AniDiary: Daily Cartoon-Style Diary Exploits Bayesian Networks

AniDiary (Anywhere Diary) uses Bayesian networks to automatically detect landmark events and summarize a user's daily life in a cartoon-style diary.

Sung-Bae Cho, Kyung-Joong Kim, Keum Sung Hwang, and In-Ji Song
Yonsei University

People love to capture their memories and experiences to create personal histories and track events,[1] as evidenced by the success of myspace.com and youtube.com. However, users must manually construct such collections, and organizing and labeling the contents is time consuming. A more traditional method for recording memories is a diary, but not everyone has the daily habit of writing in a journal. Yet these types of information storage can augment our memory[2] and help us predict future events and recall old ones. We thus aimed to create a system that could automate and complement such diary-generation procedures.

Our goal was to summarize a given user's daily life with a cartoon-style diary based on information collected from mobile devices such as smart phones. Our system, AniDiary (Anywhere Diary), addresses two main problems of typical diary systems—the huge number of events originating from the real-life log and the awkward presentation of the output. Using modular Bayesian networks, AniDiary can detect and visualize landmarks (relevant or novel events) and transform numerous logs into user-friendly cartoon images. The cartoons provide a good starting point for fine-grained searches of detailed information. For example, users can link each cartoon to rich media (photos or videos) that offer more details, reducing the search space and letting users easily recall the linked details.

This application of mobile devices promises to provide a new way for people to manage their personal information.

AniDiary

Expanding on others' research, we organized our system similarly to how human memory is structured (see the "Related Work on Life Logging" sidebar).[3] We used several Bayesian networks designed by experts to find memorable events in a modular manner. Bayesian networks are one of the most efficient ways of inferring situations given a certain amount of uncertain or partial information. After AniDiary logs and preprocesses events, it selects the most memorable ones and converts them into cartoons by selecting a set of cartoon image components from a database and composing them into a cartoon. Figure 1 shows the overall procedure.

Logging

Numerous sources are available for logging information, and only some of them require attaching additional devices to the phone. For example, ContextPhone is context-logging software for Nokia 60 smart phones,[4] and its source is available to the public. It collects information on a wide range of topics by

- logging photographs taken, music downloads,

Published by the IEEE Computer Society ■ 1536-1268/07/$25.00 © 2007 IEEE

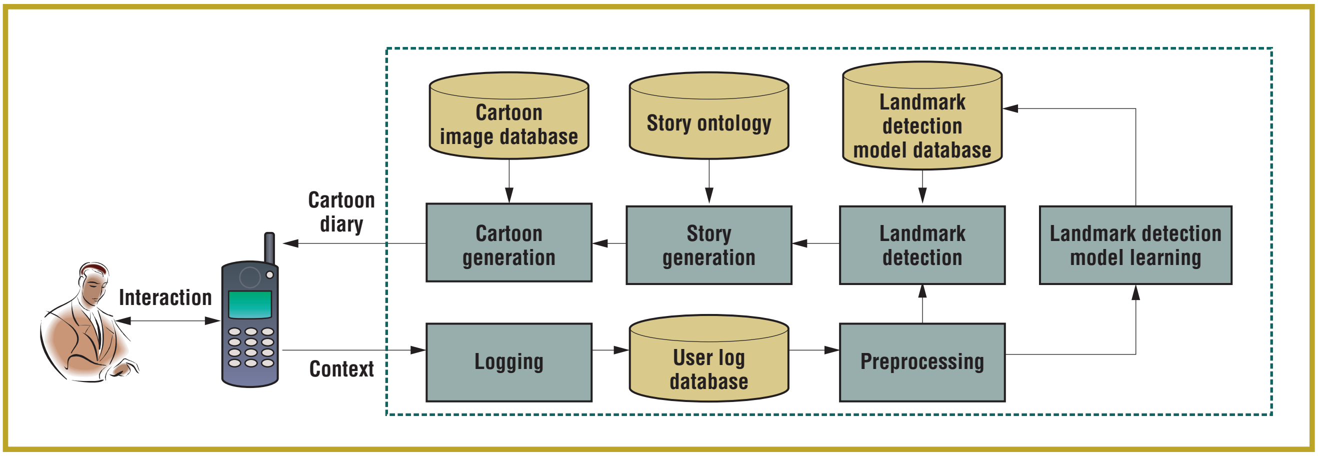


Figure 1. AniDiary's general architecture.

short-message-service use, multimedia-messaging service use, and Bluetooth use;

- monitoring the phone's battery level;
- storing call logs; and
- recording applications in use.

However, because ContextPhone is based on the Symbian operating system, we also developed a logging module for the Windows mobile environment. (In 2006, 51 percent of mobile device operating systems used Symbian and only 17 percent used Windows. However, by 2010, the Diffusion group forecasts that 22 percent of mobile devices will use Symbian and 29 percent will use Windows.[5])

Our logging system runs on Windows CE with a small GPS receiver attached to the device. The system continuously records the user's latitude and longitude and lets users easily access call logs and the address book. It also stores SMS texts, frequently compressing them according to the device manufacturer's standards (to do this, we have to contact the manufacturer). We also modified the code for a photo viewer and an MP3 player and added it to our system to log usage information. We can easily gather a photo's creation time and other low-level information from the photo's header. The system retrieves weather information from the Korean Meteorological Administration (www.kma.go.kr), and it samples the GPS and battery level once per second.

