ACTIVITY ANALYSIS WITH HIDDEN MARKOV MODEL FOR AMBIENT ASSISTED LIVING

GuoQing Yin
Vienna University of Technology
Gußhausstraße 27-29/384, 1040 Vienna, Austria
yin@ict.tuwien.ac.at

Dietmar Bruckner
Vienna University of Technology
Gußhausstraße 27-29/384, 1040 Vienna, Austria
bruckner@ict.tuwien.ac.at

ABSTRACT

In an Ambient Assisted Living (AAL) project the activities of the user will be analyzed. The raw data is from a motion detector. Through data processing the huge amount of dynamic raw data was translated to state data. With hidden Markov model, forward algorithm to analyze these state data the daily activity model of the user was built. Thirdly by comparing the model with observed activity sequences, and finding out the similarities between them, defined the best adapt routine in the model. Furthermore an activity routine net was built and used to compare with the hidden Markov model.

Keyword: Activity Analysis, Ambient Assisted Living**,** Hidden Markov Model, Forward Algorithm

1. INTRODUCTION

There are many papers about the hidden Markov model: The author in[1] explained the basic definition of a hidden Markov model with applications. Paper [2] builds a daily activity model of the elderly with a hidden Markov model. The authors in paper [3] used Bayesian posterior probability to choose the states to merge and to stop merge. The authors in paper [4] introduce the theory of the hidden Markov model and illustrate the applications in speech recognition. In paper [5] a segmental k-means algorithm is used as objective function to estimate the joint likelihood between the observation data and

the Markov state sequences. A new type of hidden Markov models is presented by the author in paper [6]. In the new models the current state is related both to the preceding state and the preceding observation. The authors in paper[7] search the dynamic transition probability parameters $A(\tau)$ and propose a moveable hidden Markov model. The hidden Markov model is used to learn the behavior of the user for building an automated system which was introduced in paper [8] and [9].

1.1 AAL, Project ATTEND

Ambient Assisted Living (AAL) aims to improve the life quality of the elderly with the help from modern technology and prolong the period of independent living in their own home. But the elderly have their own problems because of aging, such as action obstacles, memory disorder…, how can a modern technology system adapt to the elderly? [10] A system will be developed at the project ATTEND (AdapTive scenario recogniTion for Emergency and Need Detection) that extends the period of independent living of the elderly. An intelligent, adaptive network of sensors will be installed in the living environment of the user, in order to thoroughly observe the activities and behavior of the user. Then the user's activities and behavior model will be learned by the system over a time period. Based on the learned model if unusual behaviors or activities of the user are observed the system will send an alarm signal to the caregiver. Neither camera nor microphone will be used in the whole system and especially in project ATTEND because of privacy issues of the elderly. Furthermore there will be no sensor to wear on the body or to be activated by the user. The only sensors used are a motion detector or door contactor.

1.2 Contribution

In this paper at first the huge amount of raw sensor data was translated to state data. Then using a hidden Markov model, a forward algorithm builds the activities model of the user. Thirdly observed activity sequences are matched with the best adapt routine in model to find the similarities between them. Finally an activity routine net was introduced in order to compare with the hidden Markov model.

2. TRANSLATE RAW DYNAMIC SENSOR DATA TO STATE

The raw data come from a motion detector installed in the living room of the user. If there is any "movement" of the user, the motion detector will send value "1" to the controller. Otherwise the sensor value is "0". Because

the activities of the user are random so the raw data is randomly distributed over time intervals during the day and in a huge quantity (hundreds sensor data every day). In order to reduce the data amount and to get more typical activity states the raw data will be translated to state data in predefined time intervals [10].

At first the segments of daily time are divided into smaller time intervals, for example of 60 minutes each so there will be 24 time intervals in a day (for even more accuracy the time interval can be chosen smaller, such as 30 minutes or 15 minutes. In this paper 60 minutes were chosen as time interval for a better performance). Then one needs to gather the activities of the sensor in each time interval. If the total value of activities is bigger than a predefined threshold value, the activities that occurred in this time interval will be given the value "1", other less active intervals receive the value "0". Here the predefined threshold value $(T_{th}, 0 < T_{th} < 1)$ and the time interval $(T_{interval})$ are important.

