Semantic management of multiple contexts in a pervasive computing framework

Jun-Ki Min, Sung-Bae Cho *

Department of Computer Science, Yonsei University, 262 Seongsanno, Seodaemun-gu, Seoul 120-749, Republic of Korea

**A R T I C L E   I N F O**

Keywords:
- Mobile sensors
- Multiple context awareness
- Semantic network
- Dynamic Bayesian network

**A B S T R A C T**

Mobile devices can perceive greater details of user states with the increasing integration of mobile sensors into a pervasive computing framework, yet they consume large amounts of batteries and computational resources. This paper proposes a semantic management method which efficiently integrates multiple contexts into the mobile system by analyzing the semantic hierarchy and temporal relations. The proposed method semantically decides the recognition order of the contexts and identifies each context using a corresponding dynamic Bayesian network (DBN). To sort out the contexts, we designed a semantic network using a knowledge-driven approach, whereas DBNs are constructed with a data-driven approach. The proposed method was validated on a pervasive computing framework, which included multiple mobile sensors (such as motion sensors, data-gloves, and bio-signal sensors). Experimental results showed that the semantic management of multiple contexts dramatically reduced the recognition cost.

**1. Introduction**

Mobile sensors play a significant role in context-aware computing because they can easily perceive physical and biochemical signals of users (Korel & Koo, 2010). With the development of mobile sensing technologies, many studies have attempted to recognize user contexts in various perspectives and to provide situation-appropriate services (Hong, Suh, & Kim, 2009; Roy, Gu, & Das, 2010). These systems have combined various types of sensors to more accurately detect high-level contexts. For example, vision, illumination, and noise-detection sensors are used to elucidate environmental conditions, while wearable sensors, including accelerometers and gyroscopes, capture human activities.

Two major problems may arise due to the large number of sensors contained within a system (Korel & Koo, 2010). First, substantial burdens would result for power consumption, system bandwidth, and computational complexity (Krause et al., 2005; Wang, Lin, & Annavaram, 2009). The second problem is known as the “curse of dimensionality:” as the number of sensor inputs increases, the design and training of the inference model become difficult because of the highly complicated causal relationships among sensing information and contexts (Fig. 1a). To resolve these two problems, sensor selection based on the current contexts is preferred (Fabre, Appriou, & Briottet, 2001; Hwang & Cho, 2009). An effective alternative for the algorithm point of view is to decompose a multi-contexts recognition problem into multiple identification problems (called a one-versus-all (OVA) approach) and to operate some of the relevant identification models (Hong, Yang, & Cho, 2010). As shown in Fig. 1b, the decomposed models are simpler to train than is the original, while knowing and activating the modules related to the user state are the keys to the efficient operation of the system. If the system does not know which contexts need to be recognized, it has to operate all the context modules, figuring out the highest output values among them, to make a final decision. In this case, the essential problem of a large-scale computing burden remains.

In this paper, we proposed a semantic context-manager for resource-insufficient pervasive computing framework with multiple sensors. Characteristics of high-level contexts, like human activities, can be learned from perceptual psychology, in which the activities are hierarchical, organized into various levels of abstraction and temporally constrained (Zacks & Tversky, 2001). In other words, we can usefully (but not exactly) predict the contexts according to the semantic hierarchy and sequential relations among the recent situational cues. In our proposed method, we used a semantic network (SN) where the contexts and their relations were denoted as the semantic nodes and their links, respectively. As the context management module, the SN propagated activation values from the nodes of the low-level sensory information to the high-level context nodes, and then the dynamic Bayesian networks (DBNs) recognized the activated contexts.

