User Behavior Recognition based on Clustering for the Smart Home

WOOYONG CHUNG, JAEHUN LEE, SUKHYUN YUN, SOOHAN KIM* AND EUNTAI KIM
Samsung Electronics, Network Sys. Div. Internet Infra Team*
School of Electrical and Electronics Engineering
Yonsei University
134 Shinchon-dong, Seodaemun-gu, Seoul
KOREA
http://cilab.yonsei.ac.kr

Abstract: - In the vision of ubiquitous computing environment, smart objects would communicate each other and provide many kinds of information about user and their surroundings in the home. This information enables smart objects to recognize user behaviors and to provide active and convenient services to the customers. However in most cases, context-aware services are available only with expert systems. The proposed method presents general context-aware system in the smart home based on a Bayesian network (BN), which enables us handling user context probabilistically. We used fuzzy *c*-means algorithm for clustering user positions and active time to simplify BN's structure and to reduce BN's conditional probability table size. We used a virtual smart home with web based simulator to collect samples of user activities and their environments. This suggested algorithm was simulated with 10 fold cross-validation and evaluated to verify the performance of context-awareness.

Key-Words: - Behavior recognition, Context-aware, Smart home, Bayesian network, Fuzzy *c*-mean clustering

1 Introduction

Computation and access to the internet are now available in everywhere with the wireless technology, which fully interconnects all the fixed or portable computing devices. The purpose of this work is to provide more natural interaction between human and smart home to recognize user behavior, which means context-awareness. For context-aware applications, Want[1] proposed Active Badge location system, which enables call forwarding according to user location. Tapia[2] used simple and ubiquitous sensors for activity recognition.

There are also many machine learning approaches in the sensor-based context-aware system. For example Ranganathan[3] used probabilistic logic, fuzzy logic and Bayesian networks to reason about uncertainties in pervasive computing environments. Laerhoeven and Cakmakci report on using Kohonen maps, K Nearest Neighbor classification (KNN) and Markov chains to detect user activity. We developed a user behavior recognition model based on a Bayesian Network (BN). However these kinds of methods require some modification by user or experts. So we propose more natural interface between user and home based on simplified BN and fuzzy clustering algorithm.

The BNs are one of the best known methods to reason under uncertainty. In addition, the graphical nature of BNs gives us a much better intuitive grasp of the relationships among the features [4]. The BN denotes joint probabilistic distribution among variables of interest based on their probabilistic relationships. The structure of BN is a directed acyclic graph (DAG). Each node represents a random variable which has a finite discrete data set of domain and is connected with its parent's nodes. Each arc represents the conditional dependency between the parent and the descent.

The fuzzy *c*-means(FCM) clustering algorithm is an clustering technique which is separated from hard c-means that employs hard partitioning. The FCM employs fuzzy partitioning such that a data point can belong to all groups with different membership grades between 0 and 1 [5].

The rest of the paper is organized as follows: We briefly introduce the Bayesian network and fuzzy *c*-means algorithm in Section 2. In Section 3, we describe the modeling of user behavior recognition we address. In Section 4, we describe the simulation we have performed to explore the performance of the method and examine the results. We conclude and discuss our future work in Section 5.

2 Preliminaries

2.1 Bayesian Networks

A Bayesian network is a graphical model that combines graph theory, probability theory and statistics[4]. In the graph theory a directed graph composed of ***(V,E)***, where the elements of ***V*** are called nodes and the set of edges, ***E*** is ordered pairs, which represent conditional dependencies between nodes. There would be a cycle in the directed graph. A cycle is a set of edges, where the start and end nodes are the same. On the other side directed acyclic graph (DAG) is a directed graph with no cycles and the BNs are also DAG.

Suppose that ***G=(V,E)*** is a DAG. If all the nodes are conditionally independent of their non-descendant nodes given their parent nodes, then the DAG satisfies the Markov condition. This Markov condition allows us to represent the probability of the whole network with the joint distribution of the nodes. For example the probability of the whole network shown in Figure 1 is given by

$$\begin{aligned} &P(A,B,C,D,E,F,G) \\ &\quad = P(A)\ P(B)\ P(C)\ P(D \mid A,B)\ P(E \mid B,C) \\ &\qquad P(F \mid D)\ P(G \mid E) \end{aligned} \tag{1}$$

