Accepted Manuscript

Activity recognition with weighted frequent patterns mining in smart environments

Jiahui Wen, Mingyang Zhong, Zhiying Wang

PII: S0957-4174(15)00250-X
DOI: http://dx.doi.org/10.1016/j.eswa.2015.04.020
Reference: ESWA 9967

To appear in: *Expert Systems with Applications*

Activity recognition with weighted frequent patterns mining in smart environments

Jiahui Wen[a,b], Mingyang Zhong[a], Zhiying Wang[b]

[a] *School of Information Technology and Electrical Engineering, The University of Queensland, QLD 4072, Australia*
[b] *School of Computer, National University of Defense Technology, Changsha, China*

Abstract

In the past decades, activity recognition has aroused a great interest for the research groups majoring in context-awareness computing and human behaviours monitoring. However, the correlations between the activities and their frequent patterns have never been directly addressed by traditional activity recognition techniques. As a result, activities that trigger the same set of sensors are difficult to differentiate, even though they present different patterns such as different frequencies of the sensor events. In this paper, we propose an efficient association rule mining technique to find the association rules between the activities and their frequent patterns, and build an activity classifier based on these association rules. We also address the classification of overlapped activities by incorporating the global and local weight of the patterns. The experiment results using publicly available dataset demonstrate that our method is able to achieve better performance than traditional recognition methods such as Decision Tree, Naive Bayesian and HMM. Comparison studies show that the proposed association rule mining method is efficient, and we can further improve the activity recognition accuracy by considering global and local weight of frequent patterns of activities.

Keywords: Data mining, Association rule, Activity recognition, Global and local weight, Smart environments

1. Introduction

Activity recognition (de la Concepción et al., 2014; Fernández-Caballero et al., 2012) has aroused a great interest in the past decade and has been addressed by many research groups using different kinds of physical devices and reasoning techniques. The great interest in activity recognition can be explained in many ways. On one hand, because of the unprecedented growing speed of the aging population around the world (Chernbumroong et al., 2013), one can imagine that elderly health-care will cost increasingly large amount of government budget in the future. However, monitoring Activities of Daily Living (ADL) (Reisberg et al., 2001) such as sleeping, cooking and eating can help the aged to live independently at home, and detecting the abnormal situation as soon as possible can reduce the danger to the minimum extent. On the other hand, as the increasing computational capability and memory storage enable the intelligent computing units to be deployed invisibly around the environments, there is a growing interest in the area of context-awareness computing. Environment-embedded sensors make it possible to gather various context information to guide the applications to be intelligent and behave adaptively toward the benefits of the residents. Human activity is one of the most important context, and activity recognition bridges the gap between various context-aware applications and intelligent ambient sensors.

Activity recognition is related to expert and intelligent systems from two aspects. Firstly, activity recognition can be viewed as a middleware between low-level sensors and high-level context-aware applications. The high-level context-aware applications are expert systems which make decisions towards the benefits of the users by reasoning the current observations against the pre-defined domain knowledge. For example, one application may turn the smartphone into *silence* mode if the on-going activity is *meeting*. In this example, the pre-defined rule to change smartphone's mode and the activity recognition component can be regarded as *knowledge base* and *inference engine* respectively, which are the two most important subsystems in expert systems. On the other hand, activity recognition system itself can be viewed as an expert and intelligent system. It learns knowledge from the labelled data and performs inference to reason activities based on current sensor readings. Activity recognition can explicitly specify the knowledge base such as the decision rules in Decision Tree. The learned knowledge can also be implicitly specified, such as the transition probability in Hidden Markov model (HMM), the support vectors in Support Vector Machine (SVM), the weights of potential functions in Conditional Random field (CRF). The inference process depends on the machine learning techniques used for activity recognition, and it includes dynamic programming in HMM and CRF, inner product between test vector and support vectors in SVM.

Based on the activities to be recognised, there are mainly two ways of recognition using different types of sensors. One is to attach sensors to human body to capture the physical activity signals such as acceleration and angular velocity (Banos et al., 2012; Kwon et al., 2014), and then machine learning models are trained with the labelled data and used to classify the test data. The other one is to recognise high-level activities through the

Email addresses: `j.wen@uq.edu.au` (Jiahui Wen), `m.zhong1@uq.edu.au` (Mingyang Zhong), `zywang@nudt.edu.cn` (Zhiying Wang)

interactions between the people and the environments (Ordóñez et al., 2013; Chernbumroong et al., 2013; Azkune et al., 2015; Wen and Zhong, 2015). The argument for the second method is that high-level activities usually share common sets of physical actions, and are difficult to differentiate based solely on physical signals. However, these kinds of high-level activities can be characterised by the objects used by people, people's location and the time they perform the activities, and these objects can be obtained from sensors such as electrical ID tags deployed in the environments (Palmes et al., 2010; Gu et al., 2010).

Even though there are numerous ways for human activities recognition, with each addressing a certain aspect of the problems during the recognition model construction, some issues are still needed to be addressed. First of all, most of the activity recognition systems disregard the discriminative power of the features they choose. Even if some works (Banos et al., 2012; Könönen et al., 2010) use greedy algorithm to select the best group of features that are able to yield high accuracy, all the features are applied for classification if they are selected in the previous step, ignoring the fact that some features may not be informative in discriminating an activity from another. For example, the irrelevant features in the feature vector may contribute to the error when calculating the distance in instance-based classifiers. Furthermore, human activities, characterised by the sensor events in smart environments, may show some degree of overlap and are difficult to distinguish using traditional methods (Rashidi et al., 2011). Note that the *overlap* is termed as the phenomenon that different activity classes share the same set of sensor events and are difficult to differentiate solely based on the types of sensor events they triggered. However, the frequencies of sensor events may be different for the activity classes and can be used to discriminate them. For example, activity a_1 triggers sensor events $\{s_1, s_3, s_3, s_3\}$ and activity a_2 triggers sensor events $\{s_1, s_1, s_1, s_3\}$. The two activities trigger the same set of sensor events $\{s_1, s_3\}$ and are impossible to differentiate based solely on the types of triggered sensor events. However, the activities have different frequencies in these two sensor events, and these knowledge can be mined to recognise overlapped activities.

In this paper, we apply association rule mining techniques to find frequent patterns of human behaviours from annotated daily life logs and use the frequent patterns to classify the human activities based on the sensor readings. In this way, the frequent patterns of each activity are characterised by the sensors triggered more frequent by the activity than by the others. This is reasonable, since people tend to perform certain activities in the same place and use the same objects, thus trigger almost the same sensors every time they perform the activities. For example, people are always cooking in the kitchen and interacting with the kitchenware. In other words, human behaviours can be characterised by the surrounding sensor readings, and in turn, the sensor readings can be regarded as the patterns of human behaviours, thus it can be used to recognise human activities if they are frequent enough. The contributions of this paper can be concluded as follows:

1. We propose an efficient association rule mining algorithm to find the relationships between the activities and their frequent patterns in smart environments.

2. We use the association rules to build a classifier that is able to achieve a higher performance than traditional classifiers commonly used for activity recognition in smart environments.

3. We also incorporate the global and local weights of sensor events in different activities to differentiate overlapped activities.

