Figure 3 shows the clustering of the events of a refrigerator, which results in six clearly separated clusters, indicating six possible state transitions.

Because of the short measurement period of one week, some clusters have very few samples. We thus excluded all clusters with less than 20 samples. Each remaining cluster is then assigned a unique class label to identify the corresponding state transition.

For phase A, nine classes with 681 samples altogether remain, phase B contains 21 classes and 906 samples overall. We also evaluated the framework on the combination of both phases with 30 classes and 1587 samples.

As the datasets are heavily unbalanced, we take the class labels into account when splitting the dataset into a training and a test set to ensure that all classes are represented both in the training and the test set. 25 % of the samples of each class are randomly selected as test set, the remaining samples form the pool of unlabeled samples.

We also tested the active learning system on our own dataset called ISS kitchen, which was measured in our institute's kitchen. It contains eight days of measurements of a refrigerator, a coffee maker, a water kettle, a boiler and a microwave as well as the aggregate signal. As in BLUED, we

ISBN 978-0-9928626-7-1 © EURASIP 2017

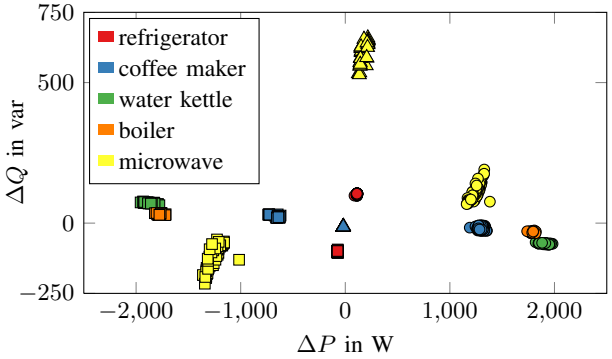

Fig. 4. All samples of the ISS kitchen dataset in the ΔP-ΔQ plane. Circles mark the switching on events of an appliance, squares mark the switching off events. Other state changes are marked with a triangle.

identified the state transitions of an appliance by clustering all events of an appliance and ensured that each class contains at least 20 samples. The dataset contains a total of 710 samples distributed across twelve classes as shown in Figure 4.

To take into account that the datasets are heavily unbalanced, we used the balanced test accuracy to evaluate the performance of our active learning system. The balanced test accuracy is calculated as the mean of the test accuracy of all classes. Each experiment was repeated 100 times with randomly chosen test sets. The mean balanced test accuracy (MBTA) across the 100 runs is used as our evaluation metric.

V. Results

A. Query Strategies

Across all tested datasets, probability-based methods consistently yield the best results and clearly outperform random sampling. Distance-based sampling and the approach by Yin yield considerably poorer results than random sampling and seem to be unfit as query strategies for active learning in our event-based NILM framework.

Figure 5 shows the MBTA of all query strategies without density weighting on phase A of BLUED against the ratio of the size of the current training set $\mathcal{D}_i$ against the number of all available training samples. As on the other tested datasets, margin sampling performs best and uncertainty sampling and entropy-based sampling perform slightly poorer. Density weighted methods are not shown as none of them exceeded the performance of margin sampling.

In Figure 6, the MBTA of margin sampling is plotted against the ratio of the size of the current training set $\mathcal{D}_i$ against the number of all available training samples. The plot shows that after the initial training, the MBTA reaches values of over 80 %. On phase A of BLUED and our own dataset, an MBTA of 100 % is reached using only 7 % of all available training samples.