Preprocessing

This stage employs standard statistical analysis to extract significant information. Because raw information isn't meaningful, we use statistical variations to detect informative situations.

For example, we infer a user's current position from the GPS value of a pair of longitude and latitude coordinates. (We "infer" because a device can sometimes lose its GPS signal in a building or a shadowed area of an urban environment, or when placed in a pocket or bag. Cell-based location positioning is available, but its accuracy is worse than with the GPS.) Using the GPS value, a Web service (http://maps.naver.com) identifies the nearest building. We then transform the raw information into semantic labels using stored information about the relationship between the GPS value and the semantic label (the building or street name). The stored information might include semantic labels such as "my home," "my office," or "my friend's home." The user can manually input this information using AniDiary's map-based visualization.

To determine discrete information using SMS text, call logs, photos, and MP3 selections, we extract patterns using simple statistical techniques, such as determining the average, maximum, and minimum values or the frequency over the time domain.

Detecting memory landmarks

In this stage, user feedback and learning procedures can reduce the overhead of manually constructing a detection model.

The idea of landmarks stems from human memory research, which has shown that the brain stores related events together as "episodes" and uses landmark events to point to each episode. The problem is identifying the landmark events and labeling them so that you can use them for future indexing. When people try to remember things, the query is often unclear and we sometimes miss related events, so we must deal with uncertainty and missing variables. A Bayesian network can address such problems by providing a robust inference based on probability theory.

SMILE (Structural Modeling, Inference, and Learning Engine) is a Bayesian network library for mobile devices (http://genie.sis.pitt.edu). Although it supports ways to implement Bayesian network inference in mobile devices easily, it can't handle the inference that comes from extremely large Bayesian networks. Because our focus is on daily life, the Bayesian network could be large enough to cause real-time errors. So, we structured the Bayesian network in a modular rather than monolithic fashion.

A daily diary's domain includes many activities, and we can't incorporate them all into a single model. So, we propose using an ensemble of multiple Bayesian networks specialized for each activity. Each model is manually designed by

Related Work on Life Logging

A key organizational principle of human memory is episodic storage and retrieval. Our brains group related events as episodes and use landmark events to recall these episodes. Finding such landmark events can also recall related items. Similarly, Bayesian networks can detect landmark events from the data stored in schedulers.

Eric Horvitz, Susan Dumais, and Paul Koch attempted to reorganize personal information storage in desktop PCs in terms of an episodic style of memory.[1] They built a single Bayesian network using data from a desktop environment for landmark detection. We've expanded this idea by designing an ensemble of Bayesian networks for landmark detection.

Our research also borrows from the Massachusetts Institute of Technology's research. The MIT Reality Mining group has developed a serendipity service, which cues informational, face-to-face interactions between nearby users who don't know each other but probably should. Their service uses the ContextPhone software,[2] and they've been collaborating with the MIT Common Sense Reasoning group to generate diaries automatically. Because the research is still in the early stages, although the Reality Mining group has made available its visualization tool for a collected log, it has yet to produce any concrete results. However, their work shows a new way of generating more interpretable high-level diaries using common sense. (Basic details about common sense knowledge appear elsewhere.[3]) Our work is based on this ontology and can be expanded to more general models using such a common sense corpus.

Other related work is the comic diary system Yasuyuki Sumi and his colleagues designed to summarize conference tours in a cartoon-style form.[4] They based their system on explicit user input including schedule information.

Nathan Eagle has also tried to develop a diary system based on log information collected from cellular phones.[5] This system showed raw information directly on a GUI, which made it difficult to intuitively understand the big picture of a given day. Nokia's Lifeblog service gives users a way to store and manage photographs, multimedia, and short-message-service messages chronologically (see www.nokia.com/lifeblog). However, it doesn't use any abstraction or summarization methods.

REFERENCES

1. E. Horvitz, S. Dumais, and P. Koch, "Learning Predictive Models of Memory Landmarks," *Proc. 26th Ann. Meeting Cognitive Science Soc.*, Lawrence Erlbaum Associates, 2004, pp. 583–588.

2. N. Eagle and A. Pentland, "Social Serendipity: Mobilizing Social Software," *IEEE Pervasive Computing*, vol. 4, no. 2, 2004, pp. 28–34.

3. P. Singh, B. Barry, and H. Liu, "Teaching Machines about Everyday Life," *BT Technology J.*, vol. 22, no. 4, 2004, pp. 227–240.

4. Y. Sumi et al., "ComicDiary: Representing Individual Experiences in a Comic Style," *Proc. UbiComp*, Springer, 2002, pp. 16–32.

5. N. Eagle, "Machine Perception and Learning of Complex Social Systems," PhD thesis, Program in Media Arts and Sciences, Massachusetts Inst. of Technology, 2005.

experts, based on the independence of random variables. Each model has its own input, intermediate, and output variables, some of which it can share with other models. We manually determine the causal relationships between the variables using the expert knowledge, and the system determines the conditional probabilities using the noisy-or method. Virtual evidence techniques let us input other models' probabilistic outputs.

Increasing the number of stages in the inference requires more computational resources, so we limited the number of stages to two. In the first stage, AniDiary inputs evidences to each Bayesian network and calculates each network's output. In the second stage, it uses the first stage's output as input evidences for the other Bayesian networks.