The reminder of this paper is organised as follows: Section 2 describes the related work. Section 3 details how to use the association rules to build a classifier, while Section 4 describes the mining process of the association rules and the experiment results are presented in Section 5. Finally, we conclude our work in Section 6.

2. Related work

2.1. Association rule and associative classifier

Traditionally, association rules mining (Rodríguez-González et al., 2013) is used to find the frequent itemsets among the historical transactions and discover unknown relationships so as to provide information for decision making or prediction (Rajasethupathy et al., 2009).

An association rule is presented as $X \Rightarrow Y$ where X and Y are disjoint set of items and are called the antecedent and consequent of the association rule respectively. Two conventional criteria that are used to evaluate an association rule are *support* and $confidence$. The support of a rule is the ratio of the transactions that contain both of its antecedent X and consequent Y, while the confidence of a rule is the ratio of transactions that contain its antecedent also contain its consequent. Only the associations rules that meet the user-specified minimum support and minimum confidence are of interest. Apriori algorithm (Agrawal et al., 1994) is the most simple and efficient association rule mining algorithm that iterates the steps of candidate generation and pruning to find the frequent itemsets, while FP-growth algorithm (Han et al., 2000) transforms all the transactions into a compact representation of a tree, avoiding the candidate generation.

Associative classification is another research topic which means to extract association rules from the training dataset and select some of them to construct the classification models, and is demonstrated in CBA, CMAR and CPAR (Chien and Chen, 2010) to achieve a better performance than traditional classifiers such as Decision Tree. Recently, many research works (Pach et al., 2008) also extend the associative classification to deal with numerical data by introducing the concept of fuzzy sets. Some others (Yan et al., 2009; Qodmanan et al., 2011) even use the genetic algorithm to learn the membership function of fuzzy logic or to mine the association rules without user-specified minimum support.

The difference between the aforementioned methods and our association rules mining methods is that, we leverage the special characteristics of the activity data in smart environments and propose an efficient rules mining method for activity recognition. This is crucial because sensor readings of the datasets from smart environments usually last for several months and contain millions of sensor event logs.

2.2. Activity recognition

Generally, the models recognizing human activities can be classified into two categories: knowledge-driven models and data-driven models. In knowledge-driven models, the activities are usually represented in the form of rules specified with common sense, and the models have an advantage in being reused among different environments. However, the limitation of the statically and strictly defined rules makes the models being unable to deal with noises and uncertain information in sensor readings (Gu et al., 2010). By contrast, data-driven models, which are trained with realistic data, are more powerful when facing the characteristics of randomness and erratic nature of human behaviours. To name a few, they include Naive Bayesian used in (Bao and Intille, 2004; Tapia et al., 2004), HMM in (Patterson et al., 2005; Van Kasteren et al., 2008), SVM in (Cook et al., 2013; Brdiczka et al., 2009; Zhan et al., 2014), Decision Trees in (Bao and Intille, 2004; Hevesi et al., 2014), KNN in (Sundholm et al., 2014; Hevesi et al., 2014) and CRF in (Vail et al., 2007; Zhan et al., 2014).

In recent years, human activity recognition has focused on areas such as less supervision, energy efficiency and activity personalisation. In less supervision, authors try to build activity recognition model using less labelled data with semi-supervised learning methods (Stikic et al., 2011; Stikic and Schiele, 2009; Stikic et al., 2008, 2009; Maekawa and Watanabe, 2011; Lee and Cho, 2014), or discover frequent activity patterns from unlabelled data (Huynh et al., 2008; Sun et al., 2014; Seitr et al., 2015). In energy efficiency, researchers try to lower the energy consumption of the activity recognition system by selecting a subset of the sensors dynamically (Gordon et al., 2012; Zappi et al., 2008; Keally et al., 2011) or change the sampling rate of the sensors adaptively (Yan et al., 2012). As for activity personalisation (Reiss and Stricker, 2013; Zhao et al., 2011; Cvetkovic et al., 2011), general activity models trained with data of various users are adapted and personalised for a specific user in order to improve the general recognition accuracy.

By contrast, this paper aims to recognise high-level activities in smart environments. The high-level activities have much more semantic meanings than low-level locomotion such as running and walking, and can better characterise daily routines of the human beings. Those semantic activity routines are much more useful for expert systems to behave adaptively towards the interests of the users, such as providing living assistance or monitoring active level for the elderly people. The difference also roots in the fact that we mine the frequent patterns that can better characterise the activities and filter out irrelevant and less informative sensor readings which may negatively affect the recognition accuracy, and we also try to distinguish overlapped activities that are not addressed in traditional activity recognition in smart environments (Van Kasteren et al., 2008; Cook et al., 2013; Tapia et al., 2004).

2.3. Frequent patterns for activity recognition

In some scenarios, activities are recognised by frequent patterns mining and matching (Huang et al., 2010). Gu et al. (Palmes et al., 2010; Gu et al., 2010) propose to recognise activities with emerging patterns, which are the sensor events that appear frequently in one activities and infrequently in the others. They dynamically segment the sensor event sequence into windows, and then computing the score for each activity by accumulating the weights of the sensor events in the windows against each activity. Although each sensor event may have different weighs against different activities, each activity is characterised by a certain sensor event with 100 percentage. However, in pervasive environment, human activities show a great degree of variation and are impossible to be characterised by a small set of sensor events with 100 percentage. By contrast, unsupervised techniques (Rashidi et al., 2011; Rashidi and Cook, 2009) are much more suitable for activity frequent patterns mining and clustering in pervasive environment where the sensor data shows a high degree of randomness and discontinuity. However, it neglects the fact that some activities may trigger the same set of sensors, and sensor readings from these sensors would be unavoidably regarded as the same pattern based on the similarity computation in the literature, which makes it difficult to distinguish overlapped activities. Lühr et al. (Lühr et al., 2007) also apply association rule mining to recognise activity in smart environments. However, they only focus on frequent sensor event sequences, which is vulnerable to the noises and the various ways that the residents perform the activities in the realistic environments.

To overcome the problems stated above, we develop an efficient Apriori-modified algorithm to mine the associations between the activities and the sensor events they triggered. We also consider the global and local weight of the sensor events in order to discriminate the activities that have the same frequent patterns but differing in sensor events frequency.

3. Associative classifier

In this section, we give the of definitions of the association rules between the activity classes and their frequent sensor event patterns, and describe the method using the association rules together with their confidences to build an activity classifier.

3.1. Activity trace

An activity trace is a set of sensor events triggered during an activity, as is shown in Table.1.

As can be seen from table, the activity trace *Sleeping* only triggers sensors *m003* and *m007*, which can be regarded as the patterns of *Sleeping*. If they are frequent enough in activity *Sleeping* and infrequent in other activities, then they are frequent patterns of activity *Sleeping*, and can be used to characterise it. The goal of the associative classifier building is to find these kinds of frequent patterns for each activity and generate association rules in order to construct the classifier. Note that frequent itemsets can be regarded as frequent patterns for one activity if they are frequent in that activity and infrequent in others.

