
This item was submitted to [Loughborough's Research Repository](#) by the author.
Items in Figshare are protected by copyright, with all rights reserved, unless otherwise indicated.

Towards adaptive control in smart homes: Overall system design and initial evaluation of activity recognition

PLEASE CITE THE PUBLISHED VERSION

<https://doi.org/10.1109/IntelliSys.2017.8324274>

PUBLISHER

© IEEE

VERSION

AM (Accepted Manuscript)

LICENCE

CC BY-NC-ND 4.0

REPOSITORY RECORD

Wang, Zelin, Safak Dogan, Milosh Stolikj, and Ahmet Kondo. 2019. "Towards Adaptive Control in Smart Homes: Overall System Design and Initial Evaluation of Activity Recognition". figshare.
<https://hdl.handle.net/2134/24473>.

Towards Adaptive Control in Smart Homes

Overall System Design and Initial Evaluation of Activity Recognition

Zelin Wang

Institute for Digital Technologies
Loughborough University London, UK
z.wang2@lboro.ac.uk

Milosh Stolikj

Irdeto B.V., The Netherlands
milosh.stolikj@irdeto.com

Safak Dogan

Institute for Digital Technologies
Loughborough University London, UK
s.dogan@lboro.ac.uk

Ahmet M. Kondo

Institute for Digital Technologies
Loughborough University London, UK
a.kondo@lboro.ac.uk

Abstract—This paper proposes an approach for adaptive control over devices within a smart home, by learning user behavior and preferences over time. The proposed solution leverages three components: activity recognition for realising the state of a user, ontologies for finding relevant devices within a smart home, and machine learning for decision making. In this paper, the focus is on the first component. Existing algorithms for activity recognition are systematically evaluated on a real-world dataset. A thorough analysis of the algorithms' accuracy is presented, with focus on the structure of the selected dataset. Finally, further study of the dataset is carried out, aiming at reasoning factors that influence the activity recognition performance.

Keywords—*smart home; machine learning; ambient intelligence; activity recognition; classification*

I. INTRODUCTION

A smart home [1] refers to an indoor living environment, where different kinds of devices, such as indoor sensors, lighting switches, devices related to providing media services, etc., can interconnect together and interface with each other accordingly, making the lives of inhabitants more comfortable. Over the years, with the rapid improvement in software and hardware techniques, machine learning methods are often found applicable to various complex problems commonly found in areas such as smart environments.

With today's techniques and services, users are expecting better living experiences from their smart homes. This is what has been generally referred to as living in a smart home environment (or smart living in brief) [2]. An essential feature of smart living is the autonomous control of devices and services based on user preference. We refer to the service responsible for such control as adaptive system control: the brain of a smart home which understands its users and adapts its services according to the user's needs.

We identify four fundamental elements in a smart home: users, sensors, actuators and controllers. Services offered in a smart home, such as multimedia stream delivered to a set of

devices [3] are considered both as sensors (e.g. type of content being played) and as actuators (play/stop/seek commands). A simplified model describing their relationship and interactions is shown in Fig. 1. Events related to the user and the environment are captured by various ubiquitous sensors. The control system is responsible for making decisions what adaptation needs to be carried out on the delivered services and the related devices. The reaction of the user is then captured by the sensors, which in turn provides information to the system control whether the previously taken action satisfied the user or not. This observation is further used by the control system to initiate the next action. This model highlights that the control system is highly complex and user-dependent. Therefore, instead of relying on rule-based type of control, we aim at implementing adaptive system control which leverages machine learning methods to learn user behaviour and preferences.

In general, there are two approaches when developing machine learning algorithms: supervised learning and unsupervised learning [4]. While supervised learning tries to infer a function based on pre-labelled training data, unsupervised learning is more suitable for brand new learning conditions where little prior knowledge is acquired or given. In this paper, we take activity recognition as a supervised learning problem, where we learn over a sufficiently large real-world dataset, whose activity labels are given for every sample. However, for adaptive control, we cannot use supervised learning since such labelled datasets are not available. Therefore, we need to consider other machine learning mechanisms, such as reinforcement learning.

Current systems for control in smart homes are either focused on specific tasks or a function as generic remote controls. For example, the NEST thermostat [5] requires user actions until it learns the user preferences. From then on, it does not need additional input from the user. A different approach is taken within the Amazon Echo [6], Google Home [7] or Apple Home [8] where devices act as central points of control for all connected devices within a home, but still

The work presented in this paper was carried out as part of CLOUDSCREENS, a Marie Curie Initial Training Networks action funded by the European Commission's 7th Framework Program under the Grant Number 608028.

require commands from the user. Limited research has been conducted on adaptive system control where a proactive way of learning upon user and indoor environment is involved for promoting user-preferred services.

