Noname manuscript No.

(will be inserted by the editor)

Discovering Varying Patterns of Normal and Interleaved ADLs in Smart Homes

Mahsa Raeiszadeh · Hooman Tahayori* · Andrea Visconti

Received: date / Accepted: date

Abstract People may do the same activity in many different ways, hence modeling and recognizing that activity based on data gathered through simple sensors like motion sensor is a complex task. In this paper, we propose an approach for activity mining and activity tracking which identifies frequent normal and interleaved activities that individuals perform. With this capability, we can track the occurrence of regular activities to monitor users and detect changes in an individual's behavioral pattern and lifestyle. We have tested the proposed method using the datasets of Washington State University CASAS and the Massachusetts Institute of Technology (MIT) smart home projects. The obtained results show considerable improvements compared with existing methods.

Keywords smart home · activity mining · normal and interleaved ADLs

Mahsa Raeiszadeh
Department of Computer Science and Engineering and IT, Shiraz University, Shiraz, Iran
E-mail: mahsa.raeiszadeh@shirazu.ac.ir

Hooman Tahayori
Department of Computer Science & Engineering and IT, Shiraz University, Shiraz, Iran
* corresponding author
Tel.: +98-71-3613 3544
E-mail: tahayori@shirazu.ac.ir

Andrea Visconti
Department of Computer Science, Universita degli Studi di Milano,via Comelico 39/41,
20135, Milano, Italy
E-mail: andrea.visconti@unimi.it

1 Introduction

Smart homes are composed of various services and technologies which are set up to improve the quality of life of occupants [1]. Smart homes, on one hand improve home comfort, convenience, security and the healthcare of the residents and on the other hand enable management of energy, water and other resources [2]. To fulfill these aims, smart homes rely on knowledge about residents' behavior and their preferences. Gathering the required knowledge is based on different types of sensors installed in the smart home and the supporting communication technologies [3]. Sensors range from simple motion sensors to video cameras. Sensed data should be transmitted where they can be processed. However, due to the volume of data that should be transmitted, costs of installation and maintenance of the sensor network, the required computational efforts for processing the gathered data, simple sensors like motion detection, light, temperature, humidity, door sensors and alike are more preferred to be exploited in smart homes. Such sensors gather numerical data that is easier to process in comparison with images that would be gathered by cameras.

Everyday routines and basic tasks performed by residents are known as Activities of Daily Living (ADLs) [4]. Analyzing the residents' recorded activities enables determining the patterns of various behaviors. By recognizing normal behavior of residents, a monitoring system can be developed that can identify anomalies, help reduce power consumption, maximize residents' comfort and so forth [5]. However, due to human intrinsic uncertainties and based on the fact that people may do the same activity in many different ways, modeling people's activities is complex.

In this paper, based on data that are gathered from simple sensors like motion and door sensors installed in smart homes, we present a supervised method for discovering activities of residents of smart homes. The discovered behavioral patterns can be used for elderly care, health care, improving safety, security, comfort, managing power consumption, etc.

To discover an individual's ADLs, we propose *Discovery Method for Varying Patterns* (DMVP) of activities in smart homes. DMVP identifies sensor event sequences that probably belong together and appear frequently enough to comprise an activity that can be tracked. DMVP combines the *frequent sequential pattern miner* to identify frequent patterns and the *Shortest Super Subsequences* (SSS) method to find a container for common patterns. Since people perform tasks in different ways, discontinuous and varied order patterns may arise. By varying pattern, we refer to such discontinuous and varied order sensor event occurrences that may all represent the same activity.

We use a classification model to predict and recognize activities in a smart environment based on the discovered patterns of activities. The testbeds for validating our work is a part of the CASAS smart home project [6] and the Massachusetts Institute of Technology (MIT) smart home dataset [7]. In the conducted tests, our proposed method has demonstrated its efficiency for identifying and its suitability for detecting discontinuous and interleaved activity pattern.

The rest of the paper is organized as follows. In section 2, related work is reviewed. In Section 3, a theoretical description of the proposed method is presented. Section 4 provides experimental analysis in the CASAS and MIT testbeds for normal and interleaved ADLs. In section 5 a discussion is presented. Section 6 concludes the paper.

2 Related Work

Activity recognition is a common process in smart home which is aimed to detect or recognize human activities and behavioral patterns in real-life environments. Existing algorithms for tracking and recognizing individual's activities mainly rely on data that are gathered from various sensors installed in smart homes.

In [4] the problem of recognizing human activity in everyday routines via supervised learning algorithms like Markov and hidden Markov models in physical environments is discussed. This paper investigates several possible ways of applying a probabilistic model to learn activities when they are performed not only in a sequential fashion, but also in complex scenarios like when different activities are interleaved together or are performed concurrently by multiple residents. A number of supervised machine learning models for Human Activity Recognition (HAR) are proposed. Naive Bayesian network has been used with promising results for offline learning of activities [7-9]. It identifies the activity that corresponds with the greatest probability to the set of sensor values that are observed. However, its performance depends on the volume of data; the larger the volume, the higher the performance. Lombriser et al. in [10] have proposed a sequential activity prediction using k-Nearest Neighborhood (K-NN) to detect everyday activities such as walking, watching TV, etc. K-NN is robust to noisy training data but as it is a lazy supervised learner, it runs slowly. Support Vector Machines (SVM) [11] and back propagation neural networks [12-13] are also used for detecting smart home residents' behaviors. Mehr et al. in [12], have investigated detecting and recognizing resident activities such as bathing, grooming, going to work, etc. by using Artificial Neural Network (ANN), Quick Propagation (QP), Levenberg Marquardt (LM) and Batch Back Propagation (BBP) algorithms.

Other researchers have employed decision trees to find logical description of the activities. This approach often generates understandable rules and if weights of the attributes are determined rationally, results can be more appealing than other methods [14]. It is shown that decision tree can deal with noisy data that is expected from sensors in smart home however, it has the problem of overfitting.

In [15] and [16], probabilistic models such as Hidden Markov Models (HMM) for modeling user activities are exploited. It is a popular method for identifying the spatiotemporal relationships between the sensor data [17]. The drawback of using the HMM model is that if the data volume is too large then it will have a long run-time [17]. Also other variations of HMM such as Hierarchical Hidden Semi-Markov Models (HHSMMs) are used for human activity recognition. HHSMM is a type of Dynamic Belief Network (DBN) for tracking the ADLs [7]. The algorithm can individualize several activities such as “asleep" and “having meals" despite of existing noisy information. Although DBNs have proven to be one of the most powerful representation method for temporal events that efficiently fuses information from multiple sensors [7], the complexity of the networks and learning algorithms makes it difficult to be applied in problems involving hundreds of sensors.

Unsupervised methods are also used for recognizing human activities. Jakkula et al. [18] have used k-means clustering approach to cluster the sensor readings into events with a centroid. K-means is easy to implement for activity recognition but it is difficult to predict the K-value. Rashidi et al. [19], have proposed a context-driven algorithm and a clustering algorithm to identify frequent patterns that may be either continues -with the same order events- or discontinuous -which might have variability in ordering of events.

The aim of [20] is to discuss the possibilities of recognizing and predicting user activities in the smart home environment. This paper suggests the combination of a pattern clustering method and artificial neural network. In [21], a hierarchical clustering algorithm is used in a distributed environment. Validation measures like entropy, coefficient of variance and time are applied to evaluate algorithm's accuracy. Although it is easy to be implemented however, for large datasets it has a long run-time and poor quality of clusters. In [22], the Expectation-Maximization (EM) algorithm is used to cluster similar objects into one group. The algorithm is simple and fast but depending on the number of input features, the number of objects, and also the number of iteration, it's performance would be variable.

