Appl Intell
DOI 10.1007/s10489-012-0359-7

EEM: evolutionary ensembles model for activity recognition in Smart Homes

Muhammad Fahim · Iram Fatima · Sungyoung Lee · Young-Koo Lee

Abstract Activity recognition requires further research to enable a multitude of human-centric applications in the smart home environment. Currently, the major challenges in activity recognition include the domination of major activities over minor activities, their non-deterministic nature and the lack of availability of human-understandable output. In this paper, we introduce a novel Evolutionary Ensembles Model (EEM) that values both minor and major activities by processing each of them independently. It is based on a Genetic Algorithm (GA) to handle the non-deterministic nature of activities. Our evolutionary ensemble learner generates a human-understandable rule profile to ensure a certain level of confidence for performed activities. To evaluate the EEM, we performed experiments on three different real world datasets. Our experiments show significant improvement of 0.6 % to 0.28 % in the F-measures of recognized activities compared to existing counterparts. It is expected that EEM would be a practical solution for the activity recognition problem due to its understandable output and improved accuracy.

Keywords Activity recognition · Evolutionary ensemble · Genetic algorithm · Smart Home

M. Fahim (✉) · I. Fatima · S. Lee · Y.-K. Lee
Ubiquitous Computing Lab, Department of Computer Engineering, Kyung Hee University, Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do, 446-701, Korea
e-mail: fahim@oslab.khu.ac.kr

I. Fatima
e-mail: iram.fatima@oslab.khu.ac.kr

S. Lee
e-mail: sylee@oslab.khu.ac.kr

Y.-K. Lee
e-mail: yklee@oslab.khu.ac.kr

1 Introduction

Over the last few years, activity recognition has become an active research area due to the wide range of human centric-applications. The aim of this research is to support an aging society by providing personalized services and assistance for performing daily life activities. The smart home is one of the best solutions to provide a level of independence and comfort in the homes of elderly people rather than requiring them to reside at health care centers [1, 20]. The advancement of sensor technology has proven itself to be robust, cost-effective, easy to install and less intrusive for inhabitants [1, 2].

Significant approaches for activity recognition use video cameras [3], wearable sensors [4] and embedded sensors [5]. However, several problems are associated with the first two approaches; for instance, video cameras are not practical due to privacy, day/night vision problems and complex environments. Wearable sensors are uncomfortable and inconvenient for users, and their accuracy depends on the body attachment position. Embedded sensors are an acceptable solution for sensing the environment without disturbing inhabitant privacy and without adding the extra burden of sensor wearing. In this paper, we introduce a more accurate activity recognition model to recognize daily life activities using embedded sensors with a human understandable rule profile.

Many researchers have designed a variety of models to recognize the activities of daily living and have greatly contributed to improvements in smart home technology. Nevertheless, despite this progress, current activity recognition models are not adequate for practical use in real world applications for a number of reasons. The most notable is low accuracy, due to domination of major activities over the minor activities. During the training phase of the model, minor activities are not properly learned; therefore, accuracy may

Published online: 14 June 2012

be decreased. In addition, the non-deterministic nature of activities (activities are performed differently due to different lifestyles, cultures, and mental and physical differences) also contributes to the decrease in classification accuracy. Furthermore, there is a general lack of confidence in the results of the activity recognition process. Confidence is crucial for successful healthcare service provisioning, so understandable rules can help practitioners to use the model for real world applications.

At present, none of the existing activity recognition models are able to comprehensively handle the aforementioned problems. Therefore, in this study, we propose a novel model to recognize human activities by designing a Genetic Algorithm (GA) based evolutionary ensemble learners that has better accuracy than the existing state-of-the-art methods. We introduce a method to process the information independently by giving equal importance to minor and major activities; hence, the problem of few occurrences of minor activities can be solved. Our proposed model provides a human-understandable activity rule profile to facilitate real world healthcare applications. In addition, we investigate an evolutionary ensemble approach to the activity recognition domain that, to the best of our knowledge, has never been applied before. We implement and evaluate our approach using real datasets collected in the House_n smart homes [2], a project of Massachusetts Institute of Technology (MIT), and the Intelligent System Laboratory (ISL) smart home [5] at the University of Amsterdam.

The rest of the paper is organized as follows. We briefly describe related work and their limitations in Sect. 2. In Sect. 3, we introduce our proposed evolutionary ensemble model and its implementation for recognizing daily living activities in smart homes. In Sect. 4, we analyze our experimental results and evaluation to show our improvements. Finally, we conclude our paper in Sect. 5.

2 Related work

Evolutionary techniques as learning classifiers have successfully solved well-known problems, such as function approximation, general prediction, classification and data mining tasks [6, 21, 22]. Matthew et al. [7] proposed an extended version of learning classifier and utilized GA to produce generalizations over the space of all possible condition-class combinations. Kuncheva et al. [8] used a GA to design the classifier fusion system and determined that, as a learner component, GA outperforms other classifier models. Also, GA has been successfully used as a learner to select optimal genes for analyzing DNA microarrays [9].

In the ensemble learning paradigm, n individual learners are trained using machine learning algorithms [10]. This solves the problem in which single learners suffer from statistical, computational and representational problems. Statistical problems arise due to high variance in the data that excessively increase the size of the search space. Computational problems occur when the training data is computationally intractable and can get stuck in local optimum. Representation problems cause biasness for learning algorithms. Both theoretical and empirical studies of ensemble learning show higher accuracy in real world applications such as spam email filters, character recognition, text categorization, face recognition, computer-aided medical diagnosis, and gene expression analysis [7].

