

# Bayesian Activity Recognition in Residence for Elders

Tim van Kasteren and Ben Kröse

Intelligent Systems Lab,  
University of Amsterdam ,  
Kruislaan 403, 1098 SJ, Amsterdam , The Netherlands

**Keywords:** Activity Recognition, Temporal Sensor Patterns, Dynamic Bayesian Networks, Context Awareness

## Abstract

The growing population of elders in our society calls for a new approach in caregiving. By inferring what activities elderly are performing in their houses it is possible to determine their physical and cognitive capabilities. In this paper we describe probabilistic models for performing activity recognition from sensor patterns. We introduce a new observation model which takes the history of sensor readings into account. Results show that the new observation model improves accuracy, but a description using less parameters is likely to give even better results.

## 1 Introduction

An important task in intelligent environment applications is known as activity recognition [6, 5, 4]. The goal of this task is to recognize the activities an inhabitant, or possibly several inhabitants, of a house are performing. The choice of activities to monitor can differ from application to application. The activities we monitor are known as *activities of daily living* (ADLs) and are widely used within the health-care domain to determine a person’s mental and physical abilities. ADLs are routine activities that people tend to do everyday, such as eating, bathing and toileting [2].

The input for the activity recognition task typically comes from a large number of simple sensors. Such sensors come in different forms such as contact switches, pressure mats or motion detectors. However, they all share the properties of being wireless, low-power consuming, binary sensors that are installed throughout the house.

The activity recognition data can be described as being noisy (incidental sensor firings), non-deterministic (activities can be performed in numerous ways) and ambiguous (similar sensor patterns can be generated by different activities). Typically probabilistic models are used for this kind of data. Previous work [5, 6] shows the use of probabilistic models works, but leaves a lot of room for improvement.

In this paper we run experiments for various parameter values to determine the effect on the performance of our models. Furthermore, we introduce a new observation model for performing activity recognition.

The rest of this paper is organized as follows. In section 2 we introduce a notation for describing data. In section 3 we describe the models used for activity recognition. Then, in

section 4, we describe the results of experiments done on a dataset provided by MIT. Finally, in section 5 we end with a conclusion.

## 2 Activities of daily life

Our goal is to recognize activities of daily living (ADLs) from sensor readings in a house. In order to discretize the time space we use intervals of a constant length  $\Delta t$ ; we will state the chosen value for  $\Delta t$  in the experiments section. We denote a sensor reading as  $y_t^i$ , indicating whether sensor  $i$  fired at least once between time  $t$  and time  $t + \Delta t$ , with  $y_t^i \in \{0, 1\}$ . In a house with  $N$  sensors installed, we define a binary vector  $\vec{y}_t = (y_t^1, y_t^2, \dots, y_t^N)^T$ . Activity instances have variable duration, but are divided in pieces of equal size  $\Delta t$ , the activity between  $t$  and time  $t + \Delta t$  is denoted as  $a_t$  (fig. 1).

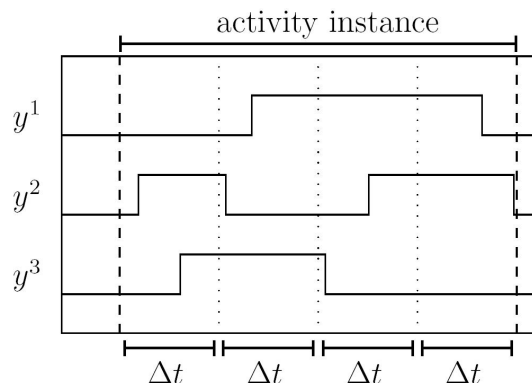


Figure 1: Showing the relation between sensor readings  $y^i$ , time intervals  $\Delta t$  and activity instances.

## 3 Probabilistic models

We carry out activity recognition in a Bayesian framework:  $p(\text{activity}|\text{sensordata}) \propto p(\text{sensordata}|\text{activity})p(\text{activity})$ . First we describe a static Bayesian model in which temporal aspects are not considered. Second we present a Dynamic Bayesian Network, in which we are able to model the temporal aspects of the activities. Third we present a model where the dynamics of the sensordata are used by taking into account a history of sensordata.

Model parameters come in the form of conditional probability tables and are learned using Maximum-Likelihood

training from supervised data [1].

We speak of an observation model when discussing the relation between the sensor data (observations) and the activity (hidden state). And of a transition model when discussing the relation between activities over time [3].