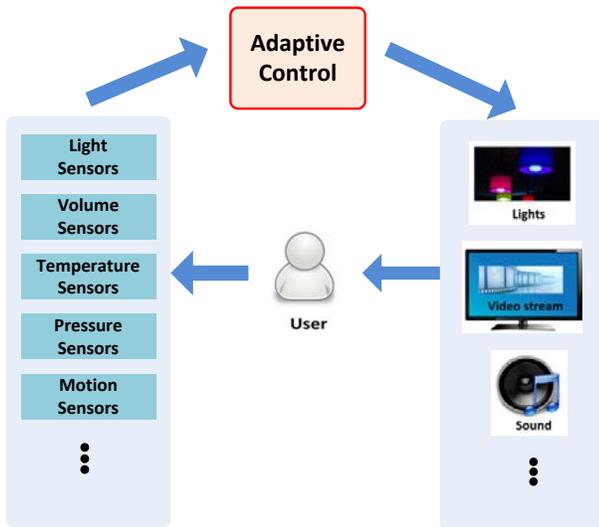


Fig. 1. The Relationship between the User and Various Devices in a Smart Home Environment

The main contribution of this paper is the design of adaptive system control for smart homes. The proposed adaptive system control uses machine learning to deduce user behaviour, and to decide which devices and services to actuate. We propose a hierarchical approach, where first an activity recognition module is used to detect the user activity based on the collected sensory input. Then, an ontology is used to discover the related devices and services to the specific user activity. Finally, the reduced set of devices is analysed by a learning algorithm to decide on the next action.

In this paper, we focus on the first two tasks. We present a survey and experimental comparison of classification algorithms for activity recognition. Then we present a sample ontology for the particular use case of adaptive system control for home multimedia delivery.

The rest of the paper is organized as follows. The related work on ontology-based indoor knowledge management and smart home activity recognition are presented in Section II. An approach of system design with regard to functional modules as well as related ontology structure is introduced in Section III. Section IV concentrates on tests of activity recognition by using a real-world dataset. Comparison of the performances of selected classifiers is measured via accuracy and computation time. Results indicate that computation complexity of a classifier may contribute to the classification accuracy, but at a cost of computation time. Besides, factors such as user's daily activity routine which influences the classification accuracy within the dataset are discussed. Finally, in Section V, we make a summary of the system adaptive control and the evaluation on activity recognition.

II. RELATED WORK

Activity recognition aims at recognizing the actions or activities of a user. In indoor spaces, usually sensors are used to collect a series of observations of the users and the ambient environment. Activity recognition algorithms then use these observations to classify the activity performed by the users. In this section, we will briefly look back on related work done in ontologies within smart environment domain and indoor activity recognition.

The concept of ontologies [9] establishes the hierarchical properties of system components, allowing association among concepts for achieving knowledge management in building intelligent systems. As a method for information management, the recent years have seen the development of ontologies in related areas such as Internet of Things [10], smart environments including multimedia deployment and context-awareness [11], etc.

A. Smart Home and Activity Recognition

Many researchers started to cast their interest in developing smart spaces in limited living environments such as building automation [12]. The Aware Home Research Initiative at Georgia Tech is one of the early research initiatives on smart environments [13]. Besides, there are several research papers showing progress in Managing an Adaptive Versatile Home (MavHome) [14], which is used to provide a customised personal environment to the users of this space. Research studies conducted within the scenario of Activity of Daily Living (ADL) [15][16] have focused on activities such as preparing a meal, washing the dishes, using the toilet, calling on the phone, etc. In these studies, activity recognition is treated as a supervised machine learning task. A summary of the classification methods used for indoor activity recognition in the previously mentioned studies, can be found in [17]. Among these approaches, the J48 Decision Tree algorithm was shown as one of the most commonly used ones, due to its high accuracy rate and low computational complexity [18]. Other commonly used classifiers include Support Vector Machine (SVM) [19], K Nearest Neighbour (KNN) [20], Naive Bayes Classifier (NBC) [21], and Hidden Markov Model (HMM) [22]. SVM aims at regulating boundaries for a class so as to decide whether the testing instance shall fall into it or not. KNN calculates the distances and finds out the closest neighbours of the test instances. NBC makes use of probability model in which it attempts to estimate the probabilities for each class and then picks the class with the maximum probability. HMM, instead of looking at the instances individually, scans the adjacent instances and finds the relations among them.

There are a few available datasets for indoor activity recognition. Centre for Advanced Studies in Adaptive Systems (CASAS) was established in the US in 2007 to focus on creating shared datasets related to indoor everyday living and healthcare activities [23][24]. Sherbrooke, generated by the Domus Lab in Canada in 2010, is a home-based living environment dataset where 7 main everyday living activities were recorded from 6 individuals [25]. The Activity

Recognition with Ambient Sensing (ARAS) dataset was generated in Turkey in 2013. Compared with the other datasets such as CASAS and Sherbrooke, the ARAS dataset has a much richer recorded activity labels as well as instances. Most published work with ARAS dataset have concentrated on multi-user detection [26][27][28]. In Section IV, we use the ARAS dataset to compare several algorithms for activity recognition.

