International Journal of Hybrid Intelligent Systems 2 (2005) 221–234
IOS Press

A hybrid personal assistant based on Bayesian networks and a rule-based system inside a smartphone

Sang-Jun Han and Sung-Bae Cho*
Department of Computer Science, Yonsei University, 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea

Abstract. As technology improves, mobile phones are becoming essential aspects of human communication. As more and more people begin to use mobile phones, various services based on mobile phone networks and high-end devices are developing. In addition, with the growth of ubiquitous computing, there have been many ongoing studies regarding novel and useful services inside smartphones. In order to develop personalized and intelligent services in these smartphones, we propose an intelligent service-providing method based on Bayesian networks and rule-based service selection. This method infers the user's state with the use of a Bayesian network and provides appropriate services based on the inferred knowledge. We also show a prototype of a personal assistant agent based on the proposed method. This prototype is applied to a group of realistic situations, in order to confirm the usefulness of the proposed method.

Keywords: Intelligent agents, Bayesian networks, personal assistants, smartphones, ubiquitous computing

1. Introduction

Recently, mobile phones have started to become essential tools for human communication. As more and more people begin to use mobile phones, various services based on mobile phone networks and high-end devices are being developed. Smartphones, which integrate the functions of personal digital assistants (PDAs) and mobile phones, have already earned a world-wide reputation as all-in-one devices: many technologies such as wireless voice/data communication, digital cameras, and multi-media players are combined into just one smartphone device.

Although smartphones are currently only being considered as high-end mobile phones, with innovative applications, they have the potential to be used for a wide range of novel services. Especially, with the rise of ubiquitous computing, the demand for personalized intelligent services on mobile devices like smartphones is increasing. However, current mobile devices have the constraints of limited processing power and awkward interfaces. AI (Artificial Intelligence) techniques, which can be applied directly to smartphones, are required to cope with these constraints and make intelligent services feasible in reality.

There are three major issues when implementing intelligent services in constrained environments. The first issue is to gather information that provides meaningful ways to measure the user's state. A good

*Corresponding author. E-mail: sbcho@cs.yonsei.ac.kr.

way to do this is by using techniques that provide sufficient information while not asking the user to provide explicit information and thus not invading the user's privacy.

The second issue is to infer and predict the user's state from the collected data. Predicting the user's state from the data can be seen as a conventional classification task. Many AI methods have been successfully applied to this problem.

The third issue deals with service selection or composition. This refers to selecting one service from a pre-defined service library, or composing novel services appropriate to the inferred user's state dynamically.

In this paper, we focus on the personal information and communication logs that are available in smartphones. We actively employ commonsense knowledge to deal with incomplete, imprecise data. Bayesian networks are used as the inferring technique to represent the expert's knowledge when coping with uncertainty. This method is more flexible, requires few resources, and can be easily extended compared to simple rule-based techniques. In order to provide services that are appropriate to the user's state, we developed a rule base for service selection. To verify our service-providing method, we also implemented a prototype agent based on the proposed method in the smartphone environment.

The rest of this paper is organized as follows. In Section 2, we give a brief overview of the kind of information sources that are available in smartphones. The proposed service-providing method is presented in Section 3. Section 4 shows a working scenario of the proposed agent in realistic situations. The conclusions and future work are discussed in Section 5.

2. Information sources in smartphones

Some available services in current intelligent agents include providing personalized help, assistance or suggestions, automation of common tasks, information filtering or retrieval, and selection of entertainment [1]. To provide these intelligent services, it is necessary to define the user's state by observing his or her behavior. For this purpose, information sources available in smartphones include personal information, communication logs, program usage patterns, and locations (as shown in Table 1). Though there is no existing research about exactly how these information sources can be utilized in smartphones, many research groups in other application domains have worked on personalized intelligent services using these information sources.

We can access various types of personal information easily because most PDAs and smartphones provide personal information management systems (PIMS) such as an address book, a schedule, a to-do-list and a memo. There have been some works on automatic management of a calendar using the user's schedule information. T. Mitchell et al. [2] have designed a calendar manager that works by learning the user's preferences with machine learning techniques. They used decision trees to learn these preferences in terms of various factors such as meeting duration, location, time, and date.

Smartphones enable various types of communication such as voice conversations, short message services (SMS), e-mail, and access to the world wide web. With information from these communications, we can collect various and highly personal communication logs. By using this type of information, we can infer social relationships and assist the e-mail program user. Boone [3] developed an e-mail management agent that performs a process of sorting, storing, and deleting e-mails automatically using the nearest neighbor and neural network classifiers.