In [23] it is shown that Self Organizing Map (SOM) can aid computer forensic investigators to determine anomalous behaviors. The Self Organizing Map (SOM) classifies objects into their appropriate clusters and get higher accuracy compared to the k-means and the EM-clustering algorithm. But, its performance decreases when the number of clusters k increases and shows poor results on large datasets.

3 Proposed Method

The success of applications of smart home like elderly care, security and comfort improvement, depends on the known preferences and behaviors of the residents. Knowing what activities individuals do and do not at a certain time and place, enables preparing the environment as the resident wishes. It also enables detecting an intrusion or deciding if any health threatening issue has happened. Toward the aim of detecting activities and modeling the behavior of the residents of a smart home, we propose a supervised method for discovering the activities that frequently occur by an individual in a smart home. We will then build models to recognize and predict these activities as they occur. Our proposed method is consisted of three phases that are schematically depicted in Fig. 1.

Phase 1 of the method is devoted to preprocessing the raw sensor data and converting them into event sequences. In phase 2, *Discovery Method for Varying Patterns* (DMVP) identifies frequent sequential patterns from event sequences. Frequent Sequential Pattern Mining Algorithm (FSPMA) as a part of DMVP, discovers the most frequent sequential patterns from the sensor event data. Then, Shortest Super Subsequences (SSS) method is applied to find the longest subsequences common to all sequences in the set of pattern sequences. In the next step, DMVP’s output is fed to a pattern classification method to model them. The output of the phase 3 enables determining when and where the activities occur, based on which related action in a smart home would be taken. In the sequel we describe each phase in details.

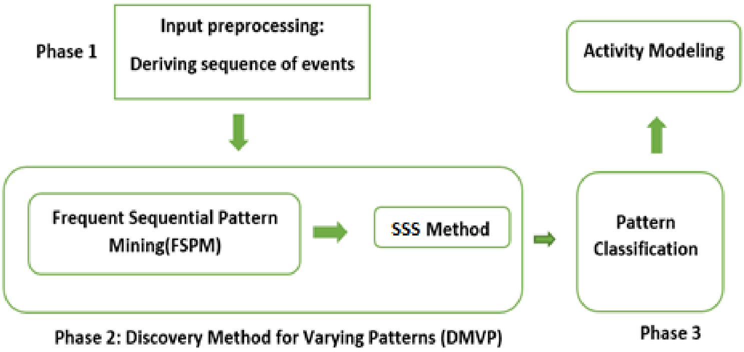

Phase 1
Input preprocessing:
Deriving sequence of events
Activity Modeling
Frequent Sequential Pattern Mining(FSPM)
SSS Method
Pattern Classification
Phase 2: Discovery Method for Varying Patterns (DMVP)
Phase 3

Fig.1 Architecture of the proposed method for discovering and tracking activities

3.1 Phase 1- Preprocessing

Preprocessing helps improving the quality of data, efficiency and ease of the mining process and consequently, improves the quality of the mining results. In the preprocessing phase, raw sensor data will be converted into sensor event sequences. An *event sequence* is an ordered sequence of sensor events, each described by a fixed set of sensor id and sensor state features.

We expect raw sensor data be represented in the descriptive form of [Date-Time, Encoded Sensor Id, Sensor Status]. Our proposed method needs the pieces of data that are mentioned in the aforementioned descriptive form to (1) order the sensed data based on their date-time stamps (2) construct sensor events.

Sensor id, sensor state and the number of activation or deactivation of the sensors are considered for deriving a sensor event. We represent an event in the format of XXXYZZZ where XXX, represents encoded ID of the sensor, Y denotes the sensor status and ZZZ indicates the number of times the sensor is activated or deactivated to that point. The status of a sensor i.e. Y, can be 1(=ON, OPEN, PRESENT, …) or 0(=OFF, CLOSE, ABSENT, …).

As an example, assume motion sensor M08 is coded as “8” in a smart home. For the first time that the sensor data related to M08 like [2011-11-04 12:43:27 M08 ON] is seen, after removing the date and time, the event “811” is derived based on the ON status of motion sensor no.8. If the next sensor data is [2011-11-04 12:43:30 M08 OFF], since it is the first time that motion sensor no.8 is deactivated, “801” will the corresponding event. Given the third sensor data represented in a descriptive form be [2011-11-04 12:43:31 M08 ON], its corresponding event would be “812”.

3.2 Phase 2 - Discovery Method for Varying Patterns (DMVP)

Phase 2 is consisted of two sub-phases namely Frequent Sequential Pattern Mining Algorithm (FSPMA) and Shortest Super Subsequence (SSS). It is notable that each of the two sub-phases of DMVP reduces the dataset. We will demonstrate each of these sub-phases in the following.

3.2.1 Frequent Sequential Pattern Mining Algorithm (FSPMA)

Frequent patterns are subsequences of events that appear in a dataset with frequency greater than or equal to some specific threshold [24]. In FSPMA first the frequent and repeated patterns of event sequences are identified. Then a model is created for recognizing activities.

A wide range of algorithms for identifying frequent sequences from a sequence database exist. PrefixSpan algorithm as a pattern growth-based approach, discovers sequential patterns in sequence database [25]. We use PrefixSpan to find event sequence patterns from discontinuous instances. PrefixSpan algorithm is shown in Fig. 2. As an example, this algorithm extracts pattern {811,711} from event sequences {901,801,811,711}, {811,711,211} and {811,202,711}, where the events are discontinuous. Unlike many other sequence mining algorithms, PrefixSpan algorithm reports a general pattern that contains all repeated conversion of a single pattern that occurs in the input dataset.

PrefixSpan Algorithm

- Input of the algorithm:

A sequence database S, and the minimum support threshold min_support.

- Output of the algorithm:

  The complete set of sequential patterns.

- Subroutine: PrefixSpan(α, L, S|α).
- Parameters:
  - α: sequential pattern,
  - L: the length of α;
  - S|α: : the α-projected database, if α ≠<>;

    otherwise; the sequence database S.
- Call PrefixSpan(<>,0,S)

Method:

1. Scan S|α once, find the set of frequent items b such that:
   - b can be assembled to the last element of α to form a sequential pattern; or
   - <b>can be appended to α to form a sequential pattern.
2. For each frequent item b, append it to α to form a sequential pattern α', and output α';
3. For each α', construct α'-projected database S|α', and call PrefixSpan(α', L+1, S|α').

Fig.2 PrefixSpan Algorithm[25]

3.2.2 Shortest Super Subsequence (SSS)

Since PrefixSpan algorithm produces sequential frequent patterns which can be subset of other discovered frequent patterns, it is needed to omit sequences that are subset of others. Thus to reduce the number of sequential patterns generated by the PrefixSpan in previous step, we use Longest Common Subsequences (LCS) method to find the shortest super sequences. Let the input sequences be X[1..m] and Y[1..n] of lengths m and n respectively. In effect, LCS (X [1 . . . m] , Y [1 . . . n]) determines the longest common subsequences for all possible combination of the input strings. The recurrence relation for relating the length of the LCS for each prefix X [1 . . . m] , Y [1 . . . n] is as follows [26]:

$$LCS[m,n]=\begin{cases}0 & m=0 \;\; or \;\; n=0\\ LCS[m-1,n-1]+1 & x[m]=y[n]\\ \max(LCS[m,n-1],LCS[m-1,n]) & x[m]\neq y[n]\end{cases} \qquad (1)$$

In effect LCS finds the length of longest common subsequences. However, we need to find the shortest super sequence i.e. the shortest sequence that contains other sequences. For example, given FSPMA finds three event sequences $\{711\}$, $\{711, 911\}$ and $\{711, 911, 701\}$. These are then fed to LCS. LCS compares $\{711, 911\}$ with $\{711\}$ and gives $\{711\}$ as a common subsequence between the two sequences. Since the number of events in a common subsequence and the number of event in $\{711, 911\}$ are not equal, $\{711, 911\}$is kept and $\{711\}$ is ignored. Then LCS compares $\{711, 911\}$ with $\{711, 911, 701\}$. With the same reasoning, $\{711, 911, 701\}$ is kept as the shortest super sequence of the event sequences.

