An activity monitoring system for elderly care using generative and discriminative models

van Kasteren, T.L.M.; Englebienne, G.; Kröse, B.J.A.

DOI
10.1007/s00779-009-0277-9

Publication date
2010

Document Version
Final published version

Published in
Personal and ubiquitous computing

Link to publication

Citation for published version (APA):
van Kasteren, T. L. M., Englebienne, G., & Kröse, B. J. A. (2010). An activity monitoring system for elderly care using generative and discriminative models. *Personal and ubiquitous computing*, *14*(6), 489-498. https://doi.org/10.1007/s00779-009-0277-9

Pers Ubiquit Comput (2010) 14:489–498
DOI 10.1007/s00779-009-0277-9

ORIGINAL ARTICLE

An activity monitoring system for elderly care using generative and discriminative models

T. L. M. van Kasteren · G. Englebienne ·
B. J. A. Kröse

Received: 31 December 2008 / Accepted: 5 October 2009 / Published online: 26 February 2010

Abstract An activity monitoring system allows many applications to assist in care giving for elderly in their homes. In this paper we present a wireless sensor network for unintrusive observations in the home and show the potential of generative and discriminative models for recognizing activities from such observations. Through a large number of experiments using four real world datasets we show the effectiveness of the generative hidden Markov model and the discriminative conditional random fields in activity recognition.

Keywords Activity recognition · Machine learning · Wireless sensor networks

1 Introduction

As the number of elderly people in our society increases so does the need for assistive technology in the home. Elderly people run into all sorts of barriers in performing their daily routines as they get older. Activities of daily living (ADLs), such as bathing, toileting and cooking, are good indicators of the cognitive and physical capabilities of elderly [10]. Therefore, a system that automatically recognizes these activities allows automatic health monitoring [3, 18, 19, 26, 35], evidence-based nursing [7] and provides an objective measure for nursing personnel [1, 36]. Such a system can also be used to support people with dementia by reminding them which steps to take to complete an activity [24]. An activity monitoring system is therefore a crucial step in future care giving.

An activity monitoring system consists of sensors that observe what goes on in the house and a recognition model to infer the activities from sensor data. In previous work a variety of sensing modalities has been used. One approach is to tag a large number of objects in a house with RFID tags. An RFID reader in the form of a bracelet is worn by the user to detect which objects are used [6]. The observed objects are used as input for activity recognition [12, 21]. Another approach is to use video. Duong et al. [5] use four cameras to capture a scene from different angles. From the videos they extract the location of a user and use it for activity recognition. Wu et al. [37] use a single camera combined with an RFID bracelet. Experiments compare the performance of a model using only video and a model using both video and RFID. The results show equal performance suggesting RFID does not add any information. A similar conclusion can be drawn from Logan et al. [16] in which RFID is compared to sensors installed in the Placelab. The Placelab is a custom-built house equipped with several hundred wall-mounted sensors, such as reed switches on doors and cupboards, temperature sensors and water flow detectors [9]. Both the RFID bracelet and the wall-mounted sensors were used to record a large dataset annotated with activities. The performance of a naive Bayes classifier using only wall-mounted sensors is higher than the performance of the same model using only RFID for almost all activities in this dataset [16].

Although RFID does not seem to give very good results, both video and wall-mounted sensors appear to be suitable sensing modalities. However, because an activity monitoring system is installed in a home setting it is important that a non-intrusive sensing modality is used. Because the acceptance of video cameras in a home is still questionable,

T. L. M. van Kasteren (✉) · G. Englebienne · B. J. A. Kröse
Intelligent Systems Lab Amsterdam, University of Amsterdam,
Science Park 107, 1098 XG Amsterdam, The Netherlands
e-mail: T.L.M.vanKasteren@uva.nl

we do not use video. Furthermore, to deploy the system on a large scale it needs to be easily installable in existing houses. Therefore, in this work we use a wireless sensor network which satisfies these criteria. We provide a detailed description of our approach and present two datasets that were recorded using this system.

The most difficult element of an activity monitoring system is the activity recognition model. This model interprets the observed sensor patterns and recognizes the activities performed. This is a challenging task because sensor data is noisy and the start and end point of an activity are not known. Typical approaches include the naive Bayes model in which sensors are modeled independently and no temporal information is incorporated [16, 27]. The hidden Markov model (HMM) adds temporal information by modeling the transition from one activity to the next [21, 32]. Several extensions on the classic hidden Markov model have been introduced, such as tracking a person's location and using that information as an additional input for activity recognition [34]. Further extensions include a switching semi-hidden Markov model [5] and a hierarchical hidden Markov model [20]. These are all generative probabilistic models that require an explicit model of the dependencies between the observed sensor data and the activities. However, in activity recognition there are long-term dependencies which are difficult to model. This problem occurs in many other fields as well, which is why recently a discriminative probabilistic model known as conditional random fields (CRF) has become very popular [11, 14, 17, 30, 33]. It has been applied to activity recognition in a home from video [28, 29] and from wall-mounted sensors [8]. However, in these works either the datasets were recorded in a lab setting or consist of only four hours of real world data. In this paper we give a comparison of generative HMMs and discriminative CRFs using several real world datasets consisting of at least 10 days of data.

This paper is organized as follows: Section 2 describes the difference between generative and discriminative probabilistic models. Section 3 presents our system for activity monitoring in detail. Section 4 describes the experiments, datasets and results for evaluating our approach. Finally, Sect. 5 sums up our conclusions.