Danah et al. [4] visualized social relationship networks using e-mail data. People can recognize hidden patterns and connections in their social relationships and make their behaviors fit more appropriately to their social situations. These studies illustrate that we can infer preferences or social relationships by

Table 1
Information sources available in smartphones

Type		Components
Personal information	Address book	Name, category, mobile phone number, home phone number, e-mail, address, home page
	Schedule table	Subject, location, start time, end time, attendees, importance, category, repeat, memo
Communication log		Name, phone number, start time, duration, contents of message
Location		Latitude, longitude, altitude, velocity

using communication log data. In [5], several intelligent applications are presented, including a personal assistant based on social network modeling and e-mail message analysis.

Program usage patterns have been commonly used to provide assistance to computer users in conventional desktop domains. These patterns can be observed from human-computer interaction and they can be used to automating tasks by learning usage patterns or inferring the user's goals and needs. The Lumiere project [6] is a representative piece of research. This project employs Bayesian networks and influence diagrams in order to infer users' goals from users' actions for supporting Microsoft Office users. Lie et al. [7] designed a user-adaptive Word assistant that predicts users' intentions and provides assistance autonomously by learning the action sequences of the particular user.

With the integration of location-aware devices like global positioning systems (GPS), location of the user becomes available in smartphones. This location information is very useful when trying to understand a user's state because the smartphone constantly moves around with the user and the user's state is significantly influenced by location. Flavia [8] utilized location information for user-adaptive storytelling in museums. She built a Bayesian network that made possible to estimate the user type from his or her location and duration at each location. In other studies, location information has been commonly used for recommendations and support for tourists.

D.J. Patterson et al. [9] proposed a GPS data method for current transportation modes such as taking a bus, driving a car, and walking. The method uses a dynamic Bayesian network model [10]. Domain knowledge about real world constraints (for example, the facts that cars are left in parking lots and buses pick up passengers only at bus stops) is incorporated into the Bayesian network model and the parameters of their networks are learned using an expectation-maximization (EM) algorithm. In experimental results against the real GPS data, the network model that was learned using the EM outperformed both the decision tree and the network model without domain knowledge.

A location-aware event planner designed by Z. Pousman et al. [11] integrates a friend finder application which displays locations on a given campus map. The user can organize social events in contextually-enhanced ways. The system also includes privacy management functionality which enables the user to manage visibility to others.

3. Intelligent agent for smartphones

The proposed agent consists of three components: an evidence collector, an inference engine, and an action selector. The evidence collector determines the state of the variables in the Bayesian network by mining meaningful events from personal information databases and call logs. The inference engine infers the user's state (such as affect or how busy the user is) using the evidence collected. The action selector chooses the agent's action appropriate to the user's current state. Figure 1 shows an overview of the proposed agent.

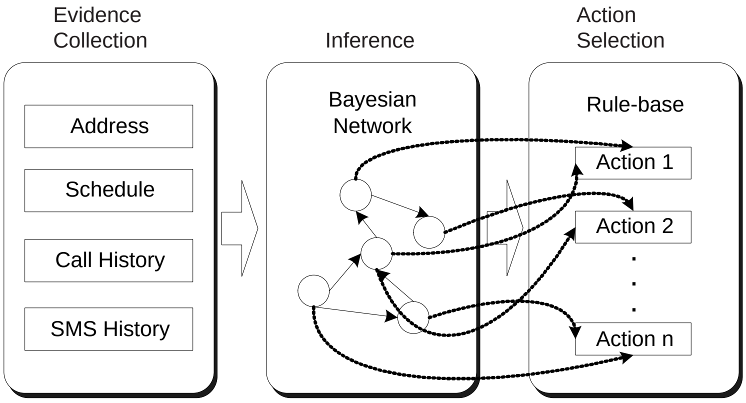


Fig. 1. The architecture of the proposed agent.

3.1. User modeling based on Bayesian networks

It is crucial to recognize the user's state in terms of emotion, goals, and needs in intelligent and pervasive services. The process of obtaining explicit input from users is very expensive. Therefore, it is necessary to collect meaningful evidence from implicit and low-level observations and infer the user's high-level state with the evidence collector. However, there is much uncertainty when attempting to recognize the user's state [6]. Bayesian probabilistic inference is one of the most famous models for inference and representation of the environment with insufficient information. The node of the Bayesian network represents a random variable, while the arc represents the dependency between the variables [12]. In order to infer the network, the structure must be designed and the probability distribution must be specified. Usually, the structure is designed by an expert while the probability distribution is calculated by an expert or by the collected data from the domain. By observing the evidence, the probability of each node can be found without excessive computation by using the Bayesian inference algorithm based on the conditional probability table and independence assumption.