In figure 2, the dotted line indicates a virtual link as well as the stream of the second stage of inference processing. Parentheses indicate the number of Bayesian networks. We used 39 Bayesian networks, broken down into four kinds:

- *Place-activity*: houses, religion, shopping, photographs, hospitals, nature, meetings, workplaces, sports, movements, food, calls, music, schools, traffic, amusement, and busy, watching, and resting activities;
- *Emotional/conditional*: joy, hunger, hot or cold temperatures, nonsense, surprise, tiredness, drunk, anger, worry, gloom, sickness, and boredom;
- *Circumstantial/situational*: space, climate, time, device, and group; and
- *Event*: anniversaries and other events.

The 39 networks contain 638 nodes, 623 links, and 4,205 conditional probability values (CPVs). Merging these into a single model results in 462 nodes (we removed duplicate nodes). The modular Bayesian networks make an inference 39 times for average of 16.6 nodes and 107.8 CPVs. Meanwhile, the single model makes one inference for 469 nodes and 4,869 CPVs. Usually, the mobile version of Bayesian networks might not deal with such large Bayesian networks owing to memory problems.

Story and cartoon generation

For story generation, a template-based method with an ontology is a good choice, but there are several difficult issues to consider.

The easiest way to present landmark

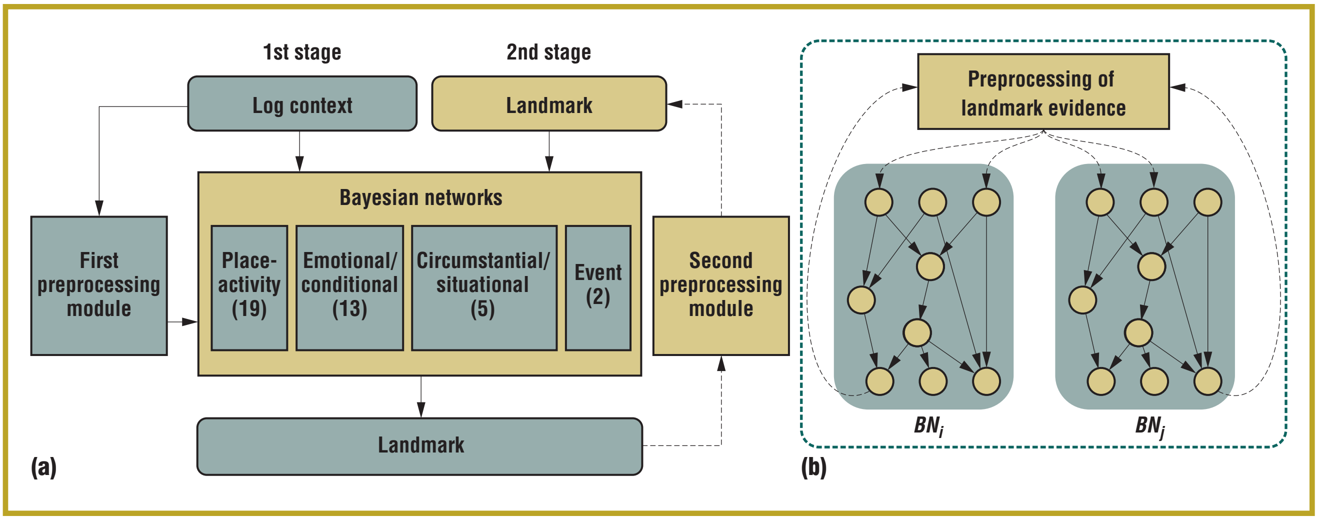


Figure 2. (a) The two-stage inference process for the cooperation of modular Bayesian networks. (b) Bayesian networks modularized for efficiency.

detection results is chronologically, but this can lead to a boring or redundant story. Reorganizing the landmarks often better captures the event. The detected landmarks are connected if cause-and-effect relationships exist between them. Such a connection results in a number of graphs with different landmark averages. AniDiary then sequentially presents the highest landmarks from each graph.

A single cartoon cut combines five images, overlaying text, subcharacters, main characters, the subbackground, and the main background (see figure 3). Professional artists prepare a set of images for each of the five image types. For example, our system contains 253 main background images. We then fuse one of these images with one of our 21 subbackground images, which represent various weather conditions. So the total number of possible background images is 253 × (21 + 1) = 5,566. The system contains 356 main characters (178 Asian and 178 Western characters). It includes 69 exaggerated images and nine animated images, and 26 subcharacters consisting of four types (man, woman, Asian, Western). The total number of images composed of the two character types is 178 × (26 + 1) × 2 = 9,612. Approximately 53 million cartoons are possible.