2 Generative versus discriminative models

In temporal classification problems, like activity recognition, we typically have a sequence of observations $\mathbf{x}_{1:T} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ and wish to infer the matching sequence of class labels $y_{1:T} = \{y_1, y_2, \ldots, y_T\}$. Generative models, like hidden Markov models and dynamic Bayesian networks, deal with this problem by explicitly modeling the relations between the observations and the class labels. More specifically, in generative models we express the dependencies among variables ($\mathbf{x}$ and $\mathbf{y}$). For example, the Markov assumption states that the current state y_t depends only on the previous state y_{t-1}. As a result we can express our belief in y_t based only on y_{t-1} and ignore all the other variables (i.e., x_t, y_{t-2}, etc.) These dependencies, or rather the independence assumptions with respect to the other variables, therefore greatly reduce the number of parameters that specify the model [2, 4]. However, a violation of dependencies, meaning there exist dependencies in the actual data that we do not model, can strongly affect the performance of the model [30].

The use of dependencies can lead to very elegant and well-performing models when a process is well understood. However, in complicated tasks it is often difficult to find the proper dependencies. Discriminative models, therefore, avoid making independence assumptions among the observations. The idea is, because observations are always given during inference anyway, there is no need to model them explicitly. Instead discriminative models directly model the discriminative boundary between the different class labels [2]. The advantage of this approach is that we can incorporate all sorts of rich overlapping features without violating any independence assumptions [25].

Conditional random fields (CRF) are temporal discriminative probabilistic models that have this property. They were first applied in language problems where features such as capitalization of a word and the presence of particular suffixes significantly improved performance [11]. Since then they have been applied to a variety of domains such as gesture recognition [17], scene segmentation [33] and the activity recognition of robots in a game setting [30]. They have also been applied to human activity recognition from video, in which primitive actions such as 'go from fridge to stove' are recognized in a lab-like kitchen setup [28, 29]. A variation of CRFs known as skip-chain CRFs has been applied to an activity dataset consisting of 4 h of sensor data [8]. In an outside setting CRFs have been applied on activity recognition from GPS data [14] in which activities such as 'going to work' and 'visiting a friend' were distinguished.

3 Activity monitoring system

Our activity monitoring system consists of a wireless sensor network and a recognition model. We give a detailed description of the wireless network nodes and the sensors used. Furthermore, we describe the generative hidden Markov model and the discriminative conditional random field for activity recognition.

3.1 Wireless sensor network

Our wireless sensor network consists of wireless network nodes to which sensors can be attached. The network nodes are manufactured by RFM and come with a firmware that includes an energy efficient network protocol allowing a long battery life. The nodes can be equipped with an analog or digital sensor and communicate wireless with a central gateway attached to a server on which all data are logged (Fig. 1). An event is sent when the digital input changes state or when a threshold on the analog input is violated. Sensors we used are: reed switches to measure open–close states of doors and cupboards; pressure mats to measure sitting on a couch or lying in bed; mercury contacts for movement of objects (e.g., drawers); passive infrared (PIR) to detect motion in a specific area; float sensors to measure the toilet being flushed; temperature sensors to measure the use of the stove or shower. Because our system is wireless the nodes and sensors can be installed using tape, allowing an easy installation.

We have recorded two datasets using our system. One consisting of 4 weeks of fully annotated data in the apartment of a 26-year-old male and another one consisting of 2 weeks of fully annotated data in the house of a 57-year-old male. Further details on the datasets can be found in the Sect. 4.

3.2 Probabilistic models for activity recognition

The sensor data obtained from the wireless sensor network need to be processed by the activity recognition models to determine which activities took place. We first present the notation we use in describing the models, then describe the hidden Markov model and then the conditional random fields.

3.2.1 Notation

The time series data obtained from the sensors are divided in time slices of constant length Δt . We denote a sensor reading for time t as x_t^i , indicating whether sensor i fired at least once between time t and time $t + \Delta t$, with $x_t^i \in \{0, 1\}$. In a house with N sensors installed, we define a binary observation vector $\mathbf{x}_t = \left(x_t^1, x_t^2, \ldots, x_t^N\right)^T$. An activity at time slice t is denoted by y_t with $y_t \in \{1 \ldots K\}$, therefore, the recognition task is to find a sequence of labels $y_{1:T} = \{y_1, y_2, \ldots, y_T\}$ that best explains the sequence of observations $\mathbf{x}_{1:T} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ for a total of T time steps.

3.2.2 Hidden Markov model

The hidden Markov Model (HMM) is a generative probabilistic model consisting of a hidden variable y and an observable variable x at each time step (Fig. 2). In our case the hidden variable is the activity performed, and the observable variable is the vector of sensor readings. Generative models provide an explicit representation of dependencies by specifying the factorization of the joint probability of the hidden and observable variables $p(y_{1:T}, \mathbf{x}_{1:T})$. In the case of HMMs there are two dependency assumptions that define this model, represented with the directed arrows in the figure.

- The hidden variable at time t, namely y_t, depends only on the previous hidden variable y_{t-1} (*Markov assumption* [22]).

