Expert Systems with Applications 36 (2009) 11675–11681

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Recommendation system using location-based ontology on wireless internet: An example of collective intelligence by using 'mashup' applications

YoungHoon Yu, JiHyeok Kim, Kwangcheol Shin *, Geun Sik Jo

School of Computer and Information Engineering, Inha University, 253 Yonghyun-dong, Nam-gu, Incheon (402-751), Republic of Korea

ARTICLE INFO

Keywords:
LBS
Recommendation system
Collective intelligence
Ontology
Mashup

ABSTRACT

Location-Based Service (LBS) is considered as a key component of upcoming ubiquitous environments. A recommendation system based on LBS is expected to be an important service in ubiquitous environments, and most hardware technologies such as location estimation of a user by using Global Positioning System (GPS), as well as hi-speed internet access through cell phones, are currently supported. However, in terms of software, most services are driven and supported by a LBS service provider only. Consequently, lack of participation of users may occur in mobile environments. In this study, we suggest a LBS knowledge base inference platform with ontology which considers the current location and available time of users. Our knowledge base supports user participation as collective intelligence. We mashed up Open Application Programming Interface (OpenAPI) for scalable implementation of the system. Through experiments, we show that a user can build up his/her knowledge base, and by using this information, the system recommends to other users appropriate information that matches the user's condition and profile through inference.

1. Introduction

Contemporary lifestyles have been transformed by mobile phones and Internet communication. Increasing use of mobile phones and Personal Digital Assistants (PDA) has allowed people to access the Internet wherever they are and whenever they want. From the Internet they can obtain information about events such as cinema, concerts, and parties as well as information on places such as city maps, restaurants, museums, and hospitals (Steiniger, Neun, & Edwardes, 2006). Location-Based Services (LBS) are information and entertainment services that are accessible via mobile devices through a mobile network and that make use of the geographical position of the mobile device.

With the appearance of Web 2.0, users are no longer merely information consumers but play roles as participators and even information providers (Anderson, 2007). Users can own data on a Web 2.0 site and exercise control over that data. This stands in contrast to old traditional websites, the sort that limited visitors to viewing and whose content could only be modified by the site's owner. To follow the current trend, service providers are starting to provide the OpenAPI (Open Application Programming Interface). Open API enables websites to interact with each other by using SOAP, Javascript, or other web technologies and users can produce their own solutions by using these technologies. These web applications, referred to as 'mashups', combine data from more than one source, such as Web service or OpenAPI providers, into a single integrated service or tool.

In this study, we present a LBS knowledge base inference system platform that supports user participation as collective intelligence. In the proposed system a user can build up his/her knowledge base by using a mobile device. This information can be used to recommend to other users appropriate information which corresponds with the user conditions and user profile on the basis of inference.

This paper is organized as follows: In Section 2, we review related works. The proposed system is outlined in Section 3. In Section 4, we present the details of the system and experimental results are given in Section 5. We conclude with directions for future work in Section 6.

2. Related works

While there is a considerable body of research related to LBS, most studies are focused on localization of mobile devices for LBS or methodologies for LBS implementations. Jose, Moreira, and Rodriues (2003) investigated the use of location-based service selection as an open and generic approach to support the development of many types of location-dependent systems. To enable such an approach, the novel AROUND architecture has been proposed as a service location infrastructure capable of allowing applications to

* Corresponding author. Tel.: +82 32 872 7445; fax: +82 32 874 1435.
E-mail address: kcshin@inha.ac.kr (K. Shin).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.03.017

select Internet services that are specifically associated with their current location.

Costa-Requena, Haitao, and Espigares (2002) proposed a novel mechanism to create a common LBS universally understood by different terminals or systems. The LBS protocol works in wired and wireless IP networks for providing LBS and reuses existing network elements and protocols.

Kim et al. (2005) focused on how LBS applications obtain integrated, dynamic, and sensitive content on different domains. They applied Semantic Web technologies to resolve these issues and designed an architecture to combine Semantic Web technologies with LBS and implemented a prototype to describe the Semantic LBS.