An advantage of user modeling based on Bayesian networks is that an expert's pre-knowledge can be utilized easily. We can easily achieve desirable performance by utilizing domain knowledge in order to define the variables and their dependency. Moreover, in the case of smartphones, we cannot use a large dataset – the data-intensive modeling technique is not feasible because most learning algorithms are too computationally demanding. Modeling with Bayesian networks is superior to rule-based modeling techniques in terms of scalability. The user model can be extended easily by adding new variables and specifying the dependency with others. On the other hand, the rule-based approach requires much more effort because we have to update several rules related to the new coverage of the user model.

3.2. Inferring the user's state

Ways to infer the user's state include affect, how busy he/she is, and how close he/she is with someone. We employed Bayesian networks in an attempt to infer these contexts. The user's context can be estimated by specifying the dependency among certain events related to the context. The design methodology of Bayesian networks is as follows.

First, we identified the events that seemed to be related to the user's context and defined them as variables in the Bayesian network. Next, we specified the states for each variable and made concrete conditions for each state. (For instance, to infer how busy the user is, we may use "the number of missed calls" as the related variable and define "many" and "few" as possible states of this variable. The state of "the number of missed calls" is "many" if the number of unanswered calls in the last two hours is more than five, and "few" if the number is less than five.) The accuracy of user modeling hinges upon how we define the variables in the Bayesian network and the state of each variable [12].

After defining the variables and their states, we constructed the structure of the Bayesian network considering the dependency among the variables. In this stage, we specified the topology and probability distributions. There are two ways to do this: automatically from data obtained with a learning algorithm, or manually from an expert's domain knowledge [13]. In this work, we constructed the distributions manually because it was difficult to gather sufficient personal information data.

The Bayesian network for this problem included 33 observable variables, whose state was specified by observing the user's behavior or personal information, and 19 unobservable variables, whose state was specified from the relationship with the other variables. All variables were comprised of two states and there were 56 dependencies that were defined among the variables. Figure 2 illustrates the Bayesian network built for this problem.

3.3. Inferring the user's affect

The Valence-Arousal (V-A) space was applied to infer the user's affect. The V-A space is a simple model that represents affect as a given position in a two-dimensional space. It has been commonly used in previous studies on affect recognition [14]. It uses two eigen-moods: valence and arousal. Every kind of affect can be described in terms of these eigen-moods. The valence axis ranges from negative to positive. For example, the arousal axis ranges from calm to excited. For instance, "anger" is considered low in valence while high in arousal. In this work, we used four simple emotions: joy, anger, sadness, and relaxed. Figure 3 describes the position of these emotions in the V-A space.

General commonsense knowledge was utilized to define the variables and the relationships among the variables in the Bayesian network. Commonsense knowledge that implies the influence of available clues upon the user's emotions was represented as statements. For example, many business schedules may influence the user's emotion in a negative way. Many positive words or emoticons in the user's incoming message box may imply that the user's emotion is more likely to be positive. These statements are presented as conditional probability tables of the Bayesian network. For example, the probability that 'Valence' is 'Negative' is set to 0.7 if there are 'Too Many Events'. Table 2 shows the variables for inferring affect and the relationships between these variables.

3.4. Inferring how busy the user is

If the user makes frequent calls, he or she is likely to be very busy. A busy user is likely to receive many missed calls or unanswered messages. With commonsense rules, we are able to discover possible clues for inferring how busy the user is and design the causal relationships among the variables. The defined variables are shown in Table 3.

3.5. Inferring how close the user is to someone

In order to infer how close the user is to people who are registered in his or her address book, we used clues such as the content of incoming text messages, the name of the group the contact is added to, the frequency and duration of phone calls, etc. We show the variables that represent these clues in Table 4.

