Energy & Buildings 187 (2019) 95–109

Contents lists available at ScienceDirect

Energy & Buildings

journal homepage: www.elsevier.com/locate/enbuild

Self-configuring event detection in electricity monitoring for human-building interaction

Milad Afzalan[a], Farrokh Jazizadeh[a,*], Jue Wang[b]

[a] The Charles E Via Jr., Department of Civil and Environmental Engineering, Virginia Tech, VA, USA
[b] Department of Civil and Environmental Engineering, Stanford University, CA, USA

ARTICLE INFO

Article history:
Received 22 October 2018
Revised 24 January 2019
Accepted 26 January 2019
Available online 28 January 2019

Keywords:
Electricity disaggregation
Event detection
Unsupervised learning
Non-intrusive load monitoring
Clustering

ABSTRACT

Monitoring the temporal changes in the operational states of appliances is a key step in inferring the dynamics of operations in smart homes. This information could be leveraged in a variety of energy management applications including energy breakdown of individual loads, inferring the occupancy patterns, and associating the energy use to occupants' activities. The operational states of appliances could be identified through detecting and classifying the events on power time-series. Despite the advancements in the field of event detection, they often require in-situ configuration of model parameters to achieve a higher level of performance according to each new context. In order to address such limitation, in this paper, we have proposed a self-configuring event detection framework for detecting the changes in operational states of appliances. The framework seeks to autonomously learn the contextual characteristics of the loads from the environment and adapt the event detection parameters. The proposed unsupervised framework couples an automated clustering for identifying the recurring motifs, which are representations of the appliances' transient power draw signatures in a given environment and a proximity-based motif matching for detecting the events. The framework was evaluated on EMBED dataset, a publicly available fully labeled electricity disaggregation dataset, collected from three apartments with different categories of the appliances. The evaluations demonstrate that the proposed event detection framework outperforms the conventional event detection in detecting the operational states of different classes of loads across different environments. The proposed framework could also facilitate human-building interactions in training smart home applications by populating motifs to infer the operations of appliances and activities of occupants.

1. Introduction

Buildings account for 74% of total electricity consumption in the US with a share of more than half for residential buildings [1]. Therefore, enabling efficient consumption of the electricity in buildings remains as a major sustainability objective. Understanding the energy consumption of appliances and their operational schedule in a building and the interactions between occupants and the appliances could bring about a number of advantages that pave the way for achieving sustainability goals in the form of both demand-side and demand-response energy management [2,3]. Among these goals, one could point to providing detailed energy information to occupants for increased awareness (e.g., [3–9]), characterizing the energy impact of occupants' activities (e.g., cooking a meal or adjusting the air conditioning setting), understanding the habitual patterns of occupants in use of appliances for smart and autonomous operations [10,11], inferring the occupancy of building units for occupancy-driven energy management [12–17], and enabling utilities to identify and target critical loads for grid reliability at high peak demand time [18,19]. At the center of all these applications is the understanding of load dynamics in buildings that reflect the individual appliance operations. Individual load operations could be monitored through direct sensing of individual appliances. However, in pursuit of scalable appliance-level analyses, electricity disaggregation [20–23], which relies on data measurement at the aggregate level at one sensing node, could be used as a cost-effective alternative to break down the aggregate load into end-use individual loads.

By inferring when an appliance changes its operational state, which is interpreted as an event on the aggregate power time-series, we could potentially identify its energy consumption, the user interaction with the appliance, and the activities of users (inferred from interacting with a series of appliances). Therefore, identifying the events on power time-series comprises an

* Corresponding author.
E-mail addresses: afzalan@vt.edu (M. Afzalan), jazizade@vt.edu (F. Jazizadeh), juew94@stanford.edu (J. Wang).

https://doi.org/10.1016/j.enbuild.2019.01.036
0378-7788/

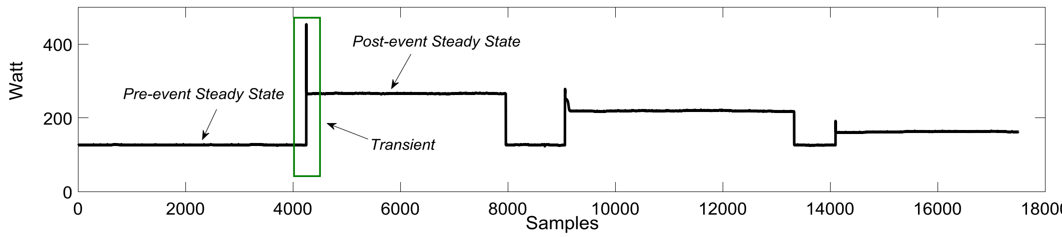

Fig. 1. Sample real power time series with a 60 Hz resolution that shows the information gain from transient power draw.

important step in characterizing the energy performance of individual loads and human-appliance interactions. In this context, an *event* denotes a change of the appliance operational state, which could be the on/off switching (e.g., in case of a lighting load) or an operational state-transition (e.g., different cycles of a washing machine). Assuming a time-series signal $P(t)$, collected at the aggregate level, an *event detector* aims to identify the timestamps associated with the change points $T = \{t_1, t_2, \ldots, t_N\}$. These timestamps will be mapped to a set of labels (corresponding to the classes that represent appliance names) through a training step. This information could be further analyzed by pattern recognition methods for processing the data into either energy breakdown estimation of appliances [24,25], inferring the trend of household's appliance use to determine the drivers of consumption and predicting future demand [26–29], or inferring the activities of occupants [30–32].

Depending on the resolution of electricity consumption data, different algorithms could be used for event detection. Increasing the resolution of the data could help improve the accuracy of the classification algorithms [20] in inferring load identities. This improvement is due to the information gain from the presence of transient data (i.e., a momentary increase in the power before reaching a steady state of power draw), which reveals more information on the dynamics of the loads [21] (Fig. 1 illustrates an example power time-series with trainset power draw data). However, increasing the resolution brings about an increase in noise interference, which results in challenges for event detectors, such as the increase in false positive detection [33]. To tackle the challenges in detecting events on high-resolution data, advanced event detection techniques were adopted and devised in the literature (e.g., [34–37]). As a common feature, these algorithms rely on a number of algorithm parameters that drive the detection statistics and the performance and thus need to be configured for different environments through a training (i.e., tuning) process. Although the configuration could be carried out through experimental efforts and across different environments, the diversity in the appliances' technologies could pose a challenge in achieving scalable performance in different environments. Therefore, there is often a need for reconfiguration of the algorithms to ensure that the better set of parameters for each new environment is set.

To contribute to the scalability of the electricity disaggregation solutions, in this study, we have proposed a framework to move towards self-configuring event detection techniques. The proposed framework is centered on enabling event detectors to learn from the data in an environment and adapt to the characteristics of the environment and its unique loads. The framework is built on populating an initial dataset for learning and leveraging the recurring motifs, reflected in the shapes of appliances' transient power draw. In its high-level concept, the framework consists of the following steps:

- Populating an initial dataset of appliances' signatures (i.e., vectors representing transient power draws) from a buffered power time series by utilizing a conventional event detector (with generalized configured parameters).
- Characterizing the recurring motifs by using an unsupervised automated clustering on detected events in the buffered data. The extracted motifs in the library represent the characteristics of appliances' signatures in the environment.
- Shifting the event detector to a motif-based matching method: events will be identified, where the shape of the power time series matches one of the motifs. Motif-matching step involves an outlier detection in comparing the power time series shape with the representative motifs in the library.

Therefore, this study contributes to the body of literature by proposing an event detection framework that seeks to automatically learn the algorithm parameters based on the context of an environment to avoid parameter-tuning in each new environment. Due to the nature of the motif detection step, which employs an automated clustering (without assuming the number of clusters), the proposed framework could also act as a classification step (mapping the events to their associated appliances). Moreover, our evaluation of the framework over the EMBED dataset [38], which is the most comprehensive labeled dataset with three building units, puts the analysis among the most comprehensive assessments.

The rest of the paper is structured as follows. In Section 2, we presented a literature review of the related studies. In Section 3, we presented the framework and discussed its components, following by Section 4, in which the results of evaluations have been presented. The paper was concluded in Section 5.

2. Background and related works

Disaggregation methodologies have been applied in different capacities with a focus on residential buildings (e.g., [21,39]) or commercial and office settings (e.g., [40–42]). Several recent efforts have focused on improved feature selection for appliance classification [43], employing deep learning methods [44,45], using alternative metrics such as current or voltage instead of the commonly used power metric for load monitoring [45–47], and leveraging the impact of device interaction (mutual operational status) for improving the disaggregation accuracy [48,49]. In this study, our focus is on methodologies that belong to class of techniques that use event detection. Event detection is a commonly used approach in the analysis of time-series data which has applications in various domains. Similarly, event detection has been used for the analysis of the power time-series (or similar electricity consumption metrics). The approach has been often adopted in the electricity disaggregation, also known as non-intrusive load monitoring, specifically in a category of efforts, described as event-based methods. In this category, the abrupt changes on the time-series are identified as events that are associated with changes in the operational states of appliances. In early studies on electricity disaggregation, the efforts were focused on developing heuristics for detecting the events of on/off appliances. Hart [50], in his seminal research, proposed a detector for on/off transitions by segmenting the normalized power values into steady or changing states. In this heuristic, using low-resolution power data, the steady states were identified by measuring the power variations against a threshold for a pre-defined number of samples on the power time series (e.g., three sample points). The segments of the time series that violate this rule are considered to be changes (i.e., events). This approach has been also used in the recent state-of-the-art disaggregation studies (e.g., [51]). In another heuristic method [52], separate profiles for different appliances were collected as the training and a set of rules were defined to find the on/off transitions by comparison with the pre-defined power ranges. The approach called for profile initialization of the appliances in-the-field.

In order to benefit from the data reflected in the transient power draws, research studies shifted to use the data from the high-resolution power time series. Increasing the resolution of the power data poses more challenges to event detection due to the presence of noise. In order to address these challenges, improved event detection algorithms were proposed. The application of the generalized likelihood ratio (GLR) test was introduced by Luo et al. [34] to improve the performance. In this method, events are detected by measuring whether samples before and after an event are coming from two different Gaussian distributions. The algorithm calls for configuring several parameters including two window sizes to calculate the parameters of probability distribution functions before and after each event and a threshold for likelihood ratio to compare against the detection statistics. Variations of the GLR algorithm for enhanced performance has been also proposed (e.g., [53]), which call for more parameters and thus configurations. Other research efforts have also introduced variants of event detection algorithms for high-resolution data to improve detection accuracy. Some of these efforts are based on the goodness-of-fit χ^2 test algorithm on power time-series [54,55], Window-with-Margins event detection method [56], and adaptive event detection [57] that detects the time limits of each transition interval. Similar to the GLR algorithm, these methods also call for a number of parameters that need to be configured for high-performance event detection. Alternative methods of event detection for alternative electricity measurement metrics have been also proposed. For example, high-resolution voltage time series were used by Patel et al. [10] for detecting events associated to switch on/off or changes of the cycle in appliances. They proposed a specialized event detection algorithm that uses a threshold for identification of the events on the voltage noise time series which required data acquisition systems with very high sampling rates.

Since extensive parameter-tuning or training imposes a barrier to the wide adoption of disaggregation technologies, a few recent studies have focused on unsupervised approaches [58,59]. Wild et al. [58] proposed an unsupervised event detector based on the kernel Fisher discriminant analysis (KFDA) using current harmonics. The event detector requires two sliding windows with the predetermined length to calculate the test statistics and a bandwidth parameter for the Kernel function. Similarly, in [59], an event detector based on a two-step clustering from graph signal processing has been proposed. The approach collects the subsequent sampling points with power measurement difference above a threshold and then applies adaptive thresholds to refine the clusters until all of them have a coefficient of variation below a specific range for the quality control.