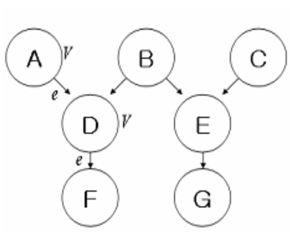


Fig. 1. Example of a Bayesian network

This DAG enables us to present Bayesian rules as simplified graph model, Bayesian network. In the BN each node represents a random variable, and the DAG represents the joint probability distribution of a set of random variables, $\boldsymbol{X}=\{X_1, X_2, \ldots, X_n\}$. We assume that a variable X_i is independent of the other variables in the network, given its parents π_i, and the attribute of each node is finite and discrete value. Using this assumption, the joint probability distribution can be factored as

$$P(X) = \prod_i P(X_i \mid \pi_i) \tag{2}$$

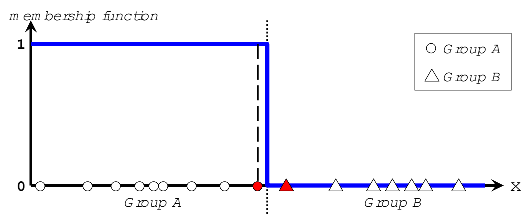
The learning BN consists of structure learning and parameter learning. In the structure learning, we find optimal connections of between the nodes. The most widely used algorithm for structure learning method is K2 learning, which was developed Coopeer and Herskovits [6]. In the parameter learning, we calculate the conditional probabilistic distribution for the given BN structure and samples of data. The most popular parameter learning method is EM algorithm [7].

2.2 Fuzzy *c*-means algorithm

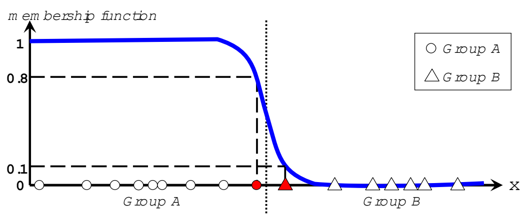
Fuzzy *c*-means (FCM) is a method of fuzzy clustering algorithm which allows one piece of data to belong to more than one cluster. This method was developed by Dunn in 1973 and improved by Bezdek[5] in 1981. Figure 2 shows the difference of hard and fuzzy *c*-means algorithm with simplified *1*-dimentional examples. Since it can handle ambiguity of clusters, this algorithm is frequently used in pattern recognition. The fundamental principles is minimization of the following objective function

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m \left\| x_i - c_j \right\|^2 \quad , \quad 1 \le m \tag{3}$$

where m is fuzzy exponent, which is any real number greater than 1, u_{ij} is the degree of membership of x_i in the cluster j, x_i is the ith of d-dimensional measured data, c_j is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center.



(a) Hard *c*-means clustering



(b) Fuzzy *c*-means clustering

Fig. 2. Examples of hard and fuzzy c-means algorithm

Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centers c_j by

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \tag{4}$$

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m \times x_i}{\sum_{i=1}^{N} u_{ij}^m} \tag{5}$$

This iteration will stop when

$$\max_{ij} \left\{ \left| u_{ij}^{(k+1)} - u_{ij}^{(k)} \right| \right\} < \varepsilon \tag{6}$$

where ε is a termination criterion between 0 and 1, whereas k are the iteration steps.

3 Constructing Context-aware System

Our approach for user behavior recognition modeling is consists of two phases: quantization by fuzzy clustering and constructing Bayesian network. The quantization process is converting the continuous data to the discrete data, which is proper for the BN. The constructing BN is composed of structure learning and parameter learning.

3.1 Quantization by Fuzzy Clustering

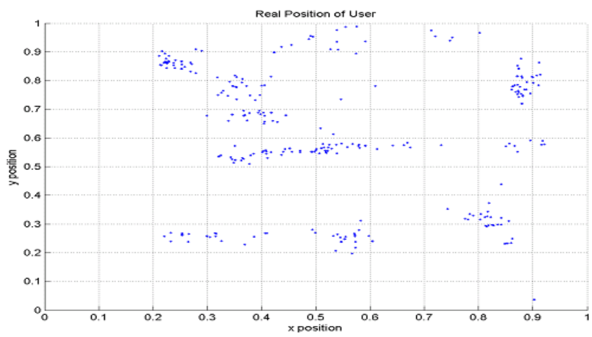
In the smart home, most of sensor data are discrete values like on/off. However there are some continuous values like time and user position. To construct Bayesian network easily and efficiently, we need to convert a continuous value to a discrete value through quantization. If we let X_q be a quantized value then a general approach to quantization can be

$$X_q = Q_r \times \left\lfloor \frac{X}{Q_r} \right\rfloor \tag{7}$$

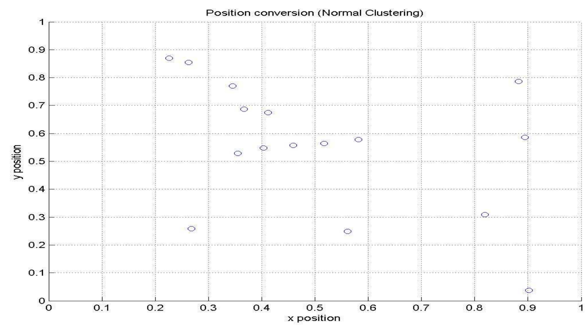
Where Q_r is a quantization size and $\lfloor * \rfloor$ is gauss value.