Normally there are three ways (Van Kasteren et al., 2008) to represent binary sensor events, but experiment on the influence

Table 1: Example of an activity trace

Timestamps	SensorID	Status	Activity
2010-11-04 00:03:50.20	M003	ON	Sleeping_begin
2010-11-04 00:03:57.39	M003	OFF	
2010-11-04 02:32:33.35	M003	ON	
2010-11-04 02:32:38.89	M003	OFF	
2010-11-04 03:42:21.82	M003	ON	
...			
2010-11-04 05:40:34.52	M007	OFF	
2010-11-04 05:40:40.48	M003	OFF	
2010-11-04 05:40:40.84	M003	ON	
2010-11-04 05:40:42.45	M007	ON	
2010-11-04 05:40:43.64	M003	OFF	Sleeping_end

of sensor event representation is out of the scope of this paper. Without loss of generality, we represent the set of sensors as:

$$S = \{s_1, s_2, \cdots, s_n\}$$

where n is the total number of sensors deployed around the environment. Similarly, the set of activities to be recognised is represented as:

$$A = \{a_1, a_2, \cdots, a_m\}$$

Given the representations above, an activity trace can be represented in the form of a transaction in frequent itemset mining:

$$T = \{a_i, S'\}$$

where a_i is the class label of activity trace T, and $S' \subseteq S$ is the set of sensor events triggered by activity a_i. Correspondingly, the notions of support and confidence of an activity trace given a set of sensor events can be defined in Eq.(1) and Eq.(2) :

$$supp(S' \Rightarrow a_i) = \frac{\sum_{T \in D} f_{S',a_i}(T)}{|D|} \tag{1}$$

$$conf(S' \Rightarrow a_i) = \frac{\sum_{T \in D} f_{S',a_i}(T)}{\sum_{T \in D} f_{S'}(T)} \tag{2}$$

and

$$f_{S',a_i}(T) = \begin{cases} 1, & S' \subset T \bigwedge a_i \in T \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $|D|$ is the total number of activity traces in the training dataset, and $\sum_{T \in D} f_{S'}(T)$ is the number of activity traces that contain S' while $\sum_{T \in D} f_{S',a_i}(T)$ is the number of activity traces contain S' and are labelled as a_i. As previously stated, the support of a rule is the percentage of activity traces that contain the literal in the rule, while the confidence is the conditional probability of the consequent (activity) given the antecedent (sensor events).

As shown in Table.1, a sensor might be triggered more than once during an activity trace. If we simply record the status of a sensor rather than its triggering frequency, information about the activity would be lost. Even though this information makes no contribution if each activity possesses the frequent patterns

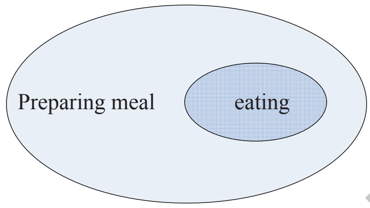

Figure 1: Example of overlapped activities

extremely different from that of the others, it is difficult to distinguish activities that share the common patterns but different in sensor event frequencies, as illustrated in Fig.1.

Assume the activity *Eating* only triggers the sensor near the table, while the *Preparing meal* triggers that sensor less often than *Eating*, which means the sensor event near the table weights more in *Eating* than in *Preparing meal*. However, if this information is neglected, the sensor event near the table would make the same contribution to recognising *Preparing meal* and *Eating*, and these two activities would be constantly mis-classified as each other.

To cope with this problem, we introduce the concept of global weights of sensor events, which are the frequencies of different sensor events in different activities. In this way, we can incorporate the frequencies of the sensor events into the calculation of the confidence of an association rule. Correspondingly, the representation of an activity trace can be extended as follows:

$$T = \{a_i, n_{s_{i1}}, n_{s_{i2}}, \cdots, n_{s_{i,n_i}}\} \tag{4}$$

where $n_i = |S'|$ is the number of sensors triggered during the activity trace a_i, and $n_{s_{i1}}$ is the times that sensor $s_{i1} \in S'$ is triggered during the activity trace, that is the frequency of the sensor event. Accordingly, the confidence of the association rule can be extended as in Eq.(5).

$$conf(S' \Rightarrow a_i) = \frac{\sum_{T \in D}(f_{S',a_i}(T) * \varphi_{S'}(T))}{\sum_{T \in D}(f_{S'}(T) * \varphi_{S'}(T))} \tag{5}$$

where

$$\varphi_{S'}(T) = n_{s_{i1}} \oplus n_{s_{i2}} \oplus \cdots, \oplus n_{s_{i,n_i}}$$

In this paper, we define $\varphi_{S'}(T)$ as:

$$\varphi_{S'}(T) = min(n_{s_{i1}}, n_{s_{i2}}, \cdots, n_{s_{i,n_i}}) \tag{6}$$

Note that we consider the frequencies of the sensor events rather than the temporal relationships among them, as people do not follow exactly the same steps every time they perform the activities in realistic scenarios. This can be best illustrated by the activity *Housekeeping* which shows the most variation in its activity traces, because there is no strict order to do the housework. Furthermore, sometimes the ongoing activity may be interrupted by other unknown steps and result in discontinuities. Therefore, it is more reasonable to cluster the sensor events and consider their intensity rather than model the temporal relations between the sensor events (Cook, 2010). Finally, different people perform the activities differently, and the temporary patterns in the sensor events can even be used to infer the

Table 2: Example of sensor noises

Activity trace No.	Sensor1	$\cdots$	Activity Label
T1	200	$\cdots$	a_1
T2	123	$\cdots$	a_1
$\cdots$	$\cdots$	$\cdots$	$\cdots$
T10	153	$\cdots$	a_1
$\cdots$	$\cdots$	$\cdots$	$\cdots$
T50	1	$\cdots$	a_2
T51	1	$\cdots$	a_3
T52	1	$\cdots$	a_4
$\cdots$	$\cdots$	$\cdots$	$\cdots$
T100	1	$\cdots$	a_2

identify of the people (Hodges and Pollack, 2007). Therefore, the temporal patterns of one person can not be scaled to others.

It is important to note that an unrelated sensor may be triggered in an activity trace due to signal noises, which is quite common in sensor network. The noises need to be filtered out as they influence the confidences of the association rules, and even affect the accuracy of classification when using the rules.

Taking Table.2 for example, the frequency of sensor1 in T50, T51, T52 and T100 is rather small and can be regarded as noise or randomly triggered by the user rather than the pattern of the other activities (e.g. a_2, a_3, a_4), otherwise the confidence of activity a_1 given the sensor event of sensor1 would be negatively affected. Therefore, we apply a threshold in order to filter out the sensor events that can be neglected, and then Eq.(5) can be extended to Eq.(7).

$$conf(S' \Rightarrow a_i) = \frac{\sum_{T\in D}(f_{S',a_i}(T) * max(0, \varphi_{S'}(T) - threshold))}{\sum_{T\in D}(f_{S'}(T) * max(0, \varphi_{S'}(T) - threshold))} \tag{7}$$

3.2. Recognise activity with local weight

All the association rules that meet the user-specified minimum support and confidence can be aggregated to build a classifier. Basically, building a classifier follows two steps. Firstly, pruning the rules that are inferior (Chen and Chen, 2008) to other rules. Secondly, further deleting the rules that do not increased the classification accuracy by applying the group of rules obtained from the first step onto the training dataset. When classifying a test instance, the rule whose antecedent satisfies the instance will be used to classify it. The purpose of the first step is to prune the redundant and conflict rules. Given Two rules, r1 and r2, r2 is redundant if it has the same consequent part as r1 and the antecedent of r1 is the subset of r2, and (1) r1 has a higher confidence than r2 or (2) they have the same confidence but r1 has a stronger support than r2 or (3) they have the same confidence and support, but r1 is generated earlier than r2. If they have the different consequent then are conflicted with each other (Chen and Chen, 2008).