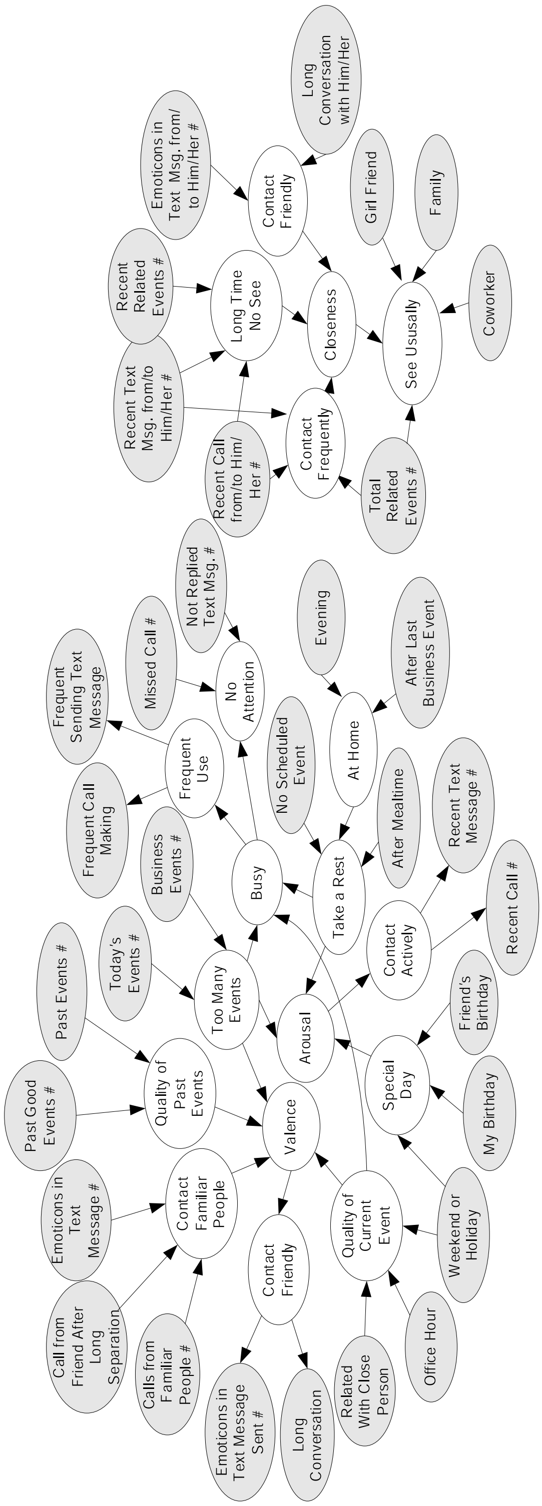


Fig. 2. An appropriate Bayesian network.

Table 2
Some variables used for inferring the user's affect

Name	Description	State	Type
Past Business Events #	The number of past business events	Many/Few	Observable
Quality of Current Event	Quality of event scheduled at now	Good/Bad	Unobservable
Today's Event #	The number of total scheduled events	Many/Few	Observable
Business Event #	The number of today's business events	Many/Few	Observable
No Event	No scheduled event at now	Yes/No	Observable
With Familiar People	Current event is related to familiar people	High/Low	Observable
Birthday	Today is his friend's birthday	Yes/No	Observable
Sent SMS Emoticon #	The number of emoticons in text message	Many/Few	Observable
Frequent Call Making	Many phone calls have been made in short time	Yes/No	Observable
Too Many Scheduled Events	There are too many events scheduled on today	Yes/No	Unobservable
Call From Familiar People #	The number of phone calls from familiar person	Many/Few	Observable

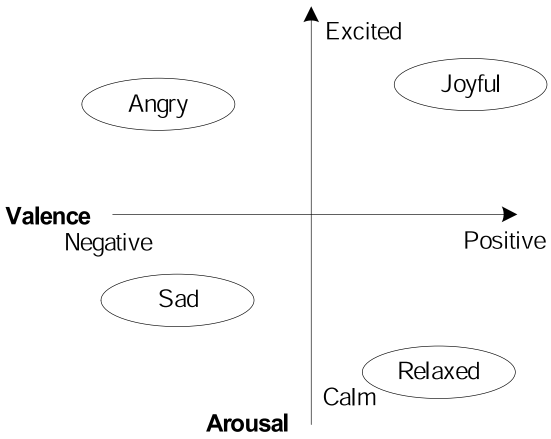


Fig. 3. Valence-arousal space.

Commonsense knowledge was used to construct the Bayesian network as follows. People who frequently make contact with or are registered in the "friends" group in the user's address book can be regarded as close. People with whom there has been little contact in a long period of time can be regarded as not close.