The state-of-the-art and most popular activity recognition techniques are based on probabilistic models like Hidden Markov Models (HMM) [11], Conditional Random Fields (CRF) [12], Bayesian Networks [2] and some other classification methods [13, 14, 19]. However, a number of difficulties and limitations remain with these approaches. The learning capability of probabilistic models depends on the observation of activity class distribution (the observed state) and the transitions between adjacent activities (transitions between states). Existing methods are unable to model activity structures such as location, value or sensor type. Furthermore, they require sufficient data to produce reliable results. Minor activities are skipped during the learning phase due to their few occurrences in the dataset. Therefore, probabilistic models are treated as a black box for recognizing activities; this is impractical for crucial applications such as healthcare.

To overcome the limitations of existing work, we propose an alternative state-of-the-art evolutionary ensembles model. We adopt the decentralized information processing feature of ensembles and investigate the suitability of GA as an ensemble learner. It has the ability to embed activity structure information, produce reliable results from small datasets and generate human understandable activity rules. For these reasons, our EEM model has the potential to work with real world healthcare applications.

3 Evolutionary ensemble model

Let $\Omega = \{S_1, \ldots, S_n\}$ be a set of n embedded sensors, e.g., {stove-sensor, refrigerator-sensor, microwave-sensor, door-sensor etc.} characterized by m attributes $a = [a_1, \ldots, a_m]^T$, where a_i includes sensor value, location and identity, to express the changes in the smart homes. In order to recognize the performed activities, we divide the daily life activities into a set of c classes $C = \{C_1, \ldots, C_c\}$. For each ensemble node (en), search space is defined as $S = (c, [S_{1a}, \ldots, S_{na}])$, where S_{ia} can be a sensor with all attributes. Similarly, rule space is defined as $R = (c, [S_{1a'}, \ldots, S_{na'}])$, where $S_{ia'}$ can be a sensor with all attributes and sensor value may be a

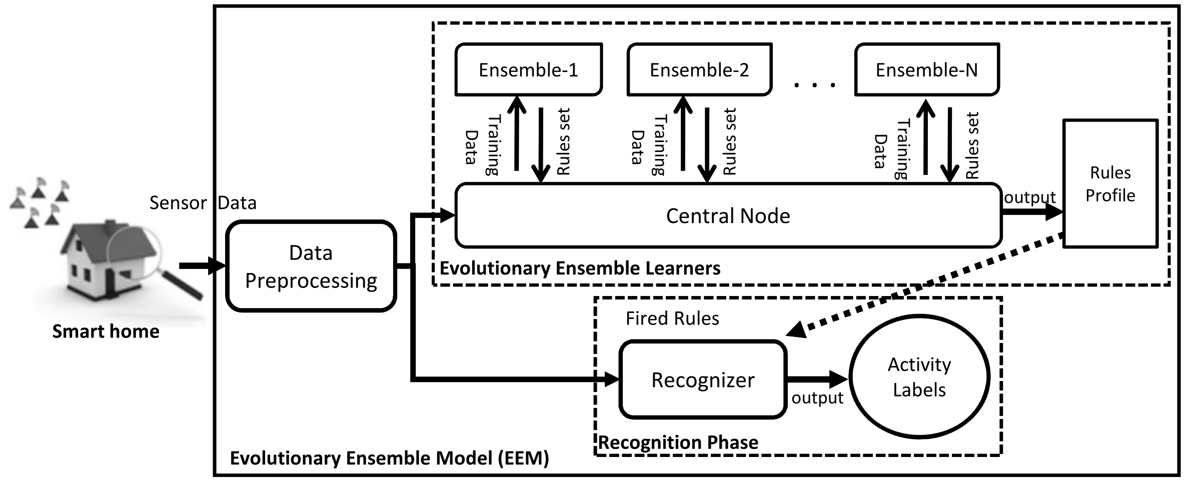


Fig. 1 The evolutionary ensemble model (EEM)

"don't care term". An Evolutionary Ensemble Learner (EL) for class c is a mapping from search space (S) to rule space (R) and is defined as:

$$\mathrm{EL}_{\mathrm{en}}^{c} : S \rightarrow R \tag{1}$$

The output of c-evolutionary ensemble learners ($\mathrm{EL}_{\mathrm{en}}$) is aggregated on the central node (cn) as a Rule Profile (RP) and depicted as:

$$\mathrm{RP}_{\mathrm{cn}} = \coprod_{c} \mathrm{EL}_{\mathrm{en}}^{c} \tag{2}$$

The proposed EEM for activity recognition is illustrated in Fig. 1.

It consists of three major components: (1) *Data preprocessing*: to represent the sensory temporal data as an observation vector (2) *the evolutionary ensemble learner*: to learn the activities during the training phase, and (3) *recognition phase*: to recognize the performed activities. Each component is described in more detail in the following sections.

3.1 Data preprocessing

In activity recognition, data preprocessing is an important step before applying any machine learning technique [15]. In the case of sensory data, preprocessing can be one of many different kinds, such as normalization of continuous values, handling missing values and streaming temporal data. In EEM, streaming temporal data is preprocessed using the event-based method. Each activity occurrence requires explicit mapping to sensor event time slices. In the dataset, performed activities starts and end time is recorded as a ground truth in an annotation file. Embedded sensors generate signals according to the subject interactions and maintain a log file with attributes start time, end time, sensor id and sensor value. We pick each performed activity time from annotation file and find all associated sensor instances from the log file to obtain training instances.

3.2 Evolutionary ensemble learners

Evolutionary ensemble learner is the training phase of EEM. It consists of ensemble nodes to learn the activities, central node for providing training data and combines the output of each ensemble to make a rule profile.