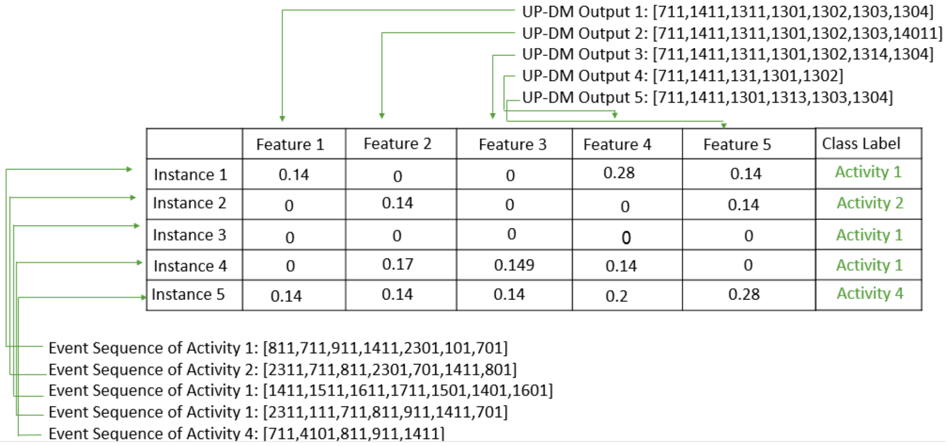
3.3 Phase 3 - Predictive Model

As stated in Sec 3.1 a sequence of sensor events - that is derived based on sensors activation and deactivations - represents an activity an individual performs. In order to create our new dataset to build model, we consider each event sequence of an activity as an instance and consider the sequential pattern that is resulted from DMVP as a feature. Also we consider activities as class labels. We use the Levenshtein (edit) distance [19] - *sim(A, B)* - to calculate the similarity between each event sequence with all sequential patterns that are resulted from DMVP-see Fig. 3.

In (2) the edit distance, *e (A, B)*, is the number of edits (insertions, deletions, and substitutions) required to transform an event sequence A into another event sequence B. Then we classify the patterns to build a model for recognizing future occurrences of activities.

$$sim(A,B) = 1 - \frac{e(A,B)}{\max(|A|,|B|)} \qquad (2)$$

We exploit Random Forest, K-NN, SVM, Decision Tree classifiers to recognize activities in a smart home. Each model is trained to recognize the patterns that correspond to the classification representatives found by pattern classification models. The values for the parameters and the effect of choosing each classifier on the performance of the proposed approach are extensively discussed in Sec.4.



UP-DM Output 1: [711,1411,1311,1301,1302,1303,1304]
UP-DM Output 2: [711,1411,1311,1301,1302,1303,14011]
UP-DM Output 3: [711,1411,1311,1301,1302,1314,1304]
UP-DM Output 4: [711,1411,131,1301,1302]
UP-DM Output 5: [711,1411,1301,1313,1303,1304]

	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Class Label
Instance 1	0.14	0	0	0.28	0.14	Activity 1
Instance 2	0	0.14	0	0	0.14	Activity 2
Instance 3	0	0	0	0	0	Activity 1
Instance 4	0	0.17	0.149	0.14	0	Activity 1
Instance 5	0.14	0.14	0.14	0.2	0.28	Activity 4

Event Sequence of Activity 1: [811,711,911,1411,2301,101,701]
Event Sequence of Activity 2: [2311,711,811,2301,701,1411,801]
Event Sequence of Activity 1: [1411,1511,1611,1711,1501,1401,1601]
Event Sequence of Activity 1: [2311,111,711,811,911,1411,701]
Event Sequence of Activity 4: [711,4101,811,911,1411]

Fig.3 Sample of new database

4 Experimental Results

We have examined our proposed method on the data that is collected in the CASAS testbed, developed by D. Cook in the project of ”Learning setting - generalized activity models for smart spaces”[6] and the MIT dataset. In this section we describe the results obtained by applying our proposed algorithm on both datasets. It is notable that our method is applicable on any data set as far as from each sensor data the descriptive form of [Date-Time, Encoded Sensor Id, Sensor Status] can be derived.

4.1 WSU CASAS Dataset

We have tested and evaluated our proposed method with WSU CASAS dataset in two different realistic scenarios to detect normal and interleaved ADLs. Fig. 4, shows the layout of the WSU CASAS project smart home that includes three bedrooms, one bathroom, a kitchen, and a living/dining room. The activities of the residents inside the smart home are recorded using motion, door, temperature and item sensors. Motion sensors are numbered and their ID is initiated by “M” - that is followed by the sensor number. State of each motion sensor can take a value of ON/OFF. “D” denotes door sensor whose state can be either OPEN or CLOSE. Temperature sensor IDs are initiated by “T” whose values indicate temperature in Fahrenheit. Item sensors are used to monitor oatmeal, raisins, brown sugar, bowl, measuring spoon whose state values can be PRESENT or ABSENT. WSU CASAS dataset is consisted of several files. Name of each file is the label of the activity in which its related sensor data are recorded. Sensor data gathered in WSU CASAS are tagged with the date and time, indicating when the data was collected, sensor id and the state of the sensor. Table 1 depicts few records of a file of this dataset as an example.

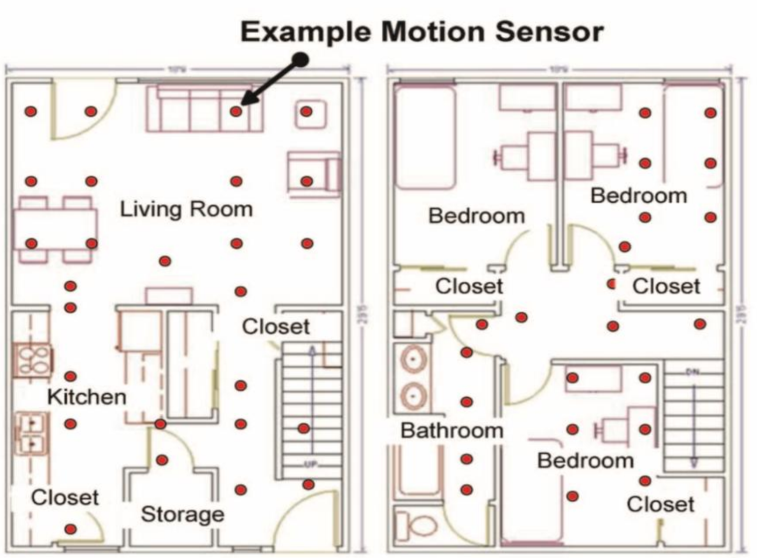


Fig.4 Layout of smart apartment used in CASAS for data collection. The positions of motion sensors are indicated by circle[19]