The degree of closeness can be inferred from all the persons who are registered in the user's address book and this is updated periodically. It is used for inferring other states rather than directly for action selection. For instance, the state of "Call from familiar person #" refers to the number of phone calls from close people and is used to infer the user's affect according to the degree of closeness of the caller.

3.6. Action selection

Actions that are appropriate to the current situation are selected using predefined rules. We defined four basic actions for the agent. These rules determine whether the agent should take an action and if so, what kind of action. Figure 4 illustrates the actions of the agent.

The condition part of the rules is a Boolean expression. The condition expression consists of the variables in the Bayesian network and the states of the variables. The action taken by the agent constitutes the result part. If the condition of the rule is met, the action is executed. The action selection rules are as follows:

Table 3
Some variables used for inferring how busy the user is

Name	Description	State	Type
Office Hour	Office hour at now	Yes/No	Observable
Weekend or Holiday	It is holiday or weekend today	Yes/No	Observable
Missed Call #	The number of missed calls	Many/Few	Observable
Use Frequently	The user uses the smarphone frequently	Yes/No	Observable
No Attention	The user does not pay attention to the smartphone	Yes/No	Unobservable
Take a Rest	The user is taking a rest	Yes/No	Unobservable
Busy	How busy the user is	High/Low	Unobservable

Table 4
Some variables used for inferring how close the user is with someone

Name	Description	State	Type
Related Events #	The number of events related with him/her	Many/Few	Observable
Long Time No See	The user has not got in touch with him/her for a long time	Yes/No	Unobservable
Recent Call #	The number of recent phone calls from/to him/her	Many/Few	Observable
Long Conversation	Talk with him/her frequently in a long time	Yes/No	Observable
Family	He/she is a family member	Yes/No	Observable
See Casually	See him/her almost everyday	Yes/No	Unobservable
Closeness	How close he/she is	High/Low	Unobservable

(1) If (Arousal = Calm && Valence = Negative && Quality of Current Event = Bad && Past Business Events # = Few)
OpenMsgWin "Do Your Best!"

(2) If (Arousal = Calm && Valence = Negative && Busy = Yes && Too Many Events = Yes)
OpenMsgWin "Cheer Up!"

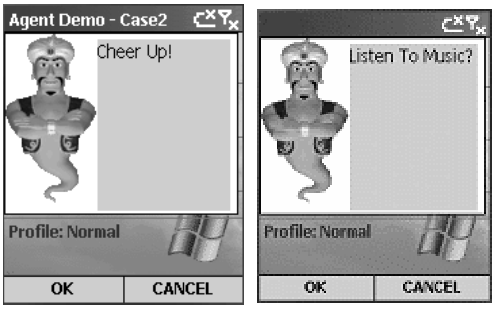
(3) If (Valence = Positive && Quality of Current Event = Good && Busy = No)
ChangeBkgnd "Bright"

(4) If (Arousal = Excited && Valence = Positive && Take a Rest = Yes)
Execute "MusicPlayer"

(5) If (Arousal = Calm && Valence = Negative && After Last Event = Yes && Take a Rest = Yes)
RecommendContact

The "Posting a message" action posts words of encouragement such as "do your best" and "cheer up!" on the home screen according to user's state. The "Execute a program" action can play music when the user is inferred as being relaxed and taking a rest. The "Change background image" action changes the background image of the home screen and it also changes the color coordination of the windows appropriate to the user's state. It changes the background image into a bright one when the user is in a good state. Finally, the "Recommend a conversation partner" action is selected if the user is considered depressed and tired. It recommends a conversation partner by randomly choosing someone who has high degree of closeness from the user's address book.

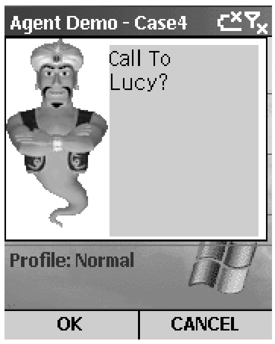
The agent is activated when each schedule begins or ends. It specifies the state of observable variables by detecting changes in personal information databases and communication logs. It updates the probability of unobservable variables by executing an inference algorithm. The updated probabilities are inputted to action selection rules and if a rule matches, the corresponding action is taken.



(a) Posting a message

(b) Playing music

(c) Changing background image



(d) Recommending a conversation partner

Fig. 4. Screen shots of agent's possible actions.