Tilson, Lyytinen, and Baxter (2004) presented a framework to help design and assess the potential of LBS and identify a viable strategy for service positioning. Individual service concepts are passed through two stage filters to simplify later analysis. These filters incorporate our growing knowledge of the social and behavioral characteristics of location-aware services along with technical and commercial considerations. They also proposed scenario analysis and other techniques for mitigating risks in decision making under the uncertainty that surrounds these services.

McDiarmid and Irvine (2004) presented a protocol that allows users to anonymously receive location-based information, within certain restrictions. The protocol can be used to guarantee user anonymity for user-pulled location-based services with a handset-based location measurement system.

In this study, we present a knowledge base recommendation LBS platform that supports user participation as collective intelligence.

3. Architecture of LBS system for recommendation

In this section, we describe the architecture of the proposed system. In our system, as Fig. 1 shows, users can register recommending information on shops into a knowledge base through a Web

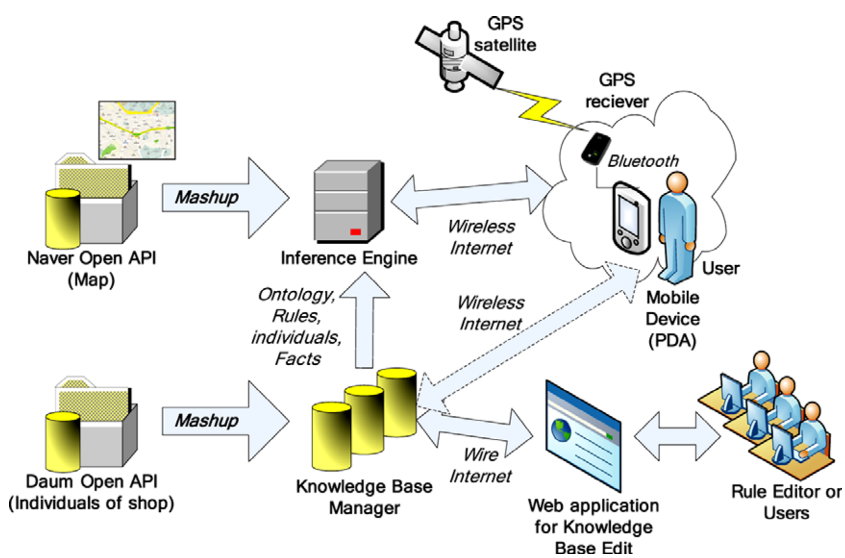

Fig. 1. System architecture.

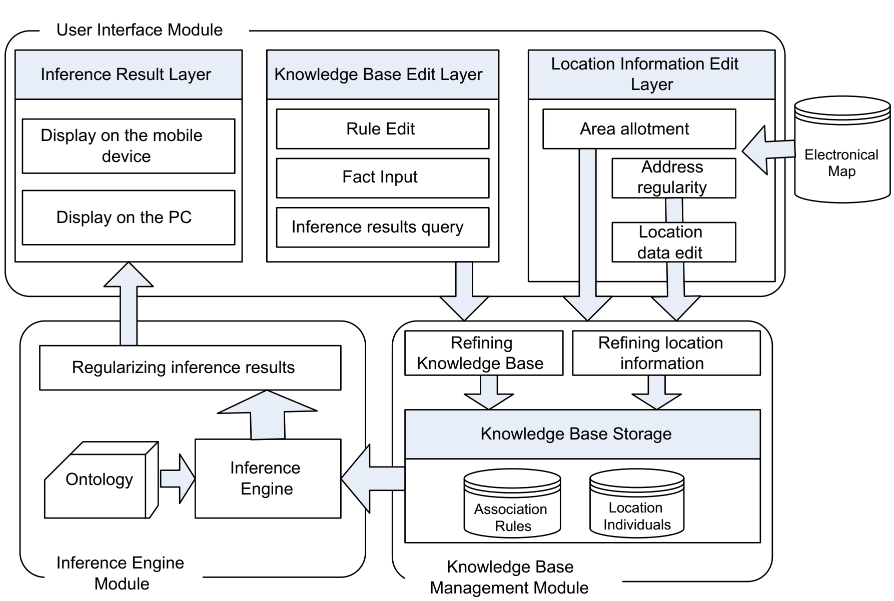

Fig. 2. The detailed architecture of our recommendation system.

