Episodic Memory for Ubiquitous Multimedia Contents Management System*

Kyung-Joong Kim, Myung-Chul Jung, and Sung-Bae Cho

Dept. of Computer Science, Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, South Korea
kjkim@cs.yonsei.ac.kr, mcjung@sclab.yonsei.ac.kr,
sbcho@cs.yonsei.ac.kr

Abstract. Recently, mobile devices are regarded as a content storage with their functions such as camera, camcorder, and music player. It creates massive new data and downloads contents from desktop or wireless internet. Because of the massive size of digital contents in the mobile devices, user feels difficulty to recall or find information from the personal storage. If it is possible to organize the storage in a style of human-memory management, it could reduce user's effort in contents management. Based on the evidence that human memory is organized as an episodic-style, we propose a KeyGraph-based reorganization method of mobile device storage for better accessibility to the data. It can help user not only find useful information from the storage but also expand his/her memory by adding user's contexts such as location, SMS, call, and device status. User can recall his/her memory from the contents and contexts. KeyGraph finds rare but relevant events that can be used as a memory landmark in the episodic memory. Using artificially generated logs from a pre-defined scenario, the proposed method is tested and analyzed to check the possibility.

1 Introduction

Personal information management is one of the hottest issues because huge number of sensors is available at this moment and they can collect all the information about users [1]. Everything about users including photo, e-mail, movie clip, computer usage, TV watching, and contexts can be stored in a unified manner [2]. However, it requires a special-purpose equipment and software to do that and has difficulty to be used generally.

Though we cannot collect everything about users, relatively easy method such as using personal mobile devices can be a partial solution for the problem. Advances in mobile computing devices have led to digital convergence. Recent mobile phones provide many functions such as MP3, camera, game, PIMS (Personal Information Management System) and so on. Using logging software [3], the user's interaction data on the phone can be stored in the inside of device or remote server and retrieved for future use. Such information can be used to enhance user's access to the contents on the phone and expand limited human's memory.

* This research was supported in part by MIC, Korea under ITRC IITA-2006-(C1090-0603-0046).

H.G. Okuno and M. Ali (Eds.): IEA/AIE 2007, LNAI 4570, pp. 796–805, 2007.

Chance discovery is to recognize a chance which is a very rare event, but with significant impact on decision making or future change [4][5]. Also, although it is not rare, finding an event implying an uncertainty of the future is important. It gives not only an awareness of chances but also an explanation about chances. This approach has been applied to various applications domains such as predicting earthquake [6], discovering new topics from WWW [7], and identifying intrusions for computer security [8].

KeyGraph is one of the most frequently used methods for chance discovery [9]. Originally, it is proposed to index terms in a set of documents and its purpose is to find the main point of the documents, not frequent terms. Because the KeyGraph is only based on the information within documents, it does not rely on the domain-specific corpus. If we can expand the meaning of sentence and documents to the more general one, it is possible to detect chance in many practical areas.

The purpose of this research is applying the chance discovery method to the organization of information stored in the personal database. Because the size of information is huge, efficient organization of information is critical to find information quickly and accurately. The idea is to organize the information similar to human's memory structure. Studies on human memory support the assertion that people use special landmarks for recall and the memory is organized by episodes of significant events [10]. Chance discovery algorithm is used to find landmark events among daily events and the whole memory is reorganized by episodes with landmarks identified. Figure 1 shows an episodic memory of personal databases.