These aforementioned event detection methods either require a training process ([10,52,54,60]) or a parameter-tuning ([34,50,53,61,62]) step, which calls for the reconfiguration of parameters for new environments and therefore pose a challenge on generalizability and wide adoption [63]. In order to tackle such limitations, in this study, we have sought to propose an event detection approach that leverages the recurring appliance signatures, obtained from an environment to adapt to the characteristics of each unique environment. The approach allows the system to automatically identify detection parameters according to the characteristics of the new deployed environment and could potentially facilitate appliance event labeling through reduced user-system interaction.

3. Self-configuring event detection framework

The proposed framework seeks to shift the detection logic from searching for abrupt changes on a time series to a motif-based detection approach. Therefore, the framework utilizes a search for the most probable transient (power draw) shapes on a time-series that are associated with appliances' state transitions, which we refer to as *motifs*. The concept of using motifs has received attention in the time-series data mining domain (e.g., [64]). In this work, we leverage and deploy the recurring motifs (i.e., time-series subsequences) that represent a collection of signatures from a specific operational state of a given appliance. In doing so, the framework uses the processed power time-series as the representative metric of aggregate electricity consumption. Fig. 2 illustrates the components and process map of the framework. There are two underlying steps: (1) self-training stage for motif processing on the buffered data, and (2) the proximity-based event detection stage. In what follows, the descriptions for different components have been provided.

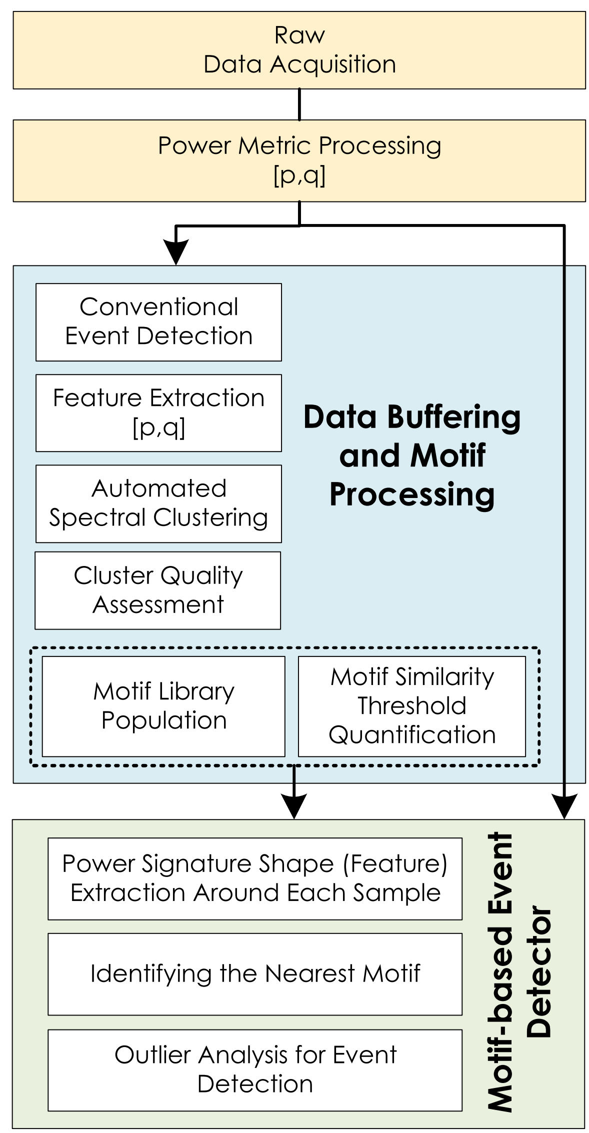

Fig. 2. The framework for the motif-based event detection.

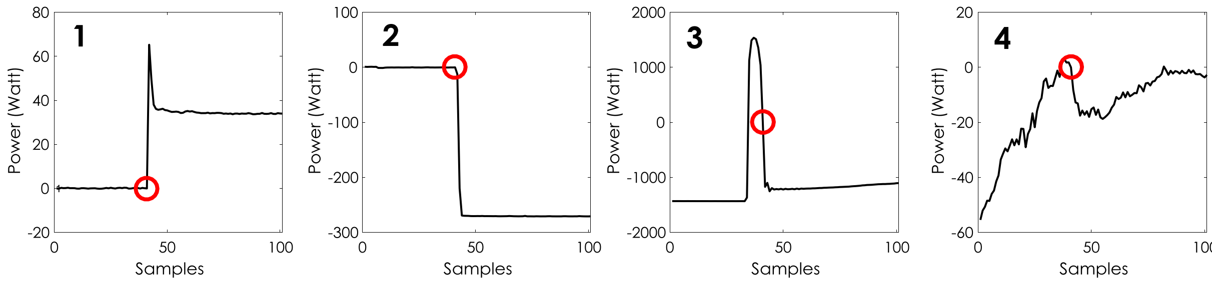

Fig. 3. Examples of extracted features (the real power component) on the buffered data.

3.1. Data buffering and motif processing

3.1.1. Conventional event detection for initial learning

The first step of this framework focuses on identifying the recurring appliance signature motifs in a given environment. Considering that the motifs represent the changes of the appliances' operational states, the number of these motifs is significantly lower compared to those of steady-state segments. Therefore, the framework leverages a conventional event detector to identify the occurrence of transient events that represent the potential motifs and prepare the dataset for contextual learning. As the nature of motifs implies, this framework leverages the information in the transient state in high-resolution data. Although any event detection algorithm for high-resolution data could be integrated into this framework, in this study, we have utilized the GLR event detector [34] as a common method, which uses a statistical test to identify events. The algorithm evaluates likelihood ratio (detection statistics) between Gaussian distributions ($S \sim N(\mu,\ \sigma^2)$), assigned to samples of data before and after each data point on the power time-series:

$$L(n) = \ln \frac{P\left(s_i|\mu_1, \sigma_1^2\right)}{P\left(s_i|\mu_0, \sigma_0^2\right)} \tag{1}$$

where s_i is the ith signal sample point, and μ_0, σ_0^2, μ_1, and σ_1^2 are mean and standard deviation in two windows before and after each data point. Time-series sample points with a likelihood ratio higher than a predefined threshold are marked as events. Accordingly, the algorithm calls for configuring the size of two windows before and after each sample point for estimating parameters of the distributions and a threshold for event detection. In some implementations of this algorithm [53], additional mechanisms for reduced false positives have been used. These mechanisms in turn call for configuring a few more parameters.

In the proposed framework, the conventional event detector will be used without specific efforts on tuning these parameters. In other words, we use parameters from the literature that are the outcome of a configuration process for a specific dataset. Therefore, the event detector, in this step, is prone to identifying wrong events (false positives), or missing events (false negatives). Accordingly, the initially collected events from the environment include a mix of both correct and wrong detections. Nonetheless, the impact of such inaccurate detections is tackled by the motif-mining approach, which has been described in the upcoming sub-sections.

3.1.2. Feature extraction

Upon identifying the initial events, a sequence of data samples that surround an event is extracted to form the feature vectors ($\boldsymbol{fv}$) representing each appliance signature motif. Two windows of pre-event samples (w_{pre}) and post-event samples (w_{post}) are used to extract the features. Different harmonic components of real and reactive power time series could be used in the feature extraction stage. In this implementation of the framework, we have focused on the fundamental frequency component of real and reactive power time series as the representation signature motifs in the vicinity of the events:

$$\boldsymbol{fv} = \{\boldsymbol{p}_1, \boldsymbol{q}_1\} \tag{2}$$

where p_1 and q_1 are real and reactive power (fundamental frequency) components, respectively. Fig. 3 illustrates some examples of transient power signatures (the real component only) representing changes in appliances' operational state. These feature vectors are normalized to have a value of zero at the point of the event so that the aggregate nature of the power time-series does not affect the comparison between two vectors.

3.1.3. Identifying recurring motifs through automated clustering

The next step includes self-learning of the recurring motifs for enhanced event detection. A critical component to achieve this goal includes automated clustering of the appliance signatures to infer the motifs. Although clustering algorithms are categorized under the unsupervised learning techniques, they commonly call for *a priori* parameters. For instance, K-Means clustering, hierarchical clustering, and mean-shift clustering require different parameters including the number of clusters, a threshold for pruning the tree, or the kernel bandwidth, respectively. Although these values could be set using the domain knowledge, the use of algorithms that need additional hyperparameters contradicts the objective of self-configuration. Therefore, we have developed autonomous clustering algorithms that obviate the need for input parameters (e.g., [65,66]). In this study, we have adopted our proposed heuristic for automated spectral clustering with application to electricity disaggregation [67]. The spectral clustering algorithm uses the eigenvalues of a Laplacian matrix from the data for dimensionality reduction and clustering in fewer dimensions [68]. Assuming a data set $X = [x_1,\ x_2,\ x_3 \ldots,\ x_n] \in R^{n\times m}$ in which n is the number of data points (i.e., all feature vectors) and m is the number of features, a similarity matrix is defined as:

$$A_{ij} = \begin{cases} \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) & i \neq j \\ 0 & i = j \end{cases} \tag{3}$$

In which σ^2 is the scaling factor. Through creating a diagonal matrix with the summation of all the elements on the ith row of A as D_{ii}, a Laplacian matrix is defined as:

$$L = D^{-\frac{1}{2}} A D^{\frac{1}{2}} \tag{4}$$

Assuming K cluster, spectral clustering uses the top K eigenvalues of L and performs clustering on the associated normalized eigenvectors of L in a lower dimensional space using k-means approach. The clustering outcome commonly calls for two input parameters: (1) the number of clusters and (2) a scaling factor that depends on the context of the analysis. In order to enable automated clustering, we have proposed heuristics to identify these two parameters. For the former, we have introduced the concept of *iterative eigengap search* (IES) that partitions a feature space using a search tree structure to reveal eigengaps at different scales

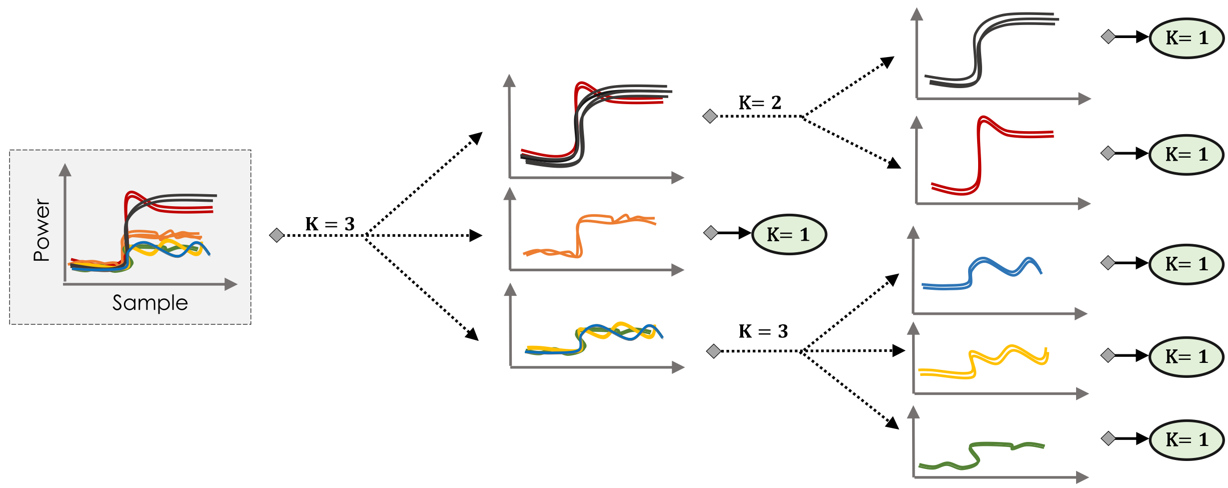

Fig. 4. The visualization of the IES for appliance signature clustering: The root nodes include the entire dataset (with 6 clusters) which is iteratively partitioned with eigengap. The leaf nodes ($K = 1$) are accepted as the final clusters.

of the feature space. For the latter, in order to learn the scaling factor from the data itself, we have proposed a principal component analysis- (PCA-) based quantification of scaling factor at different scales of the feature space. Scaling factor in this context is a parameter that controls the width of the neighborhood [69], and the value of σ defines the reference distance between connected data points within the same scale. In other words, σ is a contextual reference distance that defines if two data points should be considered similar. The outcome of the clustering process is the groups of feature vectors that represent similar operational states of appliances in the target environment. Since collecting the cluster library is a core part of the framework, we have discussed the methodology in this section (more details on the proposed automated clustering could be found in [67]).