3.2.1 Ensemble nodes

After date preprocessing step, we divide the training dataset into disjunctive subsets. Each subset belongs to a unique class that is treated as a single population in our proposed approach. The population is processed by its own GA, which is capable of handling interruptions and non-deterministic activity sequences with a mutation operator and designed encoded chromosome. To train the model we encode chromosome and apply stochastic operators of the GA in the EEM as follows:

Encoding The well-known Michigan approach [16] is used to encode the sensor values, type and locations. Every sensor in the home environment is treated as a single gene; a set of genes is a chromosome that presents a single activity rule. Each activity rule consists of two portions. The antecedent portion is the logical combination of sensor values in the form of $sensor\_value_1 \cap sensor\_value_2 \cap sensor\_value_3, \ldots, sensor\_value_n$, and the subsequent portion represents the activity class C_i. The size of the activity rule is fixed depending on the number of deployed sensors in a smart home. The encoding scheme is shown in Fig. 2.

In Fig. 2, n bits represent the sensors values, and the rightmost bit (i.e., 4) shows the activity class label. The locus of each bit provides information about the sensor location. For instance the value "1" at locus "2" represents the microwave sensor state in the kitchen.

Fig. 2 Activity rule encoding

	Stove Sensor	Microwave Sensor	Fridge Sensor	Dishwasher Sensor	Freezer Sensor	Light Sensor	.	.	.	*n*-sensors	Activity Class
value	**1**	**1**	**0**	**1**	**1**	**1**	**.**	**.**	**.**	**n**	**4**
locus	**1**	**2**	**3**	**4**	**5**	**6**	**.**	**.**	**.**	**n**	**-**

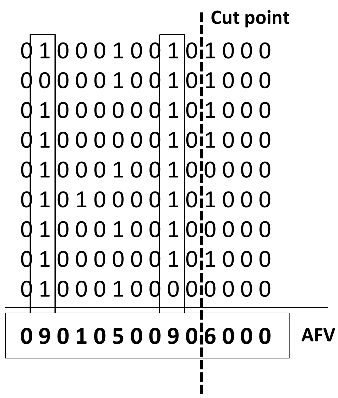


Fig. 3 AFV calculation and dynamic cutpoint operator

Selection Ranking-based selection [15] is implemented when the whole population is sorted from best to worst according to the ranked fitness values. In the proposed solution, each pair of parent selections incorporates low fitness activity rules with the best fit activity rules. After ranking, one parent is randomly selected from the top 50 % of the ranked population, while the other is randomly selected from the remaining population. This guarantees exploration of the whole search space for producing better offspring in the next generation.

Crossover Crossover is performed on the selected parents to create the new offspring. A dynamic single point crossover is applied as a reproduction operator. The cut point is decided on the basis of Augmented Feature Vector (AFV) which is calculated after each iteration (i.e., generation). The objective is to find generalized rules by identifying the most important sensors for a particular activity. It helps to avoid redundant offspring in the next generation. AFV is calculated by aggregating all the individual sensor values in the current population. The aggregated value for each sensor shows its overall importance. For example, in Fig. 3, the sensors at locus 2 and 9 have high aggregated value compared to the others. We decide to apply a dynamic cutpoint after the most important sensor values in order to carry the best portion of the activity rule to the next population. We adopt the fittest replacement mechanism to every iteration of the GA so that the entire generation is replaced with a new population by retaining the best fit in the last generation.

Fitness function After creating a new population, the next step is to measure the quality of the activity rules. We evaluated the fitness of each individual rule using reinforcement learning. The fitness function "F" evaluates the candidate rules on the basis of a reward and payoff mechanism [13] as follows:

$$\begin{aligned} F = \sum_{i}^{n}\sum_{j}^{m}\big[&reward\ (Activity\ rule_i|Search\ space_j) \\ &- payoff\ (Activity\ rule_i|Search\ space_j)\big] \end{aligned} \tag{3}$$

where,

$$reward = \begin{cases} 1 & if\ Activity\ rule \equiv Search\ space \\ & \qquad \cap\, classLabel \\ & \qquad \equiv Correct, \\ 0 & otherwise. \end{cases}$$

$$payoff = \begin{cases} -1 & if\ Activity\ rule \equiv Search\ space \\ & \qquad \cap\, classLabel \\ & \qquad \equiv Incorrect, \\ 0 & otherwise. \end{cases}$$

In (3), accuracy-based fitness function is defined to find optimal score of activity rules. In fitness score of activity rule, reward of $+1$ is added for correct classification and payoff of -1 is deducted in case of incorrect classification of each training instance.

Mutation The proposed approach inaugurates the diversity in activity rules to increase the fitness of individuals. The mutation operator assigns a "don't care" term—a value between 0, 1 and -1—on randomly selected genes of the activity rule. Interruption of sensor events is handled by introducing these don't care terms.

The stopping criterion for EEM is either a fixed number of generations or all training instances passed correctly. Later in the experiments and discussion section, we discuss the number of generations and the size of the population. The pseudocode for an ensemble node is depicted in Algorithm 1.

Algorithm 1: Ensemble Node Learner

```
Input:    C – Crossover rate
          λ – Mutation rate
          G – Number of generations
          μ – Population size
Output:   SRS – Specific Rule Set
Node Learner
  p = rand (μ)
  while (!max(G) || convg(G)) do
    fitness = fRankFitness(p)
    if !(fitness) then
      for m = 1: (⌊p(C)⌋) do
        pOne = rand(upper(p/2))
        pTwo = rand(lower(p/2))
        AFV = fAugFeatVec(p)
        Offspring = fcrossover (pOne, pTwo, AFV)
        mut = rand(⌊p(λ)⌋)
        SRS = offspring(mut)
end
```

3.2.2 Central node

A specific rule set from n-ensemble nodes is aggregated on the central node to create the activity rule profile described in Algorithm 2. It may have redundant and conflicting rules due to the overlapping region of the search spaces. So, we explicitly removed the duplicate rule instances. The problem of conflicting rules is resolved by giving priority to minor search spaces (i.e., minor activities) over major search spaces (i.e., major activities) as shown in Algorithm 3.