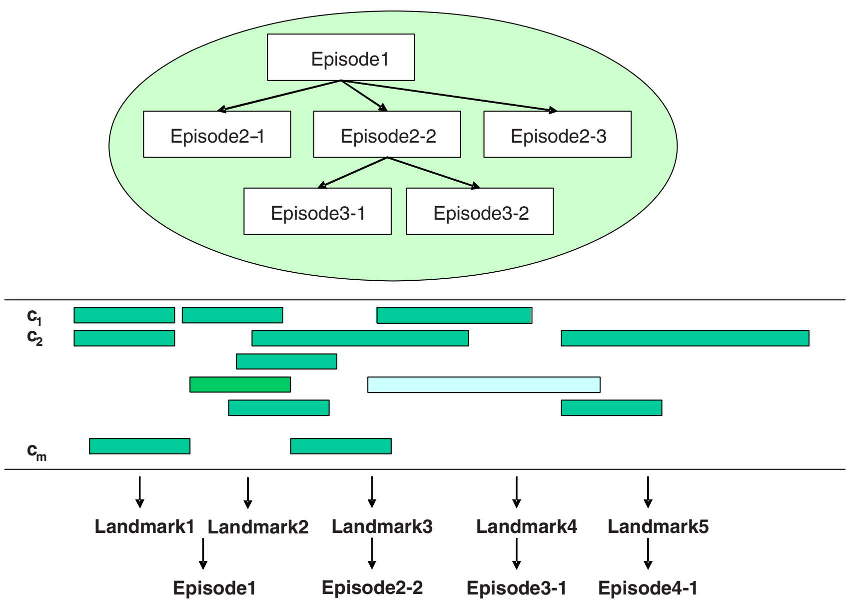

Fig. 1. Episodic memory of personal database (*c* means context and multimedia in the personal database)

In this paper, we propose a novel method to manage user's information in smartphone. User's logs such as call logs, SMS logs, multimedia logs, GPS logs and so on are recorded and combined into an integrated log format. The logs follow the prearranged procedure to find the key events and the relationship of each pair of events applying KeyGraph algorithm which detects rare but very important event based on event sequences and major changes of event environment. Because the key is unique and memorable thing, it can be used as landmarks. Each cluster is regarded as an episode. A user can explore the personal databases using the provided landmark identifiers.

2 Related Works

The initial work for the episodic memory is done by Miikkulainen who proposes an episodic memory that has functions of classifying, storing and retrieving user's memory recorded in scripts using hierarchical SOM (Self-organizing Feature Maps) [11]. Hierarchical structure enables the system to reduce time for recalling the memory.

At Microsoft research, there are many papers about personal information management. Eric Horvitz *et al.* attempt to re-organize personal information storage in desk top PC into an episodic style memory [10]. He learns Bayesian networks to detect landmark event from the data stored in outlook scheduler. Given schedule information, the Bayesian network provides the probability of landmarks of events. Bayesian networks are used for chance discovery and it can easily deal with uncertainty in the data [12]. However, Bayesian network requires relatively high computational cost compared to other simple models and it is the main reason of difficulty in the use of the model in computationally poor environments.

Bayesphone uses client-server communication for Bayesian network, its inference is done on server-side, and the results are transmitted to device through network [14]. It causes communication cost and the device must be always online. SMILE (Structural Modeling, Inference, and Learning Engine) is a Bayesian network library for mobile device [13]. Though it supports a way to implement Bayesian network inference in mobile device easily, it cannot handle inference of large Bayesian networks. KeyGraph is relatively simple model for predicting landmark and it provides a natural view on the episodic memory because each cluster can be regarded as an episode. Useful services using the well-organized personal information are very important for the success of personal information management. E. Horvitz *et al.* develop LifeBrowser and MemoryLens for more efficient access for the information on desktop [15][16]. They exploit the landmark probability inferred from the learned Bayesian networks to visualize the structure of information (stored desktop files).

MIT reality mining group develops serendipity service using the ContextPhone software [18]. The group collaborates with MIT common sense reasoning group to generate diary automatically. Because the research is at early stage, there is no concrete result about that. Only visualization tool for collected log is available in their paper. However, their work shows a new way to generate more interpretable high-level diary (interpretation) using common sense. Basic details about the common sense knowledge can be found in [19].

Recently, Sumi *et al.* develop a ubiquitous system to summarize user's experience on conference tour in a video, but his work requires too many sensors and devices to do that [17].

3 Contents Management Using KeyGraph

The framework of the proposed method consists of log collection, log integration, KeyGraph generation, KeyGraph analysis and information search. Figure 2 shows the overview of the system.

Log collection module continuously gathers application usage, call & SMS, location, device status, and public web information. The logs for several sources are sorted based on the time. The sorted log is called as integrated log. Then, KeyGraph is generated from the new log and key events are extracted. The whole memory is reorganized based on the keys and clusters. By clicking the key, user can access to the related information easily.