application for a knowledge base editor. We mash up shop information of the DAUM Open API (Daum OpenAPI, 2008). Since many users participate in building a knowledge base, the knowledge base provides more plenteous information than that of information providers. A user can transmit information on his/her location to an inference engine through a GPS installed mobile device. The inference engine then infers recommendations using the knowledge base and mashing up the map data that is provided by Naver Corp. (Naver OpenAPI, 2008). Finally, inference results are transmitted to the user through wireless Internet.

Fig. 2 shows the detailed architecture of the system, which consists of three main modules: a user interface, a knowledge base, and an inference engine.

3.1. User interface module

By using the user interface module, a user references the inference results, and also edits the knowledge base and location information. 'Edit location information' means that a user creates or modifies the information on shops, as shown in Fig. 3. Users recog-

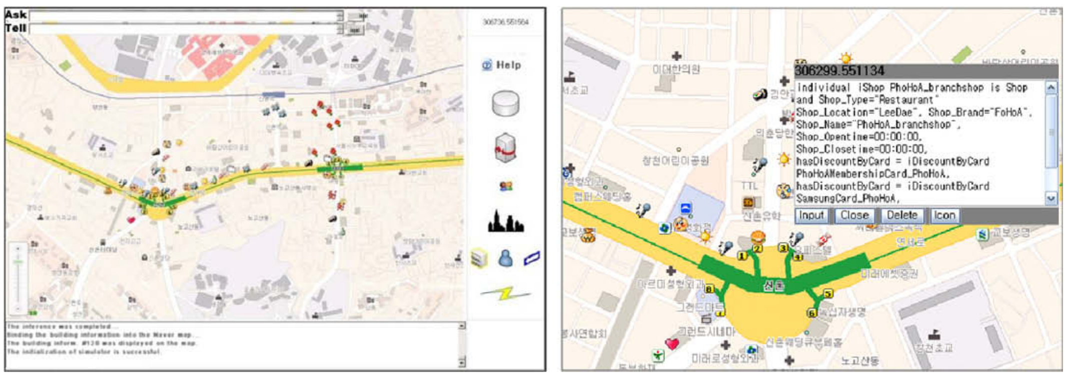

Fig. 3. Edit location information.

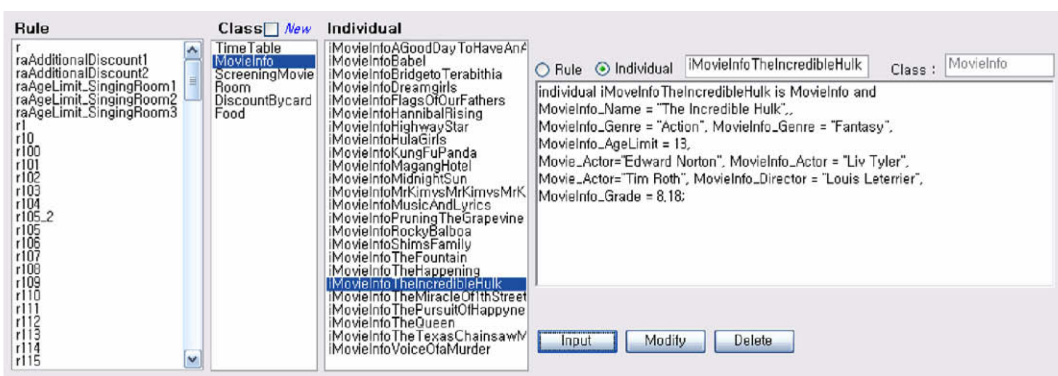

Fig. 4. Knowledge base editor.