However this general quantization approach can lead undesirable result for activity recognition in the smart home, since there are unused or useless values, e.g. 4am, at which there is no activity except sleeping. This useless value makes the conditional probability table (CPT) size too large, so the performance of this system would be degraded. To solve this problem, we quantized continuous values with FCM clustering to simplify the Bayesian network and CPT size and to remove the undesirable values.

The fuzzy quantization process consists of two steps, clustering continuous values and assigning to discrete values. In the first step, we clustered each continuous value into c categories with FCM clustering algorithm. In the second step, we assign continuous values to discrete values with the cluster number, which are same as ordered cluster centers. This means that each discrete value is one of the centers. So if the number of cluster is c_i in which a continuous value belongs then the discrete value is c_i. In this paper we used 9 quantization of time and 16 quantization of user location (Fig. 3).



(a) Real position of user



(b) Clustered centers of user positions

Fig. 3. User position conversion with FCM clustering

3.2 Constructing Bayesian Network

We have two assumptions in our model. The first one is each activity could be occurred in the restricted area. This means that some activities can be occurred near the specific sensors. The second assumption is we have no information about the dependency between sensors or activities. Only dependency between sensors and activities exists. This assumption simplifies the BN structure. Although the simplified structure the performance is still good.

As we mentioned before, the configuration of the Bayesian networks is consists of learning the network structure and parameters. In the structure learning, we used mutual information to select parent nodes. In the parameter learning, we used MAP based EM algorithm.

Although there are many algorithms for structure learning of BN such as K2, MDL causal discovery, and CaMML, we used the simple entropy method. According to our assumption, our BN structure model is restricted as a naïve Bayesian network, which assumes that every node is independent except its parent nodes, it is easy to learn the structure. We used mutual information for selecting sensor nodes concerning an act to configure the BN structure. The mutual information is defined as [8]

$$\begin{aligned} & I(Sensor; Act) \\ & \quad = H(Sensor) - H(Sensor \mid Act) \\ & \quad = H(Sensor) - H(Sensor, Act) - H(Act) \\ & \quad = \sum_{s \in Sensor} \sum_{a \in Act} p(s,a) \log_2 \left(\frac{p(s,a)}{p(s)p(a)} \right) \end{aligned} \tag{8}$$

where $H(X)$ is entropy and $H(X|Y)$ is conditional entropy.

We decided the number of connections according to computer capacity, the amount of sample size, and robustness. If we increase the connection number, then the CPT would be increased. This increased CPT size requires more computer capacity to process and more samples to learn the BN. Sometimes it means that the BN is overfitted losing the robustness. Therefore we need find the optimal number of connections. Figure 3 is the example of proposed BN structure, which composed of L area, M activities, and N sensor nodes.

The parameter learning is updating the CPTs. As mentioned before, we user simplified BN structure to reduce the size of CPTs. Since there are missed data while collecting samples for learning, we used MAP based EM algorithm for parameter learning. The EM algorithm is one of parameter learning method when there are some missing data during collecting the sensor and user activity data.

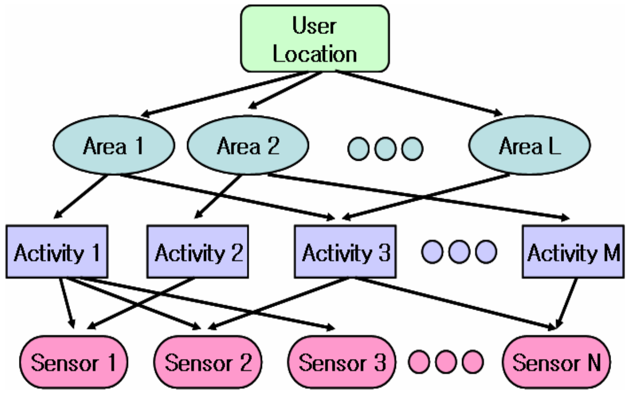


Fig. 3. The BN structure of proposed model

4 Simulation Result

In this section, the proposed method is applied to recognize system in the virtual smart home. We collected similar user activities and sensor data from the simulator for 10 days (Fig. 4). Forty two random variables are used to implement the model. Twenty one of the random variables are the state of the sensor information such as time, location, light or TV. Since the power problems of mobile device, most smart devices are power connected. The rest of variables represent user activities like sleeping, watching TV or washing. We performed simulation with 10-fold cross-validation to explore the performance of the proposed method