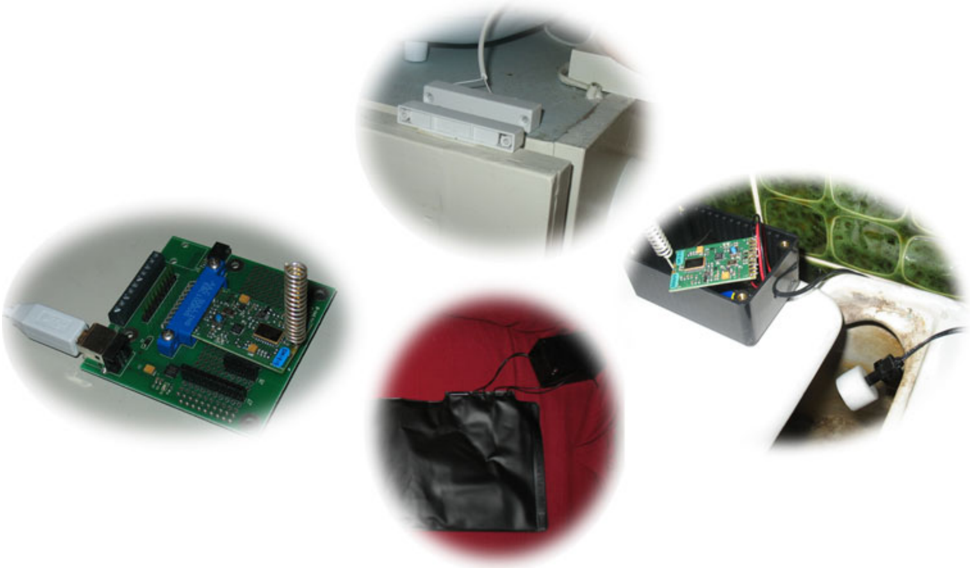

Fig. 1 Wireless sensor network installed in a home setting: a reed switch, float sensor and pressure mat communicate with a gateway which stores the incoming data on a server

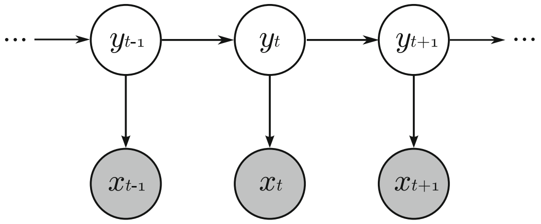

Fig. 2 The graphical representation of a HMM. The *shaded nodes* represent observable variables, while the *white nodes* represent hidden ones

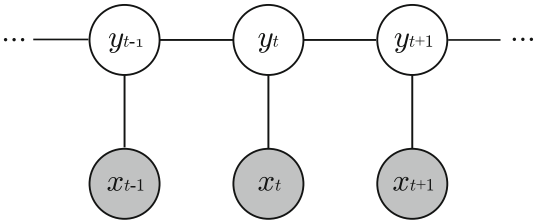

Fig. 3 The graphical representation of a CRF. The *shaded nodes* represent observable variables, while the *white nodes* represent hidden ones

- The observable variable at time t, namely x_t, depends only on the hidden variable y_t at that time slice.

The joint probability therefore factorizes as follows

$$p(y_{1:T}, \mathbf{x}_{1:T}) = p(y_1)p(\mathbf{x}_1|y_1)\prod_{t=2}^{T} p(y_t|y_{t-1})p(\mathbf{x}_t|y_t). \tag{1}$$

The different factors further specify the workings of the model. The initial state distribution $p(y_1)$ is a probability table with individual values denoted as follows

$$p(y_1 = i) \equiv \pi_i. \tag{2}$$

The observation distribution $p(\mathbf{x}_t|y_t)$ indicating the probability that the state y_t would generate observation **x_t**. In our case each sensor observation is modeled as an independent Bernoulli distribution, giving

$$p(\mathbf{x}_t|y_t) = \prod_{n=1}^{N} p\left(x_t^n|y_t\right) \tag{3}$$

$$p\left(x_t^n = v|y_t = i\right) = (\mu_{in})^v (1 - \mu_{in})^{1-v}. \tag{4}$$

The transition probability distribution $p(y_t|y_{t-1})$ represents the probability of going from one state to the next. This is given by a conditional probability table where individual transition probabilities are denoted as follows:

$$p(y_t = j|y_{t-1} = i) \equiv a_{ij}. \tag{5}$$

Our HMM is therefore fully specified by the following parameters $A = \{a_{ij}\}$, $B = \{\mu_{in}\}$ and $\pi = \{\pi_i\}$. The parameters are learned from training data using maximum likelihood. This is done for each class separately. Using these parameters we can find the sequence of activities that best fit to a novel sequence of observations. This is done by calculating $p(y_{1:T}|\mathbf{x}_{1:T})$, which can be easily calculated from the joint probability. We can efficiently find the sequence of activities that maximizes this probability using the commonly used Viterbi algorithm [22].

3.2.3 Conditional random fields

The parameters of generative models, discussed in the previous section, are learned by maximizing the joint probability $p(\mathbf{x}_{1:T}, y_{1:T})$. While, to infer the best fit to a novel sequence of observations we calculate $p(y_{1:T}|\mathbf{x}_{1:T})$. In discriminative models, like CRFs (Fig. 3), we learn the parameters by maximizing the conditional probability $p(y_{1:T}|\mathbf{x}_{1:T})$ directly. Thus, in discriminative models we learn the parameters by maximizing the same quantity used for inference. In CRFs this is calculated as follows:

$$p(y_{1:T}|\mathbf{x}_{1:T}) = \frac{1}{Z(\theta)} \exp \sum_{k=1}^{K} \theta_k \sum_{t=1}^{T} \phi_k(\mathbf{x}, \mathbf{y}, t) \tag{6}$$

where ϕ_k are the so-called feature functions and are chosen to model properties of the observations and transitions between states. The nominator of this function is straightforward and fast to compute; the complexity of the model lies in the computation of the normalisation term $Z(\theta)$, which takes into account all possible state sequences corresponding to the given observation sequence. Since in an unconstrained model the number of state sequences would grow exponentially in the length of the sequence, CRFs restrict the feature functions to model nth order Markov chains only, where n is typically 1. That is, the feature functions can be written in the from

$$\phi_k(\mathbf{x}, \mathbf{y}, t) = \phi'_k(\mathbf{x}, y_{t-1}, y_t, t) \tag{7}$$

The probability of the current state then only depends on the previous state, so that the probability of all possible state sequences depends only on all possible pairwise state combinations. This allows us to compute $Z(\theta)$ in linear time.