In contrast to those two datasets, phase B contains classes that overlap in the feature space and are not as easily distinguishable. Consequently, the MBTA increases slower for phase B and the combination of phase A and B. When using 18 % of all available training samples, the MBTA for phase B reaches the same value as when training with all available training

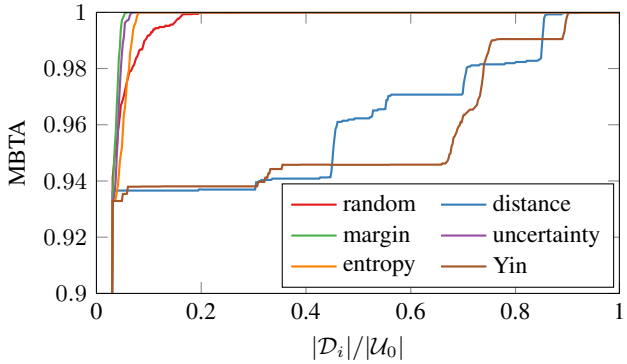

Fig. 5. MBTA of all query strategies without density weighting on BLUED phase A against the ratio of the size of the current training set $\mathcal{D}_i$ against the number of all available training samples. The initial training set is created by the combination of cluster-based and random selection. N_L is chosen as 1 for all strategies.

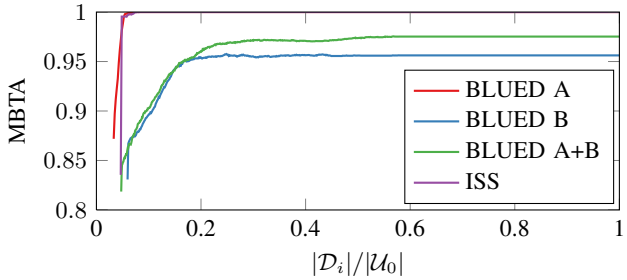

Fig. 6. MBTA of margin sampling on different datasets against the ratio of the size of the current training set $\mathcal{D}_i$ against the number of all available training samples. The initial training set is created by the combination of cluster-based and random selection. N_L is chosen as 1.

data. After the training with 22 % of the available training data, the MBTA of the combination of phase A and B reaches a stable level 0.5 percentage points below the training with all available training samples.

The good performance of margin sampling is confirmed by studies on active learning for other applications, where margin sampling often is among the most promising query strategies.

B. Selection of the Initial Training Set

Cluster-based selection of the initial training set clearly outperforms random selection on all tested datasets. Figure 7 shows the MBTA against the ratio of the size of the current training set $\mathcal{D}_i$ against the number of all available training samples for margin sampling on phase B. We found that cluster-based as well as class-based selection lead to surprisingly high initial values for the MBTA. When adding more samples to the training set, the MBTA decreases at first, but reaches higher values with a growing number of training samples.

To overcome this initial dip, we combined cluster-based selection with random selection. The initial MBTA of the combined selection is lower than the initial MBTA of cluster-based selection, but it outperforms cluster-based selection after a few active learning iterations.

On phase A of BLUED and on our own dataset, cluster-based selection and the combined selection strategy perform slightly poorer than class-based selection. However, class-based selection requires knowledge of the number of classes

ISBN 978-0-9928626-7-1 © EURASIP 2017

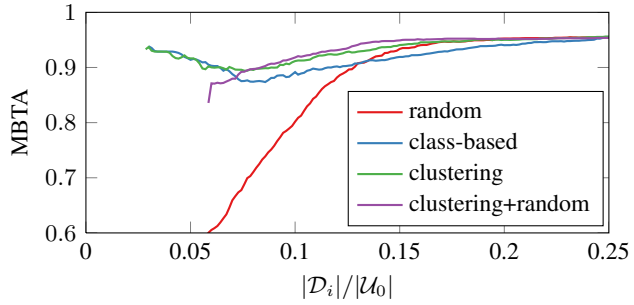

Fig. 7. MBTA of margin sampling with $N_L = 1$ on phase B of BLUED against the ratio of the size of the current training set $\mathcal{D}_i$ against the number of all available training samples for different methods to select the initial training set.

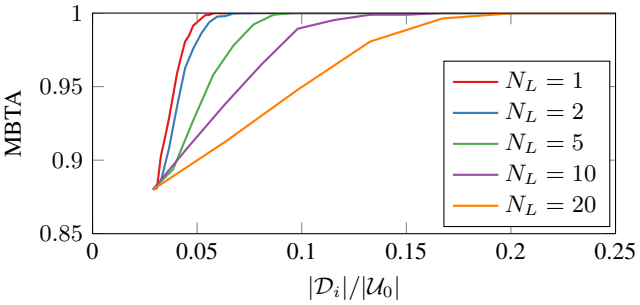

Fig. 8. MBTA of margin sampling on phase A of BLUED against the ratio of the size of the current training set $\mathcal{D}_i$ against the number of all available training samples for different numbers N_L of samples to be labeled in each iteration.

and class labels. It causes additional effort to ensure that all classes are represented in the initial training set. Therefore, we propose to use the combination of cluster-based and random selection.

