Implementation of a Geo-Semantic App by Combining Mobile User Contexts with Geographic Ontologies

Ha-Jung Lee* Yang-Won Lee**

Abstract This paper describes a GIS framework for geo-semantic information retrieval in mobile computing environments. We built geographic ontologies of POI (point of interest) and weather information for use in the combination of semantic, spatial, and temporal functions in a fully integrated database. We also implemented a geo-semantic app for Android-based smartphones that can extract more appropriate POIs in terms of user contexts and geographic ontologies and can visualize the POIs using Google Maps API (application programming interface). The feasibility tests showed our geo-semantic app can provide pertinent POI information according to mobile user contexts such as location, time, schedule, and weather. We can discover a baking CVS (convenience store) in the test of bakery search and can find out a drive-in theater for a not rainy day, which are good examples of the geo-semantic query using semantic, spatial, and temporal functions. As future work, we should need ontology-based inference systems and the LOD (linked open data) of various ontologies for more advanced sharing of geographic knowledge.

Keywords : Geographic ontology, Geo-semantic, Smartphone app, ORDBMS

1. Introduction

Development of computer and internet technologies has enriched knowledge-based society, but it is still not easy to find out pertinent knowledge from a vast amount of information. Existing information retrieval systems have some limitations to fully satisfy user's needs because the searching mechanism is mainly based on the string match for given keywords, so the search results may not be relevant to the request of users, or not be enough to solve the problems of synonyms and homonyms[16, 28]. Ontology is thought of as an alternative to such problems coming from the inexplicitness of vocabularies[20], because it can define basic terms and relations as well as the rules for combining the terms and relations in a topic area[22].

Recently, ontology is also applied to the web for more effective information retrieval and management of web resources[5]. Semantic web is the implementation of ontology on the web by building vocabularies and rules to handle a set of domain knowledge[32]. It is a new type of web technologies to pursuit high quality web service for knowledge discovery[3, 12]. As our computing capabilities are extended to mobile devices like smartphone and tablet PC[19], semantic web is also anticipated to be implemented for mobile computing.

Understanding situations or contexts of mobile users such as location, time, schedule, and weather will be very important for information retrieval on the mobile environment. Geographic information may be a key to mobile user contexts because it can answer the questions like "How is the weather around me?" or "When and where is it convenient for us to meet?" Smartphone apps

* Researcher, Nongshim Data System. lbj312@gmail.com
** Assistant Professor, Dept. of Spatial Information Engineering, Pukyong National University. modconfi@pknu.ac.kr (Corresponding Author)

using GIS also have a challenging task of discovering hidden geographic knowledge in association with mobile user contexts. Geographic ontology may be more complicated because it includes spatial information in addition to the generic ontology[6], and the geo-semantic technology should incorporate the ways of providing geographic ontologies for spatial and attribute information through the network.

Most ontology systems are based on the XML files[2, 10, 18, 28, 34] or relational DBMS[7, 11] [13, 14, 15, 17, 24, 26, 27, 30]. However, both manners are not enough for geo-semantic apps because the file-based system is not well suited to handling big data on a real-time basis, and the relational DBMS does not provide an integrated framework for spatial, temporal, and semantic functionalities. In the information retrieval using RQL (RDF query language) in an XML-based system or using SQL (structured query language) in a relational DBMS, we cannot construct a query statement that combines spatial and ontological functions at the same time. Thus, an ORDBMS (object-relational DBMS) that encompasses spatial, temporal, and semantic functions in a fully integrated way should be employed for geographic ontologies, although it has not yet been introduced into geo-semantic apps for smartphones.

In this paper, we describe a GIS framework for geo-semantic information retrieval in mobile computing environments. We built a geographic ontology that integrates mobile user contexts with POI (point of interest) information using an ORDBMS, so the space, time, and semantic functions can work together in a fully integrated database system. We also implemented a geo-semantic app for Android-based smartphones to consume the geographic ontology with the visualizations of Google Maps. The system applicability was examined through the feasibility tests for POI search based on the mobile user contexts including location, time, schedule, and weather. The flow of our study is summarized in Figure 1.

This section has briefly introduced the background and objective of our study. In section 2, we discuss the geo-sematic technologies focused on the geographic ontologies and mobile user contexts. Section 3 shows our GIS framework for the geo-semantic apps, and Section 4 presents the design and implementation of our geographic ontology using Oracle 11g. Section 5 demonstrates our geo-semantic app for Android-based smartphones working with the ontology. We summarize the study and discuss the implications and limitations of the work in Section 6.