The parameters of the CRF can be efficiently optimised by any variant of gradient descent, the gradients can be computed in linear time. We optimise the parameters θ with limited memory BFGS [15], an efficient quasi-Newton iterative optimisation method. For a novel sequence of data the sequence of activities that best fits to the data is found using the Viterbi algorithm [22].

4 Experiments

In this section we present the experimental results acquired in this work. We start by describing the objectives of our experiments, then the experimental setup and the realworld datasets. This section concludes with a presentation of the acquired results and a discussion.

4.1 Objective

The goal of our experiments is to show the effectiveness of generative and discriminative models for activity recognition in a real world setting. We compare the performance of the hidden Markov model and the conditional random fields on datasets recorded in four different homes. Furthermore, we show the effectiveness of using other sensor data representations than the raw sensor data.

4.2 Setup

In our experiments the sensor readings are discretized in data segments of length $\Delta t = 60\,\mathrm{s}$. This time slice duration is long enough to be discriminative and short enough to provide high-accuracy labeling results. We separate our data into a test and training set using a ‘leave one day out’ approach. In this approach, one full day of sensor readings is used for testing and the remaining days are used for training. We cycle over all the days and report the average accuracy.

We evaluate the performance of our models by two measures, the time slice accuracy and the class accuracy. The timeslice accuracy shows the percentage of correctly classified timeslices, the class accuracy shows the average percentage of correctly classified timeslices per class. These measures are defined as follows:

$$\text{Time slice}: \frac{\sum_{n=1}^{N}[\mathit{inferred}(n) = \mathit{true}(n)]}{N}$$

$$\text{Class}: \frac{1}{C}\sum_{c=1}^{C}\left\{\frac{\sum_{n=1}^{N_c}[\mathit{inferred}_c(n) = \mathit{true}_c(n)]}{N_c}\right\}$$

in which $[a = b]$ is a binary indicator giving 1 when true and 0 when false. N is the total number of time slices and C is the number of classes.

Measuring the time-slice accuracy is a typical way of evaluating time-series analysis. However, we also report the class average accuracy, which is a common technique in datasets with a dominant class. In these cases classifying all the test data as the dominant class yields good time-slice accuracy, but no useful output. The class average, however, would remain low, and therefore be representative of the actual model performance.

We experiment using three different sensor representations. The *raw* sensor representation gives a 1 when the sensor is firing and a 0 otherwise. The *change point* representation gives a 1 to timeslices where the sensor changes state and a 0 otherwise. Finally, the *last* sensor fired representation continues to give a 1 to the last sensor that changed state and changes to 0 when a different sensor changes state [32].

4.3 Datasets

We used four datasets in our experiments. Two datasets were recorded using our sensing system described in Sect. 3. The other two datasets were introduced in [27] and used reed switch sensors on doors, cupboards and drawers. The details of all the datasets are shown in Table 1. The floorplan of one of the apartments together with the location of the sensors is shown in Fig. 4.

Annotation in the TK26M dataset was done using a Bluetooth headset, the user recorded the start and end point of an activity by speaking into the headset. The TK57M dataset was annotated using activity diaries on several locations in the house. Both the Tap30F and Tap80F dataset were annotated using a PDA that asked the user the performed activity every 15 min, annotation was later corrected by inspection of the sensor data. The activities that were annotated in the datasets mainly consist of the basic activities of daily living (ADLs) as defined by [10]. These include bathing, dressing and toileting. Additionally, instrumental ADLs [13] such as preparing meals and taking medications were annotated. A complete list of the activities together with the number of times they occur and the percentage of time they take up in the dataset is given in Table 2.

The TK26M dataset is publicly available for download from http://www.science.uva.nl/~tlmkaste.

4.4 Results

We compared the performance of the HMM and CRF on the four datasets. Experiments were run using the ‘raw’, ‘changepoint’, ‘last’ and ‘Changepoint+Last’ sensor representation. The ‘Changepoint+Last’ representation is a

Table 1 Information about the various datasets. Two of the datasets were recorded by us, for the other two a reference is given where the dataset was introduced

Dataset	TK26M	TK57M	Tap30F	Tap80F
Age	26	57	30	80
Gender	Male	Male	Female	Female
Setting	Apartment	House	Apartment	Apartment
Duration	28 days	13 days	16 days	14 days
Sensors	14	21	77	84
Activities	10	11	14	10
Recorded by	Authors	Authors	[27]	[27]

concatenation of the changepoint and last representations. The results are shown in Table 3.

The table shows that CRFs mainly outperform HMMs on the timeslice accuracy measure, but HMMs mainly outperform CRFs on the class accuracy measure. When comparing the sensor representations we see the 'changepoint+last' sensor representation gives the best results in the case of the TK26M and TK57M datasets, but the 'changepoint' representation mainly gives the best results on the Tap30F and Tap80F datasets. The 'raw' sensor representation gives by far the worst results. Out of the datasets used the accuracies for both models are higher on the TK26M and TK57M datasets than on the Tap30F and Tap80F datasets.

These differences are better understood by looking at the confusion matrices, which show the accuracy results for each activity separately. Table 4 shows the confusion matrix for the HMM and Table 5 for the CRF, both using the 'changepoint+last' representation on the TK26M dataset. The tables show CRFs mainly perform better on the 'other' activity, while HMMs do better on the various kitchen activities.

To compare the sensor representations Table 6 shows the confusion matrix for the changepoint representation and Table 7 for the changepoint+last representation, both for HMMs on the Tap80F dataset. In the case of the 'changepoint' representation the highest accuracy is achieved for the 'other' activity. However, this activity is also mostly confused with the remaining activities meaning the majority of the timeslices were labeled as 'other'. The 'changepoint+last' representation outperforms the 'changepoint' representation on all but the 'other' and 'lunch' activity.