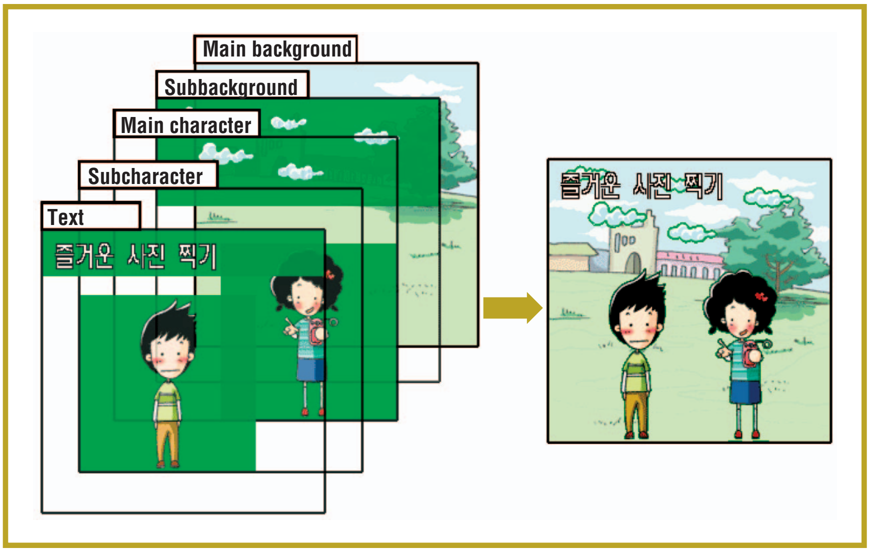


Figure 3. The composition of image components for a single cartoon cut.

AniDiary determines the number of cartoons on the basis of the threshold value for landmark detection. The larger the value, the fewer the cartoons, because fewer landmarks exceed the threshold.

Experimental results

We tested the landmark-detection module's performance using artificial data generated from predefined rules. First, we used data generated for only one day, then for 30 days. We also asked users to evaluate our cartoon images, and eventually we used real log data to evaluate system performance.

A preliminary test

We tested the proposed landmark-reasoning model using an artificial scenario (see figure 4a) to measure the performance of the manually designed modular Bayesian networks. The model incorporates prior knowledge about users and their living patterns. From the 16-hour scenario, we generated log contexts for 24 hours, and then we tested the data.

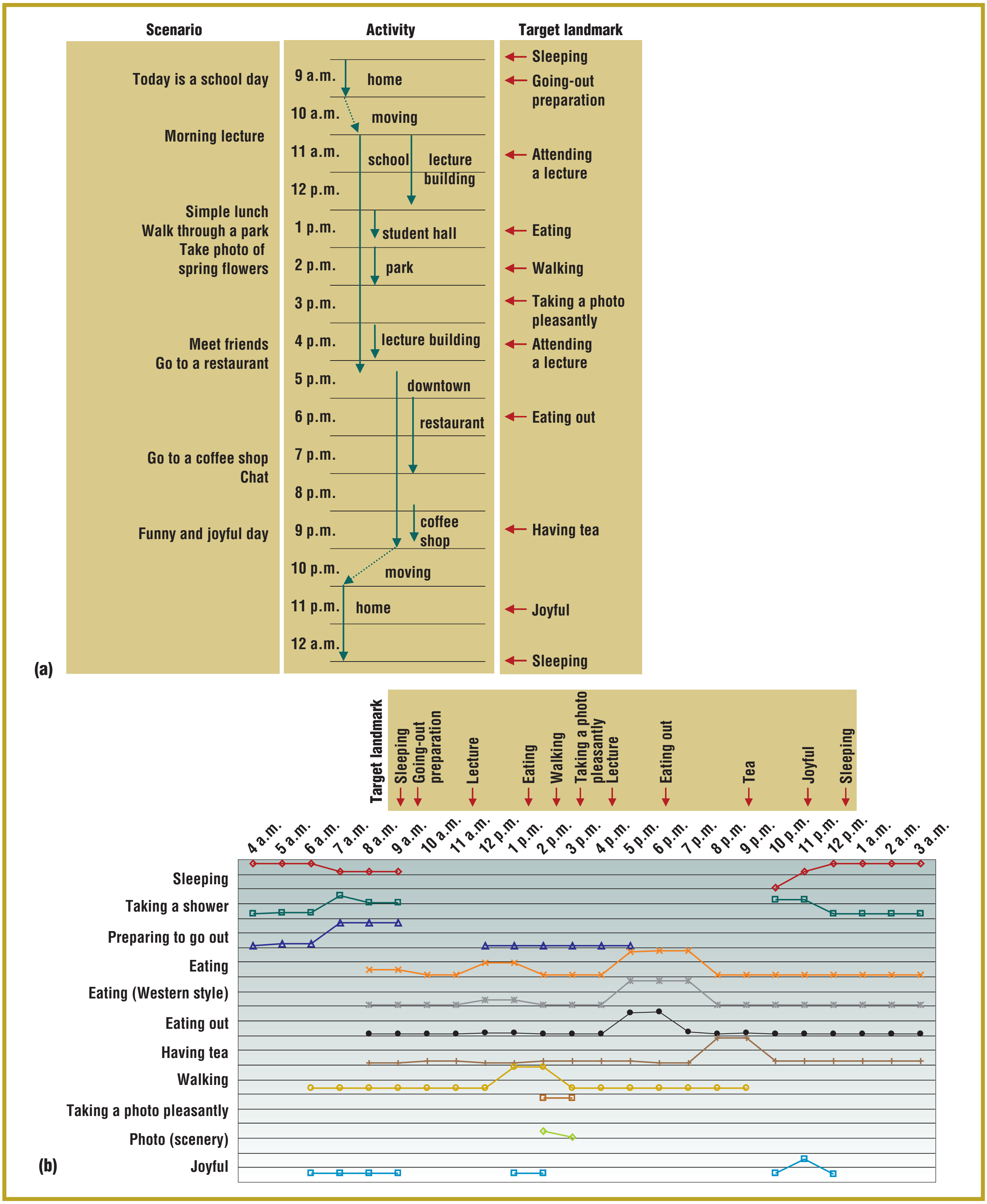


Figure 4. The probability change of each landmark node inferred in Bayesian networks. The change verifies that AniDiary makes a proper inference for the scenario: (a) A summary of a normal day using an undergraduate student's mobile device. (b) The observation of the probabilities of 11 target landmarks. The denoted time is from 4 a.m. to 3 a.m. the following day.