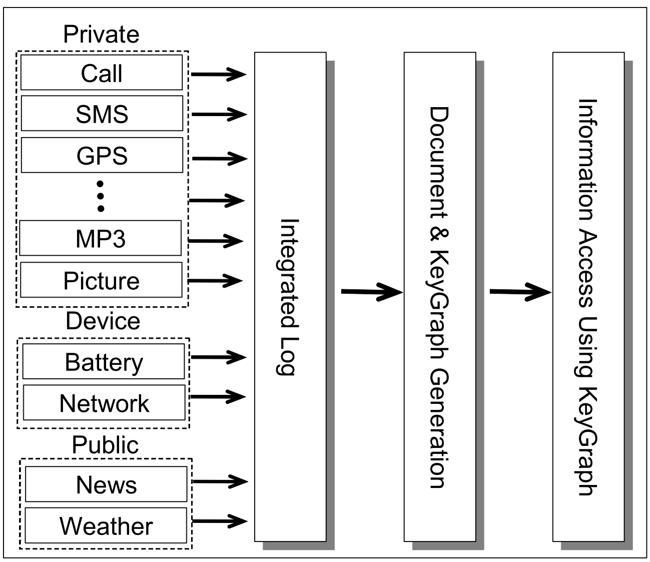

Fig. 2. System overview of the proposed method

3.1 Log Preprocessing

Time, GPS, call, SMS, photo, MP3, e-book, device status and web information can be logged into personal store. Figure 3 shows log preprocessing procedure in detail. Because GPS information is just a pair of longitude and latitude, it is required to convert them into semantic label such as name of building, street and landmark place. Pre-stored mapping table is used to do that. Other log data except location are sorted based on the time and each event is labeled as a unique ID. The naming is the

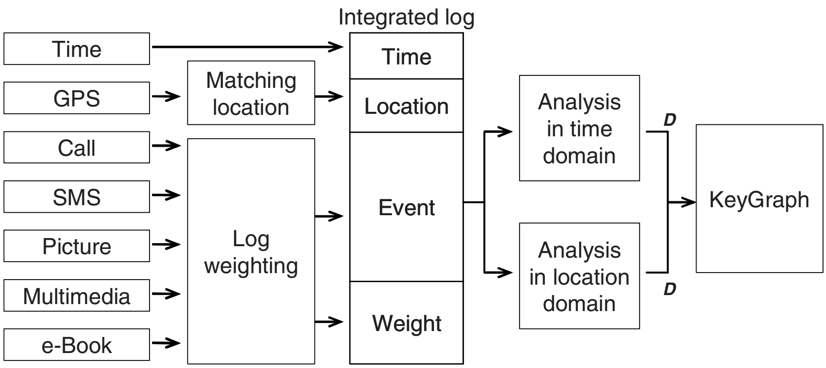

Fig. 3. Details of log preprocessing and KeyGraph generation

combination of log type and event ID. If the event is for calling and it is the 5th event of the day, the ID is Call5.

3.2 Document and Sentence Separation

Because input of the KeyGraph is a set of documents, it is required to reorganize the logs into sentences and documents. The separation of logs is done based on pre-defined rule. 24 hour data are regarded as a separated document. If the data are collected for 7 days, it means that there are 7 documents. Each document contains each day's log. Let's define the total document set as *D*.

$$D = \{d_1, d_2, ..., d_N\} \tag{1}$$

Each document is defined as follows.

$$d_i = \{e_{i1}, e_{i2}, ..., e_{iM}\} \tag{2}$$

e_{ij} means the j th event in the document. A sentence is defined as a group of events. Based on the grouping rule, there are many different ways to form the sentence. The rule set is defined as R_i.

$$R_i = \{r_{i1}, r_{i2}, ..., r_{iP}\} \tag{3}$$

Let's define a set of sentences generated from each rule as S_{ij}. The sentence set S_i for the document d_i is defined as the union of all S_{ij}.

$$S_i = S_{i1} \cup S_{i2} \cup \cdots \cup S_{ip} \tag{4}$$

Table 1 summarizes all sentence separation rules. If time is used as a rule and the threshold is 1 hour, there could be 24 sentences for a day.