2. Geo-Semantic Technologies

Ontology is an explicit, formal specification of a shared conceptualization[9]. "Conceptualization" means an abstract, simplified view of the world. The world actually refers to some phenomenon, topic, or subject area in the world. Every representation of the knowledge in an area of interest is based on a conceptualization. Every conceptualization is grounded on the concepts, objects, and the rela-
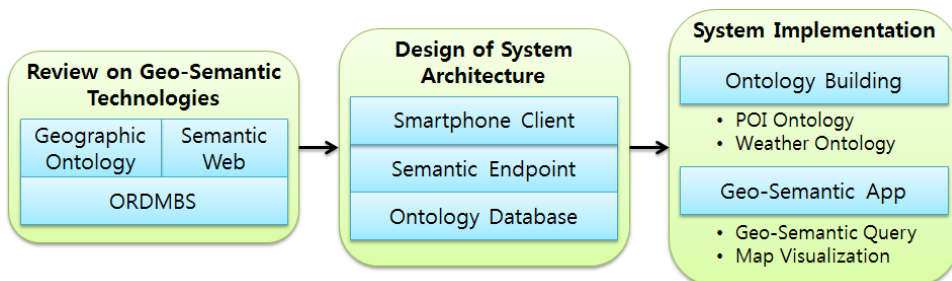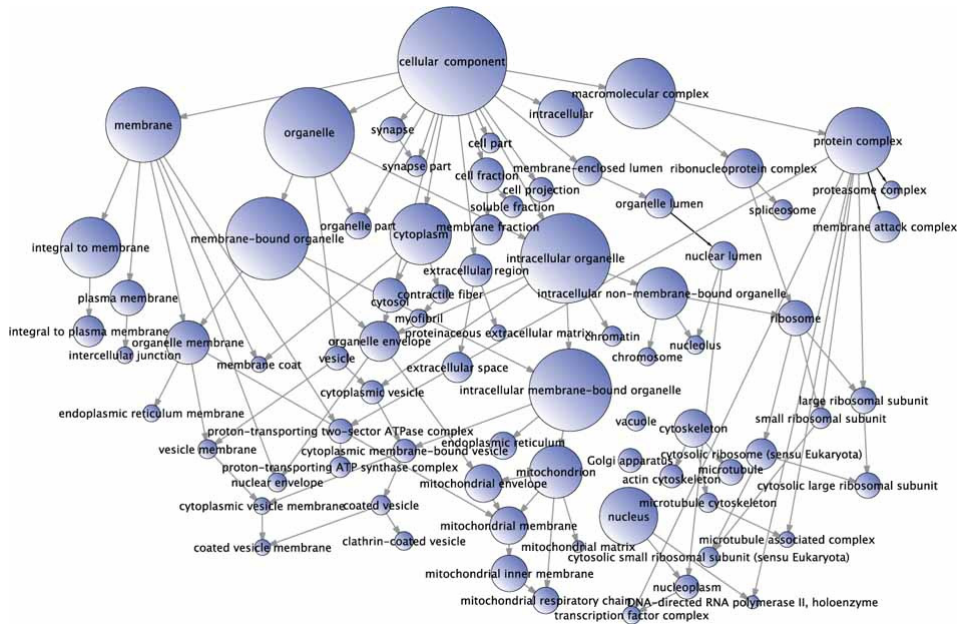

Figure 1. Flow of the study

Figure 2. Nested graph structure of ontology: A gene ontology example
(http://academic.reed.edu/biology)

tionships among them[29]. "Explicit" means that the type of concepts and the constraints on using the concepts are explicitly defined in the data structure of an ontology. "Formal" indicates that the ontology should be machine-interpretable. "Specification" means a declarative representation of the knowledge. "Shared" implies that an ontology is not supposed to represent the subjective knowledge of some individual, but it captures consensual knowledge accepted by a group or a community. In short, an ontology is the manifestation of a shared understanding of a domain that acquired a consensus, and such consensus can facilitate effective communications. This, in turn, leads to other benefits such as interoperability, reuse, and sharing[1, 8].

Legacy information retrieval systems can distinguish neither synonyms that are semantically similar but syntactically different, nor homonyms that are syntactically similar but semantically different. However, ontology can solve the problems and capture the semantic relationships among concepts in a topic area[3]. It is facilitated by the data representation languages like RDF (Resource Description Framework), RDFS (RDF Schema), and OWL (Web Ontology Language).

The RDF represents resources on the web using URI (Uniform Resource Identifier) and describes concepts and relationships of the resources. The relationship can be formalized in a statement, which includes three components, aka triple: subject, predicate, and object. A triple can be expressed in a graph structure, where a subject and an object are connected by a predicate. In a more complicated graph for RDF triples, a subject can be linked to several predicates or used as an object in other statements. In the same way, an object can become a subject in other triple statements. Such a nested graph structure is extensible like web (Figure 2), so that a set of triples can be constructed as a knowledge network. The RDF data that contains a number of individual instances can be better organized through the class modeling using RDFS. The RDFS can provide a dictionary for the vocabularies and define a specification of the

attributes in an object-oriented way, so the similar instances can be categorized and formalized as a class. The OWL provides the functionalities of reasoning for more complicated inferences using rich vocabularies and constraints to describe and interlink the resources.

Semantic web is the implementation of ontology on the web, and the geo-semantic web is an extended form of the semantic web to encompass geographic objects. The geo-semantic technologies may be more complicated because geographic objects are associated with each other in terms of the geometries as well as the attributes. Spatial relationships based on location, distance, and geometry are also very important factor in the semantic network for geographic objects. We can think of a question "Where can I find a Korean restaurant within 500-meter radius from here which opens until 10 PM and provides 20 or more parking lots?" as an example of geo-semantic query statements that include spatial and temporal operations. In this case, we need a distance operation between the two geographic objects "current location" and "Korean restaurant" and a temporal operation for "until 10 PM".

In particular, spatial operations are now related to the locations of mobile nomad equipped with smartphones. Smartphone is a kind of personalized computing device that can extract more suitable information by perceiving user's situations including his/her locations. Knowing the situation of mobile users enables a system to target the information to be delivered[33], which improves the quality of search results[4, 31]. Mobile user contexts include the current location, the nearby POIs around the current location, the time for a user schedule, the weather for the current location or a targeting place, and so on. A geo-semantic approach can handle the mobile user contexts using its spatial and temporal functions as well as its vocabularies and rules. The Oracle 11g provides the functionalities for semantic, spatial, and temporal operations in an integrated ORDBMS.