4. Working Scenario

To develop the agent prototype, we used the Microsoft Smartphone SDK 2002 and a smartphone emulator [15]. The emulator provides a good implementation of a real smartphone device containing PIMS, voice conversation, a text message service, a web browser, and an audio/video player. The Smartphone SDK provides a development environment for a Windows mobile operating system.

We observed the agent's behavior in the following situation. Our user is a thirty-year-old company

Table 5
Schedule table

Time	Title	Remark	Time	Title	Remark
09:00	Team meeting	Event 1	16:00		
09:30			16:30		
10:00	Visit client's site (Victor)	Event 2	17:00		
10:30			17:30		
11:00			18:00		
11:30			18:30		
12:00			19:00		
12:30	Lunch appointment (Smith)	Event 3	19:30		
13:00			20:00	Go to movie (Christen)	Event 5
13:30			20:30		
14:00	Project presentation	Event 4	21:00		
14:30			21:30		
15:00			22:00		
15:30			22:30		

Table 6
Address book

Group	Name	Phone number
Co-worker	Bryan	011-123-4567
	Tom	011-987-6543
	James	016-111-2222
Friend	Smith	011-333-4444
	Anderson	011-999-9999
	Paul	011-555-5555
Client	Victor	019-222-2222
	Marcus	016-333-3333
Girlfriend	Lucy	011-111-1111

employee who utilizes the smartphone actively for managing personal information, communication, and entertainment. It is a week day and five events are scheduled (as shown in Table 5.) He manages the information about his friends and colleagues with the address book program as shown in Table 6. He groups them into five categories according to kind of relationship. Tables 7 and 8 illustrate telephone conversations and text messages made throughout the course of the day. Figure 5 describes the actions taken by the agent over time.

The user's first activity is "team meeting" at 9 am. Unfortunately, he misses three calls on the way to the office because of having to rush. Although he barely arrives at the meeting on time, the agent is activated and begins the inference and action selection processes.

Newly gathered clues are as follows. "Missed Call #" is set as "Many" because the number of missed calls during the last 60 minutes was more than 4. "Office Hour" is set as "Yes" because it is a week day and between 7 a.m. and 6 p.m. "Business Events #" is set as "Many" because 3 scheduled events are grouped into the "work" category. No Event is set as "No" because "team meeting" is scheduled. "Today's events #" is set as "Many" because the number of events scheduled is more than 6. "Frequent Call Making" is "Yes" because the number of received or made calls is more than 5. These clues all

Table 7
Phone call log

Time	From	In/Out	Answered	Duration
08:49:00	Bryan	In	No	0:00
08:51:00	Bryan	In	No	0:00
08:52:00	Bryan	In	No	0:00
08:55:23	Bryan	In	No	0:00
08:59:00	Tom	In	No	0:00
09:50:02	Victor	Out	N/A	1:44
09:59:00	Victor	Out	N/A	2:00
10:05:00	Victor	In	Yes	1:00
10:00:09	Victor	In	Yes	0:50
10:30:19	Marcus	In	Yes	0:42
11:09:22	James	Out	N/A	1:01
13:25:00	James	In	Yes	0:22
19:49:00	Lucy	Out	N/A	2:12
19:53:00	Lucy	In	Yes	0:53
19:57:00	Lucy	Out	N/A	1:00
23:32:00	Lucy	Out	N/A	38:59

Table 8
Text message log

Time	From	In/Out	Contents
12:25:00	Smith	Out	Don’t you forget about our appointment?
12:27:40	Smith	In	OK. I’ll be get there in a moment -_-
12:37:33	Anderson	Out	How are you? Long time no see
12:39:00	Anderson	In	Yes fine, How are u? let’s get together some time ^^
12:42:01	Anderson	Out	OK ::) Keep in touch ~!
19:20:00	Lucy	Out	Shall I reserve the movie tickets?
19:22:50	Lucy	In	No, I have not chosen a movie ^^;
19:24:00	Lucy	Out	OK. Let’s meet in front of box office ~^^

negatively influence the user’s state.

On the other hand, there is a positive clue: the state of “Related With Close Person” is “Yes”. Even though this clue influences user’s state positively, the state of “Arousal” becomes “Low” with the probability of 0.57, the state of “Valence” becomes “Negative” with 0.7, the state of “Busy” becomes “Yes” with 0.58 by inference algorithm as shown in Fig. 6. This is because the negative clues are dominant. In this case, rule 1 matches and the agent cheers up the user by posting the “Do your best!” message on the home screen.

The agent does not know the explicit facts (such as whether the user is late for the meeting and depressed.) Nevertheless, the user’s state can be inferred properly by using implicit clues such as the number of missed calls, the number of remaining events, and the quality of the current event.