This paper has three main contributions. First, we revealed the semantic relations among contexts based on a hierarchical SN to achieve systematic and cost-efficient management of the inference modules. Second, we optimized the modeling process for the proposed method by integrating two different techniques: knowledge-driven and data-driven approaches. The SN was designed based on the domain knowledge, while the DBNs were trained with a genetic...
algorithm (GA) and an expectation–maximization (EM) algorithm. Third, we implemented a workable pervasive system integrating multiple sensors and context-monitoring user interfaces and used it to verify the proposed method. The rest of this paper is organized as follows. Background on context inference with multiple sensors is provided in Section 2. In Section 3, the proposed method is explained, and experimental results and discussion follow in Section 4. Conclusions are given in Section 5.

2. Context inference based on multiple sensors

Personal contexts, especially for task contexts, are closely related to user movement or gestures; and therefore, acceleration has been chosen as a basic feature in activity analysis (Lee & Mase, 2002). Recently, the integration of multiple sensors has been investigated to recognize diverse kinds of daily contexts, such as eating, typing, shaking hands, clapping hands, driving, brushing teeth, and washing dishes. Oliver, Garg, and Horvitz (2004) proposed the use of layered probabilistic representations for modeling office activities such as phone conversations and presentations. They collected real-time streams of evidence from video, audio, and computer interactions. Ermes, Parkka, Mantyjarvi, and Korhonen (2008) identified daily activities and sport activities using wearable three dimensional-accelerometers and a GPS receiver. Junker et al. used wearable inertial sensors for detecting daily and dietary activities from hand motions, such as shaking hands, turning on a light, opening a door, eating with a fork and knife, and drinking (Junker, Amft, Lukowicz, & Tröster, 2008).

Heterogeneous sensors are clearly beneficial to systems charged with determining complex contexts. However, additional techniques to manage the vast amount of sensory information are required with regard to the efficiency and effectiveness of a system. Schmidt et al. presented a layered architecture for context-aware adaptation based on collections of multimodal sensors (including a photodiode, two accelerometers, a passive infrared sensor, a temperature/pressure sensor, and a carbon monoxide gas sensor) in the TEA project (Schmidt et al., 1999). Their approach first analyzed situation cues using a self-organizing map and recognized the corresponding contexts with rule-based models. More recent studies have increasingly considered the embedding of direct awareness into mobile devices; therefore, minimization of the overall system cost is becoming important for the widespread use of such systems. Wang et al. (2009) proposed a hierarchical sensor-management scheme based on the state transition rules that turn sensors on and off. Krause et al. (2005) suggested a collection of selective sampling strategies to further reduce the number of required sensor readings and the computation duty cycles. In this paper, we focused on balancing the inference load by adapting the contexts of the environment and the user.

3. Semantic context management

Despite recent development in sensing and mobile processing technology, continuous perception of heterogeneous contexts is still challenging due to their complexities and the resource constraints of the pervasive computing environment. Our proposed method addresses these problems by modeling the contexts as different abstraction levels within a hierarchical structure of context management and context inference models. The SN effectively takes on the role of context manager by representing the relations among the low-level and high-level contexts, while DBNs are trained as the inference models that deal with uncertain and noisy sensory information. By considering the curse of dimensionality, we generate m (the number of contexts) modules in which each module identifies its corresponding context, rather than training a large model that classifies all contexts at once. Fig. 2 shows the context-awareness process of the proposed method.

At the training stage, input observations and target contexts are first defined as the variable nodes for the SN and DBNs. The semantic relations among the nodes for the SN are specified based on the knowledge-driven approach. In the data-driven approach, GA and EM algorithms train the structures and parameters of the DBNs, respectively, which are impractical for manual design because of their complexity. For on-line recognition, the SN transmits the activation values in three steps:

1. Propagating the states of the sensors (low-level context nodes) to the high-level context nodes.
2. Propagating the previous states of the inference engines to the high-level context nodes.
3. Propagating the states of the high-level context nodes for each other.

The DBNs are then operated in descending activation value order until the given context is identified.