1) The gathered sensor value according $T_{interval}$

$$T = \{t_{1(1)}, t_{2(0)}, t_{3(1)}, t_{4(0)}, t_{5(1)}, t_{6(0)} \dots t_{n(v)}\} \tag{1}$$

Here t_n is the time point that the motion detector send value to controller $(n \geq 1)$, v is the sensor value itself, it has value "0" or "1".

2) The activities duration between sensor values

$$\Delta T = \left(t_{n(0)} - t_{n-1\,(1)}\right) \tag{2}$$

3) The sum of the activities duration in $T_{interval}$

$$T_{sum} = \Sigma(\Delta T) \tag{3}$$

4) Deciding if the time interval gets value "1" or "0".

$$If\ T_{sum} \geq T_{th} \times T_{interval}, S_{ix} = 1 \tag{4}$$

$$If\ T_{sum} < T_{th} \times T_{interval}, S_{ix} = 0 \tag{5}$$

Here S_{ix} is the state value that the interval should take. Here "ix" is the interval count (index).

3. HIDDEN MARKOV MODEL, FORWARD ALGORITHM

A hidden Markov model can be characterized by the following parameters.

1) The number of states N.

2) The number of output distinct observation symbols for each state M.

3) The state transition probability distribution matrix $A = \{p_{ij}\}$.

$$p_{ij} = p\{Q_{t+1} = j \mid Q_t = i\}, 0 \leq p_{ij} \leq 1, \sum_{j=1}^{N} p_{ij} = 1, 1 \leq i, j \leq N. \tag{6}$$

Here Q_t is the current state at time t.

4) The state emission probability distribution matrix $B = \{b_{ik}\}$.

$$b_{ik} = p\left(O_t = k \mid Q_t = i\right), 1 \leq i \leq N, 1 \leq k \leq M \tag{7}$$

Here O_t is the output symbol at time t.

5) The initial state distribution $\pi = \{\pi_i\}$.

$$\pi_i = p\{Q_0 = i\} \tag{8}$$

According the parameter $\lambda(\pi, A, B)$ with forward algorithm we can find out the probability of an observed sequence $Q^{(t)} = \{q_1, q_2, ..., q_t\}$. Here each of the q is an observable state within the time label.

6) Get the first transition probability a_1 for $t = 1$.

$$a_1(j) = \pi(j) \times b_{jt} \tag{9}$$

Here j is the observation count of each observation set and $\sum \pi(j) = 1$.

7) For $t \geq 2$ get the transition probability $a_t(j)$

$$a_t(j) = \sum_{i=1}^{n}\left(a_{t-1}(i) \times p_{ij}\right) \times b_{jt} \tag{10}$$

8) For $t \leq T$ repeat (10). Here T is the length of the sequence.

4. ACTIVITY MODEL AND MERGING OF STATES

The raw data for 8 days was gathered from a motion detector and with the formulae (1) to (6) the data was translated to 24 states for each day. The values of the states are shown in figure 1. Each routine in figure 1 indicated the activity of the user on one day. The x-axis is the state count and the y-axis is the day count. The blue circle symbol in the figure means the state with value 0 and the green square symbol means the state with value 1.

On the left side of figure 1 are some consecutive states having the same value. These states can be merged together. The merging approach was introduced in figure 2. The right side of figure 1 indicates the activity model after merging of the states. The star symbol indicates 0 and 1 were merged together in the state value. After state merging each merged state lost the time label but has become more tolerant to adapt observation states with different values.
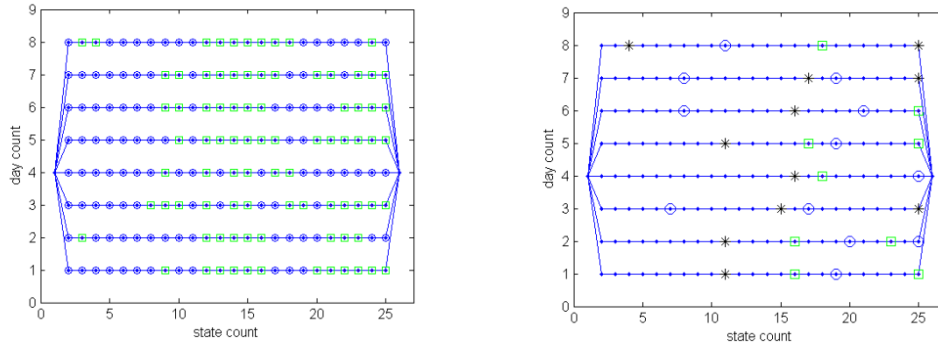