Table 1 Sample of WSU CASAS sensor data

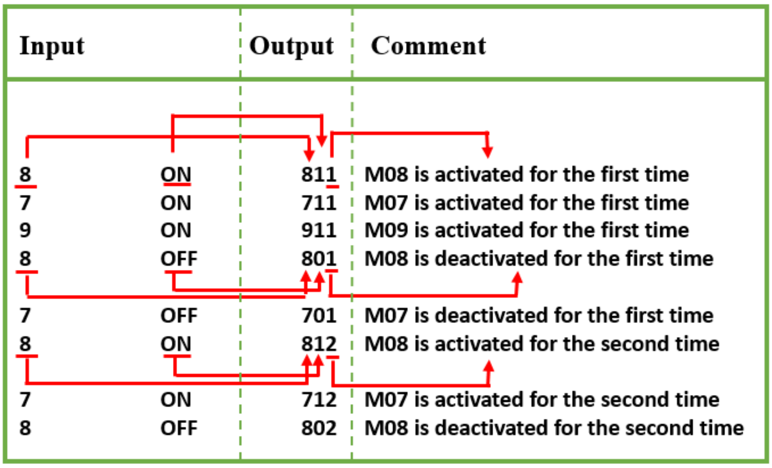
2011-11-04	12:51:01.15	M	ON
2011-11-04	12:51:01.60	D	OPEN
2011-11-04	12:52:23.87	T	12
2011-11-04	12:52:33.79	I0	PRESEN
2011-11-04	12:52:44.71	M	ON
2011-11-04	12:52:46.94	M	OFF
2011-11-04	12:52:49.87	D	CLOSE

Table 2 Encoding CASAS Sensor IDs

Sensor ID	M01	M02	M03	….	M051	I01	I02	…	I08	D01	D02	….	D12	T01	T02	T03
Encoded ID	1	2	3	…	51	52	53	…	59	60	61	…	71	72	73	74

Table 3 A part of raw sensor data for eating activity- CASAS Dataset

2011-11-04	12:43:27.41	M	O
2011-11-04	12:43:27.84	M	O
2011-11-04	12:43:28.48	M	O
2011-11-04	12:43:30.28	M	OF
2011-11-04	12:43:31.49	M	OF
2011-11-04	12:43:31.49	M	O
2011-11-04	12:43:32.18	M	O
2011-11-04	12:43:34.23	M	OF



Input		Output	Comment
8	ON	811	M08 is activated for the first time
7	ON	711	M07 is activated for the first time
9	ON	911	M09 is activated for the first time
8	OFF	801	M08 is deactivated for the first time
7	OFF	701	M07 is deactivated for the first time
8	ON	812	M08 is activated for the second time
7	ON	712	M07 is activated for the second time
8	OFF	802	M08 is deactivated for the second time

Fig 5 Event sequence derivation from the raw sensor data shown in Table 3, *Event Sequence= {811,711, 911, 801, 701, 812, 712, 802}.*

In order to derive the descriptive form of [Date-Time, Encoded Sensor Id, Sensor Status] for the collected sensor data in WSU CASAS dataset, initially we have used the sensor encoding that is depicted in Table 2. As an example, using such encoding event sequence shown in Fig. 5 is derived from Table 3 which shows raw sensor data for an eating activity.

4.1.1 Normal ADL discovery

Normal ADL refers to the cases when inhabitants in a smart home concentrate on performing only one activity at a time. These activities are performed separately, with no interleaving or interruptions. In WSU CASAS, 24 users have performed the following five normal activities:

1. Phone Calling: The participant looks up a specific number from phonebook located at the living room table and calls the number. The participant listens to the recorded message and writes down the cooking instructions.
2. Hand Washing: The participant washes hands in the kitchen sink.
3. Cooking: The participant cooks oatmeal on the stove according to the recorded instructions, adding brown sugar and raisins (from the kitchen cabinet) once done.
4. Eating: The participant eats oatmeal together with a glass of water and takes some medicine to the living room table.
5. Cleaning: The participant cleans the dishes in the kitchen sink, and return all items used to their respective locations.

Data were recorded for each of the five activities performed by all 24 users. Hence the total number of 120 sets of data was gathered; 24 sets of data for each type of activity that were manually labeled. Specifically, each set of data that belongs to one activity was labeled with the corresponding activity id. To test our method, each of the 120 datasets is being fed to the preprocessing phase to be converted to event sequences. Then we put together all event sequences that were derived for each normal activity into a separate dataset - we have 5 datasets for normal activities. 70% of each dataset is allocated for training. The remaining 30% of each dataset is referred to test datasets. Afterwards, we have applied the DMVP algorithm to each training dataset separately. In the DMVP phase, we discovered frequent sequential patterns in the sensor event data by FSPMA. We experimentally set the minimum support thresholds -minSup- of frequent event, to 0.95 and the threshold of max pattern length for each activity to the minimum length of sensor sequences of users for the intended activity. When we analyzed the sensor events, the algorithm in total discovered 18325 patterns with the lengths varying from 1 to 11 events. Then SSS method removed patterns that are subset of another patterns. Finally, DMVP's output was 583 patterns with the lengths varying from 2 to 11 events.

Using the training data, in the phase 3 we have exploited the four classification models, namely Random Forest, SVM, K-NN and Decision Tree for creating model which is then exploited to recognize future activities of the users. The four approaches are applied on the same dataset and the experimental setup is identical for all of them. Fig. 6, compares the accuracy achieved by each method. The accuracy of predicting the activity "Phone Calling", "Hand Washing" and "Cooking" showed up a maximum value with each of the four models and "Cleaning" activity get approximately the minimum accuracy prediction than the others. Calculating the average accuracy value for each classification model shows that Random Forest, SVM and Decision Tree has reached the highest accuracy level of 95% and the minimum accuracy level of 92.5% is achieved by KNN.

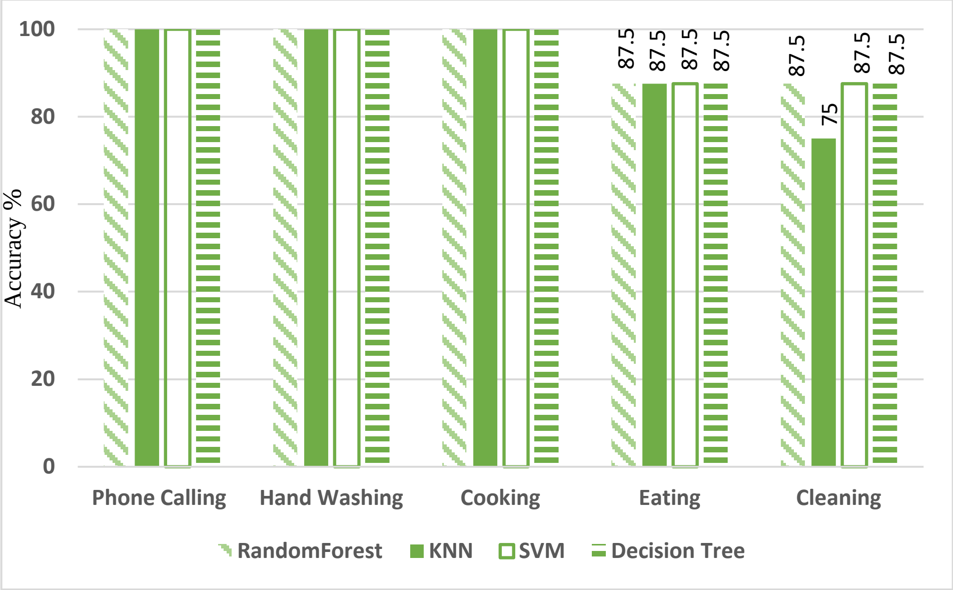


Fig.6 Comparison of the accuracy of the four predictive models for normal ADLs

4.1.2 Interleaved ADL Discovery

In our second experiment, we examined our method to identify interleaved activities which are often performed not only in isolation (i.e. sequential), but also in complex (i.e. interleaved and concurrent) manner. For example, when an individual prepares a meal, the steps to do the job, do not always follow the same sequence, rather, their order may be changed and be interleaved with steps that do not consistently appear each time. We have use CASAS dataset for interleaved activities in which 24 participants have performed eight activities in the smart apartment. The eight ADLs are as follow.