3.1. Designing a semantic network

In general, a context can be identified based on several sources of sensory information perceived from the user and environment
as described in Schmidt and Laerhoven (2001): “User sleeps = [It is dark, room temperature, silent, type of location is indoors, time is nighttime, user is horizontal, specific motion pattern, absolute position is stable].” Moreover, the current situation is closely related to the most recent activities and states in the contexts of the semantic relations among them, such as causation or prevention. Previous studies have adopted the ontology of providing common terms as building primitives for activity definitions since the ontology can capture and encode rich domain knowledge and heuristics in understandable and easily processed ways to a machine (Chen & Nugent, 2009). In this paper, a SN is hierarchically designed for modeling the relations among the sensing information and high-level contexts.

A SN is a directed graph that depicts semantic relations (links) among concepts or common sense knowledge (nodes) Roy et al., 2010. Since representing all variables is impractical, only relevant context variables are designed with the appropriate levels, as shown in Table 1. Some of the low-level contexts, like user states, would be regarded as high-level contexts in other applications; however, mainly task-contexts are considered as high-level in this paper.

Six relation types are defined as shown in Fig. 3, in which the FollowedBy, HardToFollow, and Prerequisite relations denote connections among the high-level contexts, and the remaining relations, such as Cause, Prevent, and PartOf, represent the links between the low-level contexts and the high-level contexts (see Fig. 3).

For example, “A user takes a rest or sleep when he/she gets tired” can be denoted as the Cause relation, and “[A user is outside, he/she cannot work with a computer” is represented using the Prevent relation. PartOf indicates the event components. For example, the “Meeting” context consists of several low-level contexts including location (Meeting room) and object (a Beam projector). FollowedBy and HardToFollow represent temporal constraints of the high-level contexts: “A user might want to go to the restroom after eating or drinking,” and “A user cannot attend a meeting while doing exercise.” The Prerequisite represents the “part of” conditions among high-level contexts like “A user shaking hands and clapping during the meeting.” Practical examples for construction of a SN will be given in Section 4.

### 3.2. Construction of dynamic Bayesian networks

A DBN, which is adopted as the context inference model in our system, is a graphical model that encodes probabilistic relationships among time-series variables of interest as a set of edges between nodes. DBNs have been employed in many studies of context-aware computing because of their advantageous features, such as the capability to represent a large degree of joint probability; handle uncertainties; and capable of using both of knowledge-driven and data-driven approaches for modeling (Hong et al., 2010).

Since time-series variables often lead to a complexity problem, the proposed method applies a GA for determining the optimal structure of the large number of variables (Ross & Zuviria, 2007)

### Table 1

<table>
<thead>
<tr>
<th>Level Type</th>
<th>Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor state Object that interacts with the user</td>
<td>Sensor: HeadMotion, Sensor: ArmMotion, Sensor: HandMotion MobilePhone, Projector</td>
</tr>
<tr>
<td>High-level (task context) High-level user context</td>
<td>Exercising, Driving, Drinking, Resting, Meeting, KeyboardTyping, MouseScrolling, Working, Eating, Urinating, ToothBrushing, Calling, Sleeping, HandShaking, HandClapping</td>
</tr>
</tbody>
</table>
Fig. 3. Six relations among the semantic nodes.

Fig. 4. (a) A constraint for the link directions of the DBNs where $A_p$ = past context, $A_c$ = current context, $E_p$ = past evidence, and $E_c$ = current evidence. (b) Chromosome encoding for the evolutionary optimization of a DBN structure. (c) Examples of the evolved structures of DBNs.

Fig. 5. The overall process of the three-phase activation propagation.
and trains the parameters (conditional probability tables, CPTs) using an EM algorithm with the training set. A GA is a search technique based on the evolution of a biological system that uses a fixed population size of chromosomes representing potential solutions that iteratively evolves over generations. According to the common sense of the cause-effect relationships, the current situation $A_i$ results from the past states $A_p$ and indications $E_p$ and causes the current observations $E_c$. We use this common-sense method as a constraint for the directions of the DBN links where a chromosome composed of a fixed-length binary string denotes the links, as shown in Fig. 4.