3. System Architecture

Figure 3 shows the architecture of our GIS framework for a geo-semantic app working with ORDBMS on the mobile Internet. It includes Oracle 11g for a database server, Apache Tomcat for a web server, and Android-based smartphones for mobile client. Data and ontology for mobile user contexts such as POI information and weather vocabularies are stored in the Oracle 11g using its geometry and semantic objects. Also, real-time weather information is extracted from Google server though its geocoding services. These server-side components are consumed by Android-based smartphones though an endpoint written in JSP (Java Server Pages) that carries out the brokerage between the server and smartphone clients. Mobile user contexts utilize user's schedules obtained from the calendar app on the smartphones and user's locations acquired by a built-in GPS sensor.
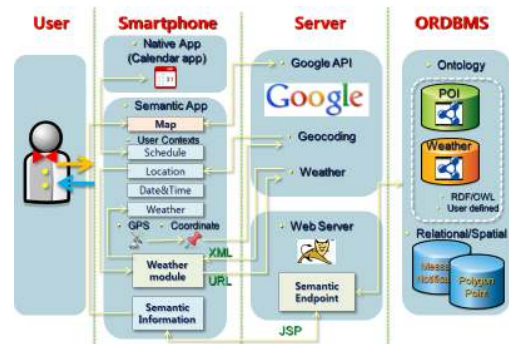

Figure 3. System architecture for geo-semantic app

Our semantic endpoint transmits SQL statements for geo-semantic query, extracts the query result, and sends it to the smartphone clients in the form of JSON (JavaScript Object Notation). The JSP *request.getParameter* function is used to obtain the client-side parameters, and the function *out.print* is used to return the JSON ar-

rays created using the *net.sf.json* package. To call the JSP from the smartphone client and obtain the JSON arrays as a response, we can use an *URLConnection* object that contains the URL of the endpoint JSP. The user parameters for a geo-semantic query are written to an *OutputStreamWriter* object to be incorporated in the URL of the JSP. The query result in JSON can be read from an *InputStream* object into a *BufferedReader* object, and the JSON data in the reader is decoded using the *org.json* package in the Android SDK (software development kit) for visual representations on the smartphone.

Implementation of geo-semantic queries in a fully integrated way can be achieved by using Oracle 11g because it provides the semantic objects for RDF, RDFS, and OWL as well as spatial and temporal objects, while other major ORDBMSs such as SQL Server, MySQL, and PostgreSQL do not yet support spatial, temporal, and semantic objects simultaneously. The RDF data model of Oracle 11g stores triples and connects them in a directed graph, where a node is a subject or object and a link is a predicate or relationship. The *SDO_RDF_TRIPLE_S* object is used for inserting a triple data into a database table (Figure 4). A subject, predicate, and object are stored as an individual row in the *RDF_VALUES$*, an Oracle internal table. At the same time, a subject and object are stored in the internal table *RDF_NODE$* while a predicate is stored in the internal table *RDF_LINK$*.
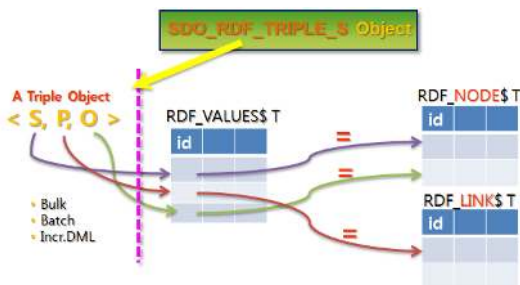

Figure 4. *SDO_RDF_TRIPLE_S* object for triple data

Spatialand temporal objects can be also used as the node, which will be combined with spatial or temporal functions in constructing geo-semantic queries.

4. Building Geographic Ontologies

We built a geographic ontology for weather and POIs to implement a geo-semantic app that reflects mobile user contexts using our system framework. We adopted the process of iterative design[23] consisting of seven steps as follows.

<Step 1> Defining domain and scope: Our study concentrates on the POI recommendation according to the contexts like location, time, weather, and schedules for smartphone users.

<Step 2> Considering existing ontologies: There are a few knowledge bases for weather information[21, 25], but their structures and vocabularies are not well suited to Google Weather API, which we employ to combine user locations and weather information. Thus, we decided to build our own ontology for weather information. We also needed to build a POI ontology for our study area around Daeyeon-dong, Nam-gu, Busan.

<Step 3> Enumerating important terms: We first selected and listed up all important words and names for weather and POIs in an unstructured form for later use in the definition of classes and the creation of instances.

<Step 4> Defining class hierarchy: Using the terms enumerated in Step 3, we defined the classes that represent general concepts and organized them in a hierarchical structure. These classes produce instances, and they construct the nodes for subjects and objects in triple data.