Table 1. Sentence separation rule

Log	Sentence separation rules
Time	T < Time < T+δ, T+δ < Time < T+2×δ, …
Location	**If** (L_new≠L_old) **then** new_sentence

3.3 KeyGraph Generation

KeyGraph extracts the important events and the causal structures among them from such an event sequence [9]. A document example is given in Eq.(5).

$$\begin{aligned}&\text{Call20\#Pic2\#Pic7\#SMS11\#.}\\&\text{EB24\#MM12\#Pic7\#Call20\#SMS11\#.}\\&\text{MM5\#SMS11\#.}\\&\text{EB24\#MM5\#.}\end{aligned} \tag{5}$$

It is composed of 4 sentences. Figure 4 shows an example of KeyGraph based on the document. Each cluster is composed of co-occurring frequent events. That is, events appearing frequently in *D* are extracted, and each pair of events that often occur in the same sentence unit is linked to each other. 'Call20, Pic7, SMS11' forms a cluster as shown in the figure. The events that are not frequent but co-occurring with multiple clusters, e.g., 'MM5' are a key event. It is rare but very important events.

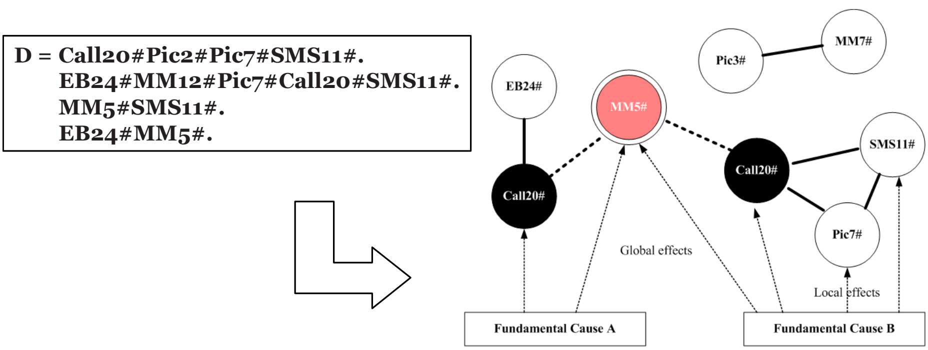

Fig. 4. An example of KeyGraph

The details of the KeyGraph generation are as follows. First, highly frequent events in the document are listed. Then, pairs of the events that are often co-occurred are extracted based on $local(e_i, e_j)$ in Eq.(6), and their link is drawn by the solid line in Figure 5. e means each event and g means the cluster that is composed of e linked by solid lines. $|e|_s$ means the count of e in sentence *S*.

$$local(e_i, e_j) = \sum_{S \in d} \min(|e_i|_S, |e_j|_S) \tag{6}$$

$global(e_i, g)$ in Eq.(7) calculates the strength between event and cluster. Event e that has the highest value summed by $global(e_i, g)$ from every cluster is extracted as key event described as Eq. (8). Links with key events is drawn by dot lines. $|g|_s$ means the count of cluster g in sentence *S*.

$$global(e_i, g) = \frac{\sum_{S \in d} |e_i|_S |g - e_i|_S}{\sum_{S \in d} \sum_{e_i \in S} |e_i|_S |g - e_i|_S} \quad (7)$$

$$where\, |g - e_i|_S = \left\{ \begin{array}{ll} |g|_S - |e_i|_S & if\ w \in g \\ |g|_S & if\ not\ w \in g \end{array} \right\}$$

$$key(e_i) = 1 - \prod_{g \in G} (1 - global(e_i, g)) \quad (8)$$

3.4 Information Access

Every key event arranges in order of time. User can simply search for his/her key events of a day and get access to their sub-events quickly. For each key event, associated contents are linked and user can access them hierarchically from the keys. If user did not know much about the detailed contents, he would only search the keys and attempt to remember about the contents. If he thinks that the contents are related to the specific key event, he can expand the searching by following the linked contents from the keys.

4 Experimental Results

The target of a scenario is daily life of an undergraduate student with smartphone, which is summarized in Table 2. The logging software of the smartphone records the events when user's activity changes. When the schedule of the day finishes, contents management software lets user know the key events and the sub-events based on their co-occurrence analysis.