4.5 Discussion

The difference in performance between CRFs and HMMs is due to the different ways these models are trained. CRFs are trained by maximizing the likelihood over the entire sequence of training data. HMMs are trained by splitting the training data according to the class labels and optimise the parameters for each class separately. As a result classes that are more dominantly present in the data have a bigger weight in the CRF optimisation. The HMM on the other hand treats each class equally, as parameters are learned for each class separately. This explains why CRFs perform so much better on the 'other' activity, since this activity takes up 12.7% of the data in the TK26M dataset. In other words, CRFs overfit on the 'other' activity since it occurs more often in the dataset. On the other hand, HMMs perform

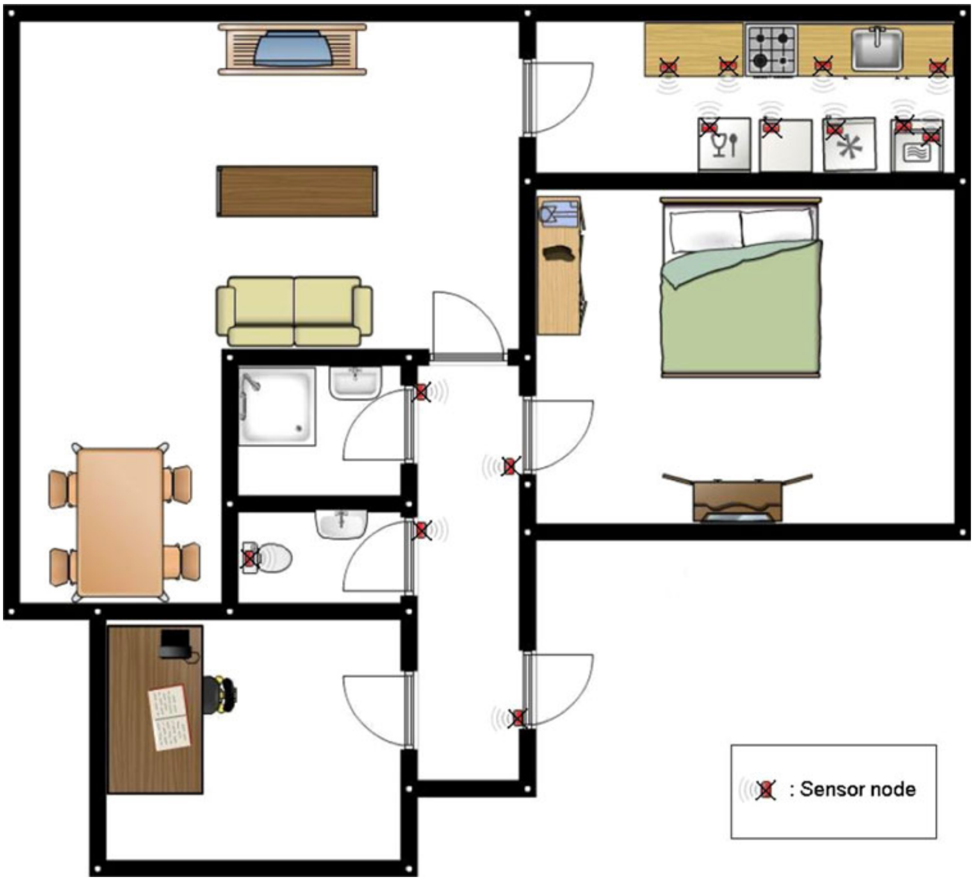

Fig. 4 Floorplan of the house of dataset TK26M, *small rectangle boxes* with an X-mark in the middle indicate sensor nodes. (created using http://www.floorplanner.com/)

Table 2 The activities that were annotated in the different datasets

TK26M			TK57M			Tap30F			Tap80F		
Activity	Num.	Time (%)	Activity	Num.	Time (%)	Activity	Num.	Time (%)	Activity	Num.	Time
Leave	34	51.3	Leave	29	50.5	Leave	12	0.4	Meds.	14	0.8
Toilet	114	1.1	Toilet 1	33	0.6	Toilet	85	3.6	Toilet	40	2.8
Shower	23	0.7	Toilet 2	22	0.4	Bathe	18	2.0	TV	15	3.2
Sleep	24	32.5	Shower	10	0.5	Groom	37	2.1	Music	18	4.8
Breakf.	20	0.3	Sleep	12	27.2	Breakf.	14	1.0	Breakf.	18	2.0
Dinner	10	1.0	Dress	15	0.4	Dinner	8	0.8	Dinner	14	2.2
Snack	11	0.1	Breakf.	10	0.5	Snack	14	0.6	Snack	16	1.4
Drink	20	0.2	Dinner	9	1.6	Drink	15	0.7	Dishes	21	1.5
Dishw.	5	0.1	Groom	28	0.8	Lunch	17	3.0	Lunch	20	3.7
Other	–	12.7	Relax	26	8.4	Dishes	7	0.3	Other	–	77.6
			Other	–	9.1	Clean	8	0.8			
						Laundry	19	1.1			
						Dress	24	0.7			
						Other	–	82.9			

The TK57M dataset contains two toileting activities because the house has two toilets

Num. number of times the activity occurs in the dataset, *Time* percentage of time the activity takes up in the dataset. Abbreviated activity names are: *Leave* leaving the house, *meds.* taking medication, *Breakf.* preparing breakfast, *Dishw.* using the dishwasher, *TV* watching TV, *Music* listening to music. All unannotated timeslices are collected in a single activity named *Other*

Table 3 Timeslice and class accuracies for hidden Markov model (HMM) and conditional random field (CRF) using various sensor representations