However, in our scenario, it is possible that a sensor event happens during different activity traces with different frequencies. Therefore, association rules built on this sensor event with different activity as consequent should not be regarded as conflict, because activities can possibly overlap with each other and share the same set of frequent patterns. In this way, by mining the frequency information about a sensor triggering, we are able to differentiate the activities that share common frequent patterns. As for the redundancy, the rule whose confidence is lower than that of others would be deleted if the antecedent of the later is the subset of that of the former, which is illustrated in the mining process.

Since each activity has its own frequent pattern, for each activity we create a group of association rules in which the activity is the consequent and the frequent patterns are the antecedent. When performing classification, for each test instance, the group of association rules are applied to aggregate the evidence against the activity based on the test instance, which is demonstrated to be able to achieve better performance than only choose one association rule for classification (Chien and Chen, 2010).

Without the loss of generality, for each activity we create a list *fplist* that stores the frequent patterns of the activity. The association rules, whose antecedents are those frequent patterns, meet the user-specified minimum support and confidence, denoted as follow:

$$fplist(a_i) = \{X|\forall x \in X, conf(x \Rightarrow a_i) \geq mconf, supp(x \Rightarrow a_i) \geq msupp\}$$

where x is a frequent pattern of activity a_i. Since the redundant rules have been pruned, the association rule with x as its antecedent must have a higher confidence and stronger support than the rules whose antecedent is the subset of x. Therefore, we order the frequent patterns of the activity in descending order in terms of their lengths. Suppose two rules, r1 and r2, the antecedent of r1 is the superset of that of r2, and then r2 would not be used to aggregate the evidence of the test instance if r1 has already been used, so as to avoid the duplicate aggregation. The algorithm of classification is illustrated in Alg.1 (line 6), assuming that we already have the ordered frequent patterns of each activity and confidences of the association rules.

Algorithm 1 Algorithm of classifying an unlabelled activity trace

Input:
Unlabelled activity trace X
fplist of each activity class
Output:
Activity class label of X

```
1:  for each a_i ∈ {a_1, a_2, ··· a_m} do
2:      temp = ∅;
3:      evidence(X, a_i) = 0
4:      for each x ∈ fplist(a_i)) do
5:          if x ⊆ X and !set(x).issubset(set(temp)) then
6:              evidence(X, a_i) += conf(x⇒a_i) / (conf(x⇒a_i)+1)
7:              temp.append(x)
8:          end if
9:      end for
10: end for
11: return a_i with maximum evidence(X, a_i) ;
```

While the global weights of the sensor events have been encoded into the confidences of the association rules in Eq.(7), local weights are still not considered. Local weight illustrates the relative frequency of a sensor event with respect to the others in the same activity trace. As one can see that, global weight measures the inter-activity trace sensor event frequency, while the local weight reflects sensor event frequency of the intra-activity trace. The local weight of a sensor event also means the degree that the sensor event dominates the activity trace. Without the local weight, all the sensor events in an activity trace are treated equally and the information of the sensor events frequencies is lost. Considering the local weight, the aggregation of the evidence in Alg.1 (line 6) can be extended as in Eq.(8).

$$evidence(X, a_i) += \frac{conf(x \Rightarrow a_i)}{conf(x \Rightarrow a_i) + 1} * \frac{NofTri(x)}{NofTri(X)} \tag{8}$$

where $NofTri(x)$ is the aggregated sensor events in frequent pattern x, and $NofTri(X)$ is the aggregated sensor events in activity trace. The ratio $\frac{NofTri(x)}{NofTri(X)}$ denotes the local weight of frequent pattern x.

4. Mining process

In the previous section, we describe how to use association rules to recognise activities, taking the global weight and local weight of sensor event into consideration. In this section, we present the algorithm of mining the association rules from the training dataset.

Generally, the mining process of the association rules can be characterised with two steps: (1) generating all the frequent itemsets that meet the minimum user-specified threshold, and (2) extracting the association rules based on the frequent itemsets. In this paper, we modify the Apriori algorithm to mine the frequent patterns of each activity, and the mining process can be divided into two steps: for each activity, we (1) joint the frequent (k-1)-itemsets together to generate the candidate k-itemsets, and (2) prune the itemsets that are infrequent and construct the frequent k-itemsets. We iterate the two steps until no new candidates can be generated, and finally all the frequent itemsets form the frequent patterns of the activity.

Traditionally, the Apriori algorithm has to pass over the database to count the frequency of the new generated candidate itemsets, which introduces lots of overhead. To boost the efficiency of the mining process, we associate each itemset with a *Tid_list*. While the *Tid_list* of an activity records IDs of all the activity traces that are labelled as the activity, the *Tid_list* of a sensor event (or an itemset) contains both of the ID of the transactions (activity traces) containing the sensor event and its corresponding frequency in those transactions. With the *Tid_list*, our algorithm can be illustrated in Alg.2.

Firstly, we create the itemset L_1 for each activity. Each element of L_1 contains only one type of sensor event which is the 1-frequent pattern for the activity. At each iteration, frequent k-itemsets are merged with each other to generate the candidate (k+1)-itemsets (line 9). After that, the Tid_list of each of the newly formed (k+1)-itemsets is obtained with the function *intersection* (line 12). Finally, the potential (k+1)-itemsets are selected to form the frequent (k+1)-itemsets. Line 14 selects frequent (k+1)-itemsets by examining whether or not the support and confidence of the corresponding association rules meet the minimum threshold.

The evaluation (line 14) of the candidate k-itemsets takes place after the computing of their *Tid_list*s, because computing the confidence of k-itemsets against an activity needs their *Tid_list*s. The confidence of the association rules can be obtained with Eq.(9), in which the function *aggre* is used to aggregate the global weight of itemsets. Eq.(9) is consistent with Eq.(7) except that we perform aggregation based on the *Tid_list* rather than going through the whole database. In special case, if the global weights of the itemsets are neglected, the only information left is the number of transactions that contain the itemset, and the function *aggre* can be replaced with the length of *Tid_list*. The function *intersection* (line 12) return the *Tid_list* of the new generated k-itemsets, and each element of the *Tid_list* consists of the ID of the transaction containing the k-itemset and the aggregated triggered times of the k-itemset in that transaction. The aggregated triggered times can be obtained based on Eq.(6).

$$conf(S' \Rightarrow a_i) = \frac{aggre(Tid\_list(S', a_i))}{aggre(Tid\_list(S'))} \tag{9}$$