<Step 5> Defining property hierarchy: Property is the vocabulary to represent the attributes and relationships of the classes. Using the terms listed in Step 3, we defined the necessary properties for weather and POIs to organize them in hierarchy.
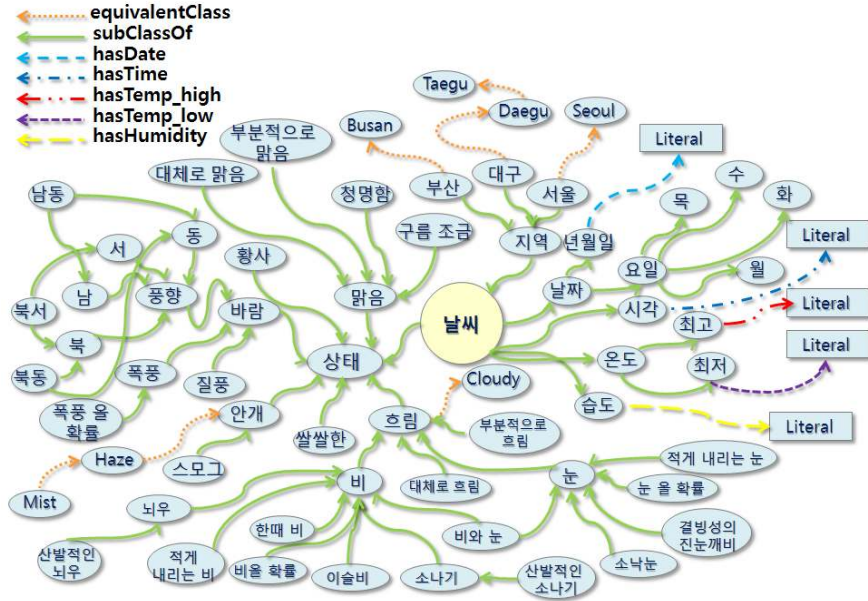
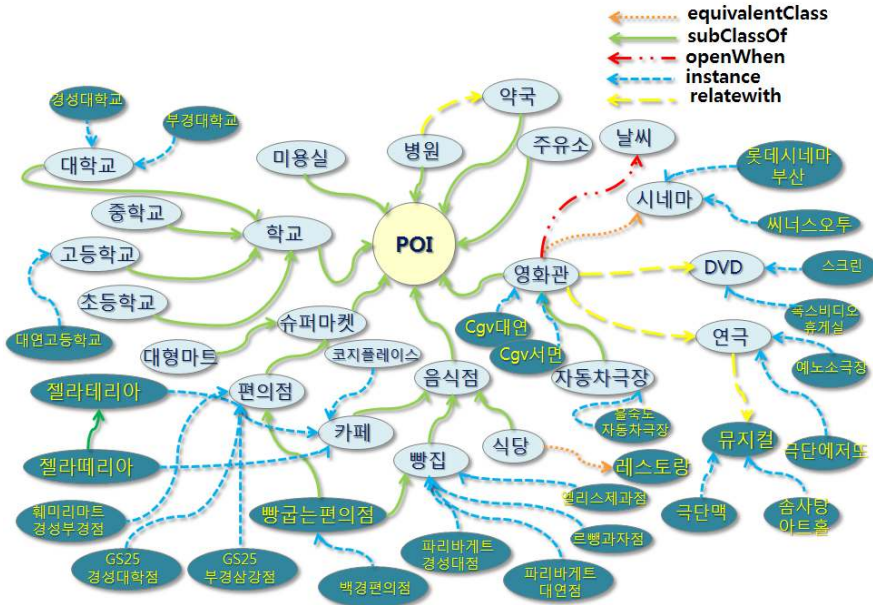Figure 5. Knowledge network for weather information

Figure 6. Knowledge network for POIs

<Step 6> Defining constraints: To improve the quality of information retrieval, the constraints such as cardinality, range, and domain are necessary for the properties. Cardinality decides how many values a property can have. The range constraint asserts that the values of property must belong to the class extension of the class description or to data values in the specified data range. The domain constraint asserts that the subjects must belong to the class extension of the indicated class description[32].

<Step 7> Creating instances: Instances should

inherit the features from its classes. As the number of instances increases, the nodes and links coming from the instances make an extended form of knowledge network.

Figure 5 and 6 are the conceptual diagrams of knowledge network for weather and POIs used in our geo-semantic app. The circle denotes classes and instances, and the line indicates the relationships for the classes and instances. For example, "Mostly cloudy" and "Partly cloudy" are a subclass of "Cloudy", and "Sporadic rain" and "Rain shower" are a subclass of "Rain". The "Paris Baguette – Daeyeon" is an instance of "Bakery" class and the "CGV – Seomyeon" is an instance of "Cinema" class. The conceptual models were physically implemented by taking account of the semantic data modeling process in Oracle 11g (Figure 7). SQL statements used in the physical modeling include the procedure of creating tables, models, and indexes followed by inserting triples and creating entailments (Figure 8). We first created the tables for storing the triple data for weather and POIs, and then executed the semantic models to apply to the triple data. Since triple data is linked each other in a graph structure, we need to create indexes on subjects, objects, and properties for the inference of semantic information. The *SDO_RDF_TRIPLE_S* object is used for inserting all the triple data in the table designated by the semantic model. The RDF, RDFS, and OWL are used in inserting classes, subclasses, and properties for new vocabularies and user-defined rules. We lastly created the entailments to make the rules in effect.