Iterative Eigengap Search (IES) Heuristic: Selecting the right number of clusters (K) is challenging and subjective in many application domains. In the case of load monitoring, K corresponds to unique and observable appliance state transitions (i.e., appliance signatures), in a given environment, for which the number of appliances as well as the number of transition states for finite state machines is not known. In spectral clustering, eigengap, a well-known heuristic, could be used for automated evaluation of the number of clusters (K) [69,70]. Let λ_i be the eigenvalues of a Laplacian matrix (calculated based on pairwise similarity distance of feature vectors in a given dataset). Eigengap δ_i is estimated through:

$$\delta_i = |\lambda_i - \lambda_{i+1}|, \ \ i = 1, \ldots, \ n - 1 \tag{5}$$

where n is the total number of feature vectors. The number of clusters (K) can be estimated by:

$$K = argmax_i(\delta_i) \tag{6}$$

In Eq. (6), the aim is to select the largest gap between kth and $(k+1)$th eigenvalues derived from the Laplacian matrix for the selection of the number of cluster. The eigengap heuristic justification has been outlined in the literature based on perturbation theory and spectral graph theory [69,71]. Although eigengap is a measure for estimating the number of clusters, it typically works well in datasets with well-separated feature spaces [69]. However, for the appliance signature features with a multi-scale and noisy nature, the resultant number of clusters is not accurate. Specifically, the differences between the signatures of appliances with smaller transient power draws are masked by signature with larger power draws resulting in clustering the signatures from different appliances and states into one cluster. Therefore, we proposed the *Iterative Eigengap Search (IES)* to search for the eigengap at different scales of the feature space and overcome the challenges of scale effect in dissimilarity quantification.

The IES uses a recursive search on a search tree structure as schematically illustrated in Fig. 4. At the first step, the entire dataset of feature vectors (root node in Fig. 4) is processed by spectral clustering (NJW algorithm [68]) and eigengap heuristic. The process continues recursively by clustering the data at each node of the tree. This process is continued until the eigengap cannot further segregate the data in any of the leaf nodes (i.e., K is estimated as 1 based on Eq. (6)). Once the clustering of all nodes is completed, the clusters at the leaf nodes will constitute the final clusters. As noted in developing the similarity matrix, a scaling parameter is required to identify the boundaries of the similarity neighborhood. In order to estimate the scale of neighborhoods in an automated manner, we employed the PCA, which deploys orthogonal transformation to map the original features into a new space with uncorrelated variables. Given that PCA sorts the transformed variables based on the maximum variance within the data, we employed PCA to account for the most variance from the major principal axes as the estimation of σ^2. This will enable us to define an approximated boundary threshold for the formation of the similarity matrix and to distinguish the similar and dissimilar data points according to their distribution at each generated node of the tree. Therefore, the value of the scaling parameter will be different at each node and will be updated to consider the similarity neighborhood of that specific subset of data points. By using this approach, although the eigengap might not initially predict the right number of clusters, it can facilitate segregating the appliance signatures into different clusters that (likely) reside in different scales, which are characterized by different power draws.

Cluster quality assessment: the task of clustering aims at obtaining high intra-cluster similarity (dense clusters) and low-inter cluster similarity (well-separated clusters). However, a number of challenges may affect the quality of appliance signature clusters which in turn may affect the event detection performance. First, due to the uncertainty in appliance signature characteristics (e.g., power draws) and the inherent challenges involved in clustering, some of the clusters may contain signatures that are distant from each other. Second, as noted earlier, the conventional event detectors are prone to detect false positives, whose feature vectors will be processed during the clustering. Accordingly, it is required to mitigate the impact of such undesired effects by examining the quality of clusters before extracting the recurring motifs. Without such consideration, some of the motifs might not represent an

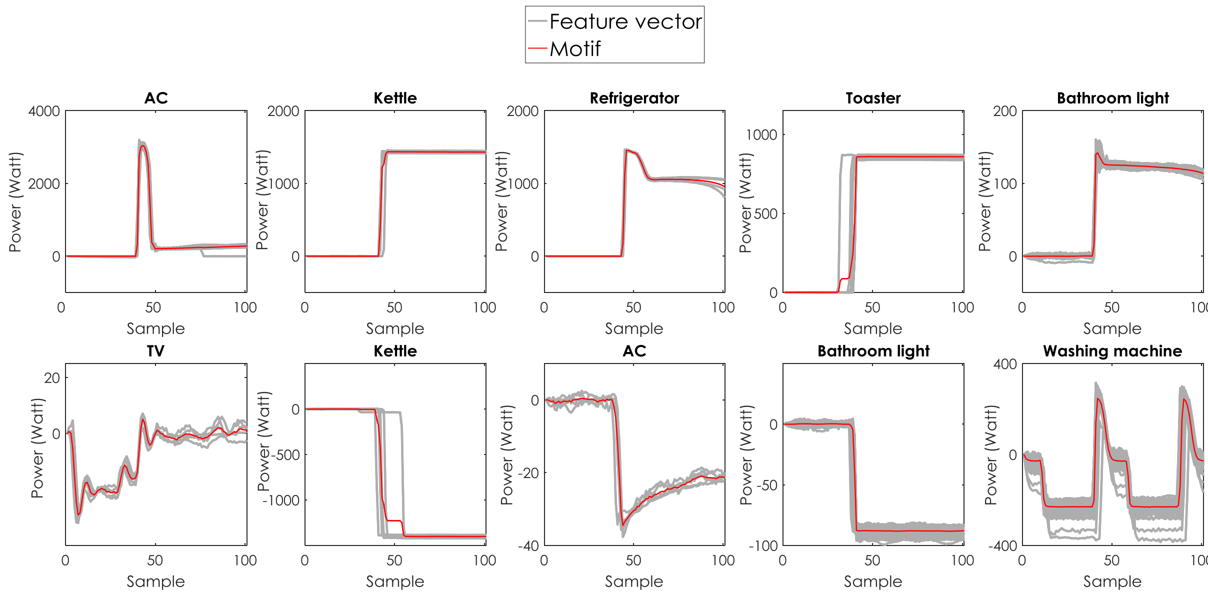

Fig. 5. Examples of feature vectors, clusters, and their associated motifs (only real power component was shown). Each feature vector contains 100 samples (corresponding to a duration of ∼1.7 s).

appliance state change in the environment. To this end, we have used a dispersion measure for cluster quality assessment [72]. Assuming m feature vectors in a cluster, the dispersion measure for cluster k (DM_k) is calculated as follows:

$$DM_k = \max dist\left(\boldsymbol{fv_i}, \boldsymbol{fv_j}\right), \ \ i, j \in 1 : m \tag{7}$$

where $dist$ ($\boldsymbol{fv}_i$, $\boldsymbol{fv}_j$) is the pairwise Euclidean distance between $\boldsymbol{fv}_i$ and $\boldsymbol{fv}_j$. According to our empirical observations, a cluster might less likely represent a recurring motif if the following condition holds [72]:

$$DM_k > S_k + \sigma_k \tag{8}$$

in which S_k and σ_k are the mean and standard deviation of the $dist$ ($\boldsymbol{fv}_i$, $\boldsymbol{fv}_j$) across all the feature vectors in cluster k. In evaluating the cluster quality, feature vectors were normalized by dividing each element by the absolute maximum value in each feature vector. Accordingly, clusters which satisfy Eq. (8) are eliminated from the motif extraction step. Other variations of Eq. (8) can be employed ($DM_k > S_k + t * \sigma_k$). However, the adjustment of t could either result in accepting more clusters, some of which might be affected by cluttered observations that do not reflect a real appliance state change (in case of t>1) or reducing the number of clusters (in case of t<1). Therefore, through empirical assessment, we used Eq. (8) based on its efficacy in selecting the clusters with useful information.

Motif library population: to extract the motifs that represent an appliance state transition, we have used the centroid vector (mean of the feature vectors) within each cluster (M_k). These motifs (M_k) play an important role in characterizing activities, human-appliance interactions, and energy consumption in an environment. The number of the motifs depends on the number of appliances, the complexity of their operational states, and the timeline of the initial data buffering. Fig. 5 illustrates examples of several extracted motifs on a sample dataset. The corresponding appliance label for each motif has been provided as well.

Given that the motif identification process could be carried out over a few days and get updated on a regular basis, the potential false negatives from the conventional event detectors will not affect the motif extraction as numerous events representing each motif are often observed in an environment.

3.2. Proximity-based event detector

As the description of the framework implies, this approach uses a proximity-based technique such as nearest neighbor for motif-based event detection. Therefore, the approach is a semi-supervised classification problem. However, to control the false positives, an outlier detection step is required for accepting or rejecting an event considering that the nearest neighbor classifier will associate every new observation with one of the motifs. Therefore, the motif-based event detection process is as follows:

– For each sample point on the power time-series, extract a feature vector (P_i) following Eq. (2). We call this feature vector the *power signature shape*.
– Identify the closest motif to the power shape using a 1NN algorithm.
– Run an outlier detection to accept or reject the power shape as a viable event.

As the process shows, the proposed framework could combine the process of event detection and classification. Therefore, upon detection of an event, if the clusters are labeled (as shown in Fig. 5) with the name of appliances in the physical environment, the load identity is also revealed that in turn could be used for other applications such as human-appliance interaction monitoring, occupancy detection, or energy consumption assessment.