Table 2. User's daily life

Time	Location	Works	Main Events
-9:00	Home		Call
9:00-10:00	Bus	To college	Mp3
10:00-12:00	Engineering hall II	A major	SMS
12:00-13:00	Cafeteria	Lunch with friend	Picture
13:00-14:00	General classroom bui lding	Liberal arts	SMS
14:00-15:00	Bench in front of engi neering building	Rest	E-book, MP3
15:00-17:00	Engineering hall I	A major	E-book
17:00-18:00	Central library	Study	Movie Clip
18:00-21:00	Sinchon	Meeting with frie nds	Call, Picture
21:00-22:00	Bus	To home	MP3
22:00-	Home		SMS

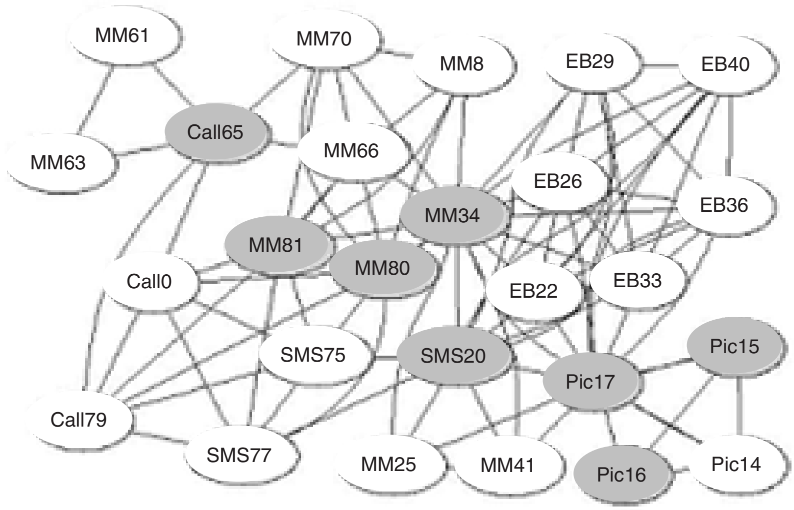

Fig. 5. KeyGraph from the scenario

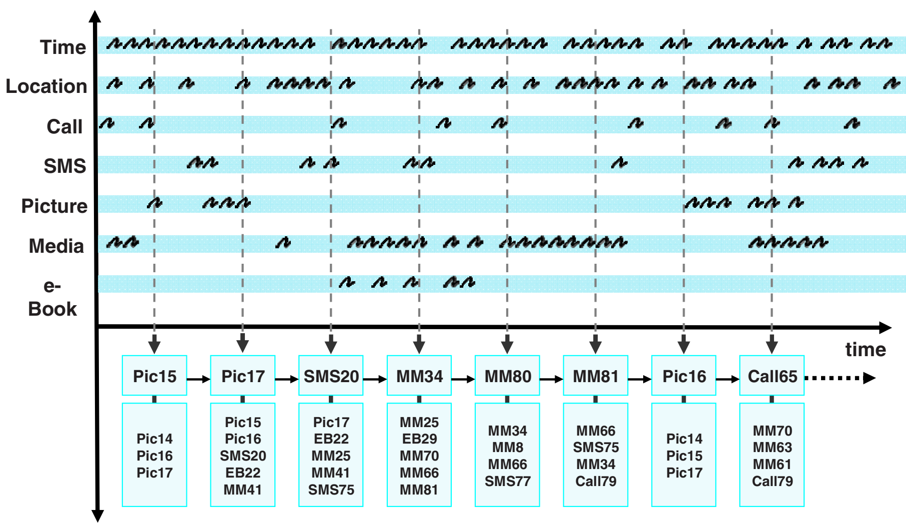

Fig. 6. The key events and the sub-events based on user's logs in time order

Figure 5 shows a part of KeyGraph based on the user's events of the scenario. The black nodes denote key events of the graph and the white nodes denote highly frequent events. "Pic15," "Pic16," and "Pic17" mean the pictures taken by smartphone with his friends in cafeteria. "SMS20" denotes friend's SMS to notify the important meeting with club friends. "MM34," "MM80," and "MM81" are impressive songs to the user. "Call65" means the events that user call to his girlfriend when the meeting is over. At this time, "Call65" is strongly linked to MP3 events because he goes home as listening MP3.

