



University of East London Institutional Repository: <http://roar.uel.ac.uk>

This paper is made available online in accordance with publisher policies. Please scroll down to view the document itself. Please refer to the repository record for this item and our policy information available from the repository home page for further information.

To see the final version of this paper please visit the publisher's website. Access to the published version may require a subscription.

Author(s): Naeem, Usman; Bigham, John.

Article title: A Comparison of Two Hidden Markov Approaches to Task Identification in the Home Environment

Year of publication: 2007

Citation: Naeem, U. Bigham, J. (2007) "A Comparison of Two Hidden Markov Approaches to Task Identification in the Home Environment" in Proceedings of the 2nd International Conference on Pervasive Computing and Applications, Birmingham, UK, 2007, pp. 383-388.

Link to published version: <http://dx.doi.org/10.1109/ICPCA.2007.4365473>

DOI: 10.1109/ICPCA.2007.4365473

A Comparison of Two Hidden Markov Approaches to Task Identification in the Home Environment

Usman Naeem, John Bigham

Department of Electronic Engineering, Queen Mary University of London, United Kingdom
usman.naeem@elec.qmul.ac.uk; john.bigham@elec.qmul.ac.uk

Abstract

In today's working world the elderly are often classified as a set of dependent people and are sometimes neglected by society. One of the ways to determine whether an elderly person is safe in their home is to find out what activities an elderly person is carrying out and give appropriate assistance or institute safeguards. This paper describes the lower tier of a two tiered approach that is being adopted. The higher tier consists of hierarchical sets of plans that model common goals and sub-goals associated with activities in daily life. The lower tier deals with recognition of tasks from the stream of sensor events. Tasks are the lowest level component of a plan. The tasks are modelled using a form of hidden Markov modelling.

Keywords: Episode Recovery, Activities of Daily Life, Elderly Care, Smart Homes.

1. Introduction

In Britain, in common with most Western societies, there is an increase in the number of elderly people and the structure and demands of society make it difficult for the children to look after their parents when they need care. Consequently the elderly are often sent to care homes where they are looked after by other people, or cared by their family who give up their work. In either case the cost is considerable, currently estimated as £17billion in the U.K for Alzheimer's disease alone. The introduction of smart homes is slowly becoming a promising solution for the elderly, extending the time that people can still lead an independent life. However, a majority of smart homes are reliant on humans to monitor the old people. If the monitoring is extensive it then tends to be intrusive, particularly if there were to be a reliance on visual monitoring. A possible solution is to introduce more automation and to develop intelligent algorithms to determine the intentions of old people as they carry out everyday Activities of Daily Life (ADL). This can be a complex process as people can often do more than one thing at once, for example,

a person can make tea as well as make toast at the same time.

There has been substantial amount of research on smart homes and identification of ADLs, and most of the research conducted to date has mostly focused on how to find out what low level activity the elderly person is currently carrying out [1]. Some of the research conducted has focussed on using 'dense sensing' [2], rather than relying on vision based systems. This is based on the observations that image analysis for the determination of complex behaviour can be difficult and also that vision systems can be very intrusive. In dense sensing, many individual objects (e.g. kettle, toaster) that are used in the every day activities to be monitored are tagged with wireless sensors or transponders, which transmit information to a computer via an RFID reader when the object is being used or touched. With this type of approach activities are generally represented by sequential models which consist of individual objects orders they would be used while the activity is being conducted by a user. This approach tends to work well for simple activities that follow a standard path of execution. However, the approach is not as successful in detecting activities that have more than one way of being carried out, or when the sensor readings unknown to the model for a particular activity arise. Approaches that discover models using data mining techniques [3] and use ontologies [4] have been used to build reliable activity models. These models are able to match an unknown sensor reading with a reference sensor reading in the model using the ontology, e.g. object Cup could be substituted for a Mug object as the model uses Cup..

The work in this paper is performing much the same function however in the context of integrating it into a framework for analysing and reasoning about the intentions of the elderly person by associating ADLs with goals and sub-goals of the person. Identification of goals gives help in determining the next ADL they are going to carry out and so is a step in providing pre-emptive assistance and feedback to the identification of ADLs themselves. The longer term goal of our work is to understand how knowledge at different levels of abstraction can be used together to determine what ADLs are currently active.