Algorithm 2 Algorithm of mining frequent patterns for activity classes

Input:
Activity traces: $T_1, T_2, \cdots, T_n$
Sensor set: $\{s_1, s_2, \cdots, s_n\}$
Activity class: $\{a_1, a_2, \cdots, a_m\}$
Output:
Frequent patterns of each activity a_i
1: **for** each $s_i \in \{s_1, s_2, \cdots s_m\}$ **do**
2: create the Tid_list for s_i
3: **end for**
4: **for** each $a_i \in \{a_1, a_2, \cdots, a_m\}$ **do**
5: create the Tid_list for a_i
6: $L_1 = \{s_i | len(Tid\_list(s_i, axis = 0) \cap Tid\_list(a_i)) \geq msupp * n \wedge conf(s_i \Rightarrow a_i \geq mconf)\}$
7: **for** $k = 1$; $L_k \neq \emptyset$; k++ **do**
8: // generate candidate (k+1)-itemset C_{k+1} from L_k
9: $C_{k+1} = \{\{s_1, \cdots, s_{k-2}, s_{k-1}, s_k\} | \{s_1, \cdots, s_{k-2}, s_{k-1}\} \in L_k \wedge \{s_1, \cdots, s_{k-2}, s_k\} \in L_k\}$
10: //computing the *Tid_list* for candidate (k+1)-itemset
11: // assuming S_k in C_{k+1} is generated from S_x and S_y in L_k
12: $Tid\_list(S_k) = intersection(Tid\_list(S_x), Tid\_list(S_y))$
13: // prune the itemsets the do not meet the minimum threshold
14: $L_{k+1} = \{S_k | S_k \in C_{k+1} \wedge len(Tid\_list(S_k, axis = 0) \cap Tid\_list(a_i)) \geq msupp * n \wedge conf(S_k \Rightarrow a_i) \geq mconf \wedge conf(S_k \Rightarrow a_i) > max(conf(S_x \Rightarrow a_i), conf(S_y \Rightarrow a_i))\}$
15: **end for**
16: frequent patterns of activity a_i: $L_1 \cup L_2 \cup \cdots \cup L_k$
17: **end for**
18: **return**

For the illustrative purpose, Fig.2 gives an example of the process mining the frequent patterns of activity a_1. Suppose

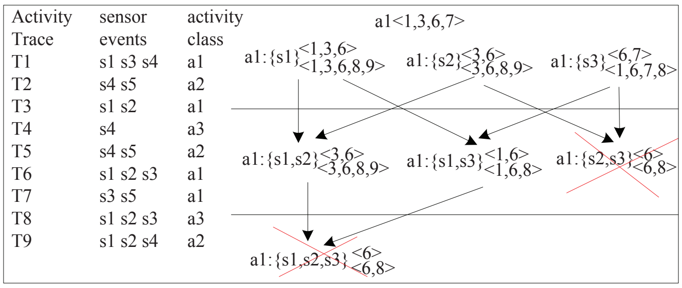

Figure 2: Example of the frequent patterns mining process

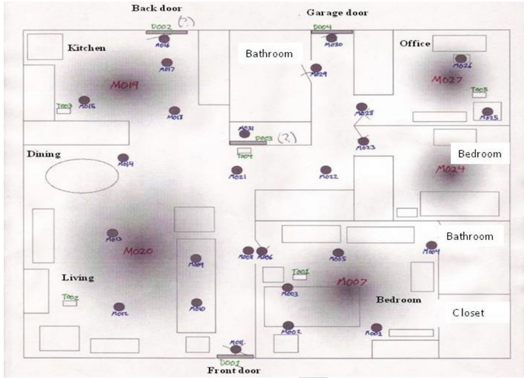

Figure 3: Sensor deployment in the smart environment

that the minimum support is 20% and the minimum confidence is 60%, each layer corresponds to each iteration in Alg.2. The right superscript is the *Tid_list* of the sensor event jointed with the activity, while the right subscript is the *Tid_list* of the sensor event only. The global weight is not incorporated in the example simply for illustrative purpose. In the first iteration, 1-itemsets ($\{s_1\}$, $\{s_2\}$ and $\{s_3\}$) are constructed with the confidence and support of the corresponding association rules meeting the minimum threshold. In the second iteration, $\{s_1\}$ is merged with $\{s_2\}$ to generate 2-itemset $\{s_1, s_2\}$, while in the third iteration $\{s_1, s_2\}$ and $\{s_1, s_3\}$ are merged together and $\{s_1, s_2, s_3\}$ is generated, after that no new candidates can be generated and the pattern mining for activity a_1 terminates. During each iteration, the new generated item sets must be evaluated in order to prune the infrequent ones. For example, $\{s_2, s_3\}$ is prune because the support of association rule ($\{s_2, s_3\} \Rightarrow a_1$) is 1/9, not meeting the minimum threshold, and $\{s_1, s_2, s_3\}$ is pruned for the same reason.

Compared with traditional mining algorithm such as Apriori, the advantages of our method reside in: (1) Instead of searching all the frequent itemsets, we are only focus on the associations between the activities and the sensor events. (2) For every itemset we create an *Tid_list*, with which we do not have to traverse the database every time to calculate the support of the candidate itemsets. The only overhead that this method introduces is the computation of the *Tid_list* of candidate itemset when merging two itemsets, and the complexity is $O(n)$. (3) With the help of *Tid_list*, the association rules can be generated during the frequent itemsets mining process using Eq.(9). More importantly, the confidences of the association rules can be used to prune the itemsets that do not meet the minimum threshold as shown in Alg.2 (line 14), resulting in a much smaller searching space. All the differences mentioned above account for the efficiency of our method, and we demonstrate the effectiveness of our method in Section 5.

5. Evaluation

5.1. Set up

In this section, we use the dataset of smart environments from the CASAS research group to validate our method. The dataset (Cook, 2010) contains the sensor readings from motion sensors, door closure sensors and temperature sensors. There is only one female resident in the home performing her daily activities, and the deployment of sensors in the house is shown in Fig.3. The datasets were annotated and the collected data has the form just the same as in Table.1.

Table 3: The activities to be recognised

Activity No.	Activity name	Activity traces
1	Bed_to_toilet	104
2	Eating	181
3	Enter/Leave_home	263
4	Housekeeping	33
5	Meal_preparation	930
6	Relaxing	1795
7	Sleeping	242
8	Working	112

The activities we are to recognise and the number of activity traces are list in Table.3. Note that activities such as *Resperate* are filtered out, because they contain too few activity traces to generate the association rules. In the realistic scenario, the frequencies of activities vary from one to another, and the supports may be unbalanced for different kinds of activities in the transactions. For example, if the number of activity traces of an activity a_i is much smaller than that of the others, then its frequent patterns may not satisfy the minimum support. To deal with this problem, we construct the transactions non-time-sequentially so that activity traces of different activities account for the same percentage of the transactions.