First, the GA process randomly generates an initial population and evaluates the fitness values of the chromosomes by calculating the context recognition accuracy of the training set. It then selects and evaluates the fitness values of the chromosomes by calculating the fitness value $v$, which is empirically determined for each relation, as shown in Fig. 3; and $l_j$ is either one or zero when the input values either do or do not satisfy a set of rules, respectively.

In the second phase, the previous DBN output for the $i$th high-level context, probability $p_i$, is accumulated to $a_i$ as follows:

$$ a_i = a_i + \sum_{k=1}^{m} w_i a_k \text{Ind}(i, k), $$

where $l_{i-1, ..., i}$ is the activation state of the $j$th low-level context. The weight value $w_i$ is empirically determined for each relation, as shown in Fig. 3; and $l_j$ is either one or zero when the input values either do or do not satisfy a set of rules, respectively.

In the second phase, the previous DBN output for the $i$th high-level context, probability $p_i$, is accumulated to $a_i$ as follows:

$$ a_i = a_i + \sum_{k=1}^{m} w_i a_k \text{Ind}(i, k), $$

where $l_j$ is the activation state of the $j$th low-level context. The weight value $w_i$ is empirically determined for each relation, as shown in Fig. 3; and $l_j$ is either one or zero when the input values either do or do not satisfy a set of rules, respectively.

In the second phase, the previous DBN output for the $i$th high-level context, probability $p_i$, is accumulated to $a_i$ as follows:

$$ a_i = a_i + \sum_{k=1}^{m} w_i a_k \text{Ind}(i, k), $$

where $l_j$ is the activation state of the $j$th low-level context. The weight value $w_i$ is empirically determined for each relation, as shown in Fig. 3; and $l_j$ is either one or zero when the input values either do or do not satisfy a set of rules, respectively.

In the second phase, the previous DBN output for the $i$th high-level context, probability $p_i$, is accumulated to $a_i$ as follows:

$$ a_i = a_i + \sum_{k=1}^{m} w_i a_k \text{Ind}(i, k), $$

where $l_j$ is the activation state of the $j$th low-level context. The weight value $w_i$ is empirically determined for each relation, as shown in Fig. 3; and $l_j$ is either one or zero when the input values either do or do not satisfy a set of rules, respectively.

In the second phase, the previous DBN output for the $i$th high-level context, probability $p_i$, is accumulated to $a_i$ as follows:

$$ a_i = a_i + \sum_{k=1}^{m} w_i a_k \text{Ind}(i, k), $$

where $l_j$ is the activation state of the $j$th low-level context. The weight value $w_i$ is empirically determined for each relation, as shown in Fig. 3; and $l_j$ is either one or zero when the input values either do or do not satisfy a set of rules, respectively.

In the second phase, the previous DBN output for the $i$th high-level context, probability $p_i$, is accumulated to $a_i$ as follows:

$$ a_i = a_i + \sum_{k=1}^{m} w_i a_k \text{Ind}(i, k), $$

where $l_j$ is the activation state of the $j$th low-level context. The weight value $w_i$ is empirically determined for each relation, as shown in Fig. 3; and $l_j$ is either one or zero when the input values either do or do not satisfy a set of rules, respectively.
4. Experiments

4.1. Sensor platform and system implementation

In this paper, we considered 15 daily user contexts in an office environment, as listed in Table 1: (Exercising, Driving, Drinking, Resting, Meeting, KeyboardTyping, MouseScrolling, Working, Eating, Urinating, ToothBrushing, Calling, Sleeping, HandShaking, HandClapping).