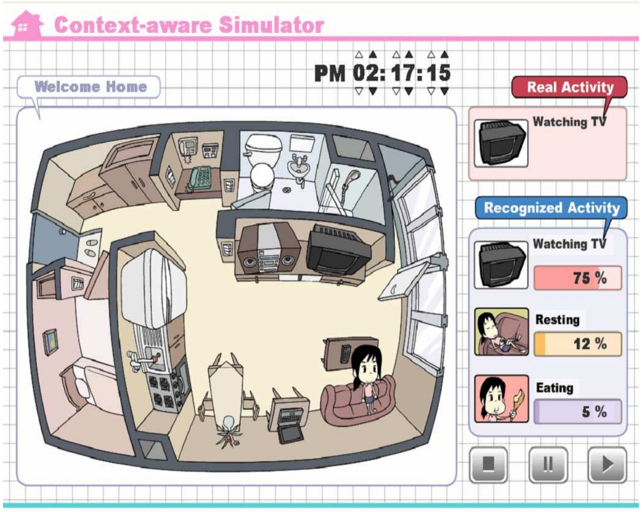


Fig. 4. The simulator of smart home

We compared the performance of proposed method with uniformly quantized case and clustered quantization but different structure case. As shown in Figure 5, the recognition rate of proposed method in percent is higher than other methods[9].

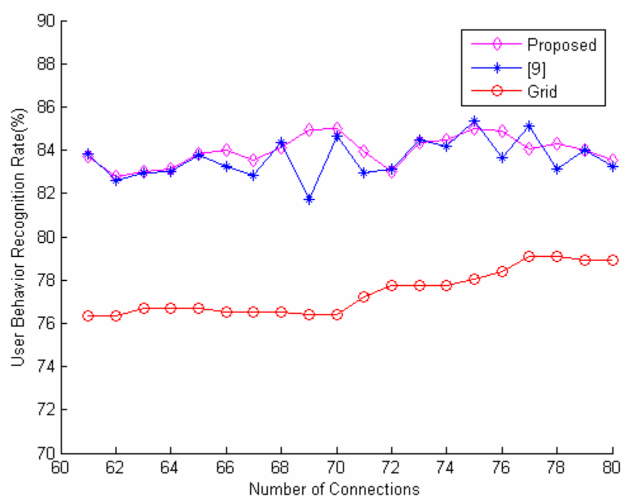


Fig. 5. User behavior recognition rate

5 Conclusion

We proposed a user behavior recognition model to provide more natural interaction between residents and home. In the constructing behavior recognition model, first it performs quantization by FCM clustering algorithm, and then learn the BN structure based on mutual entropy and the BN parameters by EM algorithm. The major advantage of this model is that after registering sensor and activity type, it construct context-aware model without other modification. This means that we can provide more comfortable interface between user and home. Considering the dependency between activities or sensors would increase the performance. We plan to address this issue in future work

Acknowledgement

This work was supported by the Ministry of Commerce, industry and Energy of Korea (HISP).

References:

[1] R. Want, A. Hopper, V. Falcãao, and J. Gibbons, “The Active Badge location system,” *ACM Transactions on Information Systems*, Jan. 1992, pp. 91-102

[2] E.M. Tapia, S.S. Intille, and K. Larson, “Activity recognition in the home setting using simple and ubiquitous sensors,” *in Proceedings of PERVASIVE 2004*, Vol. LNCS 3001, 2004, pp.158-175

[3] A. Ranganathan, J. Al-Muhtadi, R.H. Campbell, “Reasoning about uncertain con- texts in pervasive computing environments,” *IEEE Trans. on Pervasive Computing,* 2004, Vol. 3, No. 2, pp. 62-70

[4] R.E. Neapolitan, *Learning Bayesian Networks*, Prentice Hall, 2004

[5] J.C. Bezdek, *Pattern Recognition With Fuzzy Objective Function Algorithms*, Plenum Press, 1981

[6] G. Cooper, E. Hersovits, “A Bayesian method for the introduction of probabilistic networks from data, *Machine Learning*,” Vol.9, No.4. Oct. 1992, pp. 309-347

[7] S.Z. Zhang, Z.N. Zhang, N.H. Yang, J.Y. Zhang, and X.K. Wang, “An improved EM algorithm for Bayesian networks parameter learning,” *Machine Learning and Cybernetics*, Vol. 3, Aug. 2004, pp. 1503 - 1508

[8] A. Papoulis and S.U. Pillai, *Probability, Random Variables and Stochastic Processes*, Mc Graw Hill, 2002

[9] W.Y. Chung, J.H. Lee, S.H. Yon, Y.W. Cho, E.T. Kim, “Intelligent Modeling of User Behavior based on FCM Quantization for Smart home,” *Journal of Control, Automation, and Systems Engineering*, 2007, pp. 542-546