Model	Representation	TK26M		TK57M		Tap30F		Tap80F	
		Timeslice (%)	Class (%)	Timeslice (%)	Class (%)	Timeslice (%)	Class (%)	Timeslice (%)	Class (%)
HMM	Raw	57.3	49.5	32.6	20.5	57.4	24.0	14.9	17.1
	Changepoint	81.7	67.2	75.9	51.2	**85.1**	29.2	**80.0**	30.1
	Last	87.2	71.0	79.8	60.0	55.9	32.7	31.3	**34.4**
	Changepoint+Last	**91.2**	**76.0**	**80.3**	**60.4**	64.7	**35.6**	40.2	33.9
CRF	Raw	83.9	53.0	37.6	22.4	66.7	21.2	51.3	16.4
	Changepoint	91.6	62.6	82.9	47.9	**85.6**	**28.5**	**79.8**	**29.3**
	Last	94.0	59.4	86.1	49.3	83.9	24.0	74.9	17.6
	Changepoint+Last	**95.1**	**68.4**	**86.6**	**49.8**	83.7	26.3	77.2	27.1

Experiments were performed on four different datasets: TK26M, TK57M, Tap30F and Tap80F. *Bold values* the highest value in a column

better at classifying the kitchen activities. Because parameters are learned for each class separately, there is no overfitting on one particular class. However, this comes at the cost of missclassifying some of the often occurring 'other' activity. This explains why CRFs perform better in terms of the timeslice accuracy measure, while HMMs perform better in terms of the class accuracy measure.

Performance differences using the various sensor representations are due to the resulting feature spaces. We see that the 'raw' sensor representation gives by far the worst results, the 'changepoint' and 'changepoint+last' representation work best. The reason for this is that the actual state of a sensor (represented by the 'raw' representation) is not very informative, it is more useful to know when a sensor changes state (represented by the 'changepoint' representation). For example, if we want to recognize when somebody is sleeping it will be informative to know the bedroom door opened. However, many times people leave the door open once they get out of bed. The raw sensor representation will continue to give a 1 as long as the door is open, even though somebody might already be involved in a different activity. The changepoint representation solves this by only giving a 1 when the state of a sensor changes. This indicates something happened with the

Table 4 Confusion matrix for the HMM using changepoint+last representation on the TK26M dataset

	Other	Leaving	Toileting	Showering	Sleeping	Breakfast	Dinner	Snack	Drink	Dishwasher
Other	52.1	6.4	0.7	12.4	4.6	0.1	20.2	0.2	0.6	2.7
Leaving	1.6	98.2	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0
Toileting	4.2	0.5	86.0	4.5	3.4	0.0	0.8	0.3	0.3	0.0
Showering	6.4	0.4	3.4	89.1	0.0	0.0	0.0	0.0	0.0	0.8
Sleeping	2.6	0.0	0.4	0.1	96.9	0.0	0.0	0.0	0.0	0.0
Breakfast	23.6	0.0	0.9	0.0	0.0	49.1	5.7	12.3	5.7	2.8
Dinner	7.7	0.3	0.6	0.0	0.6	6.0	74.6	8.0	1.4	0.9
Snack	9.1	0.0	0.0	2.3	0.0	18.2	11.4	59.1	0.0	0.0
Drink	3.7	0.0	0.0	0.0	0.0	9.3	11.1	0.0	75.9	0.0
Dishwasher	0.0	0.0	0.0	0.0	0.0	0.0	21.4	0.0	0.0	78.6

The values are percentages

Table 5 Confusion matrix for the CRF using changepoint+last representation on the TK26M dataset

	Other	Leaving	Toileting	Showering	Sleeping	Breakfast	Dinner	Snack	Drink	Dishwasher
Other	84.6	5.5	0.5	0.4	7.0	0.3	1.4	0.0	0.2	0.1
Leaving	1.4	98.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Toileting	6.9	2.6	76.5	2.9	10.8	0.3	0.0	0.0	0.0	0.0
Showering	25.6	1.9	2.3	70.3	0.0	0.0	0.0	0.0	0.0	0.0
Sleeping	0.8	1.9	0.3	0.0	97.0	0.0	0.0	0.0	0.0	0.0
Breakfast	24.5	0.0	0.9	0.0	0.9	55.7	6.6	8.5	2.8	0.0
Dinner	34.9	1.1	0.6	0.0	0.0	1.7	58.9	0.9	0.9	1.1
Snack	20.5	4.5	0.0	0.0	0.0	13.6	15.9	45.5	0.0	0.0
Drink	27.8	1.9	3.7	0.0	0.0	3.7	13.0	0.0	50.0	0.0
Dishwasher	50.0	0.0	0.0	0.0	0.0	0.0	2.4	0.0	0.0	47.6

The values are percentages

Table 6 Confusion Matrix for the HMM using changepoint representation on the Tap80F dataset

	Other	Toileting	Take medication	Breakfast	Lunch	Dinner	Snack	Dishes	Watching TV	Listen to music
Other	94.6	0.8	0.2	1.5	0.7	0.4	0.6	0.4	0.2	0.6
Toileting	73.7	11.7	1.4	7.9	0.5	0.0	1.4	2.5	0.5	0.3
Take medication	82.2	0.0	5.4	7.6	0.0	0.0	4.9	0.0	0.0	0.0
Breakfast	52.8	0.0	0.0	30.9	15.2	1.1	0.0	0.0	0.0	0.0
Lunch	31.4	0.4	0.0	7.9	41.8	12.5	1.5	2.5	0.0	2.0
Dinner	29.4	1.4	0.0	14.2	26.1	20.2	6.1	2.0	0.0	0.6
Snack	38.4	0.0	0.0	5.9	25.3	0.9	8.1	1.6	4.7	15.0
Dishes	42.7	2.1	0.0	0.0	26.5	4.3	2.4	12.8	3.4	5.8
Watching TV	65.1	4.9	0.0	0.6	0.0	0.0	1.3	1.8	26.4	0.0
Listen to music	47.2	0.0	0.0	0.0	2.5	0.0	0.4	0.5	0.0	49.5

The values are percentages

bedroom door, however, it remains unclear whether someone entered the room or left the room. The ‘last’ sensor representation helps there. If someone enters the room the last sensor that fires is either the door to that room or any sensor inside the room. While if someone exits the room any of the other sensors outside the room are likely to fire.