In the following scenario, the user visits one of his clients’ sites. However, he gets lost near his destination and calls his client to ask the way several times. During the meeting, he receives a phone call from another client and also makes a call to a co-worker. The meeting finishes at 11:30 am as scheduled, and then he departs for his office.

Then, the agent is activated again. The frequent phone calls imply that the user may be actively doing something. The caller’s category in his address book (co-worker, client) implies that the calls are about business. By using this implicit information, the agent figures out that the user is busy and possibly feels bad. Using rule 2, the situation is matched and the agent posts the “Cheer up!” message on the home screen again.

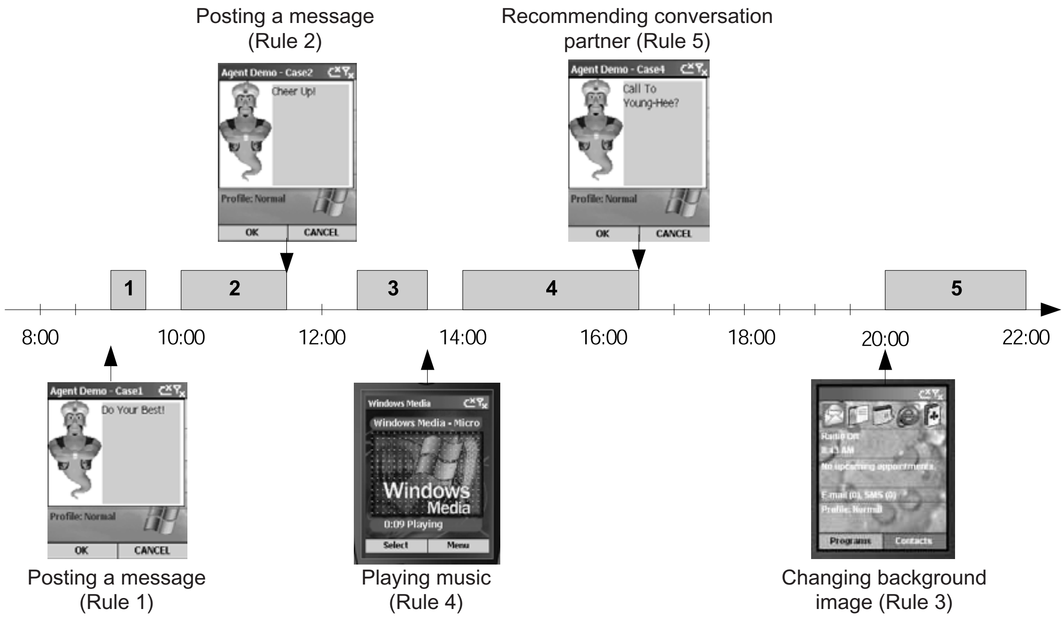


Fig. 5. Actions selected over time.

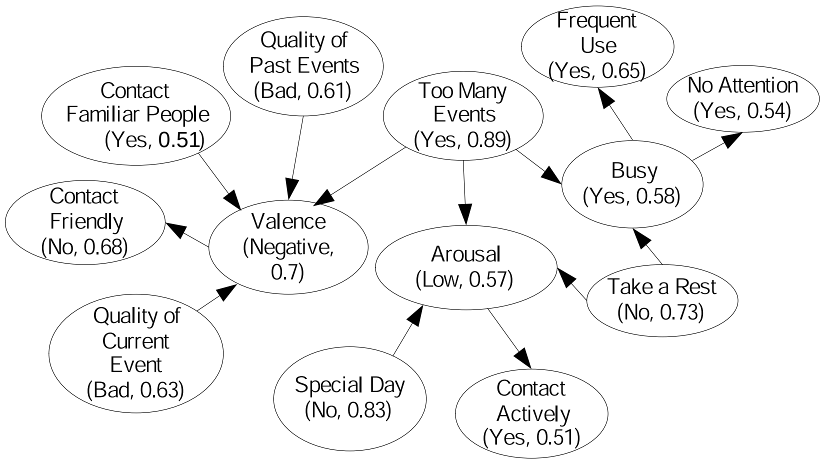


Fig. 6. Values of the network variables after the first event.