In extracting the power signature shapes, similar contiguous windows of pre-event samples (w_{pre}) and post-event samples (w_{post}), as used for motif extraction, are employed. By sliding these two windows along the time-series, for each sample point, a representative power signature shape (P_i) is extracted in the form of a feature vector. These feature vectors are normalized to have a zero value at the intersection of the feature extraction windows. The index for the closest motif (k_i^*) to each power signature shape (P_i) is identified using the 1-NN algorithm such that

$$k_i^* = \arg\min \mathrm{dist}(P_i, M_k\ ), k \in 1 : K \tag{9}$$

Outlier detection: As noted, the objective of this outlier detection is to evaluate a binary hypothesis to decide if P_i is coming from the same distribution that motif k_i^* represents. There exist different methods of outlier detection that could be used for this purpose such as comparing the distance of a new observation from the motifs (e.g., using Mahalanobis distance that accounts for the shape of signatures as well) or two-class support vector classifiers. All these methods either call for a threshold value or training process to be used for hypothesis testing. Processing the motifs through clustering, not only provides the recurring motifs but also provide an insight into the stochastic nature of the power draw at different times. The variations of the feature vectors in each cluster provide the ground for learning the outlier detection thresholds from the environment itself. Given that the clustered feature vectors represent a dense set of data points, the clustered motifs do not include outliers. This fact enables us to identify a range of acceptable thresholds for each cluster to measure the similarity of power shape P_i with the corresponding motif k_i^*. Therefore, we have formulated the event detection in this context as an outlier implementation that uses a distance metric for evaluating the hypothesis. To this end, we have used the following criteria for hypothesis testing and detection statistics. A sample point i is flagged as an event if the following similarity criterion is met:

$$\boldsymbol{D}_i < \mu_{k_i^*} + f\sigma_{k_i^*} \tag{10}$$

where $\boldsymbol{D_i}$ is the distance between the motif (k_i^*) and the power signature shape (P_i); $\mu_{k_i^*} + f\sigma_{k_i^*}$ indicates the *similarity threshold* for each motif and is learned from the content of each cluster; $\mu_{k_i^*}$ and $\sigma_{k_i^*}$ are the mean and standard deviation of the distances between the motif and each feature vector in the cluster k_i^*, and f is the parameter that controls the flexibility of the boundaries in each cluster. In this outlier detection, we have adopted the Frechet distance [73], which is defined as follows:

$$\boldsymbol{D_i} = \boldsymbol{d}\left(P_i, \mu_{k_i^*}\right) = inf\ max_{\alpha,\ \beta\ t\in[0,1]}\{\| P_i(\alpha(t)) - \mu_{k_i^*}(\beta(t)\ \|\} \tag{11}$$

In which $\|.\|$ denotes the L_2 norm and α, β: $[0,\ 1] \rightarrow [0,\ 1]$ span over all continuous increasing functions. The Frechet distance (FD) measures the similarity between time-series segments by considering the order and location of data points in the shape of the signatures, and therefore suitable in our implementation as it considers the continuity of shapes into account for distance measurement. This will make FD a more effective measure for our purpose compared to the widely used distances such as Euclidean or Mahalanobis distance that employ one-to-one mapping for calculating the distance. The pseudocode of the proposed event detection algorithm is as presented in Fig. 6.

For each sample point i,

$$P_i = S(i - w_{pre} - 1 : i + w_{post} + 1)$$

$$M_{k_i^*} \leftarrow 1NN\ (P_i, \boldsymbol{M})$$

$$\delta_i = |S_{i+1} - S_i|$$

$$\psi_i^* = FD(M_{k_i^*}, P_i)$$

If $\delta_i > \tau_\delta$ and $\psi_i^* < \mu_{k_i^*} + f\sigma_{k_i^*}$

$$E_p \leftarrow i$$

End

End

Fig. 6. Pseudo code for the motif-based event detection algorithm.

To visually demonstrate the underlying steps involved in the proposed event detector, Fig. 7 shows the process for one sample point that was detected as an event. Part (a) reflects the data buffering and motif processing to populate the motif cluster library and identify the parameters for outlier detection for each cluster (self-training process). The outcome of this part is used for the event detector in part (b), applied to all the samples for the power time-series. In this figure, steps I and II indicate the feature extraction on the buffered data. The collected features are passed to clustering (step III). In step IV, the quality of clusters is assessed. The centroids of remaining clusters were collected as motifs (step V). By calculating the statistics (average, μ_k, and standard deviation, σ_k) for observations within each cluster, similarity thresholds (step VI) for each motif is calculated (defined in Eq. (10)). In part (b), for each power sample (step VII), a power signature shape (step VIII) is extracted. Using 1NN, the closest motif to the power signature shape is determined. In step (X), the distance of the closest motif (obtained in step IX) is compared with the associated similarity thresholds (from step VI). Since the equation (Eq. (10)) holds in this case, the power shape for sample i is considered similar to the motif shown in step (X) and marked as an event. If the motifs were labeled as well, the load identity, i.e., kitchen light turn-on in this case, is revealed as well.

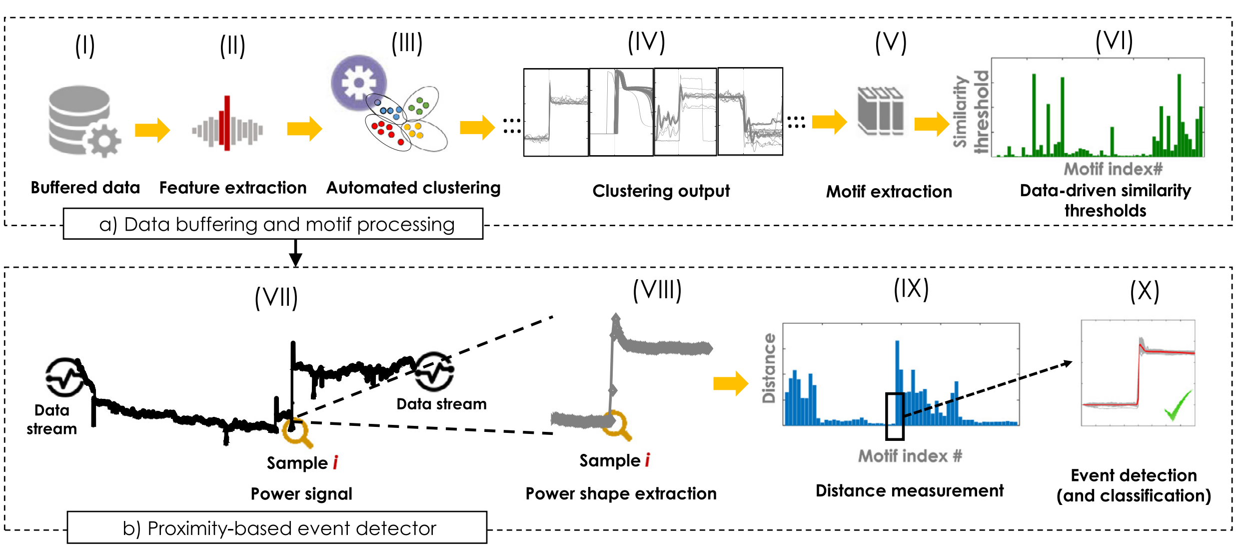

Fig. 7. Schematic diagram for the appliance self-configuring event detector using real data.

Table 1
Properties of the EMBED dataset [38].

Dataset	No. of events	No. of appliances	No. of classes[a]
Apt 1	~4400	16	66
Apt 2	~9100	20	62
Apt 3	~7800	18	68

[a] No. of classes represent the number of appliance state transitions.

4. Evaluation and results

4.1. Dataset description

To evaluate the performance of the proposed approach, we applied the algorithm to a real-world dataset of electricity disaggregation. The EMBED dataset [38] includes the aggregate power time series, collected from three apartments at the main circuit panel of each unit, for different periods varying from two to four weeks in Los Angeles, CA. The data has been fully labeled by leveraging ground truth sensors (both electricity and light) that were installed at the consumption node, and contains the timestamps as well as the corresponding appliance labels and sub-labels for different operational states of appliances. In our analysis, we have used the processed power time-series (real and reactive) at 60 Hz, which enables capturing the transient shapes between steady states. In the US, the electricity infrastructure uses a split-phase system that feeds appliances on two major circuits (i.e., phases). Therefore, in our analysis, we have pointed to performance on different phases (A and B). Table 1 shows the characteristics of appliances operations in different apartments. Details of the data collection and post-processing could be found in Jazizadeh et al. [38] and therefore were not presented here.

4.2. Performance evaluation

To characterize and quantify the performance, we have utilized the commonly used evaluation metrics [21,74] of precision, recall, and F-measure. In order to simulate the data buffering process in forming the benchmark motifs, we have divided the power data into two train and test subsets using a 75–25 ratio, respectively. It is intuitive to have a separate portion of the data for populating the motif library to avoid over-fitting and matching the power signature shapes to a motif that was already extracted from the same section. A pre-event window (W_{pre}) of size 40 (i.e., two third of a second) and a post-window (W_{post}) of size 60 (i.e. one second) were used in feature extraction. In classification studies, it has been observed that the detection performance is not highly affected by the size of the windows. A general rule of thumb for determining the size is to consider that signatures do not overlap and the transient information perseveres. Moreover, considering the fact that the size of the window for both benchmark motif identification and event detection is the same, the window size effect will be minimized. In buffering the data, GLR event detection algorithm was adopted from [53]. Leveraging the assessments in the literature, a general set of parameters for the GLR algorithm was used [75]. No specific effort was made to ensure that the GLR algorithm is tuned to its better level of performance. The data for the training subset was passed through the clustering algorithm to create the clustered motifs. An empirical value of $f = 50$ was considered for characterizing the thresholds for each cluster in Eq. (10). This value was identified through a sensitivity analysis over different values of f. A vector of $f = \{1, 5, 10, 25, 50, 75, 100\}$ was tested for the data set from Apt 1-Phase A. Fig. 8 shows the F-score results for different f values.

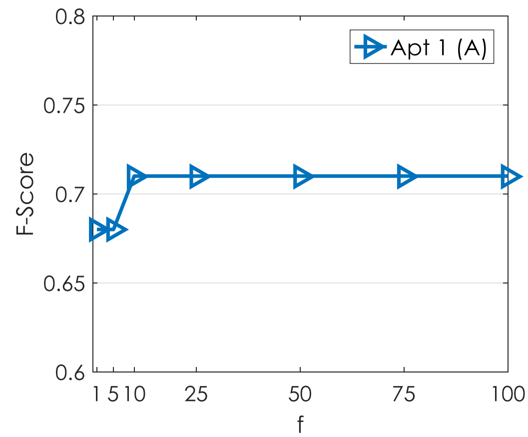

Fig. 8. Sensitivity analysis for identifying the f value on phase A of Apt 1.

4.2.1. Event detector evaluation

The performance of the proposed framework against the conventional GLR algorithm was assessed on the test data subset. The true positives (TP), false positives (FP), false negatives (FN), precision, recall, and F-score were used as the quantified performance metrics. True negatives (TN) are not presented since events are sparse on the power time-series, and the number of events is significantly lower compared to the number of instances on steady-state segments. Therefore, any event detector can achieve a very high number of TN, and the metric is not able to justify the performance. In order to quantify the metrics, $a \pm 3$ sample points (less than 0.1 s) was considered as the tolerance in the comparison between ground truth and predicted events. Table 2

Table 2
Evaluation results for the proposed event detection, compared to GLR.