B. Ontologies

The two commonly used ontologies for smart environments are the Smart Appliances REference (SAREF) [11] ontology and the Internet of Things Ontology (IoT-O) [29]. In general, the SAREF ontology provides an ontology design for an indoor smart environment scenario. It outlines the different modules that may be considered for home design, including commands, devices, services, etc. Nevertheless, the ontology is not well documented in some modules, for example, services. Moreover, the interconnection of individuals among various classes is weak, and the detailed illustrations of the object property and data property for most individuals are omitted.

Compared to the SAREF ontology, the IoT-O ontology presents a better construction in detailing the categories such as functionality, devices and family appliances. Besides, a largely enriched set of family appliances varying from lighting, kitchen devices to window shades are all distinctly categorised. Although a better ontology documentation is achieved when compared to the SAREF ontology, the IoT-O ontology lacks the sufficient association between devices and user state.

III. PROPOSED FRAMEWORK FOR ADAPTIVE CONTROL IN SMART HOME

In this section, we propose a framework for establishing adaptive system control.

In a smart home, if we inspect the typical interaction between users and devices, we can observe that the devices involved are tightly related to the particular user activity. For instance, while a user is watching television, it is likely to have control over the multimedia delivery system, the sound system, light control, shade controls etc., but highly unlikely to interact with e.g. kitchen appliances. Therefore, instead of involving all possible devices, it is intuitive to investigate only a relevant subset of the devices connected to the current user activity. This filtering step is introduced in order to reduce the number of devices and their possible states, which has to be processed by the learning agent.

Based on these assumptions, we design the adaptive control framework of three parts: activity recognition, selection of related devices and services via ontologies, and a learning agent for decision making. The interaction between the different components is shown in Fig. 2.

Sensors inside the smart home continuously gather information about the user and the environment. The input collected by these sensors (referred to as raw data) is first sent to a classifier for activity recognition. Depending on the

combined state of all sensors, the classifier predicts the activity performed by the user, encoded as an activity label. The predicted activity label is used to query an ontology, for discovering the related sensors and actuators to the given activity. The result of this query is a list of sensor and actuators within the smart home, which have functionality related to the user activity. For instance, if the activity recognition classifier predicts that a user is consuming a multimedia service (e.g. watching a television broadcast), the ontology will help associate this activity to the TV service, audio system, lighting sensor, lighting switch etc.

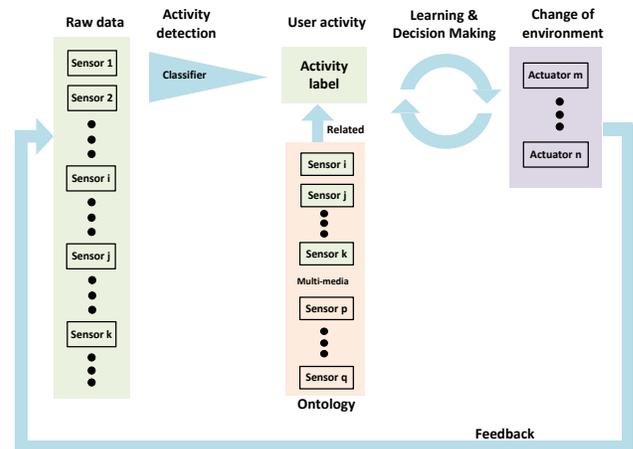


Fig. 2. Functional Modules of the Proposed System for Adaptive Control

Finally, the predicted user activity label, together with the list of related sensors, actuators, and their current states, are provided as input to the learning agent. The learning agent reasons on this input, and decides whether any actuators need to be controlled. For example, the agent can decide to control the audio volume or adjust the lighting level. The feedback whether the taken action is correct will be discovered at a later stage, by observing the user behavior. In this way, the environment is adapted continuously.

The following key objectives are needed to establish the described system:

- To select an activity recognition method suitable for the smart home environment. This includes a survey of existing methods, and their experimental evaluation on smart-home related datasets.
- To develop an ontology framework that can be used to associate devices, multimedia service and user's activity.
- To develop a control framework that can reason on the user's activity and environments, making necessary changes to the multimedia settings accordingly.

In the rest of this paper, we focus on the first two objectives. We propose a unified ontology structure to associate the user's activities with smart home devices and multimedia services, as depicted in Fig. 3. The proposed

ontology is built in Protege [30], a software environment for developing ontologies.

The three main components under *owl:Thing* are *Multimedia_service*, *User*, and *Devices*. The sensors and actuators are added under the *Devices* category. *Multimedia_service* includes the multimedia service deliveries such as movie, TV series, music, etc. In addition, the volume information and other media operations with regard to the time stamp are included into the structure. The *User* category records the user's profile, activity and speech. From the user's speech, the emotions as well as the context can be collected and used for reasoning the necessary media adaption. User activity considered here can be twofold: the main activity is related to watching a movie, whereas additional short-time activities may include making a phone call, making tea, etc.