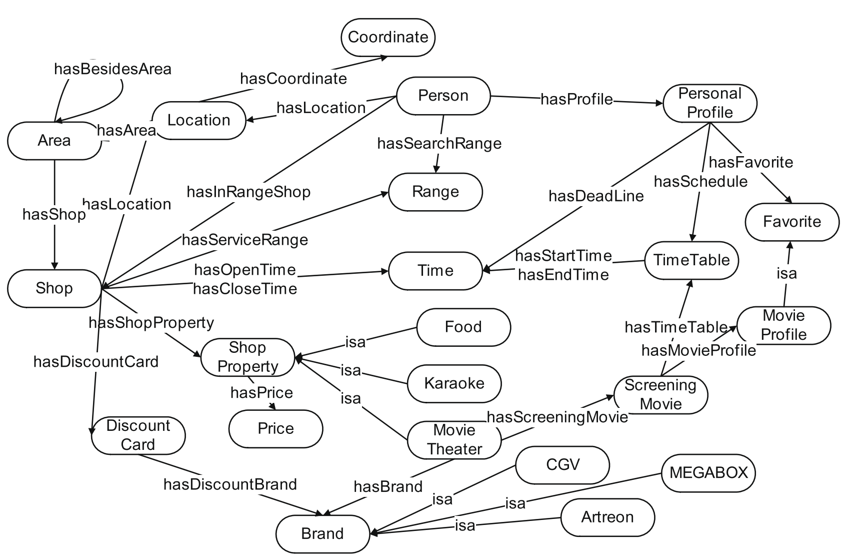

Fig. 5. Ontology structure.

nize his/her current position through GPS installed mobile devices and participate in building a knowledge base by using the knowledge base editor. Fig. 4 shows the knowledge base editor where a user modifies the contents. The knowledge base consists of rules, classes, and individuals, each of which are described in detail in Section 4.

3.2. Knowledge base manager module

The knowledge base manager plays the role of a broker that stores information on shops created or modified by users into a knowledge archive after refining the information. 'Refining' means syntactic verification of user input data and normalization of location data of user input to an appropriate coordination system that is used in our system. The knowledge archive stores information and association rules on shops. The knowledge base manager also transfers data from the knowledge archive to an inference engine.

Table 1
Association rules for location information.

Rule Shops_in_user_search_area **is**
  if Person(?User) **and** hasArea(?User, ?User_searh_area) **and**
    hasBesidesArea(?User_searh_area, ?Nearby_area_of_user) **and**
    hasShop(?Nearby_area_of_user, ?Nearby_shops) **and**
    hasServiceRange(?Nearby_shops, ? Shops_in_user_search_area) **or**
    hasInRangeShop(?User, ? Nearby_shops)
  then hasServiceShop(?User, ? Shops_in_user_search_area) **and**
    hasServiceShop(?User, ? Nearby_shops);

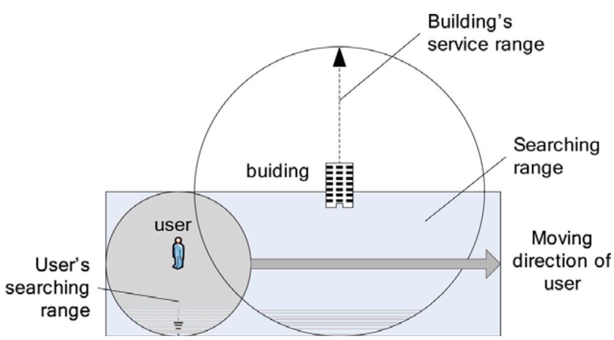

Fig. 6. Searching range of a shop and a user.

3.3. Inference engine module

We implemented a rule based inference engine that consists of static data such as ontology, rules, individuals, and facts. Static data is information or rules on shops and relationships between classes. Inference is executed with these static data and dynamic data such as the user's current position and user preferences, which are found in the user profile. The inference results, refined to be read by the user, are sent to the user interface module. A user can reference inference results on the user interface module. We explain the processes of inference in detail in the following section.

4. Details on knowledge base

In this section, we present details on the knowledge base, the key component of our system. We implemented our inference engine using Bossam (Jang & Sohn, 2004). Since Bossame is a RETE algorithm-based forward chaining inference engine, we use Buchingae (Jang, 2005) to compose association rules in Bossam between user profiles and users' current positions. To build up an ontology and the relationship between classes, we use protégé ontology editing language (Protégé, 2008).