Algorithm 2: Central Node Processing

```
Input:    X (1..M) – Training Data
Output:   RP (1..N) – Rule Profile
Central Node Processing
  ensemble[] = unqiue(actClass(X))
  for m = 1: length(ensemble) do in parallel
    x = X(ensemble[m])
    EL = fEnsNodLearn(C, λ , G, μ, x)
    RP[m] =EL
end
```

3.3 Recognition phase

This phase recognizes activities based on sensor observation and activity rule profiles. For a particular set of sensor observations, rules are fired to recognize activity class labels. In the special case when more than one rule is fired, then conflicting class labels are resolved by majority voting. The pseudocode for the recognition phase is given in Algorithm 4.

Algorithm 3: Rule Conflict Resolving

```
Input:    RP (1..M) – Rule Profile
Output:   CRP (1..N) – Compact Rule Profile
Rule Conflict Resolving
  activityClass[] = unqiue(actClass(RP))
  for m = 1: length(activityClass) do
    uRP[:,m] = unique(activityClass(m))
    rCount[m] = [activityClass(m) length(uRP[:,m])]
  CRP = uRP[]
  for i = 1:length(CRP) do
    scanRule = CRP(i)
    for j = 1:length(CRP) do
      if isequal (scanRule(i),CRP(j)) then
        if (rCount[i] > rCount[j] then
          CRP[] = delete(CRP[i])
        else
          CRP[] = delete (CRP[j])
end
```

Algorithm 4: Recognizing the activities

```
Input:    RP (1..M) – Rule Profile
          TD (1..N) – Test Data
Output:   ACL – Activity Class Label
Recognizing the activities
  for m = 1: length(RP) do
    if match(sensObser(TD), RP(m)) do
      voteList[] = vote +1
    ACL = max(voteList)
end
```

4 Evaluation and results

In this section, we present the results to evaluate and validate the EEM to measure the accuracy level of recognized activities and to investigate the feasibility of the evolutionary ensembles model in the activity recognition domain.

4.1 Data sets description

The experiments are performed on three smart home datasets, two from MIT's House_n [2] and one from ISL [5]. For MIT's House_n, datasets were recorded in two apartments by deploying 77 and 84 sensors on everyday objects. Two volunteers performed daily life activities for two weeks. The

Table 1 Characteristics of the annotated activities in the House_n smart home

Activity	MITADS1			MITADS2		
	Num.	Time	Sensor	Num.	Time	Sensor
Toileting	85	128185.7	4084	40	60494.48	1599
Washing dishes	7	10534.42	274	21	31796.88	713
Preparing breakfast	14	21307.7	645	18	27511.95	702
Preparing lunch	17	25940.1	784	20	31006.33	733
Preparing dinner	8	12385.05	329	14	21538.88	549
Preparing a snack	14	21308.83	715	16	24214.25	581
Preparing a beverage	15	22850.68	599	–	–	–
Dressing	24	36033.93	1038	–	–	–
Bathing	18	27546.28	848	–	–	–
Grooming	37	55969	1682	–	–	–
Cleaning	8	12319.98	223	–	–	–
Doing laundry	19	28950.58	945	–	–	–
Going to work	12	17997.03	584	–	–	–
Taking medication	–	–	–	14	21183.23	590
Watching TV	–	–	–	15	23223.25	667
Listening to music	–	–	–	18	28469.97	701

Table 2 Characteristics of annotated activities in the ISL smart home

Activity	Num.	Time	Sensor
Idle	–	3507.27	38
Going out	33	17304.78	83
Toileting/toilet downstairs	114	198.7833	388
Bathing	23	219.8	52
Sleeping/going to bed	24	12335.25	173
Prepare breakfast	20	55.65	122
Prepare dinner	10	325.0333	125
Get a drink	20	17.75	62

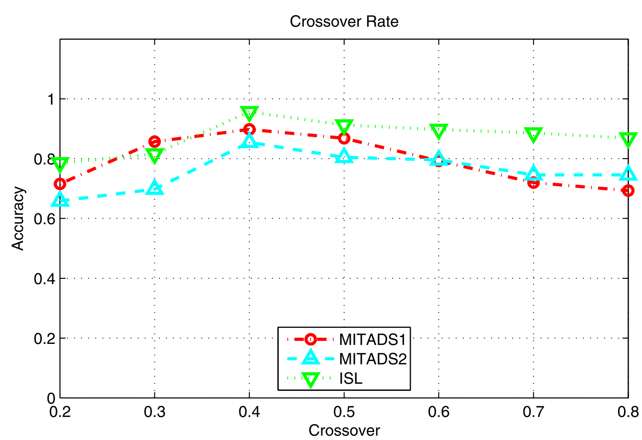


Fig. 4 Effect of crossover values

details description of the datasets and annotation method can be found in [2]. ISL data was collected from 14 binary sensors attached to the doors, cupboards, refrigerator, and toilet. A volunteer performed common household activities for 28 days.

In Tables 1 and 2 characteristic of MIT Activity Data Subject 1 (MITADS1), MIT Activity Data Subject 2 (MITADS2) and ISL dataset are shown. The 'Num' column shows activities count, 'Time' column shows the time in seconds and 'Sensor' column shows generated sensor events.