### 3.1 Static Bayesian model

In this section we assume that activity  $a_t$  is independent of previous activities  $a_{1:t-1}$  and that the sensor data  $\vec{y}_t$  is only dependent of  $a_t$ . Furthermore we assume a 'naive' Bayesian observation model:  $p(\vec{y}_t|a_t) = \prod_i p(y_t^i|a_t)$  as depicted in figure 2.

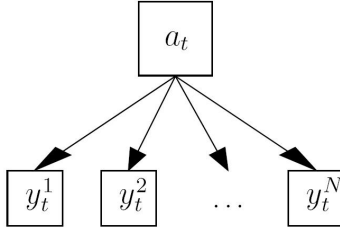


Figure 2: Static Bayesian model for activity recognition,  $a_t$  denotes the activity,  $y_t^i$  denotes data from sensor  $i$ .

### 3.2 Dynamic Bayesian Network

In this section we continue to use the 'naive' Bayesian observation model from the previous section, but additionally we use a transition model  $p(a_t | \vec{y}_{1:t-1}) = \sum_{a_{t-1}} p(a_{t-1} | \vec{y}_{1:t-1})p(a_t | a_{t-1})$  for modeling the temporal aspects of the activities as denoted in figure 3.

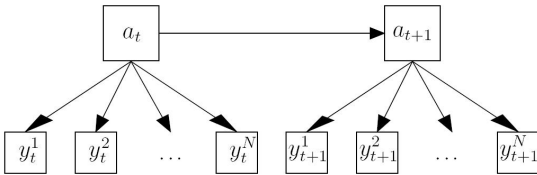


Figure 3: Dynamic Bayesian Network (DBN) for activity recognition,  $a_t$  denotes the activity,  $y_t^i$  denotes data from sensor  $i$ .

### 3.3 $k$ -Observation history matrix

The final model we introduce replaces the 'naive' Bayesian observation model with one where the dynamics of the sensor data are used by taking into account a history of sensor data. We define  $Y_{t,k}$  to be the ' $k$ -observation history matrix'. Here  $k$  is the number of timesteps we look into the past.

$$Y_{t,k} = \begin{pmatrix} y_{t-k}^1 & \cdots & y_{t-2}^1 & y_{t-1}^1 & y_t^1 \\ y_{t-k}^2 & \cdots & y_{t-2}^2 & y_{t-1}^2 & y_t^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ y_{t-k}^N & \cdots & y_{t-2}^N & y_{t-1}^N & y_t^N \end{pmatrix}$$

## 4 Experiments

To compare the performance of the models we evaluated them on a dataset made available by MIT. The dataset consists of sensor readings annotated with activities [5].

Since our models all depend on the number of sensors and the time interval  $\Delta t$  we tested how these parameters affect the performance of the models.

### 4.1 Dataset

The dataset consists of sensor readings recorded in the house of a 30-year-old woman who spent free time at home. She lives alone in a one-bedroom apartment where 77 state-change sensors were installed. The sensors were left unattended, collecting data for 14 days in the apartment. This resulted in a total of 2989 sensor firings. A total number of 13 activities were annotated through the use of a PDA.

### 4.2 Experimental setup

In our experiments we classify all 13 activities. Each activity instance is divided in data segments of length  $\Delta t$ . To separate data segments into a test and training set we used an approach known as 'leave one day out'. In this approach one full day of sensor readings is used for testing and the remaining 13 days are used for training. This is done for all of the 14 days and the average of the accuracy is taken as the end result.

Accuracy is calculated by classifying all the segments in the testset and calculate the percentage that was correctly classified.

### 4.3 Performance as a function of number of sensors

Investigating how the number of sensors affects the accuracy of the classifier is interesting simply because sensors cost money. We expected that using more sensors would result in a higher accuracy, as more sensors allow us to observe more.

In this experiment we sorted the sensors by the number of times they fired in the entire dataset. We start by using the sensors that fired most and add those that fired less. The plot in figure 4 shows the effect of the number of sensors on the accuracy for both classifiers.

This plot shows two things. First, it shows that the dynamic model outperforms the static. This is as expected because the dynamic model incorporates the temporal aspects of the activities which tells it how likely a certain activity is to follow another. Second, we see that adding more sensors does not necessarily increase the accuracy. Adding sensors that fire only incidentally will have badly estimated parameters and do not allow activities to be discriminated. Therefore, it seems wisest to use a few sensors in cleverly positioned locations.

It should be noted that in experiments by Wilson (on a different dataset) the accuracy of 'being idle' increased as the number of sensors increased [6]. However, we did not include an activity for being idle.