The lowest level activity, i.e. one that has no sub-goals, is called a *task*. The subject of this paper is restricted to the reliable identification of tasks, e.g. reducing the number of false positives while keeping the false negatives within bounds. This is achieved by low level modelling of the task using Multiple Behavioural Hidden Markov Models (MBHMMs), which is the lower tier of the two tiered approach.

2. Levels of Modelling

ADLs can correspond to simple tasks, such as “switch on kettle”, or more a complex activities such as “make breakfast”. ADLs can be nested within other ADLs. ADLs may occur in parallel with other ADLs. In the current work ADLs are modelled as plans. Plans can contain sub-plans. A plan that cannot be decomposed any further is called a task. ADL recognition is based on recognising plans from constituent tasks. When performed, a task generates sensor events, and so task recognition is based on analysing sensor data. In the literature the term “episode recovery” is sometimes used to segment a sensor stream into episodes, where each episode corresponds to a task. In the terminology used in the paper, episode recovery and task identification are synonyms.

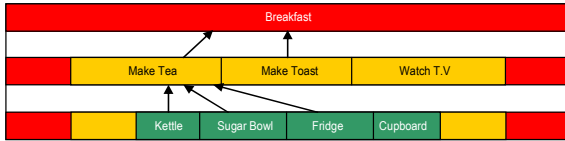


Figure 1. The structure of a Hierarchical Activity of Daily Life (HADL).

In Figure 1, the ADL “Make Breakfast” consists of a simple sequence of tasks, Make Tea, Make Toast and Watch TV, but on general these may be sequential in any order, or indeed in parallel, or some partial order. The sequences of the sensor events shown at the lowest level (Kettle Sensor, Sugar Bowl Sensor, Fridge Sensor) correspond to sensors triggered during the task “Make Tea”.

3. Hidden Markov Models

A Hidden Markov Model (HMM) is the approach used for the recognition of tasks. A HMM is a statistical model of transitions between states, where the observables (events) are related to the state that is active. The complexity of the HMM modelling relates to the abstraction level chosen for a task. Some tasks only require very simple models relating the task to a small set of events that are almost always observed, like making toast, while others may have a set of events that are only occasionally observed. Additionally, if the task

is complex the order of events can be different depending on the actual execution of the task making the HMM larger.

Two different types of HMM are compared in two episode recovery experiments. The aim was to establish which form of modelling is better suited to carry out the identification of tasks. It is vital that the recognition of the tasks is accurate, as this will have an affect on the recognition that will be carried out in the higher levels.

The first HMM model used for the episode recovery experiment was based on the automated approach developed by Wilson [5]. Wilson also describes a very interesting tool to support semi-manual episode recovery. The automated approach used a simple HMM and the Viterbi algorithm. The Viterbi algorithm computes the most likely sequence of the states. In Wilson’s approach a state corresponds to a task. A person may stay in a task over several emissions, where each emission generates an observable sensor event, before moving on to the next task (See Figure 6). The identification of the most likely sequence of tasks allows segmentation of the sensor events associated with each task. In order to measure the accuracy of the segmentation of sensor readings Wilson et al used what they term the P_k metric. This is formulated as the probability that two sensor readings at a distance of k from each other are incorrectly segmented. This statistical approach for segmenting the sensor readings was inspired from Beeferman et al [6] who used a similar approach to automatically partition text into coherent segments.

A different approach to episode recovery is also investigated using MBHMMs is described in the next section.

4. Multiple Behavioural Hidden Markov Models (MBHMM)

The hidden states on this model are the steps that are taken to complete a single task. For example, a simple model of making tea would consist of the steps: switch the kettle on, followed by putting sugar in cup then adding milk to the cup of tea. Each step may have an associated sensor reading, which may or may not be observed. A sensor reading may be repeated. A HMM is constructed for each possible ordering of the steps of the task. The hidden states are the steps. The models are called variants. So each task has a set of variant HMMs that are associated with the task. When each sensor reading is observed, its match to existing active models is established, or to new tasks starting with a step with this new observable. A time window of models are kept and only models within the time window are considered.