4.2 Performance measures

In order to evaluate our model, the three standard metrics of *precision, recall, and F-measure* are used as performance measures. They are calculated using the values of the confusion matrix [17] and computed as:

$$\text{Precision} = \frac{1}{Q}\sum_{i=1}^{Q}\frac{TP_i}{NI_i} \tag{4}$$

$$\text{Recall} = \frac{1}{Q}\sum_{i=1}^{Q}\frac{TP_i}{NG_i} \tag{5}$$

$$\text{F-Measure} = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{6}$$

where Q is the number of performed activities, TP is the number of true positives, NI is the total number of inferred labels and NG is the total number of ground truth labels.

4.3 Experiments and discussion

The model has been implemented in MATLAB 7.6. The configuration of the computer is an Intel Pentium(R) Dual-

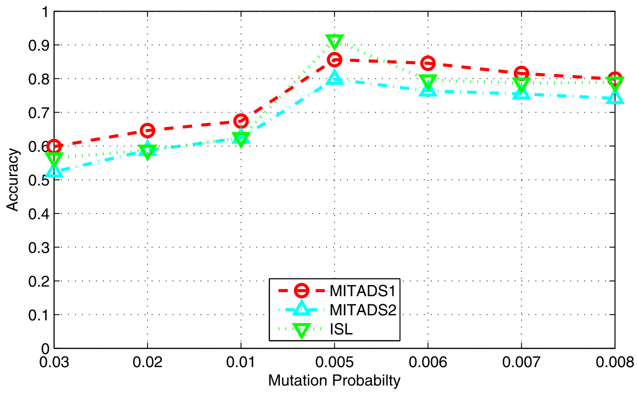


Fig. 5 Effect of mutation values

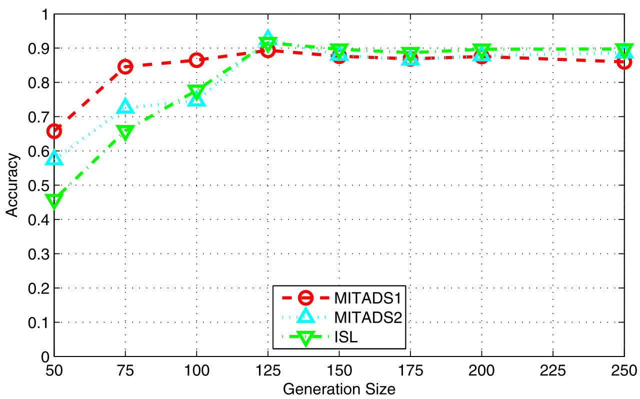


Fig. 6 Effect of number of generations

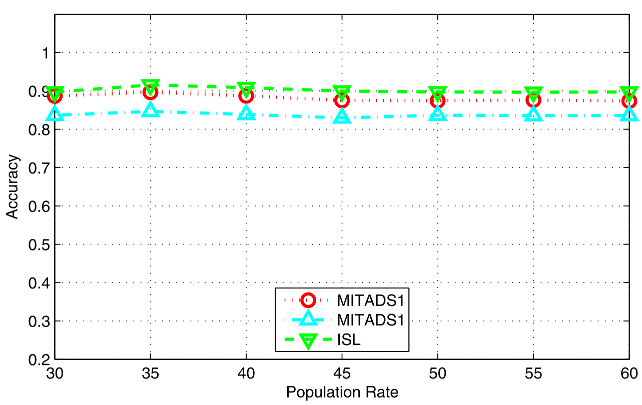


Fig. 7 Effect of population size

Core 2.5 GHz with 3 GB of memory and Microsoft Window 7. We split the dataset using the 'leave one day out' approach; therefore, the sensor readings of one day are used for testing and the remaining days for training. We evaluated different crossover rates to determine the optimal accuracy point. It is obvious from Fig. 4 that 0.4 is closer to the optimal parameter as compared to others. We analyzed the effect of the mutation rate with different values and discovered the optimal point at 0.005, as shown in Fig. 5. Small values of mutation make the solution stable, and values greater than 0.005 do not improve the accuracy.

Table 3 The confusion matrix of recognized activities in the MITADS1 smart home

Activity	Going out	Toileting	Bathing	Grooming	Dressing	Preparing breakfast	Preparing lunch	Preparing dinner	Preparing a snack	Preparing beverage	Washing dishes	Cleaning	Doing laundry
Going out	**12**	–	–	–	–	–	–	–	–	–	–	–	–
Toileting	–	**74**	5	1	–	–	–	–	2	1	–	2	–
Bathing	–	–	**18**	–	–	–	–	–	–	–	–	–	–
Grooming	–	–	–	**37**	–	–	–	–	–	–	–	–	–
Dressing	–	1	2	1	**18**	–	–	–	–	–	–	–	2
Preparing breakfast	–	–	–	–	–	**10**	–	–	1	3	–	–	–
Preparing lunch	–	–	–	–	–	1	**13**	3	–	–	–	–	–
Preparing dinner	–	–	–	–	–	–	–	**6**	2	–	–	–	–
Preparing a snack	–	–	–	–	–	2	2	3	**7**	–	–	–	–
Preparing beverage	–	2	1	–	–	1	–	–	–	**11**	–	–	–
Washing dishes	–	–	–	–	–	–	–	–	–	–	**5**	2	–
Cleaning	–	–	–	–	–	–	–	–	–	–	2	**4**	2
Doing laundry	–	2	2	–	–	–	–	–	–	1	1	–	**13**