4.1. Ontology building

Location-based recommendation service needs location and time information additionally in comparison with general recommendation services. Hence, we define an ontology as shown in Fig. 5. The center and left side of Fig. 5 represents the ontology on the location information of a user and a shop and the right side represents the ontology on time information. The lower part of the figure represents an ontology on other attributes of the shop.

Shop class has attributes such as business hours and location and as sub attributes it has the type of shop such as food, karaoke,

Table 2
Association rule for time table of movie list and available time of user.

Fact fTime **is** CurrentDateTime(2008-06-24T12:30:00);
Fact fAvailableTime **is** AvailableTime(PT5H);
Rule RecommenedMovieTheater **is**
  if hasTimeTable(?RunningMoive, ?TimeTableOfMovies) **and**
    TimeTable_StartTime(?TimeTableOfMovies, ?StartingTimeOfMovie) **and**
    [func:after(?StartingTimeOfMovie,?CurrentDateTime) = true] **and**
    [func:before(?StartingTimeOfMovie,[?CurrentDateTime + ?AvailableTime]) = true]
  then FinalRecommendationOfTheater(?Theater, ?RunningMovie,
    ?StartingTimeOfMovie, ?RecommendedMovie);

Table 3
The individuals of theater.

Individual iShopArtreon_Sinchon **is** Shop **and**
  Shop_Type = "MovieTheater",
  Shop_Location = "Sinchon", Shop_Brand = "Artreon",
  Shop_Name = "Artreon_Sinchon",
  Shop_Rating = 7, MovieTheater_TicketPrice = 7000,
  Shop_OpenTime = 06:00:00, Shop_CloseTime = 24:00:00,
  hasScreeningMovie = iScreeningMovieTheIncredibleHulk,
  hasScreeningMovie = iScreeningMovieKungFuPanda
  hasScreeningMovie = iScreeningMovieTheHappening,
  hasDiscountByCard = iDiscountByCardBCPlusCard_Artreon,
  hasServiceRange = 300, Position_X = 306742, Position_Y = 551284;

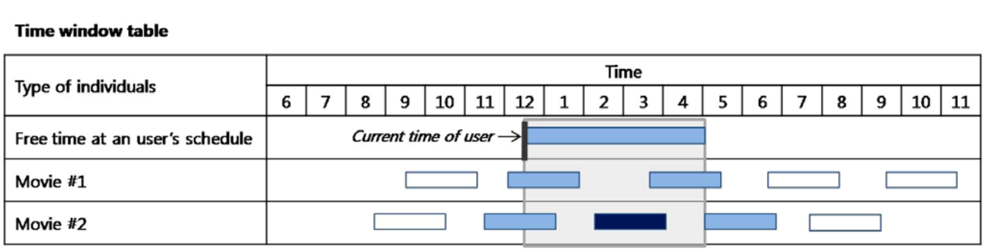

Fig. 7. Example of movie schedule and available time of user.

Table 4
The movie attribute of movie individuals.

Individual iMovieInfoTheIncredibleHulk **is** MovieInfo **and** MovieInfo_Name = "The Incredible Hulk", MovieInfo_Genre = "Action", MovieInfo_Genre = "Fantasy", MovieInfo_AgeLimit = 13, MovieInfo_Actor = "Edward Norton", MovieInfo_Actor = "Liv Tyler", MovieInfo_Actor = "Tim Roth", MovieInfo_Director = "Louis Leterrier", MovieInfo_Grade = 8.18;

Table 5
The time attribute of movie individuals.

Individual iTimeTableTheIncredibleHulk **is** TimeTable **and** TimeTable_StartTime = 2008-06-24T14:00:00, TimeTable_EndTime = 2008-06-24T15:54:00;

Table 6
The discount card attribute of movie individuals.

Individual iDiscountByCardBCPlusCard_Artreon **is** DiscountBycard and DiscountByCard_Name = "BCPlusCard", DiscountByCard_Type = 1, DiscountByCard_Value = 1500;

Table 7
Part of attributes of user individual.