Figure 1. Activity model of the user

Figure 2 describes the merge method with consecutive states. The states with value 0 are merged together and the state self transition probability distribution is $p_{ii} = 3/4$ and the state transition probability distribution is $p_{ij} = 1/4$. The state emission probability distributions are $b_{i0} = 1$ and $b_{i1} = 0$. The states with value 1 have merged together and the self state transition probability distribution is $p_{ii} = 2/3$ and the state transition probability distribution is $p_{ij} = 1/3$. The state emission probability distribution are $b_{i0} = 0$ and $b_{i1} = 1$. If the consecutive states have alternating values as in the example shown on the right side of figure 2, the self state transition probability distribution is $p_{ii} = 6/7$ and the state transition probability distribution is $p_{ij} = 1/7$. The state emission probability distributions are $b_{i0} = 4/7$ and $b_{i1} = 3/7$. Generally if Nm is the merged consecutive states amount, the self transition is $p_{ii} = 1 - 1/N_m$, and the transitions is $p_{ij} = 1/N_m$.
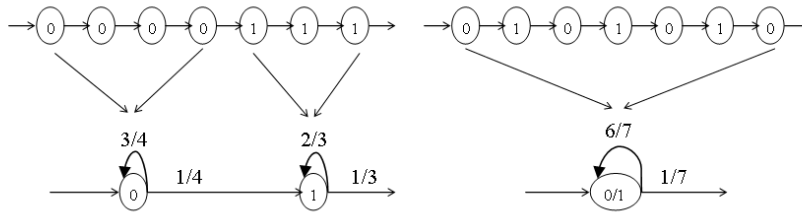
Figure 2. Merging the consecutive states

5. RESULT FROM HIDDEN MARKOV MODEL

At the right of figure 1 each merged state has parameters (A, B). Here each routine is treated as an individual routine so the initial value π is 1 for each routine, so $a_1(j) = b_{j1}$. Based on the parameters of the hidden Markov model the probability of an observation activity sequence can be found with a forward algorithm. For example the routine from the 6th day in the activity model (figure 1) was chosen as observation sequence. With forward algorithm the probability of the chosen routine matched to the hidden Markov models will be found. Figure 3 shows the result. There is a total of 8 value sequences since the chosen routine is matched to each routine of the model. The best value at state 24 is -11.98 (from the 6th routine in figure 1) and the worst value at state 24 is -24.54. There are 4 merged states (7, 15, 20, and 24) at the 6th routine on the right side of figure 1. At state 8, 16, and 21 the probability value became smaller with a discrete jump to the next state. This is because the states' value from the observation routine changed and did not adapt to the merged state and jumped to the next state with smaller states transition probability. Furthermore there are 2 routines that did not reach the last state in figure 3. This is because the last state of the 6th routine is 1 but the last states of the 2 routines (routine 2 and 4) have values 0. The emission probability is 0 at the last state and the $a_{24} = 0$. These values at state 24 will not be shown in figure 3.
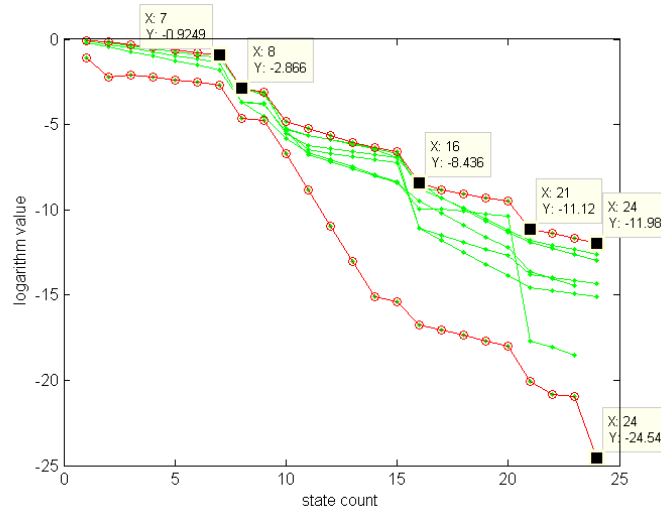
Figure 3. The probability of a chosen day matched to hidden Markov models