Out of all the datasets used the TK26M gave the best results. The reason the results are better than the TK57M dataset is due to the layout of the house and due to the

Table 7 Confusion Matrix for the HMM using changepoint+last representation on the Tap80F dataset

	Other	Toileting	Take medication	Breakfast	Lunch	Dinner	Snack	Dishes	Watching TV	Listen to music
Other	41.1	13.7	12.0	5.2	5.6	1.9	1.9	5.1	6.3	7.3
Toileting	15.6	40.2	10.5	6.7	6.3	4.4	4.4	7.6	1.3	3.0
Take medication	47.6	10.8	20.5	7.6	0.0	5.9	4.9	0.0	2.7	0.0
Breakfast	13.1	4.3	6.0	41.4	24.5	3.6	1.7	1.5	2.1	1.7
Lunch	10.8	3.3	3.9	28.4	24.6	9.8	2.4	10.2	1.4	5.2
Dinner	7.1	4.0	20.9	5.9	21.7	21.9	6.5	4.7	2.0	5.1
Snack	22.5	2.5	17.5	7.5	16.3	3.1	9.1	13.4	0.9	7.2
Dishes	8.2	6.4	0.0	5.2	22.3	4.6	6.4	38.4	2.1	6.4
Watching TV	18.0	12.0	4.6	1.0	2.6	0.7	1.7	11.2	48.3	0.0
Listen to music	18.4	2.6	4.8	0.2	12.2	1.1	0.8	3.7	2.1	54.1

The values are percentages

annotation method. As the floorplan of the TK26M house in Fig. 4 shows, there is a separate room for almost every activity. The kitchen is the only place where multiple activities are performed in a single room. This means for most activities the door sensor to the room the activity is performed in is very informative. The method of annotation used for the dataset is also relevant. For the TK26M a Bluetooth headset was used which communicated with the same server the sensor data was logged on. This means the timestamps of the annotation were synchronized with the timestamps of the sensors. In the TK57M activity diaries were used, this is more error prone because time might not always be written down correctly and the diaries have to be typed over afterwards. The large difference in performance between the two TK datasets and the two Tap datasets is most likely also because of the annotation methods used. In the Tap datasets activities were annotated using a PDA that asks the user every 15 min which activity is performed. Since our classification is done using timeslices of 60 s there can be large deviations between the time the actual activity took place (represented by the sensor data) and the time the activity was annotated. Furthermore, it was reported that the labels obtained using the PDA were not sufficient for training a classifier and annotation was added later by inspecting the sensor data [27].

In terms of future work it would be interesting to perform activity recognition using hybrid generative–discriminative models, combining the best of both worlds. It has been shown that hybrid models can provide better accuracy than their purely generative or purely discriminative counterparts [23]. Furthermore, the model parameters learned during training are specific to the house the labeled training data was recorded in. Applying the system in another house would require re-estimation of the parameters, which typically means labeled training data needs to be recorded in the new house. Since this is problematic for deploying such a system on a large scale, it is interesting to use transfer learning techniques to transfer the knowledge about activity recognition from one house to the next [31].

5 Conclusions

This paper presents an activity monitoring system consisting of sensors and an activity recognition model. We presented an unintrusive wireless sensor network and described two datasets that were recorded using this network. Furthermore, we showed the potential of generative and discriminative models for activity recognition, by comparing their performance on four real world datasets consisting of at least 10 days of data. Our experiments show three things. First, that CRFs are more sensitive to overfitting on a dominant class than HMMs. Second, that the use of raw sensor data gives bad results and that a 'changepoint' or 'changepoint+last' sensor representations gives much better results. And third, that differences in the layout of houses and the way a dataset was annotated can greatly affect the performance in activity recognition.

Acknowledgments 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 the Universiteit van Amsterdam (UvA) and the Nederlandse Organisatie voor toegepast-natuurwetenschappelijk onderzoek (TNO), and partly by the EU Integrated Project COGNIRON (The Cognitive Robot Companion).

References

1. Allin SJ, Bharucha A, Zimmerman J, Wilson D, Roberson MJ, Wactlar H, Atkeson CG (2003) Toward the automatic assessment

of behavioral disturbances of dementia. In: UbiHealth 2003: The 2nd international workshop on ubiquitous, pp 12–15