Event Detection Method	Dataset	# of events	TP	FP	FN	Precision	Recall	F-Score
Motif-based (Self-configuring)	Apt 1 (Phase A)	337	229	61	108	**0.79**	**0.68**	**0.73**
	Apt 1 (Phase B)	1075	981	19	94	**0.98**	**0.91**	**0.95**
	Apt 2 (Phase A)	665	659	63	6	**0.91**	**0.99**	**0.95**
	Apt 2 (Phase B)	1016	764	381	252	0.67	0.75	0.71
	Apt 3 (Phase A)	1734	1497	717	237	**0.68**	0.86	0.76
	Apt 3 (Phase B)	304	296	164	8	**0.64**	0.97	**0.77**
	Average					**0.78**	**0.86**	**0.81**
	Standard deviation					0.14	0.12	0.11
GLR	Apt 1 (Phase A)	337	219	106	118	0.67	0.65	0.66
	Apt 1 (Phase B)	1075	529	23	546	0.96	0.49	0.65
	Apt 2 (Phase A)	1128	659	469	469	0.58	0.58	0.58
	Apt 2 (Phase B)	1016	925	423	91	**0.69**	**0.91**	**0.78**
	Apt 3 (Phase A)	1734	1608	881	126	0.65	**0.93**	0.76
	Apt 3 (Phase B)	304	299	451	5	0.40	**0.98**	0.57
	Average					0.66	0.76	0.67
	Standard deviation					0.18	0.21	0.09

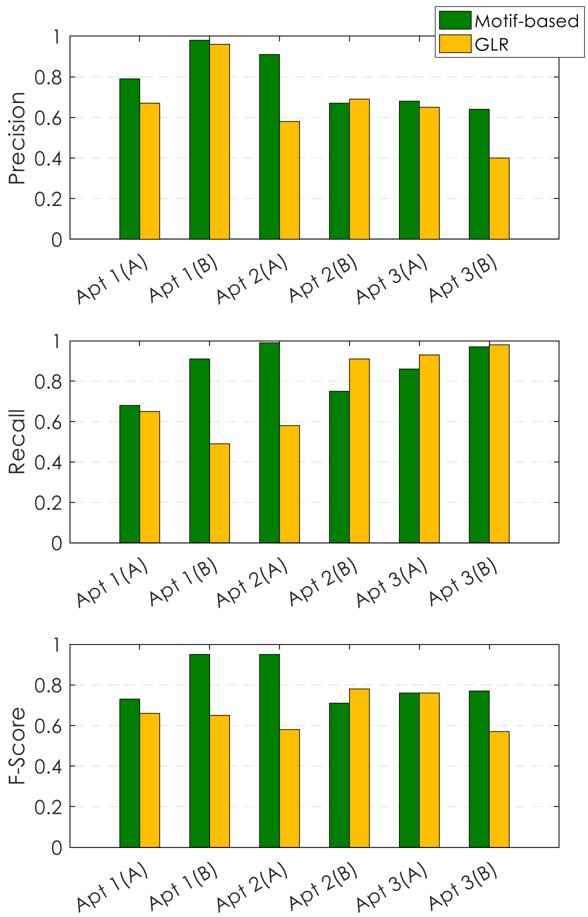

Fig. 9. Comparison of performance evaluation metrics for motif-based versus GLR event detectors.

shows the results of the evaluation. In this table, for the motif-based approach, events for the associated appliance classes that have been operated at least once in the training stage were considered for the evaluation. The higher values (i.e., better performance) for precision, recall, and F-score have been highlighted in bold.

As the results show, the proposed motif-based event detector had a promising performance without the need for parameter-tuning or a priori information for the clustering. Particularity, by using the motif-based event detection, an average F-score of 0.81 across all datasets was obtained that is higher compared to the benchmark GLR performance of 0.67. Moreover, the average value of precision (trade-off between true and false detection), and recall (trade-off between true and missed detection) is 0.78 (compared to 0.66 for GLR) and 0.86 (compared to 0.76 for GLR), respectively. The motif-based approach maintains a better balance for these metrics, considering the standard deviation across all datasets. Fig. 9 summarizes the performance metric results.

As the visual illustrations of the performance metrics, shown in Fig. 9, the motif-based approach outperforms GLR with higher precisions across different environments. This could be mainly associated with the fact that the motif-based approach reduces the false positives in event detection. However, comparison of the recall values shows an interesting trend. In some cases, the GLR shows a better recall as it is generally more sensitive the changes. However, this increase comes with a cost of reduced precision. This trade-off is better balanced by the proposed motif-based event detection.

To provide a better context for the performance of the proposed approach, in Fig. 10, we have presented visualizations of the detected events on samples of the power time series for a variety of appliances. Short sections of power time-series (less than 3 h) have been provided for each dataset for visual interpretation. The corresponding motif labels were employed for event classification as well. TP, FP, and FN are depicted based on the ground truth data.

As these graphs and quantitative evaluations show, the motif-based approach has also resulted in a number of false positives and negatives. From this limited observation, it appears that the congested areas (with multiple repeated cycles) could result in higher false negatives. Moreover, it could be seen that the missing events (i.e., false negatives) appear to have lower power draws. Depending on the application domain, the importance of detecting the operational events for appliances might vary. Therefore, we have also evaluated the performance of the motif-based event detection according to the transient power draw values for appliances. In doing so, we have illustrated the trade-off between true detection (TP) and missed detection (FN) for different power ranges in Fig. 11. As this figure shows, missed detections are mainly attributed to events that have a lower power draw, less than 100 W, and only in one case (Apt 1, phase A), some of the very large power draw events (more than 1000 W) are not detected. The parameters of the conventional event detector for populating the motif library could play a role in such observations. For example, the GLR could be set to ignore low power variations in consecutive samples to avoid false positives due to noise interference. In such scenarios, the motifs from the low power draw appliances might not be retrieved and used in motif-based event detection. In this study, we have set the minimum change in power to be equal to 25 W.

4.2.2. Appliance-level evaluation

The evaluation of the performance from the appliance type perspective is important as this knowledge could affect the performance of the end-use applications. Appliances with major power draws have a more significant role in quantifying the energy consumption, and therefore their associated events are of greater importance in that context. For example, air conditioning and heating system events mainly have higher power draws with a longer period of operations, which make them important for energy consumption assessment ([76,77]). On the other hand, events for appliances with lower power draw play a less important role in energy consumption assessment but they are important in inferring human activities and human-appliance interactions. If low-power events happen frequently and sustain long intervals of operations, their cumulative impact could be also considerable for energy consumption. For example, identifying events for some appliances such as electric range can be tangibly tied to occupants' activity (e.g., cooking). Specifically, identifying occupants' activity can provide context-aware applications in the buildings. Similarly, fine-grained monitoring of events for several multi-state appliances like the refrigerator or dishwasher can provide an insight for fault detection or demand response opportunities for the residential sector [78].

To provide an insight on the performance for different categories of appliances, we measured the number of true (TP) and missed detection (FN) for each appliance type and presented the results in Table 3. To this end, for each label type in the ground truth data, we compared the timestamps between ground truth and predicted events. In this table, the 3-digit labels represent the appliance type according to the EMBED dataset. Some appliances have only turn-on and turn-off events (e.g., lights), while others have different transition states (e.g., Air conditioning (AC) systems).

Fig. 12 illustrates the ratio of the correctly detected events for each appliance type (refer to Table 3 for label interpretation). As the results show, for apartment 1, appliances like AC, washing machine, bathroom light (which operates a fan as well), iron, and toaster have high detection accuracy, while TV, laptop, and kitchen light, all with low power draw, are hard to detect. For apartment 2, except for TV and the unknown light, the event detector achieves high accuracy for a variety of appliances like AC, range, water heater, and grill. For apartment 3, except for the laptop that was failed to be recognized, closet light, unknown light, TV, and hair iron showed an average performance. However, the event detector shows a nearly perfect detection rate for the kettle, toasters, dishwasher, AC, dryer, and some of the lights. As can be seen from Fig. 12, the event detector has a promising performance for a variety of different appliances. However, for the ones with considerably low power draws (less than 100 W), the inevitable impact of noise and artifact could be reflected into the shape of appliance signatures, which results in the reduced performance in the clustering procedure (Section 3.1.3) or the proximity-based event detector (Section 3.2).

In the presented analyses, it was assumed that the event detector was evaluated at a time period when no new appliance was added to the environment. If new appliances (with different signatures) are added to a house, their corresponding characteristics need to be added to the library. To tackle this problem, motif processing could be performed regularly in case new appliances are added to the environment to capture their characteristics. In addition, the detection of the appliance type that causes an event is an important step following the event detection for all applications of human-appliance interaction. This process could be carried out by using a classification algorithm that uses a training dataset for inferring the appliance type. Our proposed approach leverages motifs of appliance signatures from similar sets of observations associated with the generated clusters. Therefore, clusters could be used as the training data set for the aforementioned classifier considering that the clusters are labeled by the actual identity of the appliances.

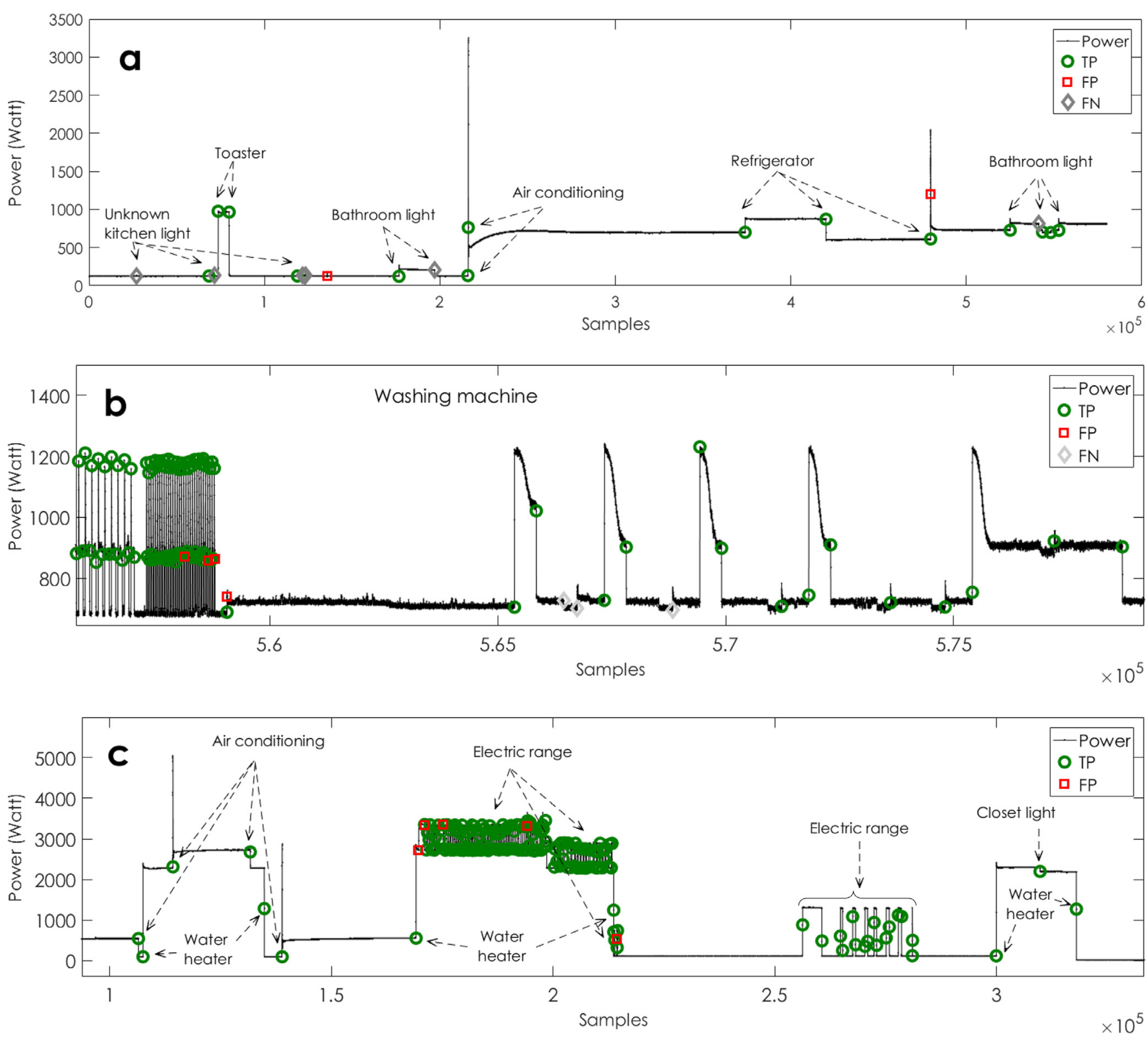

Fig. 10. Detected events using motif-based approach on samples of power time-series from: (a) Apt 1, phase A, (b) Apt 1, phase B, (c) Apt 2, phase A, (d) Apt 2, phase B, (e) Apt 3, phase A, (f) Apt 3, phase B.