Figure 6 is a part of the results of analyzing KeyGraph. The key events are extracted on time axis, and the sub-events linked with the key events are displayed. At this stage, the performance of the system is tested using synthetic data because collecting real data requires much time. It takes more than one month to get useful statistics.

5 Conclusions and Future Works

In this paper, we propose the method that can find rare but very important events based on event sequences from smartphone using KeyGraph. We define available logs in smartphone and build the integrated logs and transform them into documents using sentence generation rules. Analyzing the results of KeyGraph, user's information is arranged by key events. The usefulness of the system is evaluated based on scenario and artificially generated data. KeyGraph results show the possibility of the proposed method. As a future work, real logging software needs to be used for collecting user's data. Based on them, the system can be evaluated and tested in a systematic way. Also, it is required to define a way to handle large-scale KeyGraphs in the limited resource environment.

References

[1] Teevan, J., Jones, W., Bederson, B.B.: Personal information management. Communications of the ACM 49(1), 40–43 (2006)

[2] Gemmell, J., Bell, G., Lueder, R.: MyLifeBits: A personal database for everything. Communications of the ACM 49(1), 88–95 (2006)

[3] Raento, M., Oulasvirta, A., Petit, R., Toivonen, H.: ContextPhone: A prototyping platform for context-aware mobile applications. IEEE Pervasive Computing 4(2), 51–59 (2005)

[4] Ohsawa, Y.: Chance discoveries for making decisions on complex real world. New Generation Computing 20(2), 143–164 (2002)

[5] Ohsawa, Y., McBurney, P.: Chance Discovery. Springer, Heidelberg (2003)

[6] Ohsawa, Y.: KeyGraph as risk explorer from earthquake sequence. Journal of Contingencies and Crisis Management 10(3), 119–128 (2002)

[7] Ohsawa, Y., Soma, H., Matsuo, Y., Matsumura, N., Usui, M.: Featuring web communities based on word co-occurrence structure of communications. In: Proceedings of the 11th International Conference on World Wide Web, pp. 736–742 (2002)

[8] Koo, J.-M., Cho, S.-B.: Interpreting chance for computer security by viterbi algorithm with edit distance. New Mathematics and Natural Computation 1(3), 421–433 (2005)

[9] Ohsawa, Y., Benson, N.E., Yachida, M.: KeyGraph: Automatic indexing by co-occurrence graph based on building construction metaphor. In: Proc. Of Advanced Digital Library Conference (IEEE ADL'98), pp. 12–18 (1998)

[10] Horvitz, E., Dumais, S., Koch, P.: Learning predictive models of memory landmarks. In: 26th Annual Meeting of the Cognitive Science Society (2004)

[11] Miikkulainen, R.: Script recognition with hierarchical feature maps. Connection Science 2, 83–101 (1990)

[12] Kim, K.-J., Cho, S.-B.: Uncertainty reasoning and chance discovery. In: Chance Discovery in Real World Decisions, Springer, Heidelberg (2006)
[13] GeNIe & SMILE, http://genie.sis.pitt.edu
[14] Horvitz, E., Koch, P., Sarin, R., Apacible, J., Subramani, M.: Bayesphone: Precomputation of context-sensitive polices for inqury and action in mobile devices. User Modeling, pp. 251–260 (2005)
[15] Ringel, M., Cutrell, E., Dumais, S., Horvitz, E.: Milestones in time: The value of landmarks in retrieving information from personal stores. In: Proceedings of Interact 2003, the Ninth IFIP TC13 International Conference on HCI, pp. 228–235 (2003)
[16] Cutrell, E., Dumais, S.T., Teevan, J.: Searching to eliminate personal information management. Communications of the ACM 49(1), 58–64 (2006)
[17] Sumi, Y., Ito, S., Matsuguchi, T., Fels, S., Mase, K.: Collaborative capturing and interpretation of interactions. In: Pervasive 2004 Workshop on Memory and Sharing of Experiences, pp. 1–7 (2004)
[18] Eagle, N.: Machine Perception and Learning of Complex Social Systems. Ph. D. Thesis, Program in Media Arts and Sciences, Massachusetts Institute of Technology (2005)
[19] Singh, P., Barry, B., Liu, H.: Teaching machines about everyday life. BT Technology Journal 22(4), 227–240 (2004)