This is combinatorially expensive, but tasks are simple ADLs with a limited number steps and so have a limited number of variants and span a small set of

sensor events. This means that the window size and the number of variants are both low and so the method is tractable. Multiple models are used for each task and the one that fits the sensor readings best is chosen as identifying the task. Figure 2 below shows two variants of the task Make Tea, consisting of three sequential steps.

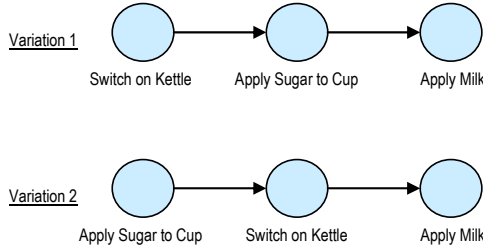


Figure 2. Two types of variations for Making Tea.

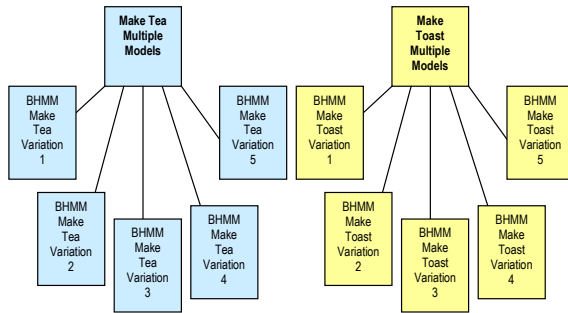


Figure 3. Any variation could be the true model, such as Make Tea Variation 3 or even Make Toast Variation 2. Whichever model has the highest probability given the observations is chosen.

One of the advantages of this approach is that even if the elderly person has not finished completing the task it is still possible for the MBHMM to determine which task is currently active. This is because the probability of being in the final state of the model is computed as each sensor reading is read. Sensor events are mapped onto a trellis (See Figure 4) where each column (T1, T2, T3...Tn) corresponds to an incoming observation and gives the probability of being in each state (step) given the incoming observation.

	T1	T2	Tn
Switch Kettle On	0.9	0	...
Apply Sugar	0.05	0.045	...
Apply Milk	0.05	0	...
	Kettle Sensor	Sugar Bowl Sensor	n

OBSERVATIONS

Figure 4. This is an example of a variation of the model of Make Tea which shows the probability of each state in the trellis after two observations.

The state transition probabilities for all the multiple models of each task are different, as the sequence of steps modelled are different. However the emission/confusion probabilities remain the same for all

the variations for each task. For example, all the different variations of the multiple models for Make Tea have the same emission probabilities.

5. Computing the probability of being at a particular state given the events

The probability calculations are illustrated by using one of the variations of Make Tea shown in Figure 5. From the model it can be seen that there can be repeated sensor events at a step. For example, though not part of this task, when filling up a kettle the kettle may go near the tap more than once. Sometimes other activities associated with sensors events unrelated to the task can intervene and these are modelled by the unexpected states.

For this particular model there are four types of sensor event, each Boolean, one for each expected step. These are “Kettle Sensor triggered” = k , “Sugar Bowl Sensor triggered” = s , “Fridge Sensor triggered” = f . A sensor event may be missing as the sensor may not be triggered. Any other sensor event that is not associated with this model will be referred to as x . The states in this model are as follows: K=Switch the Kettle On, S=Apply Sugar, M=Apply Milk, US₁=Unexpected State 1, US₂=Unexpected State 2.

Let $a_{ij}=P[\text{new state}=j | \text{old state}=i]$, these probabilities are determined by the values in the transition matrix. The values in the transition matrix have been assigned based on the sequence that should match each model. The transition probabilities for this variation of Make Tea are shown in figure 5.

Let $b_{ij}=P[\text{observe sensor reading } j | \text{in state } i]$. These probabilities are determined by the values in the confusion matrix. In order to conveniently represent the states in the probability update formula the states are labeled by numbers, viz.

K = 1, S = 2, M = 3, US₁ = 4, US₂ = 5

The sensors are also labelled.