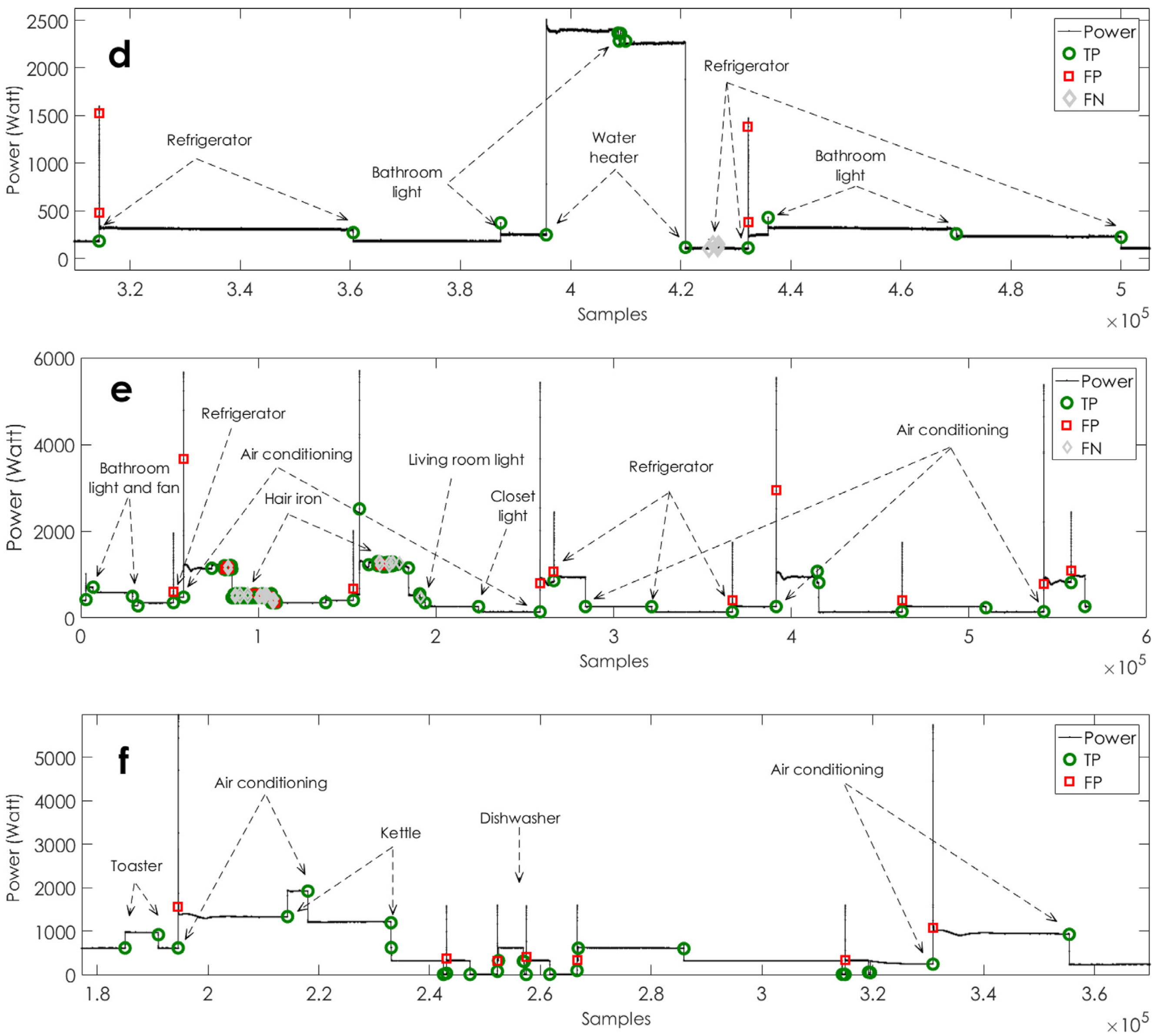

Fig. 10. Continued

We have also evaluated the impact of f on the performance of the algorithm in different environments as illustrated in Fig. 13. As shown, a value of $10 < f < 100$ leads to a relatively consistent performance, and the value of f in the selected range does not have an impact on the performance in different contexts.

4.3. Limitations

Although the proposed approach has shown an overall promising performance, there are a number of limitations to be addressed: First, the detection rate for some of the low power-draw devices including laptop, TV, and lights was observed to be lower. Fig. 12 of illustrates these observations. In general, the detection of the operations for low power-draw appliances in the presence of appliances with higher power-draw and in the context of high-resolution electricity signal could be challenging. In fact, developing techniques for detection of miscellaneous electrical loads (MEL) [79], is a topic of research that the Department of Energy has prioritized in the past few years. Second, identifying the status of the appliances is dependent on the detection of on/off events. Similar to other related studies in this domain, missing an on or off event can lead to the wrong estimation of the operational status for different applications. Given that the initial dataset (i.e., buffered data set) is populated through a conventional event-detection algorithm, the outcome of the clustering and the cluster quality analysis process could affect the detection of the events through motif-based event detection. Nonetheless, from the energy perspective, Fig. 11 of the manuscript shows that for events with higher transient power ranges (>200 W), the correct detection rate is high. Third, in our study, it is assumed that after identifying the recurring motifs from the buffered data during the self-training stage, the appliances inside the house remain the same. However, in case of adding a new appliance to the house, it is required to allow time for the motif-based approach to account for the occupant interaction with the new device, in order to learn the contextual information of the newly added load and to update the motif library, accordingly. This also holds true for appliances that have not been used during the data buffering stage. Fourth, in case of having simultaneous events (operational state transitions of different appliances with very close time difference), the shape of the feature

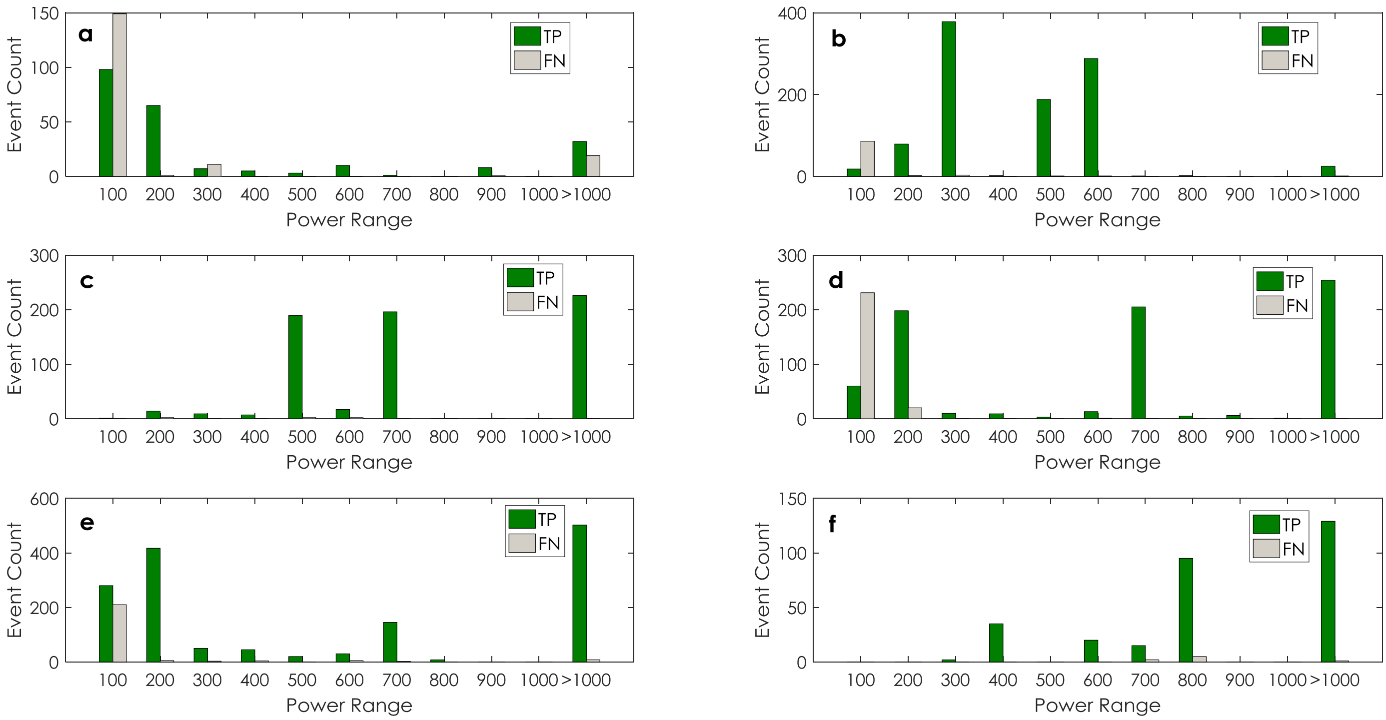

Fig. 11. Histogram of true positives versus false negatives for the motif-based event detector for different power ranges: (a) Apt 1, phase A, (b) Apt 1, phase B, (c) Apt 2, phase A, (d) Apt 2, phase B, (e) Apt 3, phase A, (f) Apt 3, phase B. The values on the horizontal axis indicate the upper bound for the power range. For example, a value of 100 indicates a power range of 0–100 W.