TABLE 1
Experimental results with synthetic data.

Class	No. of days	No. of target landmarks	True positive error rate (%)	False positive error rate (%)	False negative error rate (%)	Precision	Recall
Usual/idle	30	60	46	14	14	0.767	0.767
Unusual/idle	30	58	43	10	15	0.811	0.741
Usual/busy	30	55	41	2	14	0.953	0.745
Unusual/busy	30	60	46	8	14	0.852	0.767
Total	120	233	176	34	57	0.838	0.755

Figure 4b shows the inference results based on recorded landmark probability increments. For example, the "preparing to go out" and "taking a shower" landmarks occurred from 7 to 9 a.m., "eating" from 12 to 1 p.m. and from 5 to 7 p.m., "walking" from 1 to 2 p.m., "taking a photo pleasantly" from 2 to 3 p.m., and "eating out" and "eating (Western style)" from 5 to 7 p.m. The inference results (or probability transitions) indicate that the modular model produces appropriate probability given the scenario's artificial data.

Performance evaluation on long-term data

To produce a more realistic evaluation, we collected data for 30 days, grouping the situations into two conditions—usual/unusual and idle/busy. We generated artificial high-level contexts because controlling the raw data directly was difficult. For example, we created a context called "a lot of phone calls," which we used in place of the phone-call log data. We randomly selected two landmarks from each of the 30 days—one from the morning and the other from the afternoon.

Table 1 shows the results. We excluded the landmarks related to the default place "home" and the less significant landmarks from the main landmark set. The false-positive error of the "usual/idle" condition was high and precision was low, because the "usual"' condition included many duplicate places and landmarks. The false-positive error of the "usual/busy" condition was low because the condition included a relatively high number of landmarks that occurred frequently at the regular time such as routine SMS texts, frequent calling, and active movements. The overall recall rate was as low as 75 percent. This resulted from a lack of tuning or from hard-to-detect landmarks.

In table 1, two target objects (with few redundancies) were selected. The unusual/busy data were composed of one unusual landmark and one busy landmark.

Image generation test

To test the image-generation capability, we used a scenario in which the user went to school (listening to MP3 music), studied (with some difficulty), ate, walked, enjoyed a concert, and went out drinking. Each event had three to five possible cartoons. Figure 5a shows the cartoons describing the conditions. We didn't use the same number of cartoons for each condition because each had a different number of variations. Sometimes we needed more cartoons to show different actions for one given event. We then generated four stories (see figure 5b) and evaluated their levels of diversity and consistency in terms of event representation.

Sixteen graduate students evaluated the generated cartoons by answering four questions using a five-point scale:

Q1: These are cartoon images describing specific conditions. Please evaluate the correctness of each image given the condition (5—very correct, 4—correct, 3—average, 2—incorrect, 1—very incorrect).

Q2: Please evaluate the diversity of images given conditions (5—very diverse, 4—diverse, 3—average, 2—homogenous, 1—very homogeneous).

Q3: The four stories are composed of six cartoons, and they represent the daily life of a female student. The schedule of the day is as follows: Going to school while listening to MP3 music, studying with some difficulty, eating, walking, enjoying a concert, and going out drinking. Please rank the four stories on the basis of their measure of correctness (Give the most accurate story a 5 and the least accurate story a 1.)

Q4: Please order the four stories on the basis of the measure of fun.

Figure 6a shows the evaluation of presentational power (average 2.96) and the possibility of diverse representation of the same events (average 3.54). By selecting one cartoon randomly for each event, there are 4,500 variations ($3 \times 4 \times 3 \times 5 \times 5 \times 5$), and we randomly chose four stories (figure 6b). The correlation between questions 2 and 4 is 3.0 (positive value), indicating that funny cartoons will likely be diverse.

Evaluation using a real-life log

A female university student used her mobile phone with the logging software to evaluate the landmark detection model for 27 days. The number of sampled input contexts depended on successfully collecting GPS signals. The threshold for the landmark selection was 66 percent. We determined "correct" and "partially correct" data on the basis of the user's daily report and visualized analysis of the user's GPS log. Table 2 shows the results.