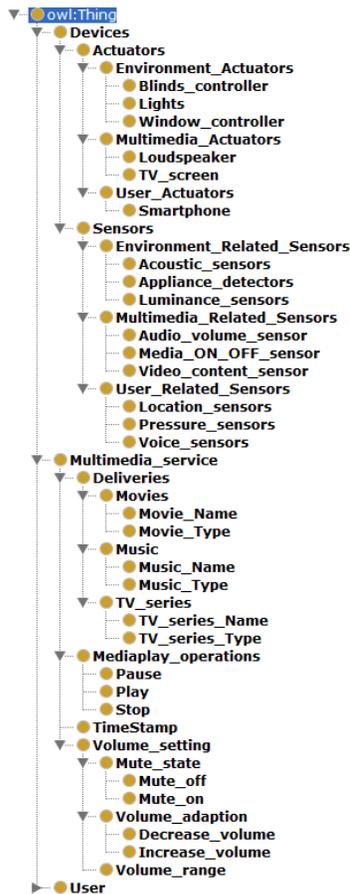


Fig. 3. Ontology for Proposed Adaptive System Control

Queries can be made such as:

- *Acoustic_sensors* has input from *Smartphone* indicating *User* has activity *Talking_on_the_phone*.
- *User* has activity *Talking_on_the_phone* makes *Volume_setting* to *Decrease_volume*.

The purpose of the proposed structure is to illustrate the association among different components needed in a typical

use case based around “watching TV”. Based on the outline of the IoT-O, the structure given in Fig. 3 can be shown as a partial expansion of original IoT-O according to the needs of the problem space to be studied.

IV. ACTIVITY RECOGNITION EVALUATION

In this section we focus on evaluation of existing algorithms for activity recognition on the ARAS dataset. Compared to the other datasets introduced in Section II, the ARAS dataset has richer living spaces, more activity categories and larger amount of training instances, making it worthwhile for our tests. Classifiers chosen here are NBC, J48 and KNN, as introduced in Section II. In addition to these classifiers, we have also added ZeroR [31] to our tests as the baseline classifier. ZeroR is a simple classification method which applies the most frequent activity label from the training set to every instance in the test set. For comparing classifiers' performances, we study the classification accuracy and processing time.

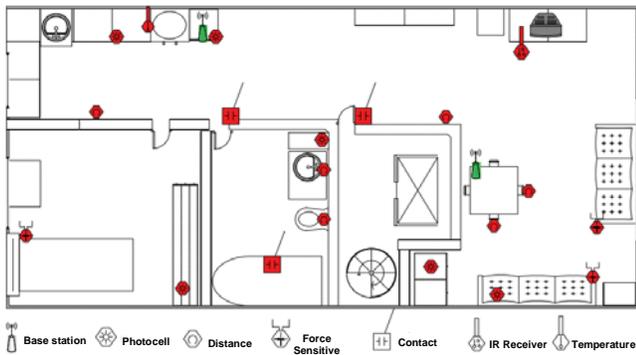
In the rest of the text, we will use the following terminology:

- Dataset: series of observations. Each observation is a set, with the current state of all sensors in the system.
- Activity label: the manually marked ‘ground truth’, i.e. the actual description (encoded) of the activity of the user(s), and the outcome of the classification algorithm.
- Activity space: set of all possible activity labels in the complete dataset.

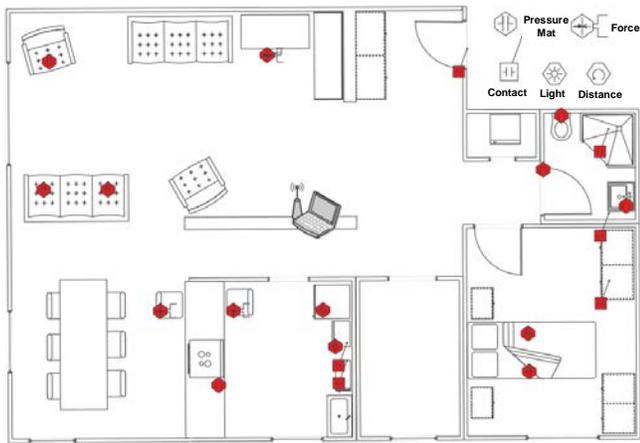
A. Dataset and Test Environment

In our experiments, we have used a real-world dataset, namely ARAS, to test and compare the performances of different classifiers in recognising activity labels. The ARAS dataset comprises sensor data collected from two indoor house environments (House A and House B), as illustrated in Fig. 4 [26]. There are 20 binary sensors located in each house, including pressure sensors, photocells, infrared sensors, temperature sensors, contact sensors and sonar distance sensors. Each house has two inhabitants and a detailed record of 27 activity labels. Both houses share a similar configuration, including a living room, a bedroom, a kitchen and a toilet. However, they differ in size and individual inhabitants. Every day, sensors' states are collected for each second, together with activity labels from the two inhabitants. A total number of 30 days' data is obtained for each house. All algorithms are implemented and compared using WEKA [32], a software tool for data pre-processing.