... Person_DiscountCard = "BCPlusCard", Person_DiscountCard = "LotteCard", Person_FavoriteMovieActor = "Will Smith", Person_FavoriteMovieDirector = "James Cameron", Person_FavoriteMovieGenre = "Action", Person_FavoriteMovieTheater = "CGV", Person_MovieTheaterConsiderFactor = "Cost", Person_MinimumMovieGrade = 7.0, Person_Occupation = "College Student", Person_Range = 100, **Fact** fDateTime **is** CurrenDateTime(2008-06-24T12:30:00); fAvailableTime **is** AvailableTime(PT2H); ...

or movie theater, as well as prices. Shop class also has an attribute on discount card information.

By using this ontology, a user, who is in or moves to the vicinity of the shops, can check the recommendation services available at the current time.

4.2. Individuals and association rules

In this part, we explain association rules for location and time, classes for shops, and users with an example of a movie theater and a college student user.

4.2.1. Association rule for location

Table 1 shows the association rules related to the location of a user and shops. 'ServiceRange' attribute sets the service range of a shop.

Fig. 6 presents a map of the service range of a shop and the search range of a user. The shop, whose service range overlaps with the search range of the user, will be recommended to the user.

4.2.2. Association rule for time

Time property is also important in our system, because business hours of shops and available time of user should be considered for a recommendation.

Table 2 shows the association rules on available time of a user and movie schedules. These association rules define current date and time, and the available time of a user for watching a movie. Through these rules, the system recommends to the user movies and a theater, where the use can view the film within his/her available time. Fig. 7 shows an example of an association rule in table 2. (In the case of Fig. 7, movie #2 will be recommended).

4.2.3. Information on shops

For easy understanding, we explain the shop class using a movie theater individual.

Table 3 shows individuals of a movie theater. These individuals have shop information such as its rating, service price, business hours, currently showing movie list, discount cards, location, and service range. (Location is represented with the GRS80 (Moritz, 1980) coordinates system since we mash up the Naver map (Naver OpenAPI, 2008)).

Table 4 shows individuals of a movie that is showing at the above theater individuals. Main actors, genre, and rating of movies are on these individuals, and this information is used for recommending movies according to the preferences of the user on the user profile.

Tables 5 and 6 show the time attribute and discount card attribute of a movie individual. In this example, the movie 'The Incredible Hulk' starts at 2:00 pm, 24 June, 2008 and ends at 3:54 pm, 24 June, 2008 and a person who has a BCPuls card can obtain a discount of ₩1500.

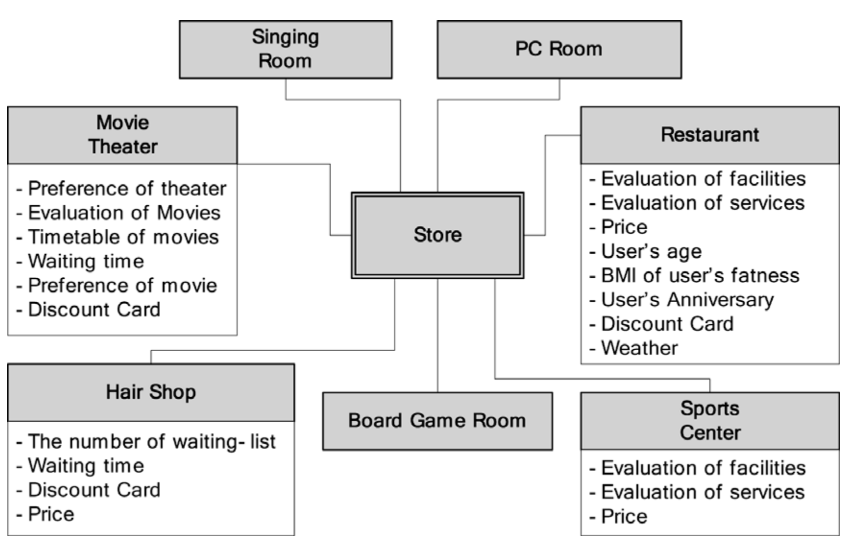

Fig. 8. Classified shops for experiment.