Table 4 The confusion matrix for recognized activities in the MITADS2 smart home

Activity	Toileting	Taking medication	Preparing breakfast	Preparing lunch	Preparing dinner	Preparing a snack	Washing dishes	Watching TV	Listening music
Toileting	**34**	1	–	–	–	1	1	1	2
Taking medication	2	**8**	–	–	1	1	2	–	–
Preparing breakfast	–	–	**18**	–	–	–	–	–	–
Preparing lunch	–	–	1	**17**	–	2	–	–	–
Preparing dinner	1	–	1	1	**11**	–	–	–	–
Preparing a snack	1	5	–	2	1	**7**	–	–	–
Washing dishes	3	2	–	–	–	–	**14**	2	–
Watching TV	1	–	–	–	–	–	1	**13**	–
Listening music	2	1	–	–	–	–	–	–	**15**

Table 5 The confusion matrix for recognized activities in the ISL smart home

Activity	Going out	Toileting	Bathing	Go to bed	Prepare breakfast	Prepare dinner	Get a drink
Going out	**33**	–	–	–	–	–	–
Toileting	–	**109**	4	1	–	–	–
Bathing	–	–	**23**	–	–	–	–
Go to bed	–	3	2	**19**	–	–	–
Prepare breakfast	–	–	–	–	**18**	2	–
Prepare dinner	–	–	–	–	3	**7**	–
Get a drink	–	–	–	–	1	1	**18**

We analyzed different generation sizes for the convergence of EEM and observed a stable point after 125 generations, as no more significant improvements were found after this point, as shown in Fig. 6. Similarly, we analyzed different sizes of population, ranging from 30 to 60 and found the optimal point at 35, as depicted in Fig. 7. On the basis of the above analysis, we determined the optimal parameters as 0.4 crossover, 0.005 mutation, 125 generations and 35 population size. The results of our experiments are summarized in Tables 3, 4 and 5.

In Table 3, the result of the proposed EEM is presented in a confusion matrix for the MITADS1 dataset. The activities 'Going out,' 'Bathing' and 'Grooming' are recognized with 100 % accuracy. The most confusion takes place during the 'Preparing a snack' and 'Cleaning' activities. These were recognized correctly half of the time but misclassified for the remaining occurrences.

In the case of MITADS2, 'Preparing Breakfast' and 'Watching TV' are recognized with the highest accuracy, while the worst recognized activity is 'Preparing a snack,' which is correctly classified seven times and confused with other activities nine times, as shown in Table 4.

The confusion matrix for ISL dataset shows that 'Going out' and 'Bathing' are recognized with the highest accuracy. The most confusion takes place in the 'Go to bed' and 'Toileting' activities. To compare our proposed EEM with one of the most recent existing methods [18] and state of the art hidden Markov model (HMM), we kept all the data settings unchanged and reported the results in Figs. 8, 9 and 10. A remarkable improvement in terms of accuracy has been achieved compared to the previous work.

As can be seen from Fig. 8, our EEM model achieves significant improvement in all recognized activities in MITADS1 except 'Doing laundry' and 'Dressing' in comparison to Jehad et al. and 'Preparing Beverage' in comparison to HMM. We achieved remarkable improvement for 'Bathing,' 'Grooming' and 'Cleaning' as major activities. For minor activity 'Washing dishes', our accuracy is more than 100 % as compared to existing methods.

In the case of MITADS2, we achieved significant improvement in all recognized activities against both existing methods except 'Taking medication' for HMM. The most noticeable improvements for major activities are in case of 'Preparing breakfast,' 'Preparing a snack,' 'Watching TV' and 'Listening to music'. In case of minor activity 'Preparing Dinner', our accuracy rate is 81 % high as compared to existing methods as shown in Fig. 9.

In Fig. 10, the ISL dataset results show either similar or high accuracy with Jehad et al. and considerable improvement is achieved for all activities in comparison to HMM; however, in the case of major activity 'Go to bed' and for minor activity 'Preparing dinner,' we achieved significant improvement compared to both existing methods.

It can be seen from Fig. 11, our proposed model EEM shows stable results. The maximum accuracy is either high or similar, while the minimum and average accuracy is always high in all three datasets.