5.2. Efficiency of mining algorithm

Since using *Tid_list* to mine the frequent patterns and generate the association rules is efficient, we compare our mining algorithm with the traditional association rule mining algorithms. The first baseline algorithm is to search the whole space with *Tid_list*, and generate the whole frequent itemsets, and then construct the association rules, referred to as with_list_searchall. The second baseline algorithm is to search the whole space without *Tid_list*, and generate the whole frequent itemsets with each iteration scanning the data base, and then construct the association rules, referred to as without_list_searchall. Searching the whole space means mining association rules not only between the sensor events and the activities, but also among the

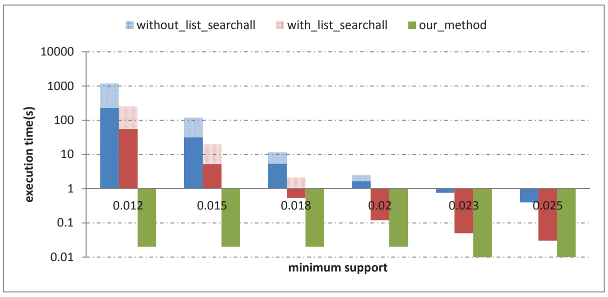

Figure 4: Mining overhead of different frequent pattern mining algorithms

sensor events, which is quite common in traditional frequent itemsets mining algorithm.

The experiments are carried out on a computer (Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz 8 processors 16G memory) running Ubuntu 12.0.4. We vary the minimum support from 0.012 to 0.025 and present the result in Fig.4 that shows the execution overhead (y-axis) of different mining algorithms as a function of the minimum support (x-axis). The execution time of the two baselines are further decomposed into the overhead of mining the frequent patterns and the overhead of generating the association rules.

From the figure we can observe that the overhead of the first baseline is constantly higher than the second one, while our method performs the best in terms of time consumption and is able to remain at a stable level across different minimum supports. Furthermore, when we decrease the minimum support in the two baseline algorithms, the execution overhead and the time spent on rules generation increase significantly. By contrast, the changes of the minimum support have little impact on our method. The reasons for the efficiency of our method are the constrained search space, the pruning policy and the simultaneous generation of frequent patterns and association rules.

5.3. Accuracy analysis

In this subsection, we compare the accuracy of our algorithm with other typical activity recognition methods such as Decision Tree, Naive Bayesian and HMM. Decision Tree is a discriminative classifier which performs classification by detecting the boundaries between different classes, while Naive Bayesian and HMM are generative classifiers which compute the degree that a test instance belongs to a certain labelled class. The difference between Naive Bayesian and HMM is that HMM considers the temporal relationships among the activities to smooth out the outliers.

Fig.5 presents the accuracy of each classifier. Our associative classifier achieves the highest average accuracy of 99.18%, significantly higher than that of Decision Tree and HMM, and much higher than that of Naive Bayesian, which are 96.29%, 95.74% and 88.68% respectively. From the confusion matrix of our method (Table.4) we can see that activities *Housekeeping* and *Relaxing* contribute nearly 85% to the misclassifications, this is because these two activities show more randomness in

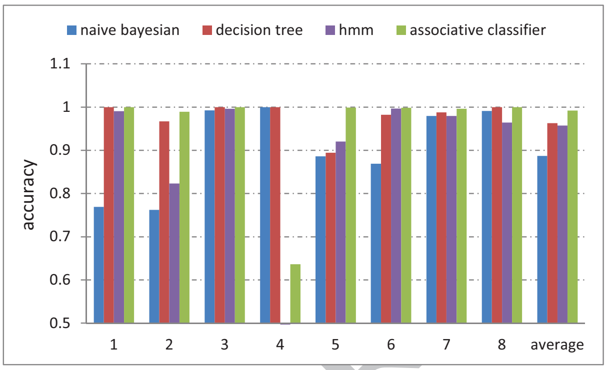

Figure 5: Accuracy of different classifiers. The x-axis shows the accuracy achieved by each activity class and the average accuracy of all the activities.

Table 4: Confusion matrix of our method

Activity	1	2	3	4	5	6	7	8
Bed_to_toilet	104	0	0	0	0	0	0	0
Eating	0	179	0	0	1	1	0	0
Enter/Leave_home	0	0	263	0	0	0	0	0
Housekeeping	0	1	0	21	7	4	0	0
Meal_preparation	0	0	0	0	929	1	0	0
Relaxing	0	0	0	2	0	1792	1	11
Sleeping	0	0	0	1	0	0	241	0
Working	0	0	0	0	0	0	0	112

the activity traces and some of their instances are unavoidably overlapped.

Interestingly in the confusion matrix (Table.5) of Naive Bayesian, many instances of other activities are mis-classified as *Housekeeping*, which accounts for most of the misclassifications. As previously stated, *Housekeeping* is one of the activities that have the most variations in the frequent patterns and overlap some of its frequent patterns with that of other activities. Even though the frequencies of the sensor events are different between *Housekeeping* and that of others, Naive Bayesian cannot capture this information and result in much more misclassifications. For example, activity *Eating* and *Housekeeping* both trigger the sensor near the table, but *Eating* triggers the table sensor much more frequent compared with other activities such as *Housekeeping*. During *Housekeeping* the residents just pass by the table when she was doing the housework. Naive Bayesian treats the events of the table sensor equally when aggregating the evidence for these two activities. Another reason accounts for the low performance of Naive Bayesian may be the assumptions of independence among features and normal distribution of feature values (Bao and Intille, 2004). The misclassifications of Decision Tree (Table.6) is much more random and evenly, and this is because Decision Tree differentiates the test instances by the boundaries and capture the conjunctions in feature values (Bao and Intille, 2004) rather than computing the probability against the class that they belong to. The *Housekeeping* activity accounts for most of the misclassification in HMM (Table.7) with the accuracy of 0%, which again explains the randomness nature of the activity.

Table 5: Confusion matrix of Naive Bayesian

Activity	1	2	3	4	5	6	7	8
Bed_to_toilet	80	0	0	24	0	0	0	0
Eating	0	138	0	26	17	0	0	0
Enter/Leave_home	0	0	261	2	0	0	0	0
Housekeeping	0	1	0	33	0	0	0	0
Meal_preparation	0	0	0	106	824	0	0	0
Relaxing	0	0	0	233	0	1560	2	0
Sleeping	0	0	0	5	0	0	237	0
Working	0	0	0	1	0	0	0	111

Table 6: Confusion matrix of Decision Tree

Activity	1	2	3	4	5	6	7	8
Bed_to_toilet	106	0	0	0	0	0	0	0
Eating	0	175	0	0	3	3	0	0
Enter/Leave_home	0	0	263	0	0	0	0	0
Housekeeping	0	1	0	33	0	0	0	0
Meal_preparation	1	81	4	1	832	7	0	4
Relaxing	0	15	1	4	7	1763	1	4
Sleeping	0	0	0	1	2	1	239	0
Working	0	0	0	0	0	0	0	112

Table 7: Confusion matrix of HMM

Activity	1	2	3	4	5	6	7	8
Bed_to_toilet	103	0	0	0	0	0	1	0
Eating	0	149	0	1	21	9	0	0
Enter/Leave_home	0	0	262	1	0	0	0	0
Housekeeping	0	12	0	0	9	12	0	0
Meal_preparation	0	43	0	21	856	10	0	0
Relaxing	0	0	0	6	0	1789	0	0
Sleeping	3	0	0	0	1	1	237	0
Working	0	0	0	2	0	2	0	108

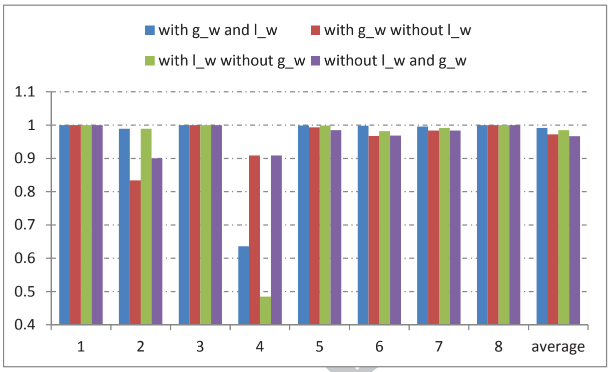

Figure 6: Accuracy comparison with or without global and local weight. Y-axis shows the accuracy while x-axis presents the accuracy achieved by individual activity and the average accuracy of all the activities.