1. Fill Medication Dispenser: The participant moves to the kitchen, retrieves a pill dispenser and bottle of pills, and follows directions to fill the dispenser.
2. Watch DVD: The participant moves to the living room, puts a DVD in the player and watches a news clip on TV.
3. Water Plants: The participant retrieves a watering can from the kitchen supply closet and waters three plants.
4. Converse on Phone: The participant answers the phone when it rings and hangs up after finishing the conversation.
5. Write Birthday Card: The participant fills out a birthday card with a check to a friend and addresses the envelope.
6. Prepare Meal: The participant moves to the kitchen and prepares cup of noodle soup in the microwave, following the directions on the package. The participant brings the soup and a glass of water to the dining room table.
7. Sweep and Dust: The participant sweeps the kitchen floor and dusts the dining and the living room using the supplies located in the kitchen closet.
8. Select an Outfit: The participant selects an outfit from the clothes closet for a job interview.

In CASAS testbed, first the participants have performed each activity at a time in sequential order. The participants then have performed all of these activities by interweaving them in any way as they like with the goal of being efficient in performing the tasks. For example, while preparing meals in the kitchen, if the phone rings in the living room, the user stops cooking, goes to the living room, attend the call and continue the task in the kitchen. In this run of performing the tasks in an interleaved manner, the order in which different activities were performed and interleaved was left to the discretion of the participant. As different participants interweaved the task differently, the dataset obtained was significantly richer. The data collected for both sets of experiments was manually labeled with the corresponding activity id.

Similar to the first experiment, there were 24 users for the eight activities, hence we have 192 datasets. Preprocessing phase converts these datasets to event sequences. Then we have applied FSPMA to 70% of data and discovered 41197 sequential frequent patterns for all eight activities, with the lengths varying from 1 to 11 events. The parameters, minSup and max pattern length for each activity were defined as in the previous experiment. After FSPMA, SSS method reduced 41197 patterns to 744 patterns, with the same lengths varying from 1 to 11 events, as of FSPMA.

Then we build the models for recognizing future activities. After the models are build, we have tested them with test data. Fig. 7, similar to Fig. 6, compares four predictive models in interleaved ADLs. The accuracy of predicting the activities “Watch DVD”, “Write Birthday card” and “Select an Outfit” showed up a maximum value in each four models. Random Forest model has reached the maximum average accuracy of 91.07%, but in contrast to the normal ADLs, SVM has shown less accuracy.

Ultimately we have compared performance of the two experiments on each prediction model. Accuracy for detecting normal ADL in any of the four models is from 92.5% to 95%, while for interleaved ADLs is 80.35%-94.64%. In Fig. 8, it is shown that Random Forest prediction model has reached the highest accuracy in the two experiments. Moreover, it can be obviously seen that higher accuracy is achieved on normal activity routines of residents.

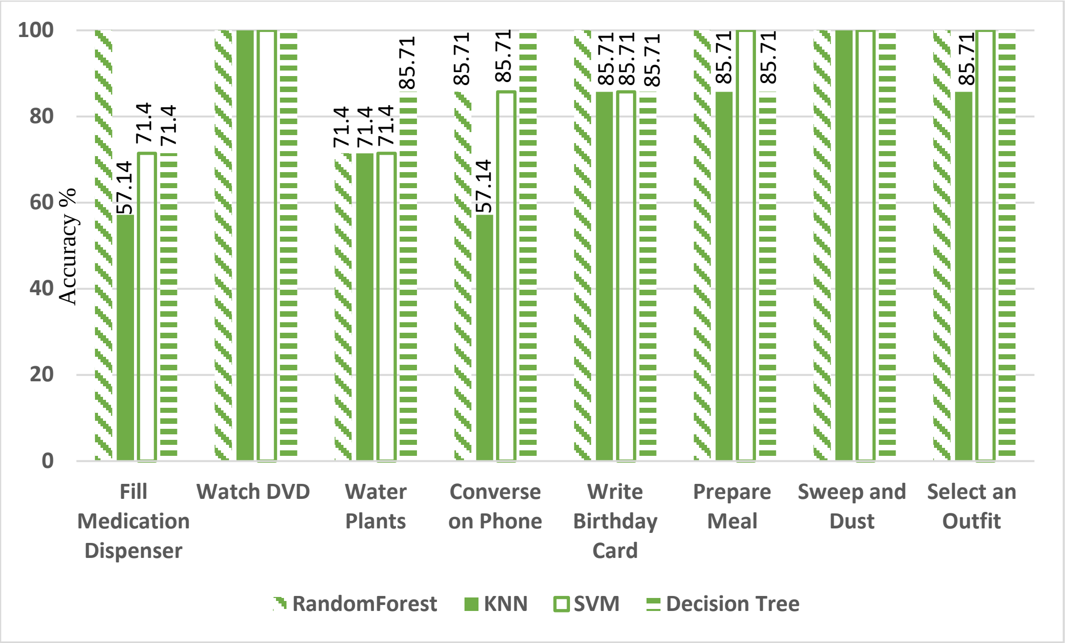


Fig.7 Comparison of the accuracy of the four models for interleaved ADL

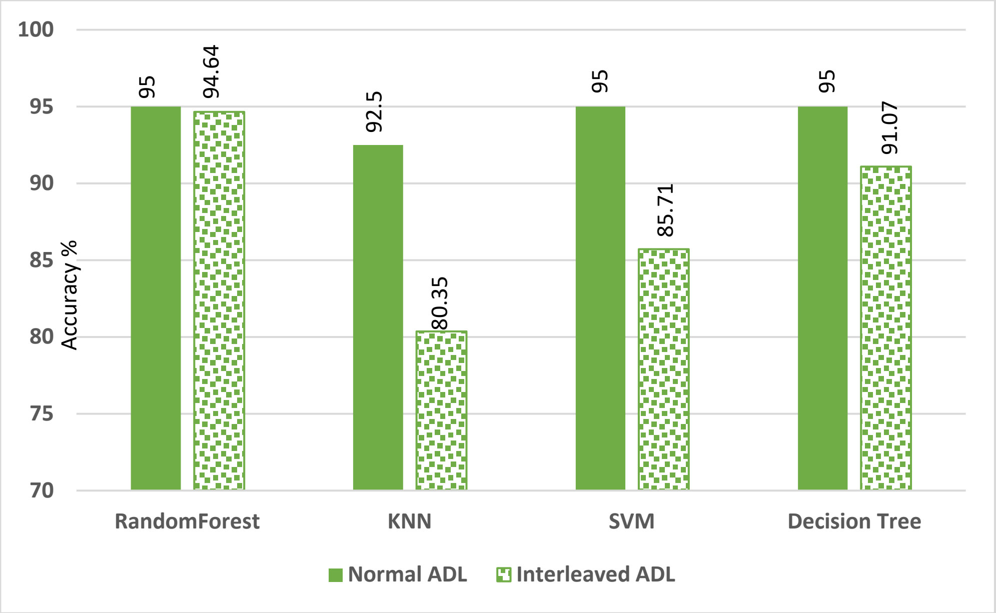


Fig.8 Comparison of the accuracy of all techniques in recognizing normal and interleaved activities

4.2 MIT Dataset

We have also examined our proposed method on the dataset that is collected in the MIT smart home testbed. Fig. 9 depicts the layout of the two smart homes in MIT project. The first subject who lived in the apartment shown in Fig. 9(a) was a 30-year-old professional woman who spent her free time at home, and the resident of the second apartment that is shown in Fig. 9(b) was an 80-year-old woman who spent most of her time at home. Both subjects lived alone in the one-bedroom apartment.

77 sensors were installed in the first subject's apartment and 84 in the second subject's apartment. The sensors were installed on doors, windows, cabinets, drawers, microwave ovens, refrigerators, stoves, sinks, toilets, showers, light switches, lamps, some containers (e.g. water, sugar, and cereal), and electric/electronic appliances (e.g. DVDs, stereos, washing machines, dish washers, coffee machines). In this project, ADLs are categorized in 13 classes. Table 4 shows the ADL classes and the number of activities each individual has performed in each category, during the project (14 days).