By combining attributes of Tables 3–6, we can obtain information on the shop 'iShopArtreon_Sinchon' as follows: "Artreon_Sinchon is a movie theater with a rating score 7 and a [₩]7000 ticket price. Its business hours are 6 am to 12 pm, and currently playing movies are 'The Incredible Hulk', 'Kung Fu Panda', and 'The Happening'. A person who has a BCPlus card can receive a discount of [₩]1500."

4.2.4. Information on user

User profiles are stored on the user class.

Table 7 shows a part of the attributes of a user profile. This individual has information on cards the user has, the user's occupation, the user's preferred actors, director, genre, and theater as well as the search range of the user and available time to watch a movie.

Information on shops and users are sent to the Bossam inference engine to infer a recommendation.

5. Simulation experiments

For experiments, we adopt the NAVER Map Open API (NAVER OpenAPI, 2008) as a digital map and Bossam (Jang & Sohn, 2004) as an inference engine. The target area we model is Sinchon, a popular shopping area of Seoul city. An experimenter moves through Sinchon and checks the recommendations on a PDA.

We classify shops into seven types and implement each ontology. Fig. 8 shows the classes applied to the shops. If we are being recommended a restaurant, the classes of current time, age of user, rating of the restaurant, and discount card are used for inference. Similarly, if we are being recommended dishes at a restaurant, the classes of weight of user, weather, and commemoration day are used. For experiments, we implement 173 association rules of these shops and store them in the knowledge base with 735 individuals and facts.

Table 8 shows the inference process using these rules on a knowledge base. We conduct experiments under the assumption that the user is a college student and he/she has interest in watching a movie.

Inference results presented in Table 8 show that a ₩1000 discount is applied for users who are college students or a ₩1500 discount for user having a discount card. If the user sets his or her available time to wait to watch the movie at 2 hours, from the current time of 13:30, the system recommends the movie 'The Incredible Hulk', which starts at 14:00, and 'Kung Fu Panda', which starts at 14:30.

Table 8
Process of inference on movie theater.

Fact8897 is ShopsInTheRange(iShopArtreon_Sinchon)
Fact87 is AdditionalDiscount(iShopArtreon_Sinchon,"CollegeStudentDiscount",1, 1000)
Fact8899 is ShopsInBusinessTime(iShopArtreon_Sinchon)
Fact8903 is ShopsOfMovieTheater(iShopArtreon_Sinchon)
Fact907 is TheatersInBusinessTime(iShopArtreon_Sinchon)
Fact29 is TheMovieListWithinAgeLimit(iMovieInfoTheIncredibleHulk)
Fact9 is TheMovieListWithUninterestingGenre(iMovieInfoTheHappening)
Fact876 is TheMovieTheaterDiscounting(iShopArtreon_Sinchon,"BCPlusCard",5500)
Fact767 is RecommendedMovieTheater(iShopArtreon_Sinchon,"BCPlusCard",5500)
Fact38971 is TheMovieTheaterDiscounting (iShopArtreon_Sinchon,"CollegeStudentDiscount",6000)
Fact38972 is RecommendedMovieTheater(iShopArtreon_Sinchon," CollegeStudentDiscount",6000)
Fact39048 is RecommendedMovieTheater (iShopArtreon_Sinchon, iScreeningMovieTheIncredibleHulk, 2008-06-024T14:00:00,"TheIncredibleHulk")
Fact39049 is RecommendedMovieTheater (iShopArtreon_Sinchon, iScreeningMovieTheKungFuPanda, 2008-06-024T14:30:00,"TheKungFuPanda")
Fact39052 is RecommendedShopWithCoordinator("Artreon_Sinchon", 306565, 551006)

Fig. 9. Recommendation results.