5.4. *Global and local weight*

We also evaluate the influence of global weight and local weight on the recognition accuracy and experiment on four settings: (1) with global and local weight(with g_w and l_w), (2) without global weight but with local weight(with l_w without g_w), (3) without local weight but with global weight(with g_w without l_w), and (4) without global weight and without local weight(without g_w and l_w).

As is presented in Fig.6, we can see that local weight improves accuracy more than global weight overall, and incorporating both of them achieves the best accuracy. Furthermore, in some activities such as *Eating*, local weight plays more important role than global weight. While in some other activities such as *Housekeeping*, global weight brings much more benefit than local weight. The reason may lay in the nature of the activities performed by the resident.

6. Conclusion

In this paper, we develop an efficient association rule mining algorithm to find the frequent patterns of human activities in smart environments. We leverage the association rule to build an activity classifier and demonstrate that it is able to achieve better performance than traditional classifier such as Naive Bayesian, Decision Tree and HMM. Furthermore, we incorporate the global weight and local weight of the sensor events to differentiate the activities that share common frequent patterns but have different sensor events frequencies.

It is not necessary that a classifier performs optimally for all given activity classification problems (Preece et al., 2009). This paper shows different ways of activity recognition by leveraging the correlations between the activities and their own frequent patterns. More importantly, the activities that have overlapped frequent patterns or even share the same frequent patterns can still be differentiated if they have different weights in the common frequent patterns.

Recognising pre-define activities has been fully addressed by many work, and detecting the basic activities such as walking and standing is meaningless because activities of daily living are more complex and various in real scene than in the experiment environment. As a result, new search areas such as finding frequent behaviour patterns and temporal relationships of activities have aroused a great interest. However, most of them only try to find the sequence frequent patterns, and the problem are equivalent to discovering frequent episodes in event sequences, which is just one aspect of human behaviours. It is conceivable that other aspects of human behaviours such as the correlations of different context information from the viewpoint of coexistence, are still needed to be modelled. Therefore, our future work is to find the frequent patterns of human behaviours in much more pervasive environment, not confining in a small set of information sources and predefined activities.

7. Reference

References

Agrawal, R., Srikant, R., et al. (1994). Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, volume 1215, pages 487–499.

Azkune, G., Almeida, A., López-de Ipiña, D., and Chen, L. (2015). Extending knowledge-driven activity models through data-driven learning techniques. *Expert Systems with Applications*, 42(6):3115–3128.

Banos, O., Damas, M., Pomares, H., Prieto, A., and Rojas, I. (2012). Daily living activity recognition based on statistical feature quality group selection. *Expert Systems with Applications*, 39(9):8013–8021.

Bao, L. and Intille, S. S. (2004). Activity recognition from user-annotated acceleration data. In *Pervasive computing*, pages 1–17. Springer.

Brdiczka, O., Crowley, J. L., and Reignier, P. (2009). Learning situation models in a smart home. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 39(1):56–63.

Chen, Z. and Chen, G. (2008). Building an associative classifier based on fuzzy association rules. *International Journal of Computational Intelligence Systems*, 1(3):262–273.

Chernbumroong, S., Cang, S., Atkins, A., and Yu, H. (2013). Elderly activities recognition and classification for applications in assisted living. *Expert Systems with Applications*, 40(5):1662–1674.

Chien, Y.-W. C. and Chen, Y.-L. (2010). Mining associative classification rules with stock trading data–a ga-based method. *Knowledge-Based Systems*, 23(6):605–614.

Cook, D. J. (2010). Learning setting-generalized activity models for smart spaces. *IEEE intelligent systems*, 2010(99):1.

Cook, D. J., Krishnan, N. C., and Rashidi, P. (2013). Activity discovery and activity recognition: A new partnership. *Cybernetics, IEEE Transactions on*, 43(3):820–828.

Cvetkovic, B., Kaluza, B., Luštrek, M., and Gams, M. (2011). Semi-supervised learning for adaptation of human activity recognition classifier to the user. In *Workshop on Space, Time and Ambient Intelligence, IJCAI*, pages 24–29. Citeseer.

de la Concepción, M. Á., Morillo, L. S., Gonzalez-Abril, L., and Ramírez, J. O. (2014). Discrete techniques applied to low-energy mobile human activity recognition. a new approach. *Expert Systems with Applications*, 41(14):6138–6146.

Fernández-Caballero, A., Castillo, J. C., and Rodríguez-Sánchez, J. M. (2012). Human activity monitoring by local and global finite state machines. *Expert Systems with Applications*, 39(8):6982–6993.

Gordon, D., Czerny, J., Miyaki, T., and Beigl, M. (2012). Energy-efficient activity recognition using prediction. In *Wearable Computers (ISWC), 2012 16th International Symposium on*, pages 29–36. IEEE.

Gu, T., Chen, S., Tao, X., and Lu, J. (2010). An unsupervised approach to activity recognition and segmentation based on object-use fingerprints. *Data & Knowledge Engineering*, 69(6):533–544.

Han, J., Pei, J., and Yin, Y. (2000). Mining frequent patterns without candidate generation. In *ACM SIGMOD Record*, volume 29, pages 1–12. ACM.

Hevesi, P., Wille, S., Pirkl, G., Wehn, N., and Lukowicz, P. (2014). Monitoring household activities and user location with a cheap, unobtrusive thermal sensor array. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 141–145. ACM.

Hodges, M. R. and Pollack, M. E. (2007). An'object-use fingerprint': the use of electronic sensors for human identification. In *Proceedings of the 9th international conference on Ubiquitous computing, Ubicomp'2007*, pages 289–303. Springer-Verlag.

Huang, P.-C., Lee, S.-S., Kuo, Y.-H., and Lee, K.-R. (2010). A flexible sequence alignment approach on pattern mining and matching for human activity recognition. *Expert Systems with Applications*, 37(1):298–306.

Huynh, T., Fritz, M., and Schiele, B. (2008). Discovery of activity patterns using topic models. In *Proceedings of the 10th international conference on Ubiquitous computing*, pages 10–19. ACM.

Keally, M., Zhou, G., Xing, G., Wu, J., and Pyles, A. (2011). Pbn: towards practical activity recognition using smartphone-based body sensor networks. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems, Sensys'2011*. ACM.