6. ACTIVITY MODEL STRUCTURE RELATED PROBABILITY

In order to compare the result obtained with the hidden Markov model another approach will be introduced in the paper. The approach is based on the state value of each routine to merge and split the routines and form a state routine net. In figure 4 if routines 1 and 2 have different state values the routine will be split to two routines. If routines 1 and 2 have same state values then these states in different routines will be merged.
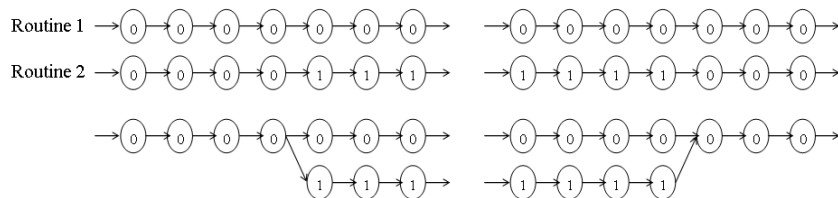

Figure 4. State routines split and merge with different state value

According to the approach introduced above the 8 state routines showed in figure 1 will be formed to the routine net. The x-axis of figure 5 is the state, each circle means one state, and it has state value "0" or "1" (the

first state 1 and the last state 26 are extra states, without any value). The y-axis is the routines number of different days (different order as in figure 1). Figure 5 shows some splitting and merging points.
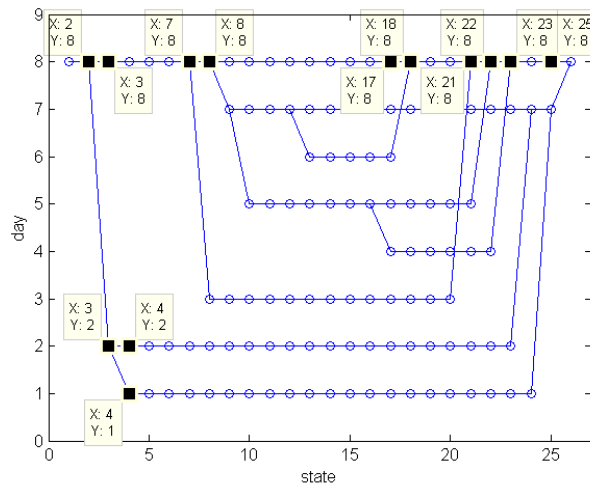

Figure 5. The state routine net in 8 days

Each forward split in figure 5 leads to change in a transition value. This happened to all the routines till they split to a single routine. The logarithm value for each single routine is the same: $\log\left(\frac{1}{\text{sum of all routes}}\right)$, here $\log\left(\frac{1}{8}\right) = -2.079$. Furthermore in spite of all the different routines ultimately the state logarithm value convergences to the same value. It is because from both the forward and backward directions all the routines split to a single routine. So the logarithm value for each routine at the last state is: $\log\left(\frac{1}{\text{sum of all routes}^2}\right)$, here $\log\left(\frac{1}{8^2}\right) = -4.159$. Using the character we can examine if the searched parameters of activity routine net are right or not. If at last state all the routine logarithm values converge to the same value, that means these paramesters are correctly searched, if not, there must be something wrong with the parameters.