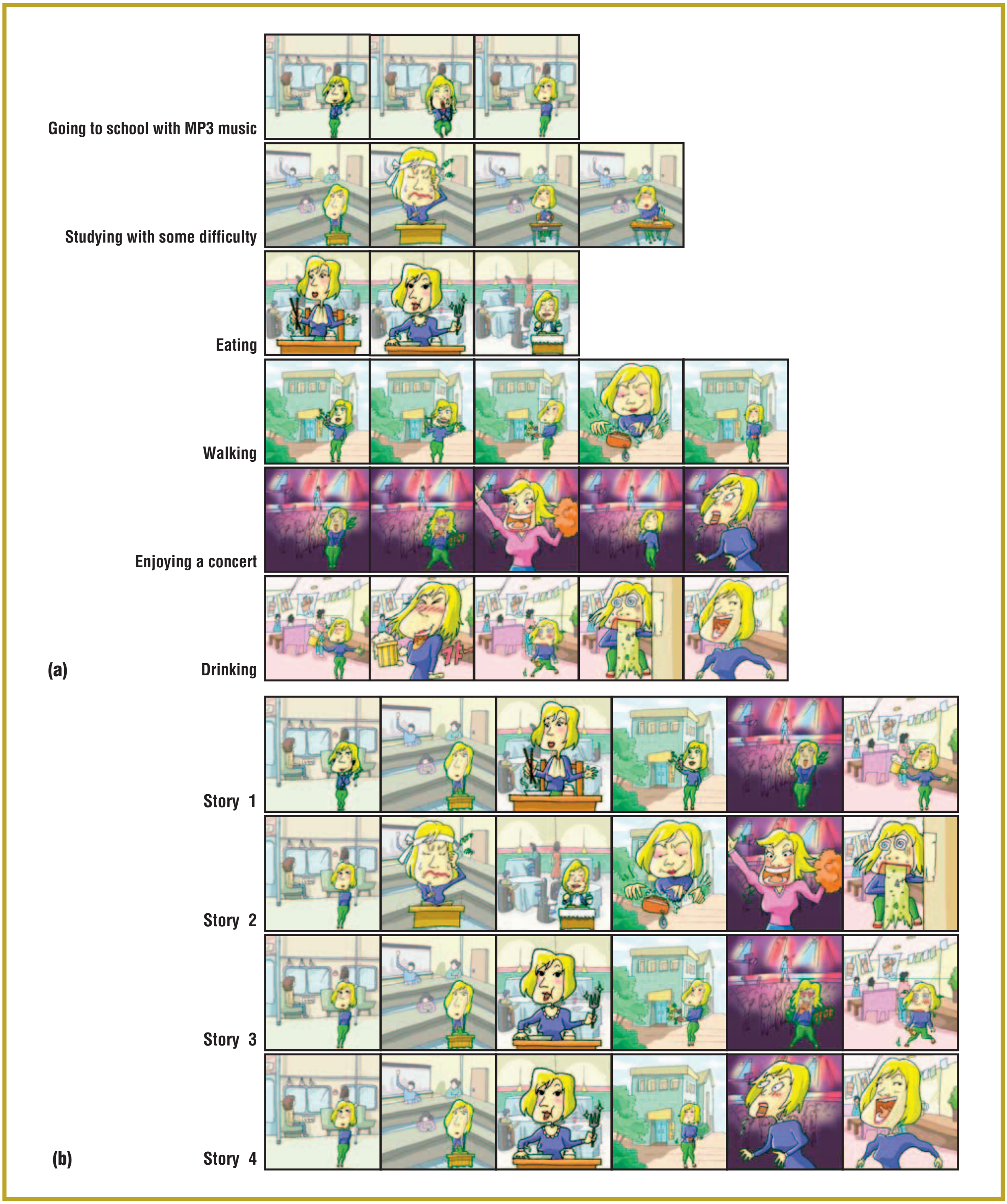


Figure 5. We used landmark examples to compose various cartoon stories: (a) the various cartoons describing the conditions and (b) the four stories we generated.

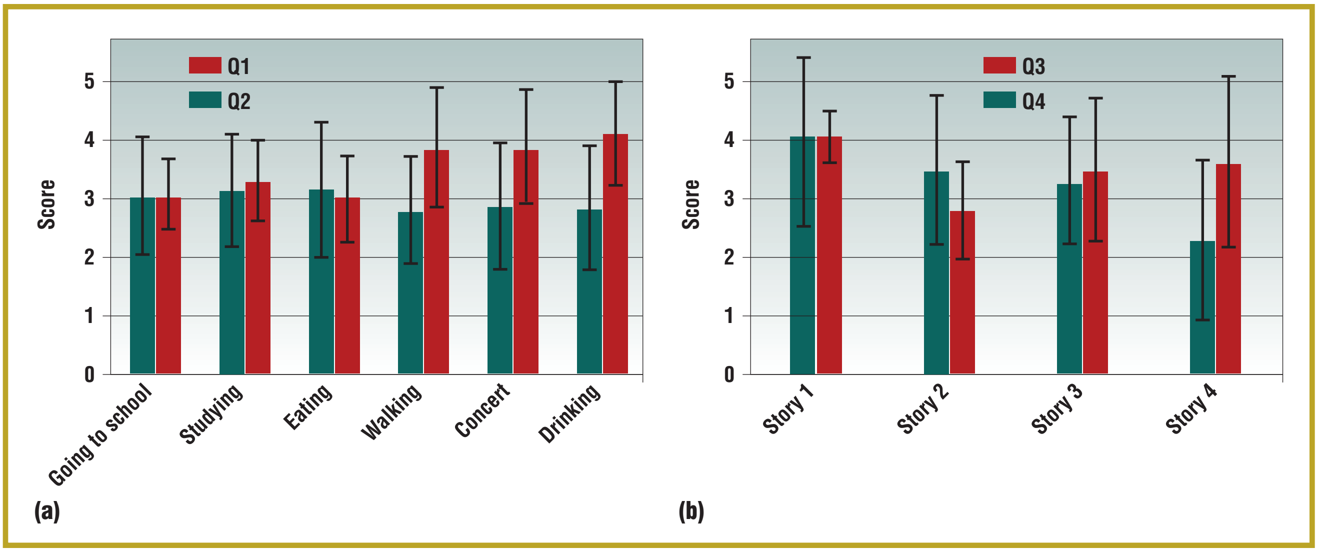


Figure 6. Evaluation of (a) the cartoons for each event in terms of their presentational power and (b) the stories and their representation of the same events. The black marks indicate the standard deviation.