$k = 1, s = 2, f = 3, x = 4$

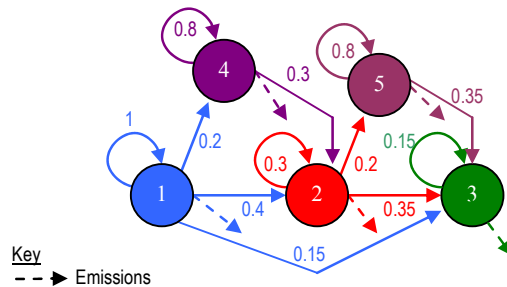


Figure 5. State Transition Diagram for Make Tea Variation Model 1.

Let O_i represent the first observation. O_i can be 1, 2, 3, or 4. After the first observation O_i , we want to update the probabilities of being in each state, i.e. compute

$P[S_i = i | O_1 = j]$ for all $i = 1, \dots, 5$, for the given j . $P[S_i = i | O_1 = j]$ means the probability of being in state i after one observation given that the first observation was j .

The probability of being in state i after the first observation, given that the observation was j can be calculated as:

$$P[S_1 = i | O_1 = j] = \frac{\alpha_i^1(j)}{\sum_{k=1}^5 \alpha_k^1(j)} \quad (1)$$

In the top line of equation 1, the number 1 indicates the number of observations that have been observed, while i represents the state of interest (e.g. Apply Sugar or Apply Milk) and j represents the incoming sensor event (e.g. Sugar Bowl Sensor triggered).

After the second observation second observation O_2 the conditional probability of being in a particular state given both observations can be computed recursively using:

$$P[S_2 = i | O_1 = j \wedge O_2 = k] = \frac{\alpha_i^2(j, k)}{\sum_{p=1}^5 \alpha_p^2(j, k)} \quad (2)$$

$$\text{where } \alpha_i^2(j, k) = b_{ik} \left\{ \sum_{p=1}^5 a_{pi} \alpha_p^1(j) \right\}$$

Similar formulae apply for the probability of being in each state after 3, 4, ... events. The MBHMM approach is derived from an approach that was developed by Veloso [7]. The aim of the approach developed by Veloso was automated recognition of robot behaviour.

6. Episode Recovery Experiments

The objective of the episode recovery experiments is to determine which tasks are active from a stream of sensor data and to compare the performance of the MBHMM with the HMM described by Wilson et al.

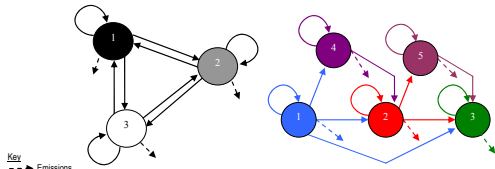


Figure 6. State Transition Diagrams (1) a HMM of the form used by Wilson, here shown for three tasks (2) a MBHMM for a variant of one task.

The left hand model in figure 6, the states correspond to the possible tasks, and after each transition (and so observation) the system can stay in the same state (task) or move to any other task. There is only one model.

The right hand in figure 6 shows one of the many models used in the MBHMM approach. Here the states correspond to steps in a task. The step with no outputs (3) to a different stage is the final state and indicates that this task has been completed. As can be seen the model allows for missing observations (steps can be bypassed) and extraneous observations (states can be entered that do not correspond to a step in the task and can emit sensor events irrelevant to the task). However, this is at the cost of having a model not only for each task, but each variant of a task.

The experiments to compare the models took place in a kitchen, with non-intrusive RFID transponders being installed around the kitchen and on its cupboards and utensils, such as the kettle, dishwasher, and toaster. The data generated from the transponders was collected by a RFID reader that is the size of match box and is worn on the finger of the subject conducting the experiment. The experiment design was the same for both sets of the experiments, with the kind of HMM being the only difference. For these experiments 10 adult volunteers had been recruited from the student community to carry out the tasks. The tasks ranged from making tea to putting dishes into the dishwasher. 10 subjects were chosen as people have different ways of carrying out a particular task as so there will be variability in the sensor stream.

The experiment was split into two parts. In the first part the subjects carried out each task in a prescribed order, while in the second part the subjects were asked to carry out each task in any order that they wished, and to record this order. The data was then analysed separately with the two types of HMM.

7. Results

The accuracy for the episode recovery experiment results was determined by a percentage of false positives and false negatives while identifying a task. Each task was performed once by each person.

Table 1. Viterbi-based Episode Recovery Experiment Results.