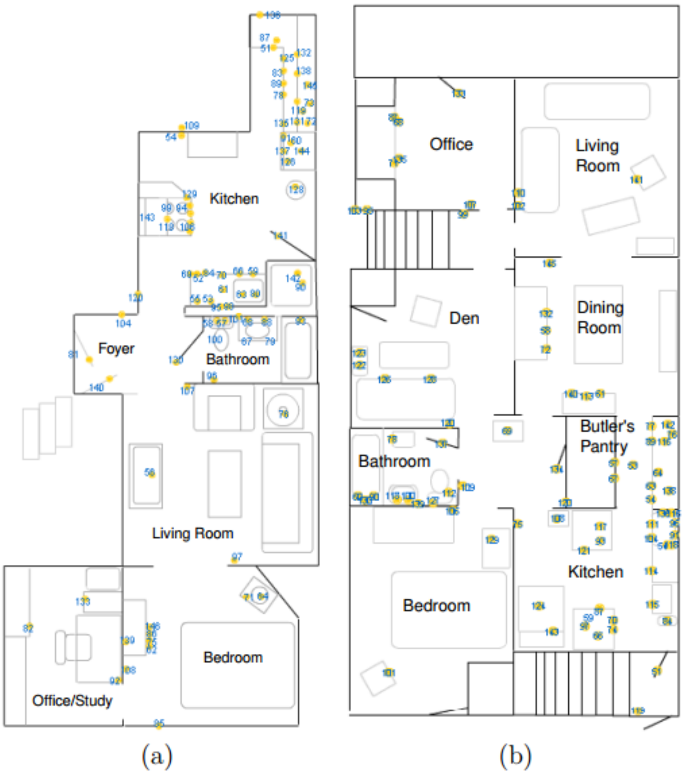


Fig.9 Layout of two smart apartments used for data collection in MIT smart home project [7]

Table 4 13 classes of ADLs individuals performed in MIT smart home project

Activity Class	Number of activities performed by Subject 1	Number of activities performed by Subject 2
Bathing	18	3
Cleaning	9	3
Doing laundry	19	0
Dressing	20	5
Going out to work	12	0
Grooming	37	2
Preparing a beverage	15	1
Preparing a snack	15	16
Preparing breakfast	14	18
Preparing dinner	8	14
Preparing lunch	17	20
Toileting	83	37
Washing dishes	8	20
TOTAL	**275**	**139**

Table 5 MIT dataset format

ACTIVITY-LABEL	DATE	START TIME	END TIME
SENSOR1-ID	SENSOR2-ID	….	….
SENSOR1-OBJECT	SENSOR2-OBJECT	….	….
SENSOR1-ACTIVATION-TIME	SENSOR2-ACTIVATION-TIME	….	….
SENSOR1-DEACTIVATION-TIME	SENSOR2-DEACTIVATION-TIME	….	….

The format of recorded sensor data in MIT dataset is shown in Table 5. Start and end time shows when the activity started and ended respectively. Sensor-id is a number representing the ID of a sensor which is unique for each sensor. Sensor-Object represents where the sensor was installed e.g. on kitchen cabinet, door, window, etc. Sensor-activation and Sensor-deactivation respectively represent the sensor activation and deactivation time in seconds. Table 6 depicts a sample of data records in MIT smart home dataset.

In order to make MIT dataset usable with our proposed method, initially we had to put the raw sensor data in the descriptive form of [Date-Time, Encoded Sensor Id, Sensor Status]. Since MIT dataset stores the time of activation and deactivation of each sensor, it is necessary to convert activation and deactivation time of each sensor to its underlying descriptive state (e.g. ON/OFF). In some cases, depending on the type of activity and the way the resident performs activity, it is possible that deactivation time of some sensors occurs after activation time of next sensors. Table 7 shows the “lunch preparation” activity which is then converted to [Date-Time, Sensor Id, Sensor Status] format, that is shown in Table 8. Then, the sensor data is converted to sequence of events. As described before, Fig. 10, shows event sequence that is derived from Table 8. At the end of phase 1, a sequence of events is extracted for each activity.

Table 6 Sample of MIT Dataset

Preparing lunch	**Day1**	**11:21:17**	**11:38:22**	
140	137	31	53	131
Door	Freezer	Toaster	Cabinet	Toaster
11:23:04	11:23:55	11:24:08	11:34:59	11:35:12
11:23:07	11:24:03	11:24:14	11:35:01	11:35:22
Dressing	**Day2**	**8:25:31**	**8:26:46**	
57	75	139		
Dress cabinet	Drawer	Jewelry box		
8:24:49	8:26:02	8:26:14		
16:45:08	8:26:11	8:26:21		
Bathing	**Day3**	**15:02:53**	**15.46.12**	
93	100	93	101	96
Shower faucet	Toilet Flush	Shower faucet	Light Switch	Exhaust Fan
15:05:46	15:07:49	15:14:09	15:40:56	15:44:07
15:14:09	16:03:54	15:36:29	16:17:07	16:17:10

Table 7 Sensor data for a lunch preparing activity-MIT dataset

Preparing lunch	Day1	11:21:17	11:38:22	
140	137	31	53	131
Door	Freezer	Toaster	Cabinet	Toaster
11:23:04	11:23:55	11:24:08	11:34:59	11:35:12
11:23:07	11:24:03	11:24:14	11:35:01	11:35:22

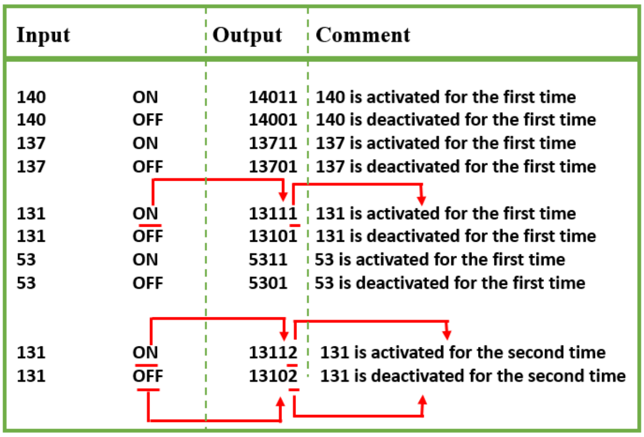
4.2.1 ADL Discovery-Subject 1

To test our method, dataset is being fed to the preprocessing phase to be converted to event sequences. Then we have applied the DMVP algorithm to 70% of dataset. In the DMVP phase, we discovered frequent sequential patterns in the sensor event data by FSPMA. We experimentally set the minimum support thresholds -minSup- of frequent event, to 0.8 and the threshold of max pattern length for each activity to the minimum length of sensor sequences for intended activity. The algorithm in total discovered 155 patterns with the lengths varying from 1 to 5 events. Then SSS method removed patterns that were subset of other patterns. Finally, DMVPs output was 32 patterns with the lengths varying from 1 to 5 events.

Setting test and train data as in previous experiment we achieved the best accuracy with Random Forest, hence, here we have only reported that in Table 9. As is shown in Table 9, the average accuracy level for the first subject is 93.64%.

Table 8 Conversion of the activity of Table 7 to [Date-Time, Sensor Id, Sensor Status] format

11:23:04	140	ON
11:23:07	140	OFF
11:23:55	137	ON
11:24:03	137	OFF
11:24:08	131	ON
11:24:14	131	OFF
11:34:59	53	ON
11:35:01	53	OFF
11:35:12	131	ON
11:35:22	131	OFF



Input		Output	Comment
140	ON	14011	140 is activated for the first time
140	OFF	14001	140 is deactivated for the first time
137	ON	13711	137 is activated for the first time
137	OFF	13701	137 is deactivated for the first time
131	ON	13111	131 is activated for the first time
131	OFF	13101	131 is deactivated for the first time
53	ON	5311	53 is activated for the first time
53	OFF	5301	53 is deactivated for the first time
131	ON	13112	131 is activated for the second time
131	OFF	13102	131 is deactivated for the second time

Fig.10 Event sequence derivation from the raw sensor data shown in Table 8 *Event Sequence= {14011, 14001, 13711, 13701, 13111, 13101, 5311, 5301, 13112, 13102}*.