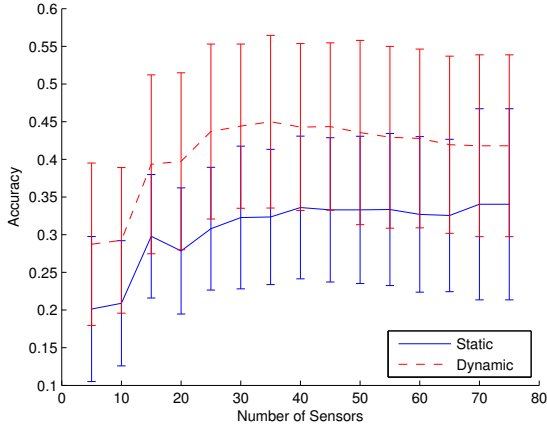


Figure 4: Plot showing the average accuracy of the static and dynamic model as a function of the number of sensors

#### 4.4 Performance as a function of the time interval $\Delta t$

The time interval  $\Delta t$  determines how our data is cut up into equal data segments. Using a very small value means sensor firings can be very precisely distinguished, but might not capture a broad enough picture. On the other hand, using a large value would provide a good summary of sensor firings within the interval, however, too large a value would cause higher ambiguity among sensor patterns. So it is difficult to choose a value that makes sense.

The error plot shown in figure 5 shows the accuracy of the static and dynamic model for various values of the time interval  $\Delta t$ .

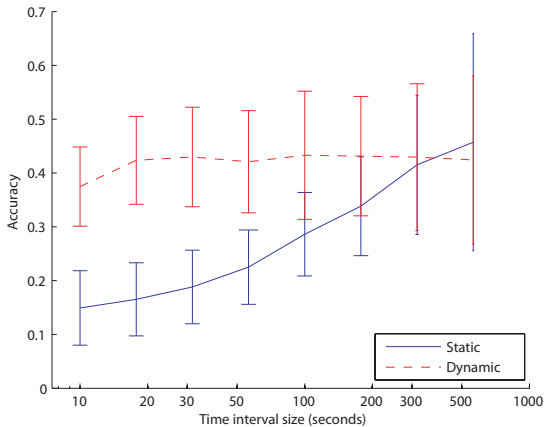


Figure 5: Plot showing the average accuracy of the static and dynamic model against time interval  $\Delta t$

We see the accuracy for the static model increases significantly as the time interval is set to higher values. However, we also see the variance of the accuracy increasing, making the results less reliable. Strangely enough, the accuracy of the dynamic model stays more or less the same. The only difference between the static model and the dynamic model is that the dynamic model incorporates transitions between activities. Therefore, we can conclude that a higher time in-

terval increases the accuracy of the observation model but decreases the accuracy of the transitional model.

To make the picture complete we include two more results. In work by Tapia the average duration of an activity was used as a time interval, this means that different time intervals are used for different activities [5]. We tested the models using this same approach, which resulted in an accuracy of 35.7% and 39.2% for the static and dynamic model respectively. Finally, we tested our approach using a time interval which exactly fits the actual duration of the activity. This is a form of cheating as this information is normally not known during inference. However, it does give an indication of how the model can ideally perform. The accuracies for this approach are 67.5% and 67.3% for the static and dynamic model respectively.

Looking at these results it remains difficult to determine what the ideal value for the time interval is. A high value gives a high mean accuracy, but with a high variance as well. Also the transitional information seems to disappear as the time interval is set higher.

#### 4.5 Performance as a function of $k$ -Observation history matrix

The idea to use an observation history makes sense as it allows the model to incorporate more information about the past. Sensors that fired in the past can provide a context for the classification of the activity. The plot in figure 6 shows the accuracy of both models using the  $k$ -Observation history matrix for various values of  $k$ , where  $k$  is the number of timesteps we look into the past.

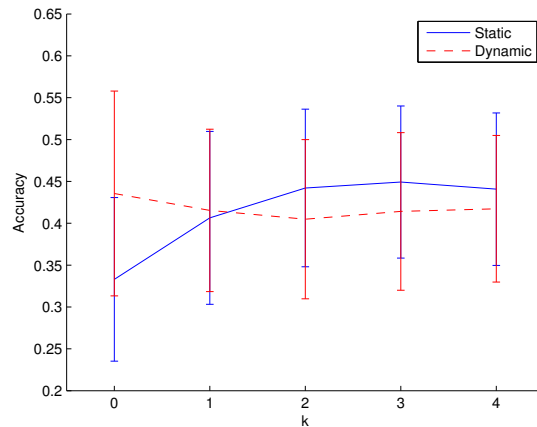


Figure 6: Plot showing the average accuracy of the static and dynamic model using the  $k$ -Observation history matrix for various values of  $k$ , where  $k$  is the number of timesteps we look into the past.