Figure 7. Semantic data modeling process

```
// Creating table
create table weather_ont_data (id Number, triple SDO_RDF_TRIPLE_S);
// Executing model
execute SEM_APIS.CREATE_SEM_MODEL('weather_m', 'weather_ont_data',
'triple');
// Crating index
create index weather_ont_data_sub_idx on weather_ont_data
(triple.get_subject());
// Inserting triples (class)
insert into weather_ont_data values (1, SDO_RDF_TRIPLE_S ('weather_m',
'http://pknu.com/weather_m/날씨', 'http://www.w3.org/1999/02/22-rdf-
syntax-ns#type', 'http://www.w3.org/2000/01/rdf-schema#Class'));
// Inserting triples (subclass)
insert into poi_ont_data values (5, SDO_RDF_TRIPLE_S ('poi_m',
'http://pknu.com/poi_m/초등학교', 'http://www.w3.org/2000/01/rdf-
schema#subClassOf','http://pknu.com/poi_m/학교'));
// Inserting triples (property)
insert into poi_ont_data values (22, SDO_RDF_TRIPLE_S ('poi_m',
'http://pknu.com/poi_m/relatewith', 'http://www.w3.org/1999/02/22-rdf-
syntax-ns#type', 'http://www.w3.org/2002/07/owl#TransitiveProperty'));
// Inserting triples (Instance)
insert into poi_ont_data values (27, SDO_RDF_TRIPLE_S ('poi_m',
'http://pknu.com/poi_m/빵집/파리바게트경성대점',
'http://www.w3.org/1999/02/22-rdf-syntax-ns#type',
'http://pknu.com/poi_m/빵집'));
// Entailment
execute sem_apis.create_entailment('owl_rix_poi', sem_models('poi_m'),
sem_rulebases('owlprime'));
```

Figure 8. Oracle SQL for geo-semantic data modeling

5. Geo-Semantic App Implementation

Using the geographic ontologies built on the Oracle 11g, we implemented a geo-semantic app in which space, time, and semantic functions work together, and the POIs are visualized with Google Maps. The geo-semantic query extracts contextually suitable information for mobile users which otherwise might be hidden under a pile of similar information.

5.1 Contextual association

A contextual information retrieval system does not recommend, for example, drive-in theaters for a rainy day or closed cinemas at midnight. In some cases, users may want the information including theater, musical theater, movie theater, and also DVD room. Such contextually suitable information can be derived by using *SEM_MATCH* or *SEM_RELATED* function in the Oracle SQL. The *SEM_MATCH* function is used in querying semantic data by a given property for triple relationships. We created "*relatesTo*" property in the form of *owl:*

transitiveProperty to express the transitivity, which means if a pair (x,y) is an instance of P, and the pair (y,z) is also instance of P, then we can infer the pair (x,z) is also an instance of P. The "*relatesTo*" property can be used in the query for cinema using the *SEM_MATCH* function, so the instances of "Theater" and "Musical theater" are also listed up according to the relationship by transitivity (Figure 9). The *SEM_RELATED* function is used for linking triples to the column of a relational table. Figure 10 shows the connection between a rain-related triple and the message "Bring an umbrella with you" and the connection between a triple about yellow dust and the message "Wear a mask when you go out."

5.2 Combination of semantic and spatial functions

When searching for a place associated with user's schedule, spatial functions such as *SDO_WITHIN_DISTANCE* and *SDO_CONTAINS* can be used for determining the spatial relationships between user's location and the nearby POIs. The *SDO_WITHIN_DISTANCE* function searches for POIs within a specified distance from a user's location and returns 'TRUE' if any POI is found (Figure 11). User's location can be obtained from a GPS sensor in a smartphone. Suppose we want to find out bakeries within 1-km radius from the current position. The combination of semantic and spatial functions provides the information that a
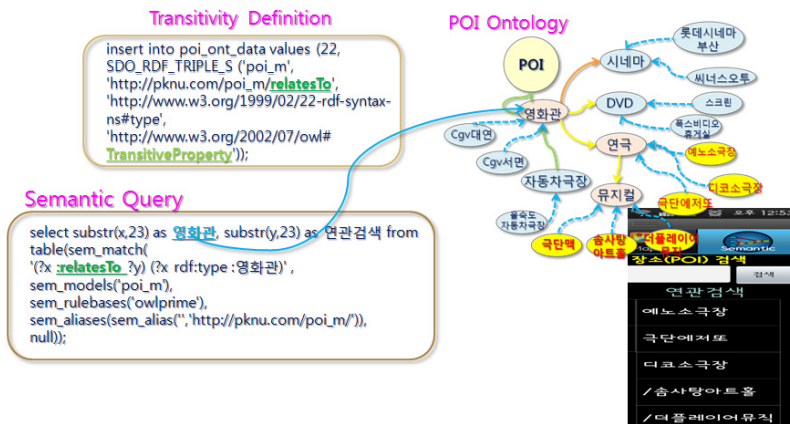

Figure 9. Contextual association discovery using *SEM_MATCH* function
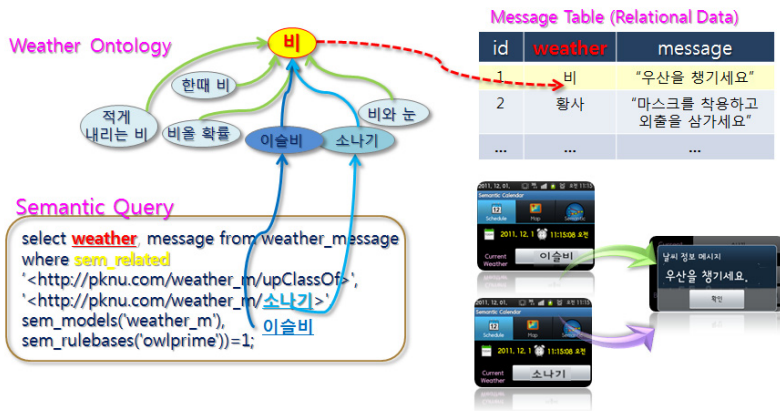

Figure 10. Contextual association discovery using *SEM_RELATED* function

regular query cannot discover. For example, 'Baekkyong CVS (convenience store)' is also included in the query result although it belongs to a class of 'CVS', because the store also bakes breads and sells them like a bakery (Figure 12). We classified such kind of CVS as 'Baking CVS', a subclass of 'Bakery'.