TABLE 2
Accuracy of the real-world data set.

Day	N_{Con}*	N_{LM}†	N_{LM}'‡	N_{HIT}**	N_{HIT}'††	N_{ERR}‡‡	R_{HIT} (%)**	R_{HIT}' (%)††	R_{ERR} (%)‡‡
24 Feb	116	72	13	3	10	0	23.1	100.0	0.0
27 Feb	167	49	15	4	11	0	26.7	100.0	0.0
28 Feb	64	50	8	3	4	1	37.5	87.5	12.5
2 Mar	202	128	18	8	10	0	44.4	100.0	0.0
4 Mar	102	53	7	1	5	1	14.3	85.7	14.3
6 Mar	86	56	12	5	3	4	41.7	66.7	33.3
8 Mar	114	92	12	3	7	2	25.0	83.3	16.7
9 Mar	103	45	7	4	2	1	57.1	85.7	14.3
15 Mar	128	76	13	4	9	0	30.8	100.0	0.0
17 Mar	46	45	8	3	3	2	37.5	75.0	25.0
21 Mar	67	40	10	4	4	2	40.0	80.0	20.0
	1195	706	123	42	68	13	34.1	89.4	10.6

* N_{Con}: No. of input data samples
† N_{LM}: No. of detected landmarks
‡ N_{LM}': No. of detected landmarks excluding duplicated and low probability landmarks (below threshold)
** N_{HIT} and R_{HIT}: No. and ratio of exactly correct landmarks, respectively
†† N_{HIT}' and R_{HIT}': No. and ratio of approximately correct landmarks, respectively
‡‡ N_{ERR} and R_{ERR}: No. and ratio of wrong landmarks, respectively

The results show that the correct ratio (R_{HIT}) is only 34.1 percent, owing to the difficulty of interpreting the real situation given a limited daily report. However, if the detected landmarks reasonably relate to the daily report's events, we refer to them as "partially correct" and view them as sufficient for generating a meaningful diary. The partially correct ratio (R_{HIT}') is 89.4 percent.

Figure 7 presents three comic diaries generated from the real log data. Figure 7a depicts the user on 27 February 2006: she went to school late (image 1) and sent an angry SMS message because of a traffic jam (image 2). After studying, she ate lunch (images 3–5). After walking around the campus (image 6), she went home (images 7 and 8). The lunch scene is duplicated because she visited the restaurant a second time to buy coffee. Our system couldn't classify the details of her behavior in this instance—it couldn't distinguish between eating lunch and buying coffee.