The next event is an appointment for lunch with his friend. They exchange text messages to make sure of the appointment time. His friend sends one more text message to find out exactly when he will arrive. The agent is activated when the user parts from his friend after lunch. This action signals the end of the third event. The agent regards the user as relaxed and likely to take a break due to the exchange of text messages with a close person, many emoticons in the text messages, lunchtime, and no scheduled work. Using rule 4, the situation is matched and the agent starts to play music appropriate for taking a rest.

"Project presentation" is the user's last business event. It keeps him tense for about two hours. At the scheduled end of the last event, the user prepares to leave his office and the agent wakes up. The agent infers the user's state as not good, but he is starting to relax because of the following facts: no more remaining scheduled work, many scheduled events on this day, and long duration of the last event. Then rule 4 matches: his girlfriend is selected as his conversation partner.

After his daily routine, he is scheduled to meet for a movie with his girlfriend. Leaving his office, he discusses the meeting place by exchanging text messages. When he meets her, the activated agent concludes that he is not busy, feels good about spending time with a very close person, and many emotions in recent text messages. Finally, rule 5 matches: the background image is changed to a much brighter one.

5. Conclusions

In our experiments, Bayesian networks were built on the assumption of a general commonsense tendency. Personalization was not included in this study, so the agent did not consider individual differences. In future work, we will explore how personalization can be achieved by adjusting the probability distributions in the Bayesian network with different learning algorithms. Rule-based action selection techniques can achieve reasonable performance somewhat easily. However, the user can become able to predict the agent's action after a given period of time because it selects the same action in similar situations. In future work, we will also attempt to adopt more flexible action selection techniques like behavior selection networks. Other information sources also have to be exploited for more accurate user modeling.

Acknowledgements

This research was performed for the Ubiquitous Computing Research Program funded by the Ministry of Information and Communication of Korea.

References

[1] P. Maes, Agents that Reduce Work and Information Overload, *Communications of the ACM* **37**(7) (1994), 31–40.
[2] T.M. Mitchell, R. Caruana, D. Freitag, J. McDermott and D. Zabowski, Experience with a Learning Personal Assistant, *Communications of the ACM* **37**(7) (1994), 80–91.
[3] G. Boone, *Concept Features in Re:Agent, an Intelligent Email Agent,* in Proceedings of the 2nd International Conference on Autonomous Agents, 1998, 141–148.
[4] D. Boyd and J. Potter, *Social Network Fragments: An Interactive Tool for Exploring Digital Social Connections,* in Proceedings of SIGGRAPH, San Diego, CA, July 2003, 27–41.
[5] A. Lockerd and E. Arroyo, *Personal Data for Personal Use: Case Studies in User Modeling for Context-Aware Computing,* AAAI Fall Symposium on Etiquette for Human Computer Interaction, Sept. 2002, 81–84.
[6] E. Horvitz, J. Breese, D. Heckerman, D. Hovel and K. Rommelse, *The Lumiere Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users,* in Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Madison, WI, 1998, 256–265.
[7] J. Liu, C.K. Wong and K.K. Hui, An Adaptive User Interface Based on Personalized Learning, *IEEE Intelligent Systems* **18**(2) (2003), 52–84.
[8] F. Sparacino, *Sto(ry)chastics: a Bayesian Network Architecture for User Modeling and Computational Storytelling for Interactive Spaces,* in Proceedings of the 5th International Conference on Ubiquitous Computing, Seattle, WA, October 2003, 54–72.
[9] D. Patterson, L. Liao, D. Fox and H. Kautz, *Inferring High-Level Behavior from Low-Level Sensors,* Proceedings of the FifthInternational Conference on Ubiquitous Computing, Seattle, WA, October, 2003, 73–89.
[10] Z. Ghahramani, Learning Dynamic Bayesian Networks, *Lecture Notes in Artificial Intelligence* **1387** (1997), 168–197, Springer-Verlag, London.

[11] Z. Pousman, G. Iachello, R. Fithian, J. Moghazy and J. Stasko, Design Iterations for a Location-Aware Event Planner, *Personal and Ubiquitous Computing* **8**(2) (2004), 117–225.
[12] E. Charniak, Bayesian Networks without Tears, *AI Magazine* **12**(4) (1991), 50–63.
[13] F.V. Jensen, *Introduction to Bayesian Networks*, Springer-Verlag New York, 1996.
[14] R. Picard, Affective Computing, *Media Laboratory Perceptual Computing TR 321*, MIT Media Laboratory, 1995.
[15] Microsoft, Windows Mobile Software for Smartphone http://www.microsoft.com/windowsmobile/smartphone/default.mspx, 2005.