5.3 Combination of semantic, spatial, and temporal functions

Temporal functions can also work together with the semantic and spatial functions in our geo-semantic framework. Most POIs have their own business time, which can be an important element in searching for POIs in terms of the availability of the POIs. For the combination of semantic,

spatial, and temporal functions, the POI table includes the information about the type of POI, geographic location, and administrative district as well as opening and closing time. A drive-in theater is a good example for the combination of semantic, spatial, and temporal functions. Suppose we want to find out nearby cinemas that open at midnight in the Daeyeon-dong area. In this case, a spatial function *SDO_RELATE* with a mask of *CONTAINS* is used to search for the POIs within a specified administrative district. The function *SDO_CONTAINS* can be used alternatively to the combination of the *SDO_RELATE* function and the *CONTAINS* mask. If it rains, drive-in theaters are not included in the query result because the weather ontology filters them out using
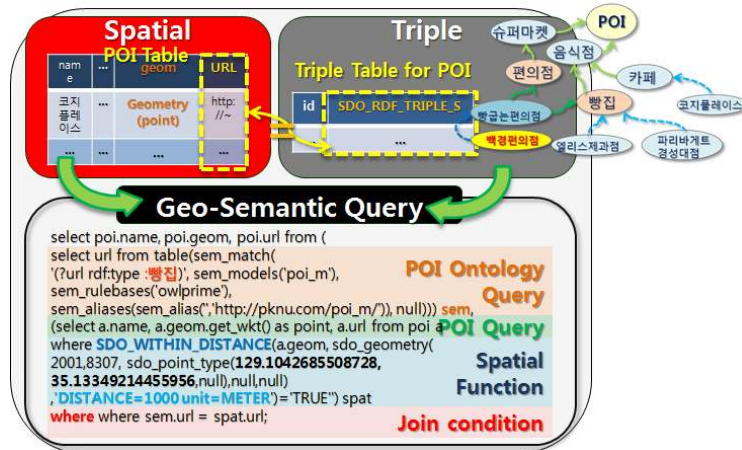

Figure 11. Geo-semantic query using spatial functions

| (a) regular query | (b) geo-semantic query |

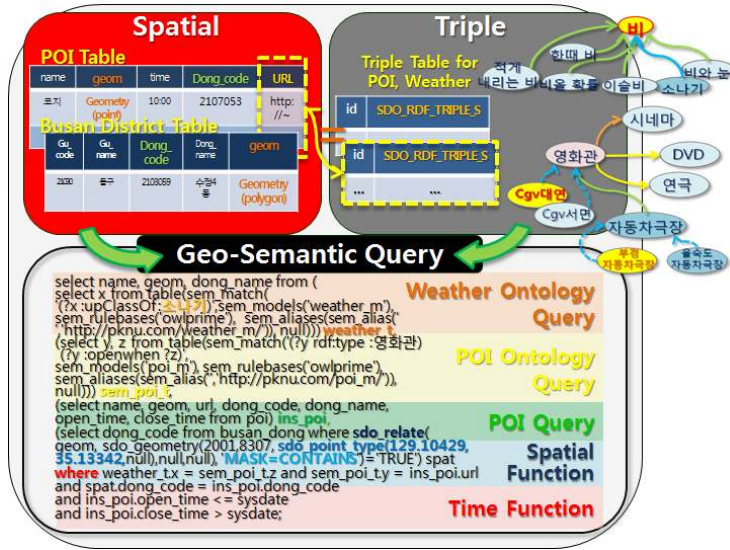Figure 12. Comparison between regular and geo-semantic queries

Figure 13. Geo-semantic query using spatial and temporal functions

(a) Rainy   (b) Not rainy

Figure 14. Query result according to weather condition

the weather information around the user's location. Also, the opening and closing time of each cinema are taken into consideration through the temporal operation in Figure 13. Query results according to the weather condition are compared in Figure 14. We can additionally get the information about "Pukyong Drive-in Theater" if does not rain.

5.4 Smartphone app

Our smartphone app was implemented using Android SDK under Eclipse IDE (Integrated Development Environment). In Figure 15, the app first shows a loading screen (a) and the main screen includes the tabs for schedule, map, and