Tasks	Predefined False + [%]	Predefined False - [%]	Any State Order False + [%]	Any State Order False - [%]
Make Tea	100	0	65	35
Make Toast	100	0	50	50
Drink Water	70	30	50	50
Make Coffee	100	0	60	40
Warm up Meal	60	40	40	60
Defrost Food	100	0	60	40
Wash/Dry Dishes	10	90	10	90
Have a Cold Glass of Water	30	70	20	80

Wash Dishes Dishwasher	100	0	60	40
Have a snack (Biscuit)	100	0	50	50

The Viterbi-based episode recovery experiment results (Table 1) show that a majority of tasks like “Make Tea”, “Make Toast” and “Make Coffee” were correctly determined when the subjects carried out the tasks in the prescribed order. In contrast, when the subjects carried the task in the order chosen by the student then this had led to significant downfall in terms of accurately determining the task.

In the predefined order of tasks experiment, the accuracy rate for “Wash/Dry dishes” and “Have Cold Glass Water” was very low in comparison to the other tasks. This was because these tasks did not have a sensor reading which was exclusive to that task. For example, the dishwasher has a sensor that is only triggered when undertaking a dishwashing task and not for any other task. So triggering the dishwasher sensor leads to a high probability that a dishwashing task is taking place. However, the task of having a cold glass of water needed a water dispenser/cooler sensor to be triggered. This sensor event is mutually exclusive from sensors associated with all other tasks but still the accuracy rate was only 10%. This is because the other states (steps) of this task were very similar to the states of having a glass of water from the tap. Therefore, whenever the glass of water task was identified then this lead to the having cold glass of water task going unnoticed.

Table 2. MBHMM Episode Recovery Experiment Results

Tasks	Predefine False + [%]	Predefine False - [%]	Any State Order False + [%]	Any State Order False - [%]
Make Tea	100	0	95	5
Make Toast	100	0	95	5
Drink Water	80	20	80	20
Make Coffee	100	0	95	5
Warm up Meal	90	10	85	15
Defrost Food	100	0	90	10
Wash/Dry Dishes	90	10	85	15
Have a Cold Glass of Water	80	20	80	20
Wash Dishes Dishwasher	100	0	95	5
Have a snack (Biscuit)	100	0	80	20

The results from the MBHMM episode recovery experiment (Table 2) and Figure 6 show an improved

level of accuracy for task recognition that is conducted in any state order and the results are more encouraging than the results gathered with the Viterbi based approach.

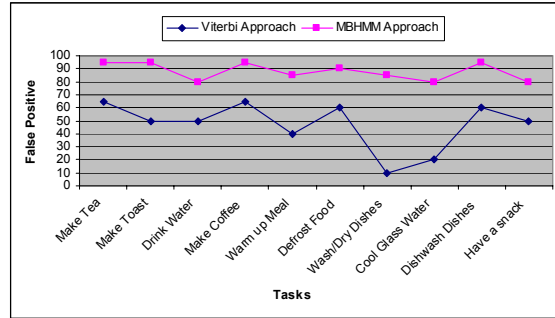


Figure 7. Viterbi approach vs. MBHMM approach, Comparison of experimental results for any state order.

The reason why this approach had higher accuracy rates than the Viterbi-based approach is because the MBHMMs model different feasible orderings of sensor readings while simpler model of Wilson does not impose any order. Additionally, Wilson’s approach does implicitly assume that each task generates, on average the same number of sensor readings however performed, and indeed the probability of returning to the same state is chosen so that the expected number of visits to a state corresponds to this average. This is a rather strong assumption.

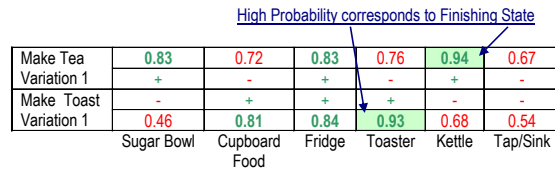


Figure 8. This indicates why the MBHMM approach is able to accommodate concurrent tasks. As well as being able to map all the variations, this approach can determine when a finishing state is about to be reached or has been reached, as the probability of each task increases when a related sensor event occurs and decreases when an unknown event for a specific model occurs.