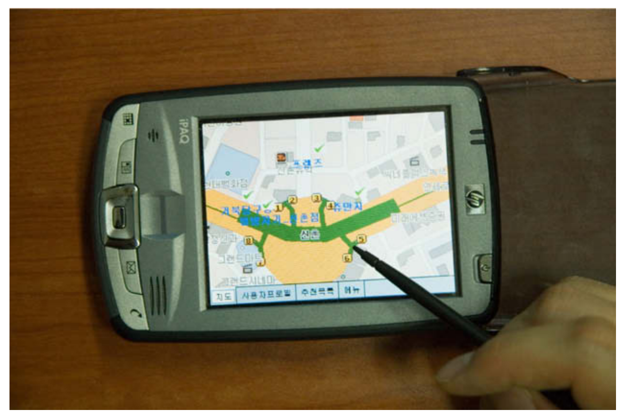

Fig. 10. Mobile interface.

Fig. 9 shows the interface for presenting inference results to the user and Fig. 10 shows the results on a wireless PDA.

6. Conclusion

One of the core services of a ubiquitous environment is Location-Based Systems (LBS). To date, research on LBS has mainly focused on context aware services by service providers. These approaches are not easily applied to real situations, since they are limited by a lack of information. In this study, we model the ontology with consideration of the user's position and the available time and implement a knowledge base using this ontology. The ontology also has information about shops for recommendations and personal profiles of users. A user can manage the information through the user interface module. The knowledge base, where knowledge is accumulated by users, is extended as collective intelligence and the accumulated knowledge is vaster, more diverse, and more up-to-date than that of a service provider. This approach of collective intelligence is an effective implementation of LBS and should provide the most recent information. We test our system under the assumption that the user, who wants to watch a movie, is in a specific area. To recommend a list of theaters and movies to the user, our system executes an inference procedure with information on the starting time of movies, the current time and current position of the user, information on movies and theaters, and the user profile. We mashed up OpenAPI for the map data and shop information for scalable implementation of the system.

Our system shows that personalized recommendation can be provided to a user by inferring a knowledge base in a wireless internet environment. In future work, in order to design more shops that are typically encountered in daily life, we plan to extend the ontology to include various classes for these shops.

Acknowledgement

This work was supported by the Brain Korea 21 project.

References

Anderson, P. (2007). What is Web 2.0? Ideas, technologies and implications for education. *JISC Technology and Standards Watch*.

Costa-Requena, J., Haitao, T., & Espigares, I. (2002). Consistent LBS solution in next generations of mobile internet. In *Proceedings of the ninth international conference on parallel and distributed systems* (pp. 637–642). Helsinki.

Daum OpenAPI. (2008). *OpenAPI: Search API*. Daum Communications Corp. <http://dna.daum.net>.

Jang, M. (2005). *Buchingae: A rule language for the Web*. <http://www.geocities.com/jangminsu/buchingae-rule-language.html>.

Jang, M., & Sohn, J. C. (2004). Bossam: An extended rule engine for OWL inferencing. *Lecture Notes in Computer Science, 3323*, 128–138.

Jose, R., Moreira, A., & Rodriues, H. (2003). The AROUND architecture for dynamic location-based services. *Mobile Networks and Applications, 8*, 377–387.

Kim, J. W., Kim, J. Y., Hwang, H. S., Park, S. S., Kim, C. S., & Park, S. G. (2005). The semantic Web approach in location based services. *Lecture Notes in Computer Science, 3481*, 127–136.

McDiarmid, A., & Irvine, J. (2004). Achieving anonymous location-based services. In *Proceedings of the vehicular technology conference* (pp. 2970–2973). Glasgow.

Moritz, H. (1980). Geodetic reference system. *Bulletin Geodesique, 54*(3), 395–405.

NAVER OpenAPI. (2008). *Service API: Map API*. NHN Corp. <http://openapi.naver.com>.

Protégé. (2008). *The Protégé Project, Stanford Center for Biomedical Informatics Research*. <http://protege.stanford.edu>.

Steiniger, S., Neun, M., & Edwardes, A. (2006). *Lecture notes: Foundations of location based services*. University of Zurich.

Tilson, D., Lyytinen, K., & Baxter, R. (2004). A framework for selecting a location based service (LBS) strategy and service portfolio. *System Science*, 78–87.