Compared with the previously reported work related to the ARAS dataset, our aim is focused on the comparison of performances of the different types of classifiers and in-depth dataset analysis such as human factors and ambient influence, rather than multi-user related activity recognition studies via a refined algorithm of a certain classifier.



(a) House A



(b) House B

Fig. 4. Houses' Layout and Location of Sensors in the ARAS dataset [26]

B. Dataset Pre-processing and Test Design

1) Dataset pre-processing:

We use a combined label of two inhabitants. A combined label refers to that two user activities are being considered at the same time irrespective of the individual users.

The original dataset has 27 activity categories for each individual, where distinctions of labels are made among activities such as *preparing breakfast*, *preparing lunch* and *preparing dinner*. However, only based on the sensor information, it is hard to separate those labels reflecting similar behaviours, because they would trigger the same state of the sensors. Besides, there is no typical sensor targeting at some activities such as surfing the Internet. Labels for these activities are only based on the user's report in that period. These conditions may lead to a higher noise level in the dataset (multi-mapping), which would influence the performance accuracy of the classifiers.

To decrease this influence, an approach is to introduce the grouped dataset, in which similar activities are grouped together. The same approach was taken in [26], where 6 frequent activity labels are selected, namely *going out*, *eating*, *sleeping*, *relaxing*, *personal hygiene* and *others*. By combining

the activity labels of both inhabitants, the total activity space contains 21 activities.

2) Comparison metrics:

We collect the classifiers' classification accuracy and processing time in each test. Classification accuracy is defined as the ratio between the number of correctly classified instances and the number of the whole instances in a test set. Processing time is the overall time needed for a classifier to produce a test result, mainly including the time for establishing and training the classification model on the training set and then labelling of the test set. Results presented in the following sub-sections are plotted by taking into account the classification accuracy, including maximum (*max*), minimum (*min*), average, and standard deviation (σ).

3) Test design:

During training and testing, all data are shuffled, i.e., the order of the observations in the training set and testing set is not preserved. Training and testing of each classifier is done on each observation individually, without looking into sequences of observations or time.

a) Training and testing on the same day: First, to grasp general knowledge of the classifiers' performances on the ARAS dataset, both training and testing have been carried out on each of the recorded days. In this test scenario, the observations in each day were split into training and testing data. Then, for each day, we trained all four classifiers on the training data, and evaluated their accuracy on the testing data for that day. The split between the training and testing data was repeated 10 times, i.e., the idea of 10-fold cross validation. Overall, we report the average accuracy of each classifier over all 30 days.

b) Training and testing on separate days: Second, we resorted to separating the ARAS dataset into training and testing data on a daily basis, i.e. using data from certain days as training data, and the rest as testing data. However, in the ARAS dataset, the activity labels indicated within each day are a subset of the whole activity label space. As a result, the training models from *a)* are not based on the full activity space. An activity space means that instances inside the space include all possible activity labels. In fact, the trained classification model should be tested on various new data to evaluate its stability and adaptation capacity. Therefore, we first selected a set of days which covered the full activity space, and used this set for training. Then, we used the data in rest of the days as a testing set.

TABLE I. TRAINING SETS GENERATION

	House A (grouped)	House B (grouped)
Training set 1	Day1, Day3, Day6, Day7	Day1, Day2, Day16, Day17
Training set 2	Day2, Day4, Day5, Day13	Day3, Day5, Day7, Day20
Training set 3	Day8, Day9, Day11, Day12	Day4, Day10, Day14, Day19

We have conducted this part of the tests on the grouped dataset where each individual has 6 activity labels, as mentioned before. As from one single training set is not sufficient to show stability of the test result, we select 3

training sets in each house, all coming from different days, to build the training models. Table I presents the choice of days to generate the training sets in House A and House B.

C. Results and Discussions

1) Training and testing on the same day:

Fig. 5 shows the performance of the classifiers on House A and House B, with comparison of 27-label categories (original dataset) to 6-label categories (grouped dataset). Each bar in the figure represents the average accuracy measured throughout 30 days, together with the positions of **max**, **min**, and **2 σ** -range around average accuracy.

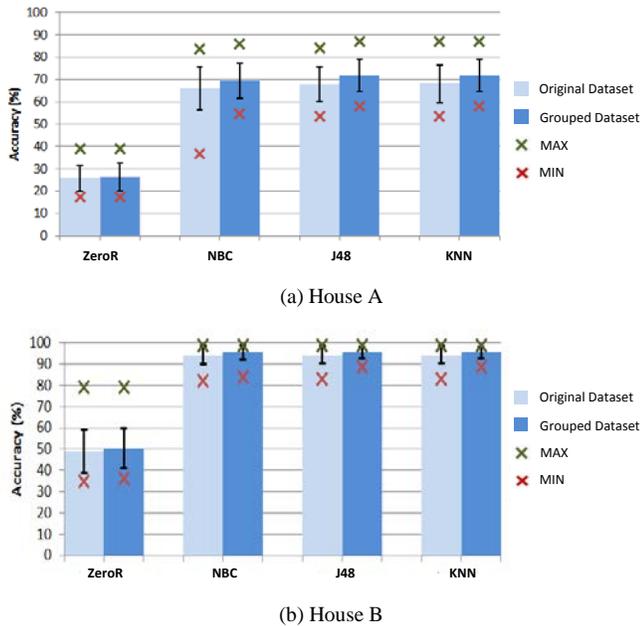


Fig. 5. Classification Accuracy Comparison between Original Datasets and Grouped Dataset based on Training and Testing on the Same Day