The plot clearly shows an increase in accuracy for the static model. However, the accuracy of the dynamic model remains more or less the same. It is strange that the dynamic model does not enjoy the same increase in accuracy as the static model, because changing the size of the observation history matrix does not affect transitional information. So although the use of an observation history matrix works for

the static model, we lose the advantage we had with the dynamic model.

#### 4.6 Comparison with results from other researchers

So far we have tested several parameters for our models to learn which value optimizes the performance. To see how much better our models perform, we compare them with the results of another researcher. In the work of Tapia the same dataset and the same scheme was used for training and testing the models. However, Tapia used a different method for calculating the performance of his models. He introduced three different accuracy methods, arguing that different applications have different requirements.

The first method expresses the 'percentage of time that activity is detected'. This measures the percentage of time that the activity is correctly detected for the duration of the labeled activity. The second method is referred to as 'Activity detected in the best interval'. This measures whether the activity was detected at the end of the real activity or within an interval of 7.5 minutes before or after the end. The delay interval of 7.5 minutes was chosen as it was the average delay observed in the dataset. Finally the third method is referred as 'Activity detected at least once'. This measures if an activity was detected at least once for the duration of the activity label. For a more detailed explanation of these accuracy methods please refer to page 67-69 of [5].

We implemented all three methods and compare our best combination of parameter values with the best results Tapia posted in his work. Our best results were obtained by using 30 sensors, a time interval of 150 seconds and a history size of 1 (table 1).

Method	Tapia	Static	Dynamic
Time	0.2273	0.3033	0.3457
Best Interval	0.5780	0.5440	0.5430
Detected	0.3861	0.5838	0.5565

Table 1: Comparison of the best results from our static and dynamic Bayesian models with the best results posted in work by Tapia [5]

## 5 Conclusion

In this paper we ran experiments for various parameter values to determine the effect on the performance of our models. Furthermore, we introduced a new observation model for performing activity recognition.

In testing the number of sensors used, we see that adding more sensors does not necessarily increase the accuracy. Sensors placed at very rarely used locations hardly ever fire, therefore it requires very large datasets to get an accurate parameter estimation for these sensors. On the one hand the use of large datasets is discouraging, on the other hand sensors placed in rarely used locations might be very descriptive for one particular activity. It would be interesting to investigate whether a model can be created that can be trained with less samples while maintaining the descriptive quality of such sensors.

By trying various time interval values we learned that this parameter greatly affects the accuracy and should therefore be chosen carefully. However, choosing a proper value is not obvious as shown by the results of the experiment. Although choosing a large time interval results in a high mean accuracy, the variance becomes large as well making the classifier less reliable. Overall the notion of time interval seems counter intuitive. It asks us to cut up a time series signal into pieces, without having a good rationale for using a particular value. We are therefore interested in how an event based model would perform and will focus on this in future work.

We showed how the use of an observation history increases the accuracy in the case of the static model. The use of the observation history allows our model to capture more correlations in the sensor pattern. Ideally we would like our model to fully observe all the possible correlations in a sensor pattern (for example by releasing the independence assumption among sensors). However, the difficulty with each possible approach is that the number of parameters becomes too large to get accurate estimations. If a representation for capturing all such correlations in a sensor pattern can be used as an observation model, for example by using a probabilistic distribution, it is expected to increase the accuracy for activity recognition models dramatically.

## Acknowledgements

This work is part of the Context Awareness in Residence for Elders (CARE) project. The CARE project is partly funded by the Centre for Intelligent Observation Systems (CIOS) which is a collaboration between UvA and TNO, and partly by the EU Integrated Project COGNIRON (The Cognitive Robot Companion).

## References

- [1] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, August 2006.
- [2] S. Katz, T.D. Down, H.R. Cash, and et al. Progress in the development of the index of adl. *Gerontologist*, 10:20–30, 1970.
- [3] Kevin Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, UC Berkeley, 2002.
- [4] Donald J. Patterson, Dieter Fox, Henry A. Kautz, and Matthai Philipose. Fine-grained activity recognition by aggregating abstract object usage. In *ISWC*, pages 44–51. IEEE Computer Society, 2005.
- [5] Emmanuel Munguia Tapia. Activity recognition in the home setting using simple and ubiquitous sensors. Master's thesis, Massachusetts Institute of Technology, 2003.
- [6] Daniel H. Wilson. *Assistive Intelligent Environments for Automatic Health Monitoring*. PhD thesis, Carnegie Mellon University, 2005.