Fig. 8 The MITADS1 activity recognition results

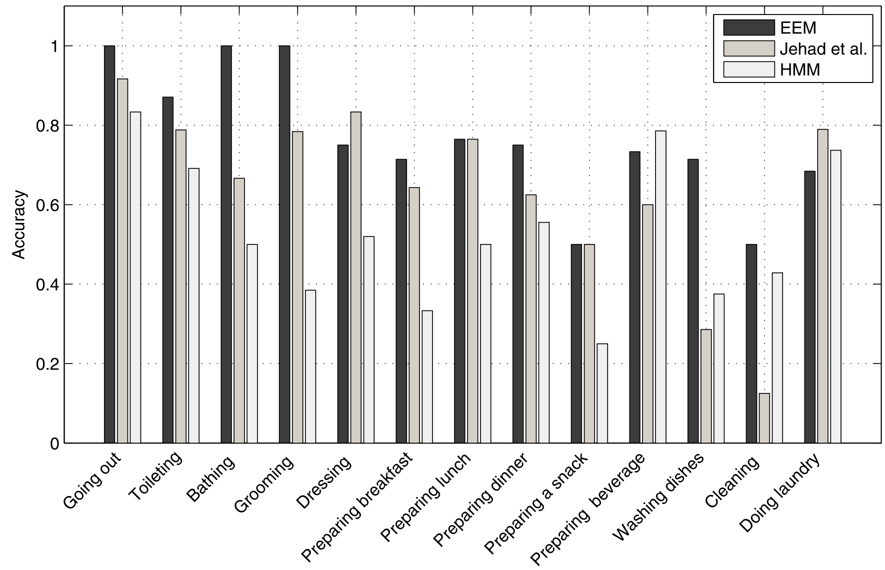


Fig. 9 The MITADS2 activity recognition results

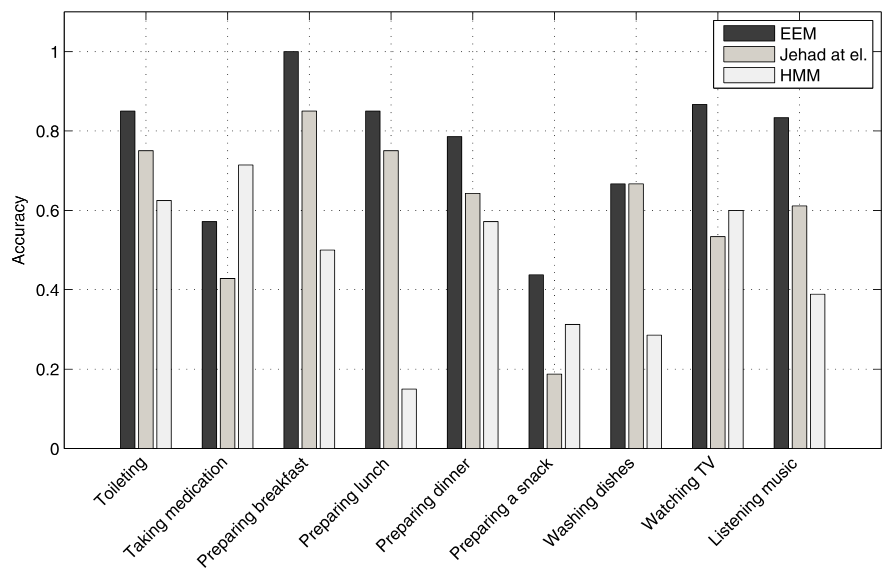
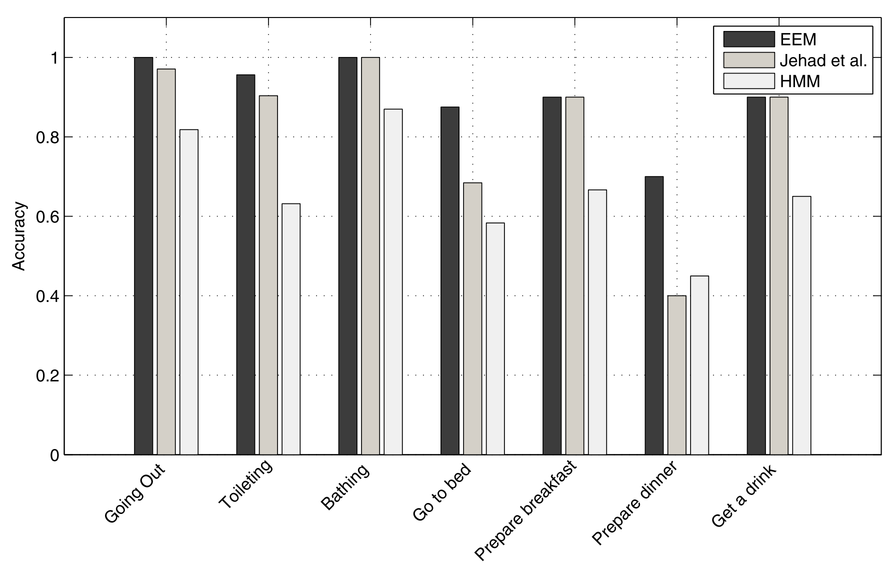


Fig. 10 The ISL activity recognition results



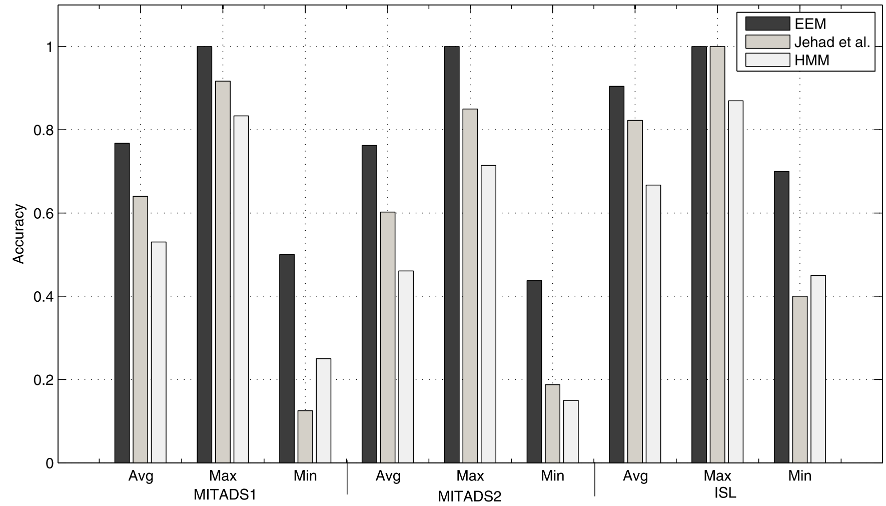


Fig. 11 Accuracy comparison (Avg, Max and Min)

Table 6 Precision, Recall, F-measure and Accuracy

Dataset	Model	Precision	Recall	F-measure	Accuracy
MITADS1	EEM	0.7515	0.6909	0.7199	0.7678
	Jehad et al.	0.6668	0.6401	0.6532	0.6401
	HMM	0.5308	0.5321	0.5314	0.5303
MITADS2	EEM	0.7721	0.7624	0.7672	0.7623
	Jehad et al.	0.6550	0.6204	0.6372	0.6022
	HMM	0.5015	0.4587	0.4792	0.4608
ISL	EEM	0.8997	0.9044	0.9020	0.9044
	Jehad et al.	0.8264	0.8065	0.8163	0.8226
	HMM	0.7130	0.6575	0.6843	0.6670

On the basis of the confusion matrices presented in Tables 3, 4 and 5, we computed four performance measures: precision, recall, F-measure and accuracy, as shown in Table 6. For all the datasets, the EEM performed better for all four measures. The highest accuracy of 16 % is achieved in the case of MITADS2 with a 13 % increase in F-measure in comparison to Jehad et al. In case of HMM, we achieved 30 % more accurate results for ISL dataset with 28 % increase in F-measure.