The MBHMM highly improved the accuracy of determining the task that is currently active. A reason why the MBHMM outperforms the Viterbi approach is because the objective of the Viterbi algorithm is to determine the most likely sequence, whereas the MBHMM determines the probability of being in a finish state of a model given the window of observations being considered. As well as that this approach was able to solve the problem of missing sensor readings to a certain extent. This is because the models that were constructed for each activity modelled the possibility of an unexpected sensor event occurring between expected sensor events. The idea of the

unexpected state being modelled is similar to the concept of profile hidden Markov models, where any unexpected sequence data which occurs in a DNA motif is substituted with an insertion [8]. However this approach only works to a certain extent where the task models are small and manageable (e.g. Make Tea). If a task model is large and has a few missing sensor readings then the ontology approach seems more efficient. Yet as seen in the results the unexpected state approach does allow for different variations of one task to be detected better than existing approaches.

This paper is reporting on early work. Significant improvements to the process of episode recovery are still required and these results are seen as only indicative. Different forms of modelling and enhancements are being developed and evaluated. Work on plan recognition based on task identification is also underway, and feedback from the planning models to support task identification is the next step.

8. Conclusion and Future Work

In this paper we described current work at the lower level of modelling in a two tier modelling scheme that models ADLs with plans and tasks in terms of the events they can generate. The results from the episode recovery experiments indicated that the MBHMM was more effective than the Wilson et al's automated approach. However these are only preliminary experiments and further work on segmenting the sensor events is needed to see if the MBHMM is the appropriate approach for the recognition of tasks in the lower level modelling. Other approaches to episode recovery are being investigated such as the use of N-grams, which is a technique that arises from work on anomaly detection. This is a simple technique that counts the frequencies of different sequences in a sliding window over a set of events. It is hoped to augment the N-gram technique with information regarding the time intervals between sensor events. More sophisticated HMMs will be developed and in particular the tool that allows different HMMs (the different models and each of their variants of the MBHMM type described) to listen for new events over a sliding window of time will be developed further in order to find more flexible approaches to episode recovery, without premature commitment to a particular partition.

References

- [1] Ogawa, M. Ochiai, S. Shoji, K. Nishihara, M. Togawa, T. (2000). "An attempt of monitoring daily activities at home". *Proceedings of the 22nd Annual EMBS International Conference of the IEEE*, July 23-28, 2000, Chicago IL. Volume 1, Page(s):786 – 788
- [2] Philipose, M. Fishkin, K.P. Perkowitz, M. Patterson, D.J. Kautz, H. Hahnel, D. (2004). "Inferring activities from

- interactions with objects". *IEEE Pervasive Computing Magazine*, Volume 3, Issue 4, Oct-Dec 2004 Page(s):50 - 57
- [3] Wyatt, D. Philipose, M. Choudhury, T. (2005). "Unsupervised Activity Recognition using Automatically Mined Common Sense". *Proceedings of the Twentieth National Conference on Artificial Intelligence*, July 9 – 13, 2005, Pittsburg, PA. Page(s):21 - 27
- [4] Munguia-Tapia, E. Choudhury, T. Philipose, M. (2006). "Building Reliable Activity Models using Hierarchical Shrinkage and Mined Ontology". *Proceedings of the Fourth International Conference on Pervasive Computing*, May 7 – 10, 2006, Dublin, Ireland. Page(s):17 - 32
- [5] Wilson, D.H. Wyaat, D. Philipose, M. (2005). "Using Context History for Data Collection in the Home". *Proceeding of PERVASIVE 2005: ECHISE Workshop*, May 2005, Munich, Germany
- [6] Beeferman, C. Berger, A. Lafferty, J. (1999). "Statistical Models for Text Segmentation". *Special Issue on Natural Language Learning*, C. Cardie and R. Mooney, eds. 34(1-3): 177-210
- [7] Han, K. Veloso, M. (1999). "Automated robot behavior recognition applied to robotic soccer". *Proceeding of the Ninth International Symposium*, 2000, Springer-Verlag, London. Also in the *Proceedings of IJCAI-99 Workshop on Team Behaviors and Plan Recognition*.
- [8] Eddy, S.R. (1998). "Profile hidden Markov models (Review)". *Bioinformatics*, Volume 14, Page(s):755-763 Department of Genetics, Washington University School of Medicine, USA