The sensor platform used in this paper is shown in Fig. 6a, and it integrates the following components: two data gloves; an armband on a right arm; five sets of accelerometers and gyroscopes on the head, both arms and both wrists; and a notebook to collect the sensor data and run the implemented modules. Data gloves (5DT Inc., http://www.5dt.com) capture the degree of finger bending from five channels on the left hand and 14 channels on the right hand. Bodymedia’s armband (http://www.sensewear.com/BMS/solutions_bms.php) senses the user’s bio-signals related to emotion and stress, including skin temperature, heat flux, and Galvanic skin response. To track human motion, we use an Xsens Xbus Kit (http://www.xsens.com), which includes a three-axis accelerometer and a three-axis gyroscope for each sensor node. The system segments the incoming sensory data using a sliding-window, and preprocesses the segmented signals via smoothing, normalization, and quantization.

Fig. 7. (a) Semantic network designed for the proposed system and (b) an example of the activation result where the inactive (activation value <0.5) nodes are denoted as small nodes.

4. Experiments

4.1. Sensor platform and system implementation

In this paper, we considered 15 daily user contexts in an office environment, as listed in Table 1: (Exercising, Driving, Drinking, Resting, Meeting, KeyboardTyping, MouseScrolling, Working, Eating, Urinating, ToothBrushing, Calling, Sleeping, HandShaking, HandClapping).

The sensor platform used in this paper is shown in Fig. 6a, and it integrates the following components: two data gloves; an armband on a right arm; five sets of accelerometers and gyroscopes on the head, both arms and both wrists; and a notebook to collect the sensor data and run the implemented modules. Data gloves (5DT Inc., http://www.5dt.com) capture the degree of finger bending from five channels on the left hand and 14 channels on the right hand. Bodymedia’s armband (http://www.sensewear.com/BMS/solutions_bms.php) senses the user’s bio-signals related to emotion and stress, including skin temperature, heat flux, and Galvanic skin response. To track human motion, we use an Xsens Xbus Kit (http://www.xsens.com), which includes a three-axis accelerometer and a three-axis gyroscope for each sensor node. The system segments the incoming sensory data using a sliding-window, and preprocesses the segmented signals via smoothing, normalization, and quantization.

Fig. 7. (a) Semantic network designed for the proposed system and (b) an example of the activation result where the inactive (activation value <0.5) nodes are denoted as small nodes.
For the inputs to the SN, we use several state-based features, as listed in Table 1, where the states of the sensors are extracted based on the rules and thresholds. The missing semantic variables, such as environment state and object state, are tagged by the user via the monitoring interface. In the case of DBNs, the preprocessed signals from the sensors are quantized into three symbols by using a k-means clustering algorithm.

We implemented a monitoring interface that can determine the state of the system by displaying the sensing signals and context recognition results, as shown in Fig. 6c. The sensing information

---

**Table 1**

| Activity | # of current evidencenodes (|Ec|) | # of past evidencenodes (|Ep|) | # of conditional probabilities (SEp|Ep|/C2 + (SA/C2 SEc) |Ec|) |
|---|---|---|---|---|
| Driving | 49 | 7 | 37 + 2 | (2 + 3) × 49 |
| HandClapping | 59 | 2 | 37 + 2 | (2 + 3) × 59 |
| ToothBrushing | 55 | 10 | 37 + 2 | (2 + 3) × 55 |
| Drinking | 73 | 8 | 37 + 2 | (2 + 3) × 73 |
| Sleeping | 29 | 8 | 37 + 2 | (2 + 3) × 29 |
| Eating | 4 | 3 | 37 + 2 | (2 + 3) × 4 |
| KeyboardTyping | 61 | 1 | 37 + 2 | (2 + 3) × 61 |
| MouseScrolling | 75 | 11 | 37 + 2 | (2 + 3) × 75 |
| Meeting | 9 | 3 | 37 + 2 | (2 + 3) × 9 |
| Urinating | 3 | 9 | 37 + 2 | (2 + 3) × 9 |
| Exercising | 6 | 13 | 37 + 2 | (2 + 3) × 3 |
| Resting | 20 | 5 | 37 + 2 | (2 + 3) × 20 |

**Table 2**

Correlations among the contexts (+: Positively correlated, -: Negatively correlated, and -: Weakly correlated).