Könönen, V., Mäntyjärvi, J., Similä, H., Pärkkä, J., and Ermes, M. (2010). Automatic feature selection for context recognition in mobile devices. *Pervasive and Mobile Computing*, 6(2):181–197.

Kwon, Y., Kang, K., and Bae, C. (2014). Unsupervised learning for human activity recognition using smartphone sensors. *Expert Systems with Applications*, 41(14):6067–6074.

Lee, Y.-S. and Cho, S.-B. (2014). Activity recognition with android phone using mixture-of-experts co-trained with labeled and unlabeled data. *Neurocomputing*, 126:106–115.

Lühr, S., West, G., and Venkatesh, S. (2007). Recognition of emergent human behaviour in a smart home: A data mining approach. *Pervasive and Mobile Computing*, 3(2):95–116.

Maekawa, T. and Watanabe, S. (2011). Unsupervised activity recognition with user's physical characteristics data. In *Wearable Computers, 2011. ISWC'11. International Symposium on*, pages 89–96. IEEE.

Ordóñez, F. J., Iglesias, J. A., De Toledo, P., Ledezma, A., and Sanchis, A. (2013). Online activity recognition using evolving classifiers. *Expert Systems with Applications*, 40(4):1248–1255.

Pach, F. P., Gyenesei, A., and Abonyi, J. (2008). Compact fuzzy association rule-based classifier. *Expert systems with applications*, 34(4):2406–2416.

Palmes, P., Pung, H. K., Gu, T., Xue, W., and Chen, S. (2010). Object relevance weight pattern mining for activity recognition and segmentation. *Pervasive and Mobile Computing*, 6(1):43–57.

Patterson, D. J., Fox, D., Kautz, H., and Philipose, M. (2005). Fine-grained activity recognition by aggregating abstract object usage. In *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*, pages 44–51. IEEE.

Preece, S. J., Goulermas, J. Y., Kenney, L. P., Howard, D., Meijer, K., and Crompton, R. (2009). Activity identification using body-mounted sensorsa review of classification techniques. *Physiological measurement*, 30(4):R1.

Qodmanan, H. R., Nasiri, M., and Minaei-Bidgoli, B. (2011). Multi objective association rule mining with genetic algorithm without specifying minimum support and minimum confidence. *Expert Systems with applications*, 38(1):288–298.

Rajasethupathy, K., Scime, A., Rajasethupathy, K. S., and Murray, G. R. (2009). Finding persistent rules: Combining association and classification results. *Expert Systems with Applications*, 36(3):6019–6024.

Rashidi, P. and Cook, D. J. (2009). Keeping the resident in the loop: Adapting the smart home to the user. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 39(5):949–959.

Rashidi, P., Cook, D. J., Holder, L. B., and Schmitter-Edgecombe, M. (2011). Discovering activities to recognize and track in a smart environment. *Knowledge and Data Engineering, IEEE Transactions on*, 23(4):527–539.

Reisberg, B., Finkel, S., Overall, J., Schmidt-Gollas, N., Kanowski, S., Lehfeld, H., Hulla, F., Sclan, S. G., Wilms, H.-U., Heininger, K., et al. (2001). The alzheimer's disease activities of daily living international scale (adl-is). *International Psychogeriatrics*, 13(02):163–181.

Reiss, A. and Stricker, D. (2013). Personalized mobile physical activity recognition. In *Proceedings of the 2013 International Symposium on Wearable Computers*, pages 25–28. ACM.

Rodríguez-González, A. Y., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., and Ruiz-Shulcloper, J. (2013). Mining frequent patterns and association rules using similarities. *Expert Systems with Applications*, 40(17):6823–

6836.

Seitr, J., Chiu, W.-C., Fritz, M., Amft, O., and Troster, G. (2015). Joint segmentation and activity discovery using semantic and temporal priors. In *Pervasive Computing and Communications (PerCom), 2015 IEEE International Conference on*. IEEE. To appear.

Stikic, M., Larlus, D., Ebert, S., and Schiele, B. (2011). Weakly supervised recognition of daily life activities with wearable sensors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(12):2521–2537.

Stikic, M., Larlus, D., and Schiele, B. (2009). Multi-graph based semi-supervised learning for activity recognition. In *Wearable Computers, 2009. ISWC'09. International Symposium on*, pages 85–92. IEEE.

Stikic, M. and Schiele, B. (2009). Activity recognition from sparsely labeled data using multi-instance learning. In *Location and Context Awareness*, pages 156–173. Springer.

Stikic, M., Van Laerhoven, K., and Schiele, B. (2008). Exploring semi-supervised and active learning for activity recognition. In *Wearable Computers, 2008. ISWC 2008. 12th IEEE International Symposium on*, pages 81–88. IEEE.

Sun, F.-T., Yeh, Y.-T., Cheng, H.-T., Kuo, C., and Griss, M. (2014). Nonparametric discovery of human routines from sensor data. In *Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on*, pages 11–19. IEEE.

Sundholm, M., Cheng, J., Zhou, B., Sethi, A., and Lukowicz, P. (2014). Smart-mat: recognizing and counting gym exercises with low-cost resistive pressure sensing matrix. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 373–382. ACM.

Tapia, E. M., Intille, S. S., and Larson, K. (2004). Activity recognition in the home using simple and ubiquitous sensors. In *In Pervasive*.

Vail, D. L., Veloso, M. M., and Lafferty, J. D. (2007). Conditional random fields for activity recognition. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, page 235. ACM.

Van Kasteren, T., Noulas, A., Englebienne, G., and Kröse, B. (2008). Accurate activity recognition in a home setting. In *Proceedings of the 10th international conference on Ubiquitous computing*, pages 1–9. ACM.

Wen, J. and Zhong, M. (2015). Activity discovering and modeling with labeled and unlabeled data in smart environments. *Expert Systems with Applications*.

Yan, X., Zhang, C., and Zhang, S. (2009). Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support. *Expert Systems with Applications*, 36(2):3066–3076.

Yan, Z., Subbaraju, V., Chakraborty, D., Misra, A., and Aberer, K. (2012). Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach. In *Wearable Computers (ISWC), 2012 16th International Symposium on*, pages 17–24. IEEE.

Zappi, P., Lombriser, C., Stiefmeier, T., Farella, E., Roggen, D., Benini, L., and Tröster, G. (2008). Activity recognition from on-body sensors: accuracy-power trade-off by dynamic sensor selection. In *Wireless Sensor Networks*.

Zhan, K., Faux, S., and Ramos, F. (2014). Multi-scale conditional random fields for first-person activity recognition. In *Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on*, pages 51–59. IEEE.

Zhao, Z., Chen, Y., Liu, J., Shen, Z., and Liu, M. (2011). Cross-people mobile-phone based activity recognition. In *Twenty-Second International Joint Conference on Artificial Intelligence, IJCAI'2011*, volume 11, pages 2545–250. Citeseer.

Highlights

We propose an efficient frequent activity patterns mining in smart environments
We build an accurate activity classifier based on the mined frequent patterns
We distinguish overlapped activities with global and local weights of sensor events
We use publicly available dataset of smart environments to validate our methods