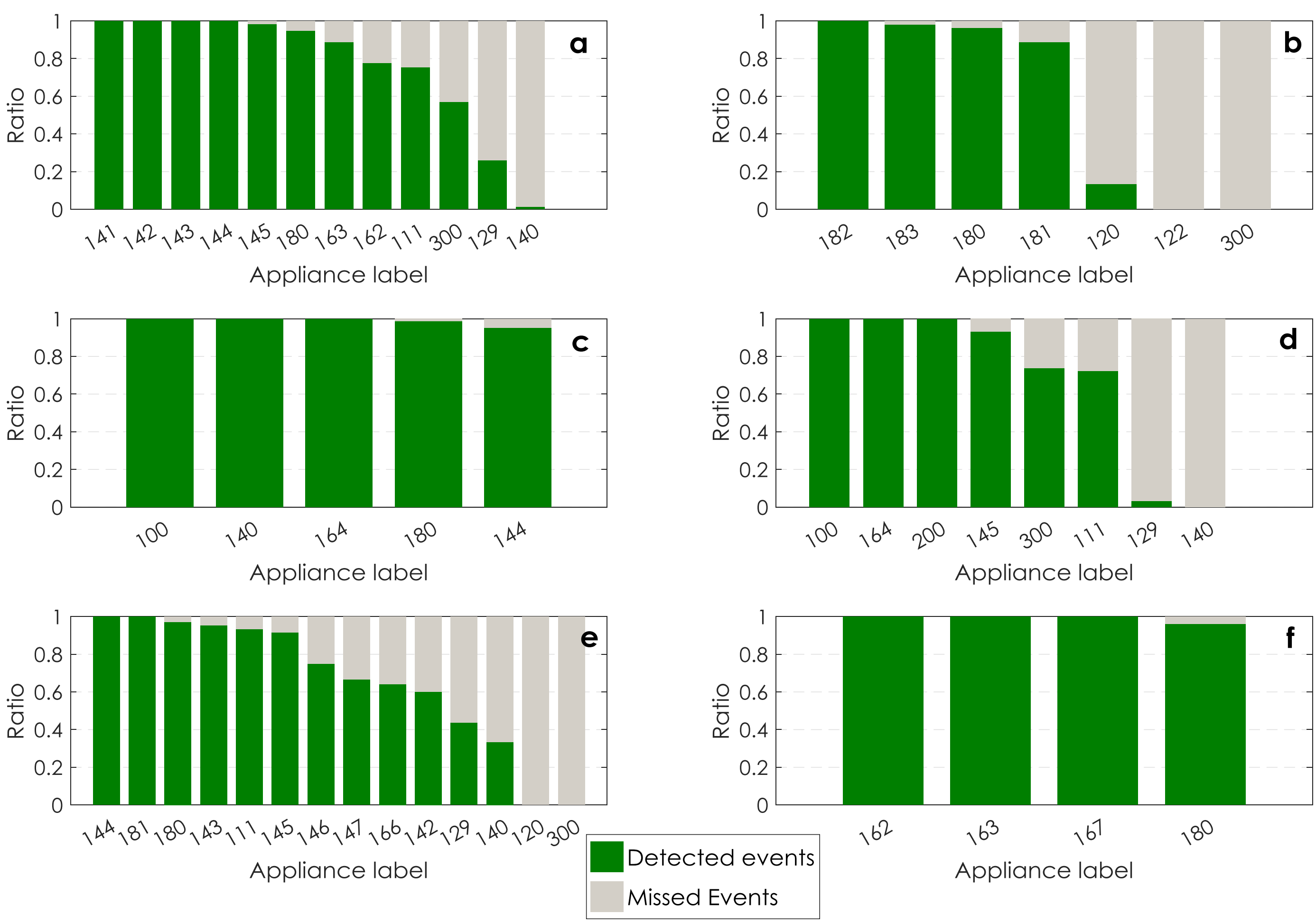

Fig. 12. The distribution between TP and FN rates for detected events for each appliance type: (a) Apt 1, phase A, (b) Apt 1, phase B, (c) Apt 2, phase A, (d) Apt 2, phase B, (e) Apt 3, phase A, (f) Apt 3, phase B.

Table 3
Detection rate from appliances' perspective for the test part of the dataset.

Dataset	Label	Appliance	# of events	True Positive	False Negative
Apt 1 (Phase A)	111	Refrigerator	65	49	16
	129	TV	96	25	71
	140	Unknown kitchen light	77	1	76
	141	Kitchen light 1	5	5	0
	142	Kitchen fan light	8	8	0
	143	Kitchen light 2	2	2	0
	144	Bathroom light 1	12	12	0
	145	Bathroom light 2	61	60	1
	162	Kettle	9	7	2
	163	Toaster	9	8	1
	180	Air conditioning	38	36	2
	300	Unknown	28	16	12
Apt 1 (Phase B)	120	Laptop	15	2	13
	122	LCD monitor	6	0	6
	180	Air conditioning	27	26	1
	181	Hair dryer	9	8	1
	182	Iron	14	14	0
	183	Washing machine	948	931	17
	300	Unknown	56	0	56
Apt 2 (Phase A)	100	Electric range	217	217	0
	140	Bathroom light	6	6	0
	144	Closet light	21	20	1
	164	Water heater	27	27	0
	180	Air conditioning	394	389	5
Apt 2 (Phase B)	100	Electric range	174	174	0
	111	Refrigerator	455	329	126
	129	TV	63	2	61
	140	Unknown light	46	0	46
	145	Bathroom light	131	122	9
	164	Water heater	34	34	0
	200	Grill	75	75	0
	300	Unknown	38	28	10
Apt 3 (Phase A)	111	Refrigerator	869	810	59
	120	Laptop	12	0	12
	129	TV	16	7	9
	140	Unknown Light	6	2	4
	142	Closet light	10	6	4
	143	Kitchen light	44	42	2
	144	Living room light	19	19	0
	145	Bathroom light and fan	107	98	9
	146	Bedroom lamp	20	15	5
	147	Living room lamp	6	4	2
	166	Hair iron	334	214	120
	180	Air conditioning	277	269	8
	181	Hair dryer	11	11	0
	300	Unknown	3	0	3
Apt 3 (Phase B)	162	Kettle	24	24	0
	163	Toaster	36	36	0
	167	Dishwasher	33	33	0
	180	Air conditioning	211	203	8

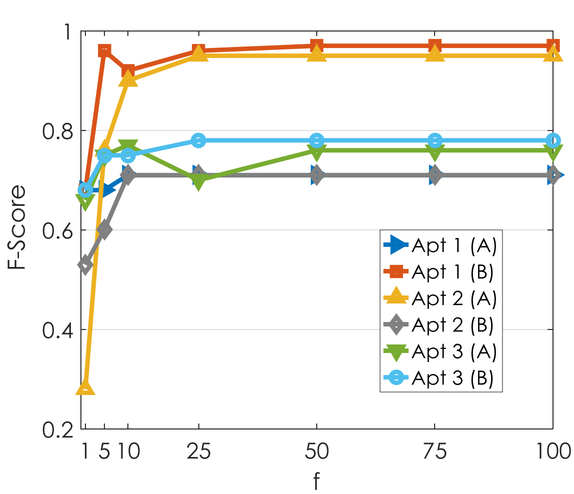

Fig. 13. Analysis of the impact of *f* value across different datasets.

vectors will be impacted by the simultaneous events, which can also affect the clustering and the motif-based event detection step. However, considering the fact that the length of the feature vectors was set to a limited duration that preserves the transient shape (less than 2 s), the chance of observing a large number of simultaneous events in practice will be low.

5. Conclusion

Fine-grained monitoring of operations of the household appliances through power time-series requires the knowledge on the timing of events. However, event detectors typically rely on models that are configured to a specific environment, in which they are deployed. In this paper, we present the shift from event detection based on abrupt changes in time-series of representative power metrics to a motif-based approach. Motifs are represented by clustered signatures, which represent the transient power draws due to change in operational states of appliances in an environment. Motif-based event detection facilitates the self-configuration of algorithms in a new environment. The realization of the proposed approach calls for data-driven techniques that automatically populate a motif library in an unsupervised manner and use a similarity threshold for comparison between clustered motifs and the new observations. The framework in this study uses a combination of a proposed automated spectral clustering, 1NN classifier, and an outlier detection based on the Frechet distance. The outlier detection autonomously learns its parameters from the clustered signatures in an environment that represent the motifs. The evaluation of the framework power time series on EMBED dataset's three apartments over two weeks demonstrated a promising performance. Overall, the proposed algorithm outperforms the GLR algorithm without the need for detection statistics parameter-tuning in new environments.

As described in this study, the motif-based approach leverages a dense set of observations in the form of a cluster to identify motifs and quantify the event detection thresholds for different clusters. Therefore, the new detected events are associated with a set of observations from the environment that either reflects the automatic change in the operational state of an appliance or a change that is caused by occupants. Accordingly, the framework also provides the ground for more efficient communication with the users in an environment for learning the activities. The framework could be used for facilitated training of a machine learning framework that identifies when different appliances are being used without the need to ask for several inputs for one appliance. Therefore, leveraging the proposed framework for facilitated training of machine learning frameworks that infer the disaggregated energy consumption of appliances or associating the energy consumption to activities of occupants comprise the future direction of this research.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.enbuild.2019.01.036.

References

[1] EIA. (2017). Electricty Explained-Use of Electricty. Available: https://www.eia.gov/energyexplained/index.php?page=electricity_use.
[2] J. Froehlich, E. Larson, S. Gupta, G. Cohn, M. Reynolds, S. Patel, Disaggregated end-use energy sensing for the smart grid, IEEE Pervasive Comput. 10 (1) (2011) 28–39.
[3] R.G. Pratt, et al., The Smart Grid: An Estimation of the Energy and CO_2 Benefits, Pacific Northwest National Laboratory (PNNL), Richland, WA (US), 2010.
[4] K.C. Armel, A. Gupta, G. Shrimali, A. Albert, Is disaggregation the holy grail of energy efficiency? The case of electricity, Energy Policy 52 (2013) 213–234.
[5] D. Parker, D. Hoak, A. Meier, R. Brown, How much energy are we using? Potential of residential energy demand feedback devices, in: Proceedings of the Summer Study on Energy Efficiency in Buildings, Asilomar, CA, American Council for an Energy Efficient Economy, 2006.
[6] B. Neenan, J. Robinson, R. Boisvert, Residential electricity use feedback: a research synthesis and economic framework, Technical report, Electr. Power Res. Inst. 3 (2009).
[7] K. Herter, S. Wayland, Behavioral Experimentation with Residential Energy Feedback Through Simulation Gaming, California Energy Commission, 2009 Technical Reports CEC-500-2009-XXXPIER Buildings End-Use Energy Efficiency Program.
[8] S. Darby, The effectiveness of feedback on energy consumption, A Review for DEFRA of the Literature on Metering, Billing and direct Displays, vol. 486, 2006.
[9] C. Fischer, Feedback on household electricity consumption: a tool for saving energy? Energy Effic. 1 (1) (2008) 79–104.
[10] S.N. Patel, T. Robertson, J.A. Kientz, M.S. Reynolds, G.D. Abowd, At the flick of a switch: detecting and classifying unique electrical events on the residential power line, International Conference on Ubiquitous Computing, Springer, Berlin, Heidelberg, 2007, pp. 271–288.
[11] T. Hargreaves, M. Nye, J. Burgess, Keeping energy visible? Exploring how householders interact with feedback from smart energy monitors in the longer term, Energy Policy 52 (2013) 126–134.
[12] E. Hailemariam, R. Goldstein, R. Attar, A. Khan, in: Real-Time Occupancy Detection Using Decision Trees With Multiple Sensor Types, Society for Computer Simulation International, 2011, pp. 141–148.
[13] W. Kleiminger, C. Beckel, T. Staake, S. Santini, Occupancy detection from electricity consumption data, in: Proceedings of the Fifth ACM Workshop on Embedded Systems For Energy-Efficient Buildings, Roma, Italy, 2013.
[14] W. Kleiminger, C. Beckel, S. Santini, Household occupancy monitoring using electricity meters, in: Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, ACM, 2015, pp. 975–986.
[15] D. Chen, S. Barker, A. Subbaswamy, D. Irwin, P. Shenoy, Non-intrusive occupancy monitoring using smart meters, in: Proceedings of the Fifth ACM Workshop on Embedded Systems For Energy-Efficient Buildings, Roma, Italy, 2013.
[16] A. Akbar, M. Nati, F. Carrez, K. Moessner, Contextual occupancy detection for smart office by pattern recognition of electricity consumption data, in: Proceedings of the IEEE International Conference on Communications (ICC), 2015, pp. 561–566.
[17] R. Brown, N. Ghavami, M. Adjrad, M. Ghavami, S. Dudley, Occupancy based household energy disaggregation using ultra wideband radar and electrical signature profiles, Energy Build. 141 (2017) 134–141.
[18] P. Yi, X. Dong, A. Iwayemi, C. Zhou, S. Li, Real-time opportunistic scheduling for residential demand response, IEEE Trans. Smart Grid 4 (1) (2013) 227–234.
[19] K.X. Perez, M. Baldea, T.F. Edgar, Integrated HVAC management and optimal scheduling of smart appliances for community peak load reduction, Energy Build. 123 (2016) 34–40.
[20] M. Zeifman, C. Akers, K. Roth, Nonintrusive appliance load monitoring (NIALM) for energy control in residential buildings: review and outlook, IEEE Trans. Consum. Electron. 57 (1) (2011) 76–84.
[21] M. Zeifman, K. Roth, Nonintrusive appliance load monitoring: review and outlook, IEEE Trans. Consum. Electron. 57 (1) (2011).
[22] S.M. Tabatabaei, S. Dick, W. Xu, Toward non-intrusive load monitoring via multi-label classification, IEEE Trans. Smart Grid 8 (1) (2017) 26–40.
[23] A. Zoha, A. Gluhak, M.A. Imran, S. Rajasegarar, Non-intrusive load monitoring approaches for disaggregated energy sensing: a survey, Sensors 12 (12) (2012) 16838–16866.
[24] A. Cominola, M. Giuliani, D. Piga, A. Castelletti, A.E. Rizzoli, A hybrid signature-based iterative disaggregation algorithm for non-intrusive load monitoring, Appl. Energy 185 (2017) 331–344.
[25] R. Bonfigli, E. Principi, M. Fagiani, M. Severini, S. Squartini, F. Piazza, Non-intrusive load monitoring by using active and reactive power in additive factorial hidden Markov models, Appl. Energy 208 (2017) 1590–1607.
[26] M. Bedir, E. Hasselaar, L. Itard, Determinants of electricity consumption in Dutch dwellings, Energy Build. 58 (2013) 194–207.
[27] R.V. Jones, K.J. Lomas, Determinants of high electrical energy demand in UK homes: appliance ownership and use, Energy Build. 117 (2016) 71–82.
[28] K. Cetin, P. Tabares-Velasco, A. Novoselac, Appliance daily energy use in new residential buildings: use profiles and variation in time-of-use, Energy Build. 84 (2014) 716–726.
[29] G. Flett, N. Kelly, A disaggregated, probabilistic, high resolution method for assessment of domestic occupancy and electrical demand, Energy Build. 140 (2017) 171–187.
[30] L. Stankovic, V. Stankovic, J. Liao, C. Wilson, Measuring the energy intensity of domestic activities from smart meter data, Appl. Energy 183 (2016) 1565–1580.
[31] S. De Lauretis, F. Ghersi, J.-M. Cayla, Energy consumption and activity patterns: an analysis extended to total time and energy use for French households, Appl. Energy 206 (2017) 634–648.
[32] S. Ahmadi-Karvigh, B. Becerik-Gerber, L. Soibelman, A framework for allocating personalized appliance-level disaggregated electricity consumption to daily activities, Energy Build. 111 (2016) 337–350.
[33] A. Faustine, N.H. Mvungi, S. Kaijage, K. Michael, "A survey on non-intrusive load monitoring methodies and techniques for energy disaggregation problem," arXiv preprint arXiv:1703.00785, 2017.
[34] D. Luo, L.K. Norford, S.R. Shaw, S.B. Leeb, Monitoring HVAC equipment electrical loads from a centralized location–methods and field test results/discussion, ASHRAE Trans. 108 (2002) 841.
[35] Y. Jin, E. Tebekaemi, M. Berges, L. Soibelman, Robust adaptive event detection in non-intrusive load monitoring for energy aware smart facilities, in: Proceedings of the ICASSP, 2011, pp. 4340–4343.
[36] H. Azzini, R. Torquato, L.C. da Silva, Event detection methods for nonintrusive load monitoring, in: Proceedings of the PES General Meeting| Conference & Exposition, IEEE, 2014, pp. 1–5.
[37] K. Barsim, R. Streubel, B. Yang, Unsupervised adaptive event detection for building-level energy disaggregation, in: Proceedings of the Power and Energy Student Summt (PESS), Stuttgart, Germany, 2014.
[38] F. Jazizadeh, M. Afzalan, B. Becerik-Gerber, L. Soibelman, EMBED: a dataset for energy monitoring through building electricity disaggregation, in: Proceedings of the Ninth International Conference on Future Energy Systems, ACM, 2018, pp. 230–235.
[39] A. Cominola, M. Giuliani, D. Piga, A. Castelletti, A. Rizzoli, A hybrid signature-based iterative disaggregation algorithm for non-intrusive load monitoring, Appl. Energy 185 (2017) 331–344.
[40] S. Henriet, U. Simsekli, B. Fuentes, G. Richard, "A generative model for non-intrusive load monitoring in commercial buildings," arXiv preprint arXiv:1803.00515, 2018.
[41] B. Kalluri, A. Kamilaris, S. Kondepudi, H.W. Kua, K.W. Tham, Applicability of using time series subsequences to study office plug load appliances, Energy Build. 127 (2016) 399–410.
[42] H.N. Rafsanjani, C.R. Ahn, J. Chen, Linking building energy consumption with occupants' energy-consuming behaviors in commercial buildings: non-intrusive occupant load monitoring (NIOLM), Energy Build. 172 (2018) 317–327.