**Table 3**

Complexities of the inference models (SEp/Ep: # of states in a past evidence node, SEc/Ep: # of states in a current evidence node, and SA: # of states in the current context node).

**Fig. 8.** Evolutionary trajectories for the short-term context modules with the exception of Driving (long-term) and Urinating (short-term). The fitness value denotes the accuracy of the training set.
is shown on the left window; and the center window visualizes the current states of the modules, such as the activation of the contexts and the inference sequences. The activation value is shown as a bar-type graph on the right side of the corresponding context node; the node is denoted as a brighter color when it is activated (activation value >0.5).

4.2. Model estimation

In order to represent the semantic relations among the contexts using the SN, we first defined the domain space to be considered (as shown in Table 1) and then analyzed the cause-effect relations among the contexts based on the domain knowledge (see Table 2).

### Table 4

<table>
<thead>
<tr>
<th>Selection method</th>
<th># of evaluations</th>
<th>Inference time (ms)</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>15</td>
<td>539.8</td>
<td>94.5 ± 0.8</td>
</tr>
<tr>
<td>Proposed method</td>
<td>1.55</td>
<td>66.4</td>
<td>91.7 ± 1.0</td>
</tr>
</tbody>
</table>

### Table 5

<table>
<thead>
<tr>
<th>Time</th>
<th>Scenario</th>
<th>Corresponding activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>6:00</td>
<td>Wake up in the morning</td>
<td>ToothBrushing</td>
</tr>
<tr>
<td>8:00</td>
<td>Drive a car to go to work</td>
<td>Driving</td>
</tr>
<tr>
<td>9:00</td>
<td>Work with a computer at the office</td>
<td>KeyboardTyping, MouseScrolling, and Working</td>
</tr>
<tr>
<td>10:00</td>
<td>Meet colleagues for a business</td>
<td>HandShaking and Meeting</td>
</tr>
<tr>
<td>11:00</td>
<td>Meeting (missed call occurred)</td>
<td>HandClapping while Meeting</td>
</tr>
<tr>
<td>12:00</td>
<td>After the meeting, check the missed</td>
<td>Calling</td>
</tr>
<tr>
<td></td>
<td>call</td>
<td></td>
</tr>
<tr>
<td>12:00</td>
<td>Go to lunch</td>
<td>MealEating</td>
</tr>
<tr>
<td>13:00</td>
<td>Go back to the office and take a rest</td>
<td>ToothBrushing and Sleeping</td>
</tr>
<tr>
<td>14:00</td>
<td>Work with a computer at the office</td>
<td>Working</td>
</tr>
<tr>
<td>16:00</td>
<td>Drink a coffee and go to the rest</td>
<td>Resting, WaterDrinking, and Urinating</td>
</tr>
<tr>
<td></td>
<td>room</td>
<td></td>
</tr>
<tr>
<td>16:00</td>
<td>Go back to work</td>
<td>Working</td>
</tr>
<tr>
<td>18:00</td>
<td>Leave the office and go to the gym</td>
<td>Exercising</td>
</tr>
</tbody>
</table>

**Fig. 9.** Degrees of activation in the 1-day simulation. Semantically connected contexts are drawn on each graph.
The correlations between the target contexts and the user’s physical movements are also described in the table. Note that five sets of accelerometers and gyroscopes sensed the movements of a user’s head and arms, and the data-gloves captured the shape of the hands.

Based on the variables and their relations, we designed a SN that consisted of 31 semantic nodes (15 high-level contexts and 16 low-level contexts) and 48 relations (seven Causes, six Prevents, 18 PartOfs, seven FollowedBy, and four Pre-requisites), as illustrated in Fig. 7a. After the SN propagated the activation values for an arbitrary input set, some of the semantic nodes had higher activation values, as shown in Fig. 7b, indicating that the corresponding context modules needed to be operated.