5 Conclusion and future work

Accurate activity recognition and understandable output are very important for many practical and healthcare applications. Nevertheless current approaches to activity recognition do not handle the problems of major/minor activities and their non-deterministic nature. To solve these problems, we investigated an evolutionary technique with the ensemble paradigm. We proposed a novel model to distinguish major and minor activities and to address their non-deterministic nature. The model is evaluated on three publically available smart home datasets, and the optimal parameters for in-depth investigation are determined. In all of the experiments, an approximately 6–30 % higher accuracy is observed. Experimental results demonstrated that handling the issues discussed above consistently increased accuracy for each considered activity.

In this study, only a single inhabitant is considered at a time; therefore complications may arise due to the presence of several residents in a home. This limits the applicability of this model at present; however, the generic nature of training and implementation will lead to the success of EEM for conceivable complex situation. Our future plan includes handling multiple residents and recognizing interleaved and parallel activity under the framework of evolutionary ensembles.

Acknowledgement This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2011-0030823)

References

1. Parisa RC, Diane JH, Lawrence B, Maureen S (2011) Discovering activities to recognize track in a smart environment. IEEE Trans Knowl Data Eng 23(4):527–539

2. Munguia TE, Intille SS, Larson K (2004) Activity recognition in the home setting using simple and ubiquitous sensors, PERVASIVE
3. Pavan T, Chellappa R, Subrahmanian VS, Udrea O (2008) Machine recognition of human activities: a survey. IEEE Trans Circuits Syst Video Technol 18(11):1473–1488
4. Vinh LT, Lee S, Xuan HL, Hung QN, Hyoung K, Manhyung H, Lee YK (2011) Semi-Markov conditional random fields for accelerometer-based activity recognition. Appl Intell 35(2):226–241
5. Kasteren T, Noulas A, Englebienne G, Krose B (2008) Accurate activity recognition in a home setting. In: Proceedings of the 10th international conference on ubiquitous computing, New York, 2008
6. Shi LD, Shi YH, Gao Y, Shan L, Bin Y (2011) XCSC: a nove approach to clustering with extended classifier system. Int J Neural Syst 21(1):79–93
7. Bull L, Studley M, Bagnall A, Whittley I (2007) Learning classifier system ensembles with rule-sharing. IEEE Trans Evol Comput 11(4):496–502
8. Kuncheva LI, Jain LC (2000) Designing classifier fusion systems by genetic algorithms. IEEE Trans Evol Comput 4(4):327–336
9. Kim KJ, Cho SB (2008) An evolutionary algorithm approach to optimal ensemble classifiers for DNA microarray data analysis. IEEE Trans Evol Comput 12(3):377–388
10. Folino G, Pizzuti C, Spezzano G (2010) An ensemble-based evolutionary framework for coping with distributed intrusion detection. Genet Program Evol Mach 11(2):131–146
11. Cuntoor NP, Yegnanarayana B, Chellappa R (2008) Activity modeling using event probability sequences. IEEE Trans Image Process 17(4):594–607
12. Liao L, Choudhury T, Fox D, Kautz H (2007) Training conditional random fields using virtual evidence boosting. In: Proceedings of the International Joint Conference on Artificial Intelligence, Hyederabad, 2007
13. Chen L, Nugent CD, Wang H (2011) A knowledge-driven approach to activity recognition in smart homes. IEEE Trans Knowl Data Eng 99:1
14. Zhu C, Cheng Q, Sheng W (2010) Human activity recognition via motion and vision data fusion. In: IEEE asilomar conference on signals, systems, and computers, Pacific Grove, 2010
15. Mitchell T (1997) Machine learning. McGraw Hill, Columbus
16. Puig AO, Mansilla EB (2008) Evolutionary rule-based systems for imbalanced data sets. Soft Comput 13(3):213–225
17. van Kasteren TLM, Alemdar H, Ersoy C (2011) Effective performance metrics for evaluating. In: Second workshop on context-systems design, evaluation and optimisation, Italy, 2011
18. Jehad AMS, Lee YK, Lee S (2010) A smoothed naive Bayes-based classifier for activity recognition. IETE Tech Rev 27(2):107–119
19. Parisa R, Cook DJ (2010) Mining and monitoring patterns of daily routines for assisted living in real world. In: Proceedings of the 1st ACM international health informatics symposium, New York, 2010
20. Popescu M, Florea E (2008) Linking clinical events in elderly to in-home monitoring sensor data: a brief review and a pilot study on predicting pulse pressure. J Comput Inf Sci Eng 2(2):180–199
21. Xing H, Qu R (2011) A compact genetic algorithm for the network coding based resource minimization problem. Appl Intell 36(4):809–823
22. Shin KS, Jeong YS, Jeong MK (2011) A two-leveled symbiotic evolutionary algorithm for clustering problems. Appl Intell 36(4):788–799