[43] N. Sadeghianpourhamami, J. Ruyssinck, D. Deschrijver, T. Dhaene, C. Develder, Comprehensive feature selection for appliance classification in NILM, Energy Build. 151 (2017) 98–106.

[44] R. Bonfigli, A. Felicetti, E. Principi, M. Fagiani, S. Squartini, F. Piazza, Denoising autoencoders for non-intrusive load monitoring: improvements and comparative evaluation, Energy Build. 158 (2018) 1461–1474.

[45] L. De Baets, J. Ruyssinck, C. Develder, T. Dhaene, D. Deschrijver, Appliance classification using VI trajectories and convolutional neural networks, Energy Build. 158 (2018) 32–36.

[46] A.S. Bouhouras, P.A. Gkaidatzis, E. Panagiotou, N. Poulakis, G.C. Christoforidis, A NILM algorithm with enhanced disaggregation scheme under harmonic current vectors, Energy Build. 183 (2019) 392–407.

[47] Y. Liu, X. Wang, L. Zhao, Y. Liu, Admittance-based load signature construction for non-intrusive appliance load monitoring, Energy Build. 171 (2018) 209–219.

[48] M. Aiad, P.H. Lee, Non-intrusive load disaggregation with adaptive estimations of devices main power effects and two-way interactions, Energy Build. 130 (2016) 131–139.

[49] M. Aiad, P.H. Lee, Unsupervised approach for load disaggregation with devices interactions, Energy Build. 116 (2016) 96–103.

[50] G.W. Hart, Nonintrusive appliance load monitoring, Proc. IEEE 80 (12) (1992) 1870–1891.

[51] A. Iwayemi, C. Zhou, SARAA: semi-supervised learning for automated residential appliance annotation, IEEE Trans. Smart Grid 8 (2) (2017) 779–786.

[52] L. Farinaccio, R. Zmeureanu, Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses, Energy Build. 30 (3) (1999) 245–259.

[53] M. Berges, E. Goldman, H.S. Matthews, L. Soibelman, K. Anderson, User-centered nonintrusive electricity load monitoring for residential buildings, J. Comput. Civ. Eng. 25 (6) (2011) 471–480.

[54] Y. Jin, E. Tebekaemi, M. Berges, L. Soibelman, Robust adaptive event detection in non-intrusive load monitoring for energy aware smart facilities, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2011, pp. 4340–4343.

[55] L. De Baets, J. Ruyssinck, C. Develder, T. Dhaene, D. Deschrijver, On the Bayesian optimization and robustness of event detection methods in NILM, Energy Build. 145 (2017) 57–66.

[56] H.A. Azzini, R. Torquato, L.C. da Silva, Event detection methods for nonintrusive load monitoring, in: Proceedings of the PES General Meeting| Conference & Exposition, IEEE, 2014, pp. 1–5.

[57] K.S. Barsim, R. Streubel, B. Yang, Unsupervised adaptive event detection for building-level energy disaggregation, in: Proceedings of the Power and Energy Student Summt (PESS), „ Stuttgart, Germany, 2014.

[58] B. Wild, K.S. Barsim, B. Yang, A new unsupervised event detector for non-intrusive load monitoring, in: Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP), IEEE, 2015, pp. 73–77.

[59] B. Zhao, L. Stankovic, V. Stankovic, On a training-less solution for non-intrusive appliance load monitoring using graph signal processing, IEEE Access 4 (2016) 1784–1799.

[60] G.L. Grinblat, L.C. Uzal, P.M. Granitto, Abrupt change detection with one–class time-adaptive support vector machines, Expert Syst. Appl. 40 (18) (2013) 7242–7249.

[61] C.S. Hilas, I.T. Rekanos, P.A. Mastorocostas, Change point detection in time series using higher-order statistics: a heuristic approach, Math. Probl. Eng. 2013 (2013).

[62] A.N. Milioudis, G.T. Andreou, V. Katsanou, K.I. Sgouras, D.P. Labridis, Event detection for load disaggregation in smart metering, in: Proceedings of the Fourth IEEE/PES Innovative Smart Grid Technologies Europe (ISGT EUROPE), IEEE, 2013, pp. 1–5.

[63] F. Jazizadeh, M. Afzalan, J. Wang, Self-configuring Event Detection for Electricity Disaggregation, The 4th International Workshop on Non-Intrusive Load Monitoring, 2018.

[64] L. Ye, E. Keogh, Time series shapelets: a new primitive for data mining, in: Proceedings of the Fifteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2009, pp. 947–956.

[65] F. Jazizadeh, B. Becerik-Gerber, M. Berges, L. Soibelman, An unsupervised hierarchical clustering based heuristic algorithm for facilitated training of electricity consumption disaggregation systems, Adv. Eng. Inform. 28 (4) (2014) 311–326.

[66] F. Jazizadeh, B. Becerik-Gerber, M. Berges, L. Soibelman, Unsupervised clustering of residential electricity consumption measurements for facilitated user–centric non-intrusive load monitoring, in: Computing in Civil and Building Engineering, 2014, pp. 1869–1876.

[67] M. Afzalan, F. Jazizadeh, "An automated spectral clustering for multi-scale data," arXiv preprint arXiv:1902.01990, 2019.

[68] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: Proceeding of the Advances In Neural Information Processing Systems, 2002, pp. 849–856.

[69] U. Von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416.

[70] M. Meila, in: Spectral Clustering: A Tutorial for the 2010's, CRC Press, 2016, pp. 1–23.

[71] B. Mohar, Some applications of Laplace eigenvalues of graphs, in: Graph Symmetry, Springer, 1997, pp. 225–275.

[72] F. Jazizadeh, Building energy monitoring realization: context-aware event detection algorithms for non-intrusive electricity disaggregation, in: Proceedings of the Construction Research Congress, 2016, pp. 839–848.

[73] H. Alt, M. Godau, Computing the Fréchet distance between two polygonal curves, Int. J. Comput. Geom. Appl. 5 (01n02) (1995) 75–91.

[74] J. Liang, S.K. Ng, G. Kendall, J.W. Cheng, Load signature study – Part I: basic concept, structure, and methodology, IEEE Trans. Power Deliv. 25 (2) (2010) 551–560.

[75] K.D. Anderson, M.E. Bergés, A. Ocneanu, D. Benitez, J.M. Moura, Event detection for non intrusive load monitoring, in: Proceedings of the Thirty-Eighth Annual Conference on IEEE Industrial Electronics Society, IEEE, 2012, pp. 3312–3317.

[76] I. Rahman, M. Kuzlu, S. Rahman, Power disaggregation of combined HVAC loads using supervised machine learning algorithms, Energy Build. 172 (2018) 57–66.

[77] C. Deb, M. Frei, J. Hofer, A. Schlueter, Automated load disaggregation for residences with electrical resistance heating, Energy Build. 182 (2019) 61–74.

[78] M. Pipattanasomporn, M. Kuzlu, S. Rahman, Y. Teklu, Load profiles of selected major household appliances and their demand response opportunities, IEEE Trans. Smart Grid 5 (2) (2014) 742–750.

[79] U. S. Department of Energy, Miscellaneous Electric Loads: What Are They and Why Should You Care?, 2016. Available: https://www.energy.gov/eere/buildings/articles/miscellaneous-electric-loads-what-are-they-and-why-should-you-care.