In order to determine the optimized structure for the context-inference DBNs, GA parameters were set as 50 individuals, a 0.8 selection rate, 0.8 crossover rate, and 0.1 mutation rate, and the evolutionary cycle was repeated through 50 generations. For the evolved DBNs, the average numbers of the conditional probability, $E_c$ and $E_p$ and were 273,516, 35, and 7, respectively, as shown in Table 3.

Generally, Drinking, KeyboardTyping, MouseScrolling, Eating, Urinating, ToothBrushing, Calling, HandShaking, and HandClapping can be regarded as short-term contexts, while the rest are long-term states that would be more affected by previous sensory observations. As presented in Table 3, the short-term DBNs (with the exception of Driving and Urinating) evolved to be sensitive to the current observations (with a larger number of $E_c$); their evolutionary trajectories are shown in Fig. 8.

Although the context modules were optimized based on the GA, they still have considerable complexities; therefore, operating all modules for every sliding-windowing-segment was difficult in the pervasive environment. As a consequence, the selective inference approach of the proposed method is essential to systems based on multiple sensors and contexts.

4.3. Model evaluation and application

A male participant wearing the sensor platform performed the goal activities ten times over a period of two minutes to train the system. We also collected a validation dataset from the user, which included a total of 5645 samples. The performance of the proposed method was compared with the winner-takes-all DBNs, which operates all the DBNs and recognizes a current context as the highest probability model without using the context manager. As shown in Table 4, the proposed method ran eight times faster than did the winner-takes-all DBNs, while losing only 3% of its accuracy.

In addition to the off-line evaluation, the proposed method was verified via the on-line application. The user equipped the system during his daily activities, which are listed in Table 5. He woke in the morning and drove his car to the office. At the morning meeting, he shook hands with the attendees and clapped his hands for the presenter. After lunch, he took a nap and went out for exercise before the end of the day. Fig. 9 shows the changes in the values of the activation states with times, where the activation results roughly reflected the user’s situation.

We also confirmed that the activations for the contexts of the causal (FollowedBy) or inclusion (Pre-requisite) relationship changed similarly to each other, while the contradictory contexts of the HardToFollow relationship showed negative correlations in their changing activation values. Meanwhile, the long-term contexts had higher activation values than did the short-term ones since they aggregated the activations over a relatively long time.

Fig. 10 shows the number of inference executions and the time of the proposed method. When using the conventional system (without the selective inference approach), all of the 15 models have to be evaluated to recognize a context. For the proposed method, however, an average of 1.5 models operated per segment.

5. Conclusions

One important aspect of context-awareness is the construction of efficient models to allow operation in a mobile environment. Recognizing all the contexts without considering the user situation is time-consuming and thus impractical, especially for targeting multiple contexts with heterogeneous sensors. This paper proposed a semantic management method with multiple context modules. In order to manage the context modules, the method activated them
using a semantic network that represents the relations among the high-level and low-level contexts based on the knowledge-driven approach. DBNs were employed as the inference modules, and a genetic algorithm was used to optimize their structures. The proposed method was implemented on a platform with multi-modal sensors, and experimental results showed that the method was capable of reducing the complexity of a context-aware system.

Since we focused only on modeling the semantic manager and implementing a workable system, there are still many experimental and analytical issues to be considered in future work, such as validating the performance with challenging datasets, analyzing evolutionary results, and comparing the proposed method with conventional approaches. The proposed method also can be extended to manage the sensor modules using its semantic activation scheme, which would be useful for resolving battery issues in mobile environments.

Acknowledgments

This research was supported by the Original Technology Research Program for Brain Science through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0018948). The authors would like to thank Mr. S.-I. Yang and Dr. J.-H. Hong for their help in implementing the system in this paper.

References