Table 9 Accuracy result for first and second subject ADLs with the RandomForest predictive model

Activities	First Subject (by RandomForest)	Second Subject (by RandomForest)
Bathing	100%	100%
Cleaning	88.88%	66.66%
Doing laundry	89.47%	0
Dressing	95%	80%
Going out to work	100%	0
Grooming	100%	50%
Preparing a beverage	100%	100%
Preparing a snack	86.66%	87.5%
Preparing breakfast	100%	100%
Preparing dinner	87.5%	85.71%
Preparing lunch	82.35%	80%
Toileting	100%	97.20%
Washing dishes	87.5%	85%
TOTAL	**93.64%**	**71.70%**

4.2.2 ADL Discovery-Subject 2

Similar to the first experiment, preprocessing phase converted the data set to event sequences. Then we have applied FSPMA to training set and discovered 81 sequential frequent patterns for all activities, with the lengths varying from 1 to 10 events. The parameters, minSup and max pattern length for each activity were defined as in the previous experiment. After FSPMA, SSS method reduced 81 patterns to 30 patterns, with the same lengths of events in FSPMA. The next step of the process is to recognize and track the discovered activities as they occur in the smart apartment. The test data are obtained similar to subject 1 experiment. We built our model based on the discovered activities and applied the algorithm to the test data to identify activities.

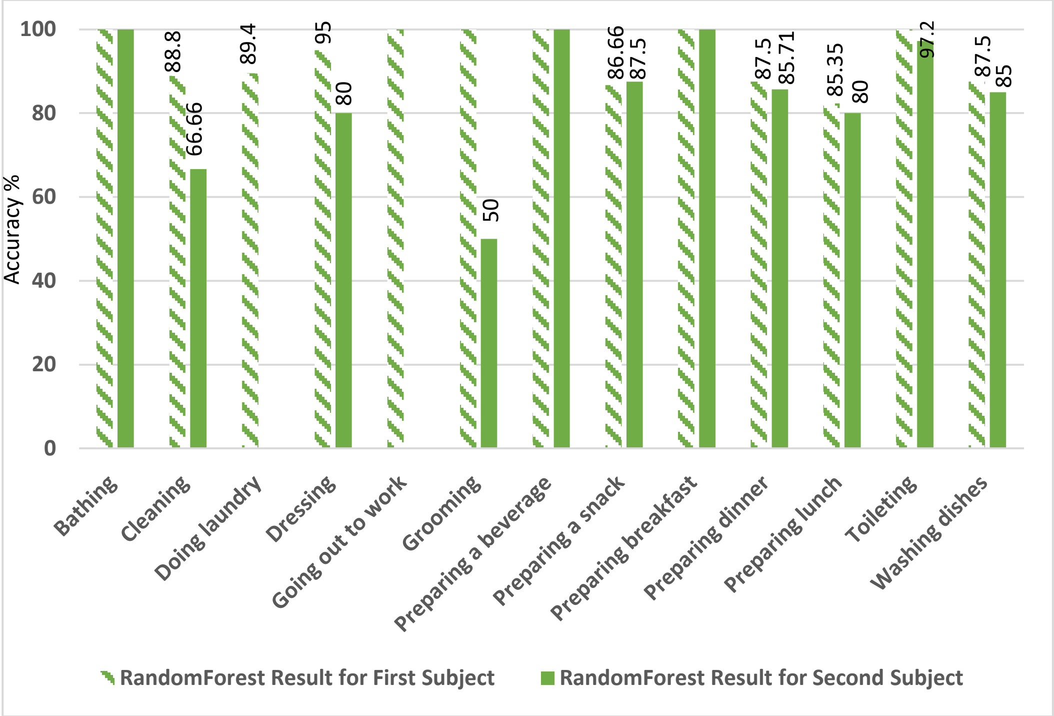


Fig.11 Comparison of the accuracy of recognizing ADLs for subject 1 and 2

Second column of Table 9, demonstrates that, the accuracy of predicting the activity “Bathing”, “Preparing a beverage” and “Preparing breakfast” has achieved a maximum value of 100% in RandomForest model and “Grooming” activity, unlike the previous experiment with achieved the minimum accuracy prediction. The total accuracy result for this subject has reached 71.70%. Fig. 11 shows the overall accuracy of each activity separately by prediction model.

The overall accuracy of activity detection for the first subject has been higher than for the second subject. This can be demonstrated according to these three reasons (1) the number of sensors that were noisy was higher in the second subject apartment [7], and (2) the quality of the labeling for the first subject was higher and (3) the first subject spent less time at home and the sensor firings were not as complex as those for the second subject.

5 Discussion

We have compared our proposed method with [19],[7],[15],[12] and [20]. Results are reported in Table 10. In [15], a supervised method for recognizing activities when they are interwoven in complex and realistic scenarios based on WSU CASAS testbed is introduced. This approach has proposed two probabilistic techniques Naive Bayes and HMM for activity recognition. The aim of this approach is providing a tool to automatically monitor and assist elderly individuals. However, its accuracy to detect both normal and interleaved ADL in contrast to our method is lower.

In Rashidi et al.'s work [19], an unsupervised automated approach for activity tracking that identifies frequent activities of individuals in smart home is introduced. Their algorithm is similarly validated on CASAS testbed with which they have reached the accuracy of 73.8% and 77.3% respectively for normal and interleaved ADLs. Although their approach can be useful for health monitoring but since the nature of their model is unsupervised, choosing the right number of clusters and correlating the cluster of sensor reading to the activities is difficult.

Another unsupervised learning method called k-pattern clustering was proposed based on WSU CASAS testbed in Bourobou's work [20] to predict user activities that utilizes artificial neural network. Although it is an efficient method to group and identify the user activity model, however it has a clustering problem similar to [19]. The accuracy of this method that is reported for normal ADLs has reached 78%.

In Mehr et. al's work [12], three algorithms of artificial neural networks, namely Quick Propagation (QP), Levenberg Marquardt(LM) and Batch Back Propagation(BBP), have been used for HAR. These algorithms are validated in MIT dataset smart home. The achieved results demonstrate that Levenberg Marquardt algorithm has better performance (by 92.81% accuracy) than Quick Propagation and Batch Back Propagation algorithms but they all have reached lower accuracy in contrast to our proposed method. The reason lies in intrinsic disadvantages of ANN such as slow convergence speed, less generalizing performance, arriving at local minimum and over-fitting problems.

In [7], a supervised method for recognizing activities based on MIT smart home testbed is introduced. This approach has used a probabilistic technique like Naive Bayes classifier for HAR. However, its accuracy to detect both subject 1 and subject 2 ADLs in contrast to our method is lower.

We have calculated precision, recall, F-score in addition to accuracy for our proposed method that are reported in Table 11. The obtained results show the effectiveness of the proposed method.

6 Conclusion

In this paper, we have introduced a novel method to recognize ADLs in smart environments. Our proposed method consists of three phases namely preprocessing, DMVP and predictive modeling. Preprocessing converts raw sensor data to event sequences which are then fed to DMVP to discover frequent activities that naturally occur in homes and then classification model is applied to predict activities. We have exploited four models to learn activities when they are performed not only sequentially but also when they are interleaved with others. We have applied our proposed method on normal and interleaved activities. The predictive model works effectively and reached the average accuracy of 94.37% for normal ADLs and 87.94% for interleaved ADLs for WSU CASAS dataset and 93.64% for the first subject and 71.70% for the second one in MIT dataset. We have compared our work with other works which have used the same datasets of WSU CASAS and MIT for normal and interleaved activities and the results shows our proposed method outperforms them.