It can be noticed from Fig. 5 that overall, the performance of the classifiers applied on the House B data is better than that of House A, based on higher accuracy and lower standard deviation levels. As the baseline classifier, ZeroR presents the worst accuracy level in both cases, while the others are achieving almost double the levels of ZeroR accuracy, which is about 70% and 93% in House A and House B cases, respectively. Also, among the three high-achieving classifiers, KNN ranks the first in terms of accuracy compared to the performances of J48 and NBC.

TABLE II. CLASSIFIERS' COMPUTATION TIME ON ONE-DAY DATA

Classifier	Computation Time
ZeroR	Less than 5 seconds
NBC	About 30 seconds
J48	About 1 minute
KNN	About 2 hours

However, when it comes to the computation time, as illustrated in Table II, KNN spends approximately 2 hours to complete the training and testing on 1 day's data (86,400 instances in total); J48 takes around 1 minute; NBC takes 30 seconds when the same computational power is used. As expected, the simplest method provided by ZeroR is the fastest, completing the computation within 5 seconds, but at the expense of largely reduced accuracy.

Improvement is visible in the results associated to House A by grouping the activity labels into 6 categories, with around 3% increase in average accuracy and around 2% decrease in standard deviation. When it comes to House B, we can note the same trend, but not as obvious as that in House A.

When conducting the tests, we have also noticed that the inhabitants of House A have a much richer activity range compared to that in House B. This observation leads to one hypothesis that the richness of activities of inhabitants may influence the classification accuracy. To explore this further, we have calculated the number of combined activity labels per day in each house. As a result, we have found that on average, this number is about twice the size in House A associated data compared to that in House B data, as shown in Table III.

TABLE III. AVERAGE NUMBER OF DAILY ACTIVITY LABELS

	House A (original)	House B (original)	House A (grouped)	House B (grouped)
Average number of daily activities	43	23	18	9

To visualise the influence on classification accuracy due to the number of daily combined activities, we have plotted a graph using the records in House A, based on the accuracy performance of the J48 classifier, as illustrated in Fig. 6.

Again, the bars in Fig. 6 show the classification accuracy levels obtained for all 30 days. The green line with its respective vertical axis on the right indicates the fluctuation of number of labels, computed for the combined activity labels throughout those days. From several selected points of attention such as Day2, Day9, Day15, Day23, and Day29, we can see that a richer record of activities on a particular day is likely to result in degraded classification accuracy, which is in accordance with the earlier proposed hypothesis.



Fig. 6. Daily Classification Accuracy vs. Number of Activity Labels based on J48 Test from House A

2) Training and testing on separate days:

Using the same parameters as in Fig. 5, Fig. 7 presents the test results of NBC and J48 using the training sets described in Table I. KNN is not illustrated due to its significantly large computation time involved. The bars indicating classification accuracy shown in Fig. 7 are based on the average classification accuracy on the remaining 26 days, which are used as the test sets.

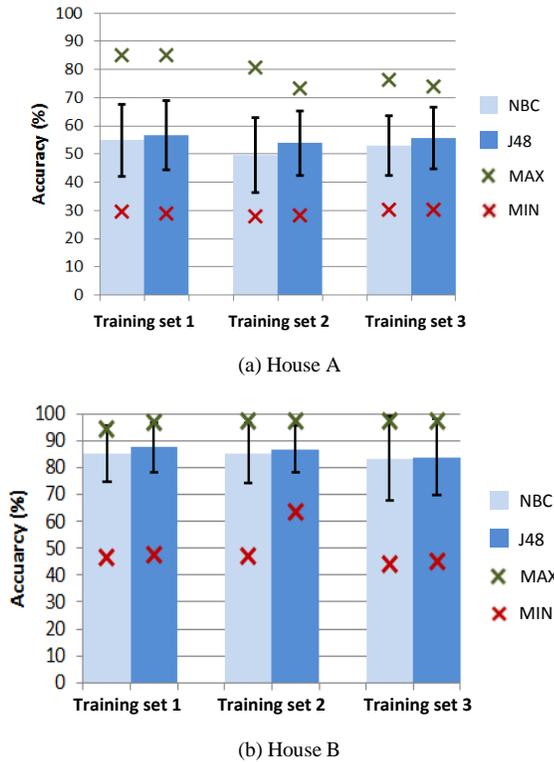


Fig. 7. Classification Accuracy Comparison between NBC and J48 Based on Training and Testing on Separate Days