geo-semantic search. The schedule tap (b) is linked with the native calendar app to extract user's own schedule information such as title, time, location, and the description of the schedule. A weather message is notified by the weather ontology working with the Google Weather API (c). The location button brings a user to the map for the location of the registered schedule (d). We can adjust the map scale and add/remove map layers (e). Current location is updated on the fly according to the movement of the user (f). We can use the geo-semantic search to get the POI information we need (g). The query result of nearby cinemas in a rainy day represents a geo-semantic information retrieval using seman-
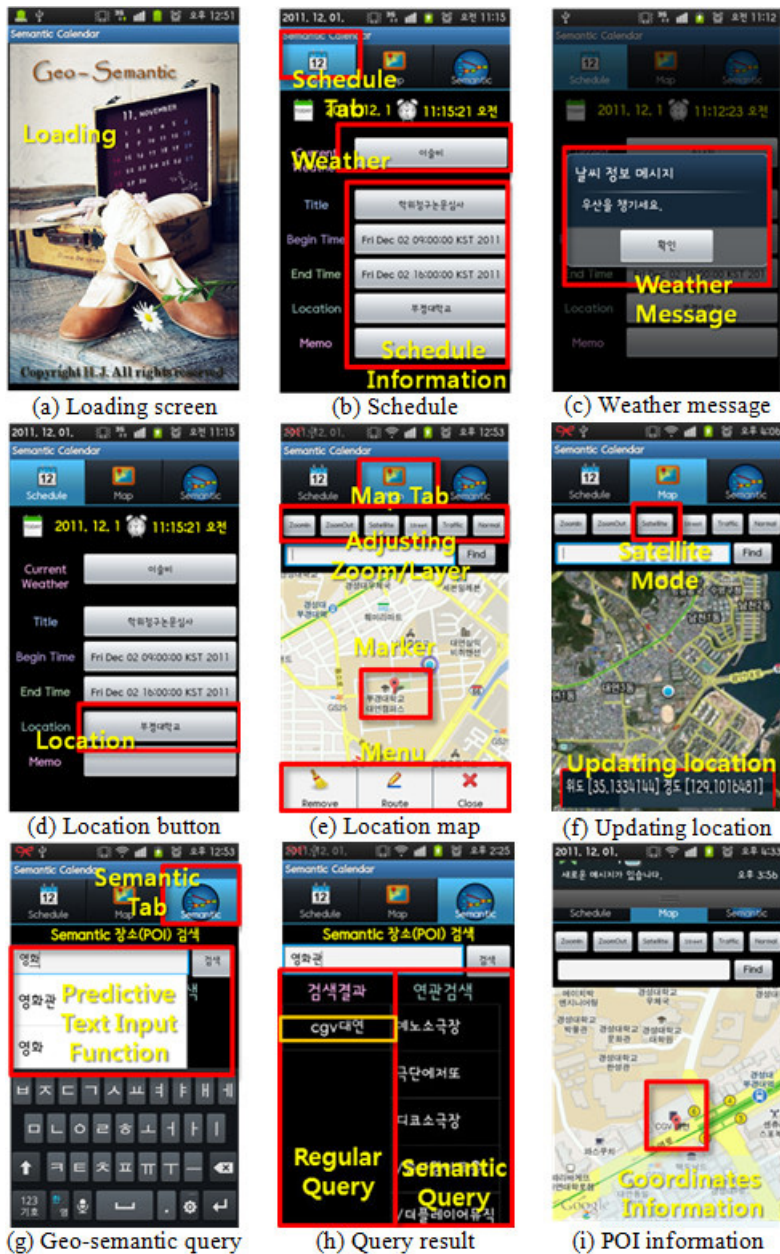
Figure 15. User interface of the smartphone app

tic, spatial, and temporal functions (h). The POIs are displayed in a map on the smartphone (i).

6. Conclusions

This paper described a GIS framework for geo-semantic information retrieval in mobile computing environments. We built geographic on- tologies for POI and weather for the combination of semantic, spatial, and temporal functions in a fully integrated database system. We also im- plemented a geo-semantic app on the Android-based smartphones that can search for more appropriate POIs in terms of user contexts and geographic ontologies and can visualize them with Google Maps. The feasibility tests showed

our app can provide pertinent POI information according to the user's location, time, schedule, and weather. A baking CVS for a bakery search and a drive-in theater for a not rainy are good examples for geo-semantic query. An ORDBMS that possesses spatial, temporal, and semantic functions can be an alternative to the existing ontology systems because it can handle bigger data in an integrated way and is compatible with any platform such as desktop, web, and mobile.

Mechanisms for such information retrieval could be also implemented without geo-semantic technologies, but more powerful functionalities including semantic reasoning and LOD (linked open data) can be achieved by the ontology and semantic approaches. Therefore, as future work, the inference functionalities to cope with more advanced information retrieval should be integrated with our method. Inference engines or semantic reasoners include commercial and open-source software such as Apache Jena, OntoBroker, OWLIM, and RacerPro, which can be a useful tool for building more advanced geographic ontologies. The LOD for the interoperability of various ontology systems located everywhere should be also carefully examined for more open knowledge exchanges. URI definition for revealing domain ontologies and the crawler development for collecting necessary ontology data will be prior steps for the LOD-based GIS interoperability.

References

[1] Agarwal, P. 2005, Ontological Considerations in GIScience, International Journal of Geographical Information Science, 19(5):501-536.

[2] Ahn, M. H; Kwon, J. H. 2007, Ontology based Context-Aware Recommendation System using Concept Hierarchy, Journal of Korean Society for Internet Information, 8(5):81-89.

[3] Broens, T. H. F; Pokraev, S. V; van Sinderen, M. J; Koolwaaij, J; Costa, D. P. 2004, Context-Aware, Ontology-Based Service Discovery, Proceedings of the 2nd European Symposium on Ambient Intelligence, 72-83.

[4] Buriano, L; Marchetti, M. 2006, The Role of Ontologies in Context-Aware Recommender Systems, Proceedings of the 7th International Conference on Mobile Data Management, 80.