Table 10 Comparison of the proposed method with [7], [12],[15], [19] and [20]

	Method	Dataset	Accuracy
Proposed Method	Supervised (DMVP+RandomForest)	CASAS	Normal ADL: 92%-95% Interleaved ADL: 80%-94%
		MIT	Subject 1: 93.64% Subject 2: 71.70%
Singla [15]	Supervised	CASAS	Interleaved ADL Naive Bayes: 66.08% HMM: 71%
Rashidi [19]	Hybrid Unsupervised (Clustering + HMM)	CASAS	Normal ADL: 73.8% Interleaved ADL: 77.3%
Bourobou [20]	Hybrid Unsupervised (K-pattern clustering +NN)	CASAS	Normal ADL: 78%
Tapia [7]	Supervised (Naive Bayes Classifier)	MIT	Subject 1: 60.6% Subject 2: 41.09%
Mehr [12]	Supervised (Algorithms of ANNs)	MIT	Subject 1 QP : 89.23 % LM: 92.81% BBP:87.61%

Table 11 Four measures of Accuracy, Precision, Recall, F-score for studied cases

	Accuracy	Precision	Recall	F-Score
CASAS Normal ADL	95%	96%	95%	95.4%
CASAS Interleaved ADL	94.64%	96.25%	94.64%	95.4%
MIT Subject 1	93.64	97%	96%	96.5%
MIT Subject 2	71.70%	88.44%	93.52%	90.9%

References

[1] M. Raeiszadeh and H. Tahayori, “A novel method for detecting and predicting resident’s behavior in smart home,” in *6th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS).*, 2018, pp. 71–74.

[2] P. Mahya, H. Tahayori, and A. Sadeghian, “An Online Demand Response EMS with Anomaly Usage Detection,” *5th IEEE Int. Conf. Smart Energy Grid Eng.*, pp. 271–275, 2017.

[3] Z. Liouane, T. Lemlouma, P. Roose, F. Weis, and H. Messaoud, “An Improved Extreme Learning Machine Model for the Prediction of Human Scenarios in Smart Homes,” *Springer J. Appl. Intell.*, vol. 48, no. 8, pp. 2017–2030, 2017.

[4] H. Mshali, T. Lemlouma, M. Moloney, and D. Magoni, “A Survey on Health Monitoring Systems for Health Smart Homes,” *Elsevier Int. J. Ind. Ergon.*, vol. 66, pp. 26–56, 2018.

[5] A. Munoz, J. C. Augusto, A. Villa, and J. A. Botia, “Design and evaluation of an ambient assisted living system based on an argumentative multi-agent system,” *Pers. Ubiquitous Comput.*, vol. 15, no. 4, pp. 377–387, 2011.

[6] P. Rashidi and D. J. Cook, “Keeping the resident in the loop: Adapting the smart home to the user,” *IEEE Trans. Syst. Man, Cybern. Part ASystems Humans*, vol. 39, no. 5, pp. 949–959, 2009.

[7] E. M. Tapia, S. S. Intille, and K. Larson, “Activity recognisation in Home Using Simple state changing sensors,” *Pervasive Comput.*, vol. 3001, pp. 158–175, 2004.

[8] O. Brdiczka and J. L. Crowley, “Learning Situation Models in a Smart Home Learning Situation Models in a Smart Home,” *IEEE Trans. Syst. Man, Cybern. Part ASystems Humans*, vol. 39, no. 1, pp. 56–63, 2016.

[9] T. van Kasteren and B. Krose, “Bayesian activity recognition in residence for elders,” *3rd IET Int. Conf. Intell. Environ. (IE 07)*, vol. 2007, pp. 209–212, 2007.

[10] C. Lombriser, N. B. Bharatula, D. Roggen, and G. Tröster, “On-body activity recognition in a dynamic sensor network,” *Proc. ICST 2nd Int. Conf. Body Area Networks*, pp. 1–6, 2007.

[11] M. Allahviranloo and Will Recker, “Daily activity pattern recognition by using support vector machines with multiple classes,” *Transp. Res. Part B*, vol. 58, pp. 16–43, 2013.

[12] H. D. Mehr, H. Polat, and A. Cetin, “Resident Activity Recognition in Smart Homes by Using Artificial Neural Networks,” *Smart Grid Congr. Fair (ICSG), 2016 4th Int. Istanbul*, 2016.

[13] J. Favela, M. Tentori, L. A. Castro, V. M. Gonzalez, E. B. Moran, and A. I. Martinez-Garcia, “Activity recognition for context-aware hospital applications: Issues and opportunities for the deployment of pervasive networks,” *Mob. Networks Appl.*, vol. 12, no. 2–3, pp. 155–171, 2007.

[14] U. Maurer, A. Smailagic, D. P. Siewiorek, and M. Deisher, “Activity recognition and monitoring using multiple sensors on different body positions,” *Int. Work. Wearable Implant. Body Sens. Networks*, pp. 4–7, 2006.

[15] G. Singla and D. J. Cook, “Interleaved Activity Recognition for Smart Home residents,” *5th IEEE Int. Conf. Intelliget Environ.*, pp. 145–152, 2009.

[16] M. R. Alam, M. B. I. Reaz, and M. A. Mohd Ali, “SPEED: An inhabitant activity prediction algorithm for smart homes,” *IEEE Trans. Syst. Man, Cybern. Part ASystems Humans*, vol. 42, no. 4, pp. 985–990, 2012.

[17] L. Liao, D. Fox, and H. Kautz, “Location-Based Activity Recognition using Relational Markov Networks,” *19th Int. Jt. Conf. Artif. Intell.*, pp. 773–778, 2005.

[18] V. R. Jakkula and D. J. Cook, “Using Temporal Relations in Smart Environment Data for Activity Prediction,” *24th Int. Conf. Mach. Learn.*, 2007.

[19] P. Rashidi, D. J. Cook, L. B. Holder, and M. Schmitter-Edgecombe, “Discovering activities to recognize and track in a smart environment,” *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 4, pp. 527–539, 2011.

[20] S. T. M. Bourobou and Y. Yoo, “User activity recognition in smart homes using pattern clustering applied to temporal ANN algorithm,” *Sensors (Switzerland)*, vol. 15, no. 5, pp. 11953–11971, 2015.

[21] a. Riabov, Z. L. Z. Liu, J. L. Wolf, P. S. Yu, and L. Z. L. Zhang, “Clustering algorithms for content-based publication-subscription systems,” *23rd Int. Conf. Distrib. Comput. Syst. 2003. Proceedings.*, pp. 133–142, 2003.

[22] P. Rashidi and D. J. Cook, “Transferring Learned Activities in Smart Environments,” *Ambient Intell. Smart Environ. Intell. Environ.*, vol. 2, pp. 185–192, 2009.

[23] M. S. Olivier, H. M. Tillwick, H. S. Venter, and S. Africa, “Using Self-Organizing Maps for Anomalous Behavior Detection in A Computer Forensic Investigation,” *Inf. Comput. Secur. Archit. Res. Gr.*, 2005.

[24] Z. Zhao, D. Yan, and W. Ng, “Mining probabilistically frequent sequential patterns in large uncertain databases,” *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1171–1184, 2014.

[25] J. Pei *et al.*, “PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth,” *Int. Conf. Data Eng.*, pp. 215–224, 2001.

[26] K. K. S. A.Kotal, “A New Algorithm to Find Longest Common Sub-sequences,” *Int. J. Sci. c Eng. Res.*, vol. 4, no. 5, pp. 664–669, 2013.