Compared with Fig. 5, there is an overall drop of classification performance in Fig. 7, indicating a decrease in the average classification accuracy of around 10~15%, together with an increase in standard deviation of about 15%. 2%~5% fluctuations in accuracy exist among different training sets. Reasons are as follows. First of all, the training model utilised here has been established on a training set using 4 days' data. This has led to an increase in the noise level, where the instances related with the same sensor states may have been associated with more activity labels. We refer to this behavior as multi-mapping, and we explore it further in the next section. Second, the testing set, which have come from a totally different day, may have contained instances whose labels indicate a new combination of sensor states, that did not appear in the training set. The results of Fig. 7 show that the difference in classification accuracy measured based on the House A and House B data is still large, with around 30% on average. Apart from the complexity in daily activities as stated before, another probable issue may lie in the multi-mapping problem of the dataset.

3) Multi-mapping discussion

To further study the phenomenon of multi-mapping in the entire datasets associated with the two houses, we have collected the following information for every combination of sensors' states:

- All possible labels found in the dataset.
- The instance(s) for each of those labels.

By examining all sensors' state combinations, a sum of multi-mapping instances can be calculated, which is then divided by the total instances in 30 days, i.e., the multi-mapping percentage. This value has been computed as 99.65% for House A and 72.05% for House B. This result shows that multi-mapping is common in the dataset, indicating that the deployed sensors cannot be always strongly correlated to a particular labelled user activity. However, it is worth noticing that in most cases, there is only one label which occupies the majority instances of a certain combination of sensors' states, which from one aspect, reduces the influence on the classification accuracy.

From our investigations, we have concluded that multi-mapping phenomenon exists and occupies a large portion of the ARAS dataset. The data recorded for House A experiences a more serious multi-mapping problem than that for House B, reflecting a much interleaved relationship between the activity labels and sensors' states. This also contributes to an increase in noise level in the dataset and leads to the confusion of selecting correct labels to test instances for a classifier.

V. SUMMARY

This paper presents the design of a system for adaptive control in smart home environments. The presented approach combines activity recognition, ontologies and machine learning, for establishing autonomous control over devices within a smart home environment. The paper focuses on the first two components of the framework. First, it presents an ontology, which connects user-related concepts with device related concepts in smart homes. Second, a study is conducted for evaluating the performance of different activity recognition algorithms on the publicly available ARAS dataset. The ARAS dataset comprises sensor data collected from two indoor house environments (House A and House B), over a period of 30 days. The three widely used classifiers, NBC, J48 and KNN, are chosen, and are evaluated together with the ZeroR classifier as a baseline. Two types of experiments are performed: training and testing on data collected from the same day, and separated training and testing data on different days.

Based on 10-cross validation of the first experiment, KNN demonstrated the highest classification accuracy, but has been determined to consume significant amount of computation time. J48 and NBC rank the second and third in achieving high accuracy levels, relatively close to that achieved by KNN, while only consuming 8% and 4% of the time required in KNN, respectively. When considering the standard deviation, all three classifiers show a similar range, indicating a unified level of error margin.

To measure the classification models in a more general manner, activity spaces, each containing 4 days' data, have been introduced as a training set, with all other days used as test data. The test results show that on average, NBC and J48 show a lower accuracy and wider dispersion than in the first experiment.

In general, the classification accuracy in House B is much better than that in House A, indicating a 30% difference. The major performance gap observed between the classification accuracies in two houses has driven us to make further analysis on the dataset. It has been noted that due to different daily routines, the inhabitants of House A are likely to produce a much richer activity labels per day. Moreover, when the multi-mapping phenomenon within the dataset is considered, House A associated data has indicated a more non-linear relationship between the sensors' states and activity labels. Furthermore, the physical factors such as sensor type, sensor location and hardware loss are potential culprits to influence the activity notation, causing an increase in the noise level. From these studies, we conclude that the nature of the data, and individual behaviour, has as much influence on accuracy as does the classification algorithm.