2. Barber D (2010) Bayesian reasoning and machine learning, Cambridge University Press
3. Barger T, Brown D, Alwan M (2003) Health status monitoring through analysis of behavioral patterns. In: 8th congress of the Italian Association for Artificial Intelligence (AIIA) on ambient intelligence, Springer-Verlag, pp 22–27
4. Bishop CM (2006) Pattern recognition and machine learning (Information Science and Statistics). Springer
5. Duong TV, Bui HH, Phung DQ, Venkatesh S (2005) Activity recognition and abnormality detection with the switching hidden semi-Markov model. In: CVPR '05: Proceedings of the 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). IEEE Computer Society 1:838–845
6. Fishkin KP, Philipose M, Rea A (2005) Hands-on rfid: wireless wearables for detecting use of objects. In: ISWC 2005: Proceedings of the ninth IEEE international symposium on wearable computers, IEEE Computer Society, Washington, DC, USA, pp 38–43
7. Hori T, Nishida Y, murakami S (2006) Pervasive sensor system for evidence-based nursing care support. ICRA, pp 1680–1685
8. Hu DH, Pan SJ, Zheng VW, Liu NN, Yang Q (2008) Real world activity recognition with multiple goals. In: UbiComp '08: Proceedings of the 10th international conference on ubiquitous computing, ACM, New York, NY, USA, pp 30–39
9. Intille SS, Larson K, Tapia EM, Beaudin J, Kaushik P, Nawyn J, Rockinson R (2006) Using a live-in laboratory for ubiquitous computing research pervasive computing, pp 349–365
10. Katz S, Down T, Cash H et al (1970) Progress in the development of the index of adl. Gerontologist 10:20–30
11. Lafferty J, McCallum A, Pereira F (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proc. 18th international conf. on machine learning, Morgan Kaufmann, San Francisco, CA, pp 282–289
12. Landwehr N, Gutmann B, Thon I, Philipose M, DeRaedt L (2007) Relational transformation-based tagging for human activity recognition. In: Malerba D, Appice A, Ceci M (eds) In: Proceedings of the 6th international workshop on multi-relational data mining (MRDM07), Warsaw, Poland, pp 81–92
13. Lawton MP, Brody EM (1969) Assessment of older people: self-maintaining and instrumental activities of daily living. Gerontologist 9(3):179–186
14. Liao L, Fox D, Kautz H (2007) Extracting places and activities from GPS traces using hierarchical conditional random fields. Int J Robot Res 26(1):119–134
15. Liu DC, Nocedal J (1989) On the limited memory bfgs method for large scale optimization. Math Program 45(3):503–528
16. Logan B, Healey J, Philipose M, Tapia EM, Intille SS (2007) A long-term evaluation of sensing modalities for activity recognition. In: Ubicomp, pp 483–500
17. Morency LP, Quattoni A, Darrell T (2007) Latent-dynamic discriminative models for continuous gesture recognition. In: Computer vision and pattern recognition, 2007. CVPR '07. IEEE Conference on, pp 1–8
18. Nambu M, Nakajima K, Kawarada A, Tamura T (2000) The automatic health monitoring system for home health care. In: Information technology applications in biomedicine, 2000. In: Proceedings 2000 IEEE EMBS international conference on, pp 79–82
19. Ohta S, Nakamoto H, Shinagawa Y, Tanikawa T (2002) A health monitoring system for elderly people living alone. J Telemed Telecare 8:151–156
20. Oliver N, Garg A, Horvitz E (2004) Layered representations for learning and inferring office activity from multiple sensory channels. Comput Vis Image Underst 96(2):163–180
21. Patterson DJ, Fox D, Kautz HA, Philipose M (2005) Fine-grained activity recognition by aggregating abstract object usage. In: ISWC, IEEE Computer Society, pp 44–51
22. Rabiner LR (1989) A tutorial on hidden markov models and selected applications in speech recognition. Proc IEEE 77(2):257–286
23. Raina R, Shen Y, Ng AY, McCallum A (2004) Classification with hybrid generative/discriminative models. In: Thrun S, Saul L, Schölkopf B (eds) Advances in neural information processing systems 16. MIT Press, Cambridge, MA
24. Si H, Kim SJ, Kawanishi N, Morikawa H (2007) A context-aware reminding system for daily activities of dementia patients. In: ICDCSW '07: Proceedings of the 27th international conference on distributed computing systems workshops, Washington, DC, USA, IEEE Computer Society, pp 50
25. Sutton C, McCallum A (2006) Introduction to statistical relational learning, chapter 1: an introduction to conditional random fields for relational learning. MIT Press (available online)
26. Tamura T, Togawa T, Ogawa M, Yoda M (1998) Fully automated health monitoring system in the home. Med Eng Phys 20:573–579
27. Tapia EM, Intille SS, Larson K (2004) Activity recognition in the home using simple and ubiquitous sensors. In: Pervasive computing, second international conference, PERVASIVE 2004. Vienna, Austria, pp 158–175
28. Truyen TT, Bui HH, Venkatesh S (2005) Human activity learning and segmentation using partially hidden discriminative models. In: HAREM
29. Truyen TT, Phung DQ, Bui HH, Venkatesh S (2008) Hierarchical semi-Markov conditional random fields for recursive sequential data. In: Neural information processing systems (NIPS)
30. Vail DL, Veloso MM, Lafferty JD (2007) Conditional random fields for activity recognition. In: International conference on autonomous agents and multi-agent systems (AAMAS)
31. van Kasteren T, Englebienne G, Kröse B (2008) Recognizing activities in multiple contexts using transfer learning. In: Proceedings of the AAAI fall symposium on AI in eldercare: new solutions to old problems. AAAI Press, ISBN=978-1-57735-394-2
32. van Kasteren T, Noulas A, Englebienne G, Kröse B (2008) Accurate activity recognition in a home setting. In: UbiComp '08: Proceedings of the 10th international conference on ubiquitous computing, ACM, New York, NY, USA, pp 1–9
33. Verbeek J, Triggs B (2008) Scene segmentation with crfs learned from partially labeled images. Adv Neural Inf Process Syst 20:1553–1560
34. Wilson D, Atkeson C (2005) Simultaneous tracking and activity recognition (star) using many anonymous binary sensors. In: Pervasive computing, third international conference, PERVASIVE 2005. Munich , Germany, pp 62–79
35. Wilson DH (2005) Assistive intelligent environments for automatic health monitoring. PhD thesis, Carnegie Mellon University
36. Wilson DH, Philipose M (2005) Maximum a posteriori path estimation with input trace perturbation: algorithms and application to credible rating of human routines. In: IJCAI, pp 895–901
37. Wu J, Osuntogun A, Choudhury T, Philipose M, Rehg JM (2007) A scalable approach to activity recognition based on object use. In: ICCV