[5] Cha, S. J; Choi, Y. J; Lee, K. C. 2010, Development of Search Method using Semantic Technologies about RESTful Web Services, Journal of Korea Spatial Information System Society, 12(1):100-104.

[6] Dransch, D. 2005, Activity and Context – A Conceptual Framework for Mobile Geoservices, In Meng et al. Eds., Map-based Mobile Services, 31-44.

[7] Gali, A; Chen, C. X; Claypool, K. T; Uceda-Sosa, R. 2004, From Ontology to Relational Databases, Lecture Notes in Computer Science, 3289:278-289.

[8] Gašević, D; Djurić, D; Devedžić, V. 2006, Model Driven Architecture and Ontology Development, Springer, New York.

[9] Gruber, T. R. 1993, A Translation Approach to Portable Ontology Specification, Knowledge Acquisition, 5(2):199-220.

[10] Jo, D. W; Choi, J. W; Kim, M. H. 2009, SPARQL Query Tool for Using OWL Ontology, Journal of Korea Society of Computer Information, 11(68):21-30.

[11] Jo, S. H. 2007, The Design and Implementation of an Effective OWL Storage and Management System based on the RDBMS, Master Thesis, Korea National University of Education.

[12] Kim, E. K; Nam, Y. J. 2004, The Comparative Study on the Methodologies of Building Ontology toward Semantic Web, Journal of Information Management, 35(2):57-85.

[13] Kim, S. H; Lee, S. W. 2009, Semantic RDF Data Storage and SQL Query Processing in RDBMS, Proceedings of the Korea Computer Congress 2009, 36(1):134-138.

[14] Lee, H. J; Shim, J. H. 2008, Logical Design of

Product Ontology in Relational Databases, Journal of Society for e-Business Studies, 14(1):81-92.

[15] Lee, J. M. 2006, Efficient Storing and Searching Method in Case-based Reasoning using Ontology, Master Thesis, Sungkyunkwan University.

[16] Lee, K. I. 2011, The Evolution of Search, Semantic Retrieval, Korea Association for ICT Promotion magazine, 5(1):24-29.

[17] Lee, M. J. 2005, A Product Ontology Model and its Design and Implementation on Relational Database Systems, Master Thesis, Sookmyung Women's University.

[18] Lee, S. H. 2010, Development of Subsurface Spatial Information Model with Cluster Analysis and Ontology Model, Journal of The Korean Association of Geographic Information Studies, 13(4):170-180.

[19] Lee, Y. W; Park, K. H. 2004, Design and Implementation of a Mapping Middleware for Wireless Internet Map Service, Journal of Geographic Information System Association of Korea, 12(2):165-179.

[20] Lee, Y. W. 2007, A Semantic Web Service for Tourism Information over the Mobile Web, Journal of the Korean Geographical Society, 42(5):788-807.

[21] Moran, K; Claypool, K. 2010, Building the NNEW Weather Ontology, Available at http://www.aixm.aero/gallery/content/public/2010_05_Conference.

[22] Neches, R; Fikes, R. E; Finin, T; Gruber, T. R; Senator, T; Swartout, W. R. 1991, Enabling Technology for Knowledge Sharing, Artificial Intelligence Magazine, 12(3):36-56.

[23] Noy, N. F; McGuinness D. L. 2001, Ontology development 101: A Guide to Creating Your First Ontology, Available at ftp://ftp.ksl.stanford.edu/pub/KSL_Reports/KSL-01-05.pdf.gz.

[24] Pan, Z; Heflin, J. 2003, DLDB: Extending Relational Databases to Support Semantic Web

Queries, Proceedings of Workshop on Practical and Scaleable Semantic Web Systems, 109-113.

[25] Romo, J; Chacon, C; Romero, R. 2010, Semantic Support for Weather Sensor Data, Proceedings of CAHSI 2010 Annual Meeting, 83-85.

[26] Seo, S. J. 2007, Automatic Ontology Schema Extraction from a Relational Data Model, Master Thesis, Soongsil University.

[27] Shim, H. C. 2009, Design and Implementation of Electronic Approval Search System based on Ontology, Master Thesis, Sogang University.

[28] Sun, B. K; We, D. H; Han, K. R. 2009, A Study on Paper Retrieval System based on OWL Ontology, Journal of Korea Society of Computer Information, 14(2):169-170.

[29] Timpf, S. 2002, Ontologies of Wayfinding: A traveler's Perspective, Networks and Spatial Economics, 2(1):9-33.

[30] Trißl, S; Leser, U. 2005, Querying Ontologies in Relational Database System, Proceedings of the Second International Workshop on Data Integration in the Life Sciences, 3615:63-79.

[31] Vallet, D; Castells, P; Fernandez, M; Mylonas, P; Avrithis, Y. 2007, Personalized Content Retrieval in Context Using Ontological Knowledge, IEEE Transaction on Circuits and Systems for Video Technology, 17(3):336-346.

[32] W3C, 2013, Semantic Web, Available at http://www.w3.org/standards/semanticweb/

[33] Weibenberg, N; Voisard, A; Gartmann, R. 2004, Using Ontologies in Personalized Mobile Applications, Proceedings of the 12th Annual ACM International Workshop on Geographic Information Systems, 229-238.

[34] Yoo, D. H; Suh, Y. M. 2008, An Ontology-based Hotel Search System using Semantic Web Technologies, Journal of Society for e-Business Studies, 13(4):71-92.