In order to find out the similarity between an observation activity sequence to the activity routine net, the day routine 7 in the figure 1 was chosen (as an example). It was used to match with all of the 8 days and the result appear in figure 6. The logarithm value indicated with a green line and circle in figure 6 is the matching result of routine 7. It has a maxmum value -4.159. If the matching states of different routines have different values, so a "not adapt transition value (such as 0.1)" will be sent to the transition

matrix A at the state. Figure 6 indicated that the logarithm value between the chosen day to the other matching day reduced rapidly from -8.764 to -24.88.
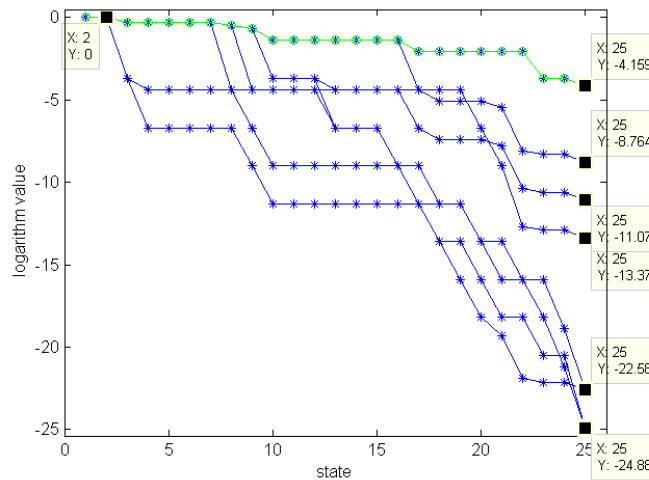

Figure 6. Matching the logarithm with chosen observation routine in 8 days

7. CONCLUSION AND FURTHER WORK

In this paper a hidden Markov model and an activity routine net were introduced to analyze the daily activity of the user. The activity routine net is sensitive but lacks tolerance. With the increased number of routines a crossover problem emerged. The activity model form hidden Markov model is more concise, structured, and more tolerant. In the next work similar daily routines will be merged. That should result in an easier but more inclusive and tolerant model.

8. REFERENCES

[1] J. Bilmes, *What HMMs can do*, UWEE Technical Report, Number UWEETR-2002-0003, January 2002.

[2] GuoQing Yin, and Dietmar Bruckner, Daily activity model for ambient assisted living. *Presented at Doctoral Conference on Computer, Electrical and Industrial System, DoCEIS 2011,* Costa de Caparica, Portugal, February 21-23, 2011.

[3] A. Stolcke, and S. Omohundro, Hidden Markova model induction by bayesian model merging. In C. L. Giles, S. J. Hanson, and J. D. Cowan

(Eds.), *Advances in Neural Information Processing System 5* (p11-18). San Mateo, CA: Morgan Kaufman, 1993.

[4] L.R. Rabiner, and B.H. Juang, An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(1), p4-16, 1986.

[5] Biing-Hwang Juang, and L.R. Rabiner, The segmental K-Means algorithm for estimating parameters of hidden Markova models. *IEEE Transactions on Acoustics, Speech, and Singal Processing,* 38(9), p1639-1641, 1990.

[6] Yujian Li, Hidden Markov models with states depending on observations. *Pattern Recognition Letters*, 26, p977-984, 2005.

[7] Bongkee Sin, and Jin H. Kim, Nonstationary hidden Markov model. *Signal Processing*, 46, p31-46, 1995.

[8] Dietmar Bruckner, Brian Sallans, and Gerhard Russ, Probability construction of semantic symbols in building automation systems. *Presented at 2006 IEEE International Conference of Industrial Informatics INDIN*, Singapore, August 16-18, 2006.

[9] Dietmar Bruckner, and Brian Sallans, Behavior learning via state chains from motion detector sensors. *Presented at Bio-Inspired Models of Network, Information and Computing Systems, 2007. Bionetics 2007. 2nd*, Vienna, December 10-13, 2007.

[10] G. Yin, and D. Bruckner, Daily activity learning from motion detector data for ambient assisted living. *Presented at the 3rd International Conference on Human System Interaction*, Poland, May 13-15, 2010.