REFERENCES

- [1] D. J. Cook and S. K. Das, *Smart Environments: Technology, Protocols and Applications*. 2005.
- [2] Li Jiang, Da-you Liu, and Bo Yang, "Smart home research," *Proc. 2004 Int. Conf. Mach. Learn. Cybern. (IEEE Cat. No.04EX826)*, vol. 2, no. August, pp. 659–663, 2004.
- [3] S. Solaimani, W. Keijzer-Broers, and H. Bouwman, "What we do - and don't - know about the Smart Home: An analysis of the Smart Home literature," *Indoor Built Environ.*, vol. 24, no. 3, pp. 370–383, 2015.
- [4] N. J. Nilsson, "Introduction to Machine Learning," *Mach. Learn.*, vol. 56, no. 2, pp. 387–99, 2005.
- [5] G. Hernandez, O. Arias, D. Buentello, and Y. Jin, "Smart Nest Thermostat: A Smart Spy in Your Home," *Black Hat USA*, pp. 1–8, 2014.
- [6] "Amazon Echo - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Amazon_Echo. [Accessed: 21-Feb-2017].
- [7] Google Home - Made by Google, "No Title." [Online]. Available: <https://madeby.google.com/home/>. [Accessed: 21-Feb-2017].
- [8] A. M. W. Gebhart, "Apple's Home app makes it easy to control your home from your phone." [Online]. Available: <https://www.cnet.com/uk/news/exploring-apples-home-for-homekit/>. [Accessed: 21-Feb-2017].
- [9] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Sci. Am.*, vol. 284, no. 5, pp. 34–43, 2001.
- [10] Z. Song, A. A. Cárdenas, and R. Masuoka, "Semantic middleware for the internet of things," in *2010 Int. of Things*, 2010.
- [11] L. M. Daniele, F. T. H. den Hartog, and J. B. M. Roes, "Study on Semantic Assets for Smart Appliances Interoperability: D-S4: FINAL REPORT," *1.0*, 2015.
- [12] J. Schein, "An information model for building automation systems," *Autom. Constr.*, vol. 16, no. 2, pp. 125–139, 2007.
- [13] I. A. Essa, "Ubiquitous sensing for smart and aware environments," *IEEE Pers. Commun.*, vol. 7, no. 5, pp. 47–49, 2000.
- [14] V. Jakkula and D. J. Cook, "Anomaly detection using temporal data mining in a smart home environment," *Methods Inf. Med.*, vol. 47, no. 1, pp. 70–75, 2008.
- [15] L.-C. F. C-H. LU, "Improving performance of activity recognition via reciprocal model cooperation on an ADL infrastructure," vol. 13, no. 2. p. 4017, 2014.
- [16] I. H. Bae, "An ontology-based approach to ADL recognition in smart homes," *Futur. Gener. Comput. Syst.*, vol. 33, pp. 32–41, 2014.
- [17] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu, "Sensor-based activity recognition," *IEEE Trans. on Syst., Man and Cyb. Part C: Applications and Reviews*, vol. 42, no. 6. pp. 790–808, 2012.
- [18] D. J. Cook, "Learning setting-generalized activity models for smart spaces," *IEEE Intell. Syst.*, vol. 27, no. 1, pp. 32–38, 2012.
- [19] C. Cortes and V. Vapnik, "Support-Vector Networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [20] P. Cunningham and S. J. Delany, "K -Nearest Neighbour Classifiers," *Mult. Classif. Syst.*, pp. 1–17, 2007.
- [21] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, vol. 9, no. 2. 1995.
- [22] S. R. Eddy, "What is a hidden Markov model?," *Nat. Biotechnol.*, vol. 22, no. 10, pp. 1315–1316, 2004.
- [23] G. Acampora, D. J. Cook, P. Rashidi, and A. V. Vasilakos, "A survey on ambient intelligence in healthcare," *Proc. IEEE*, vol. 101, no. 12, pp. 2470–2494, 2013.
- [24] G. Sprint, D. J. Cook, and D. L. Weeks, "Toward Automating Clinical Assessments: A Survey of the Timed Up and Go," *IEEE Rev. Biomed. Eng.*, vol. 8, pp. 64–77, 2015.
- [25] B. Chikhaoui, S. Wang, and H. Pigot, "A New Algorithm Based On Sequential Pattern Mining For Person Identification In Ubiquitous Environments," *KDD Work. Knowl. Discov. from Sens. Data*, pp. 19–28, 2010.
- [26] H. Alemdar, H. Ertan, O. D. Incelt, and C. Ersoy, "ARAS Human Activity Datasets In Multiple Homes with Multiple Residents," *Proc. 7th Int. Conf. Pervasive Comput. Technol. Healthc. Work.*, pp. 232–235, 2013.
- [27] H. Alemdar, "Multi-Resident Human Behaviour Identification in Ambient Assisted Living Environments," *Proc. 16th Int. Conf. Multimodal Interact. - ICMI '14*, pp. 369–373, 2014.
- [28] I. A. Emi and J. A. Stankovic, "SARRIMA: Smart ADL Recognizer and Resident Identifier in Multi-resident Accommodations," *Proc. Conf. Wirel. Heal.*, p. 4:1–4:8, 2015.
- [29] M. Ben Alaya, S. Medjiah, T. Monteil, and K. Drira, "Toward semantic interoperability in oneM2M architecture," in *IEEE Communications Magazine*, 2015, vol. 53, no. 12, pp. 35–41.
- [30] N. F. Noy, M. Sintek, S. Decker, M. Crubézy, R. W. Fergerson, and M. A. Musen, "Creating semantic web contents with protégé-2000," *IEEE Intell. Syst. Their Appl.*, vol. 16, no. 2, pp. 60–71, 2001.
- [31] B. Daniel and M. Boshernitsan, "Predicting effectiveness of automatic testing tools," *Proc. 23rd IEEE/ACM Int. Conference on Autom. Software Eng.*, pp. 363–366, 2008.
- [32] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten, "The WEKA data mining software: An update," *SIGKDD Explor.*, vol. 11, no. 1, pp. 10–18, 2009.