Figure 7b shows the user on 5 March 2006 in her hometown (image 1). After

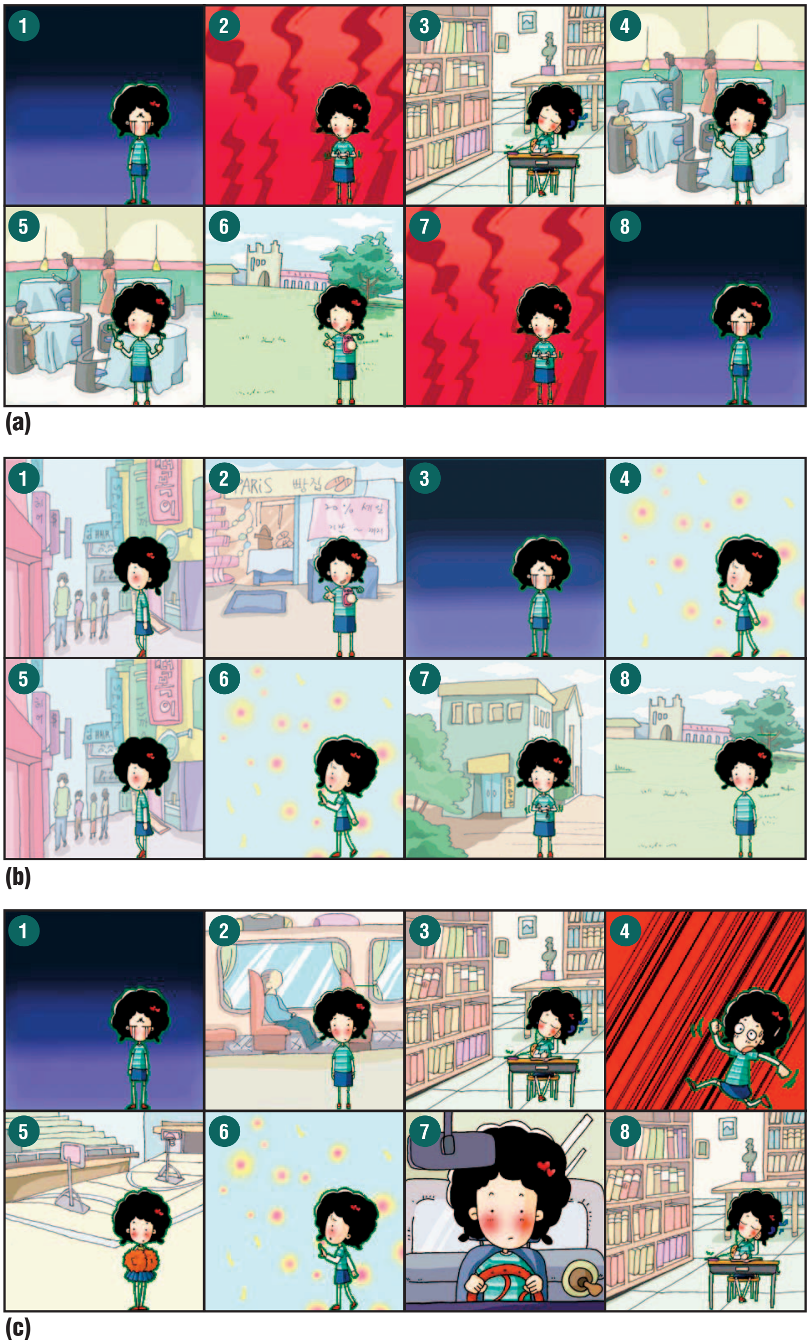


Figure 7. Comic diaries generated from real log data for (a) 27 February, (b) 5 March, and (c) 9 March.

walking around for a while (image 2), she took a bus downtown (image 3). Because the bus ride was long, the cartoon represents the ride with a person who is sad. However, our location-mapping system couldn't provide rich enough information to accurately represent her hometown, which is in the country, so it failed to generate successful output for the rest of the day.

Figure 7c shows the user on 9 March 2006. She took part in a university concert as a staff member. Image 5 correctly represents the main event. Image 4 illustrates how busy she was before the concert. Most of the images correctly represent the day's main events, even though there are some awkward images due to the limited data.

Our long-term goals are to evaluate the system using real logs collected over a long time period from real subjects and to apply our techniques to online communities of personal, virtual space such as Cyworld, MySpace, and orkut. In particular, Cyworld (of SK Communications in South Korea) has attracted many young people because it lets them easily buy items to decorate their blogs (90 percent of South Koreans in their 20s have registered with this site). We also hope to develop a more sophisticated learning algorithm for landmark detection for personalized detection models.

ACKNOWLEDGMENTS

This research was supported in part by the Samsung Advanced Institute of Technology and MIC (Korea) under ITRC IITA-2005-(C1090-0501-0019).

REFERENCES

1. Y.-H. Rhee, J.-H. Kim, and A. Chung, "Your Phone Automatically Caches Your Life," *ACM Interactions*, July–Aug. 2006, pp. 42–44.

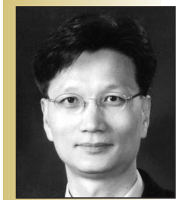
2. S. Vemuri, and W. Bender, "Next-Generation Personal Memory Aids," *BT Technology J.*, vol. 22, no. 4, 2004, pp. 125–138.

3. E. Horvitz, S. Dumais, and P. Koch, "Learning Predictive Models of Memory Landmarks," *26th Ann. Meeting Cognitive Science Soc.*, Lawrence Erlbaum Associates, 2004, pp. 583–588.

4. M. Raento et al., "ContextPhone: A Prototype Platform for Context-Aware Mobile Applications," *IEEE Pervasive Computing*, vol. 4, no. 2, 2005, pp. 51–59.

5. L. Allen, "Advanced Mobile Operating Systems: Comparative Analyses & Forecast," *The Diffusion Group*, 2005.

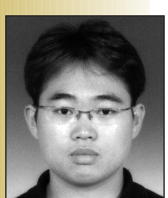
the AUTHORS



Sung-Bae Cho is a professor in Yonsei University's Department of Computer Science. His research interests include neural networks, pattern recognition, intelligent man-machine interfaces, evolutionary computation, and artificial life. He received his PhD in computer science from KAIST (the Korea Advanced Institute of Science and Technology). He's a senior member of the IEEE and a member of the Korea Information Science Society, ACM, IEEE Computational Intelligence Society, IEEE Computer Society, and IEEE Systems, Man, and Cybernetics Society. Contact him at the Dept. of Computer Science, Yonsei Univ., 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea; sbcho@cs.yonsei.ac.kr.

Kyung-Joong Kim is a researcher in the Yonsei University's Department of Computer Science. His research interests include evolutionary neural networks, pattern recognition, evolutionary computation, and artificial life. He received his PhD in computer science from Yonsei University. He's a member of the IEEE and ACM. Contact him at the Dept. of Computer Science, Yonsei Univ., 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea; kjkim@cs.yonsei.ac.kr.

Keum-Sung Hwang is a researcher at the Softcomputing Laboratory and a doctoral student in computer science at Yonsei University. His research interests include Bayesian networks and evolutionary algorithms for context-aware computing and intelligent agents. He received his MSc in computer science from Yonsei University. He's a student member of the IEEE. Contact him at the Dept. of Computer Science, Yonsei Univ., 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea; yellowg@cs.yonsei.ac.kr.



In-Ji Song is a master's student in computer science at Yonsei University. His research interests include intelligent agents and adaptive user interfaces. He received his BS in computer science from Yonsei University. Contact him at the Dept. of Computer Science, Yonsei Univ., 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea; schunya@sclab.yonsei.ac.kr.