We observed that the performance of the combined approach does not deteriorate noticeably, if the initial clustering finds fewer clusters than there are classes.

Compared to other active learning applications, we use a rather small initial training set. The high initial MBTA and the good performance of the initial clustering on the tested datasets can be explained by the compact and distinguishable representation of the classes in the feature space.

C. Number of Samples to Be Labeled in Each Iteration

The number N_L of samples to be labeled in each iteration influences the practicability of the active learning system. If N_L is small, the classifier has to be retrained very often. On the other hand, if more samples are selected for labeling in each iteration, the average uncertainty across those samples decreases, which could lead to a lower performance of the active learning system.

Our results show that for probability-based query strategies, increasing N_L leads to a higher overall number of training samples necessary to achieve a comparable MBTA value. This behaviour was observed across all datasets and probability-based methods. Figure 8 shows this behaviour for margin sampling on BLUED phase A as an example.

With only one sample labeled in each iteration, probability-based methods require the fewest samples to achieve an MBTA comparable to the training with the whole training set. Despite the increased computational cost caused by the frequent retraining of the classifier, we therefore propose to only label one sample in each iteration.

This choice is not uncommon for active learning. In some applications, however, it is preferable to use a larger number of samples to label in each iteration to improve the diversity in the labeled samples and to reduce the computational cost.

VI. Conclusion

In our study, we showed that active learning can reduce the amount of labeled data needed to train a support vector machine for event classification by around 80%, depending on the data set even by over 90%. Our results also show that the improvement by active learning heavily depends on the employed query strategy. We came to the conclusion that query strategies based on class probabilities, especially margin sampling, work very well for the described setup and are robust with respect to the samples chosen for the initial training. Distance-based methods, on the other hand, yield very poor results. Concerning the actual active learning setup, we found that creating the initial training set by clustering the unlabeled data, selecting the sample closest to each class centroid and adding several random samples is the most promising method. The labeling of only one sample per iteration is a reasonable choice.

The cross-dataset evaluation of our framework showed reproducible results concerning the query strategies, the selection of the initial training set and the number of samples to be labeled in each iteration.

References

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

[2] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 1, pp. 76–84, February 2011.

[3] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors*, vol. 12, no. 12, pp. 16 838–16 866, 2012.

[4] X. Yin, "An active learning framework for non-intrusive load monitoring," in *Proceedings of the 3rd International Workshop on Non-Intrusive Load Monitoring*, Vancouver, Canada, May 2016.

[5] K. Anderson, A. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Berges, "BLUED: a fully labeled public dataset for Event-Based Non-Intrusive load monitoring research," in *Proceedings of the 2nd KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, Beijing, China, Aug. 2012.

[6] D. Lewis and W. Gale, "A sequential algorithm for training text classifiers," in *Proc. of the ACM SIGIR Conf. on Research and Development in Information Retrieval*. Springer-Verlag, 1994, pp. 3–12.

[7] T. Scheffer, C. Decomain, and S. Wrobel, "Active hidden markov models for information extraction," in *Proceedings of the 4th International Conference on Advances in Intelligent Data Analysis*, ser. IDA '01. London, UK: Springer-Verlag, 2001, pp. 309–318.

[8] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[9] B. Settles and M. Craven, "An analysis of active learning strategies for sequence labeling tasks," in *Proc. of the Conf. on Empirical Methods in Natural Language Processing (EMNLP)*. ACL, 2008, pp. 1069–1078.

[10] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. AAAI Press, 1996, pp. 226–231.

ISBN 978-0-9928626-7-1 © EURASIP 2017