

Web-Based Inference Rules for Processing Conceptual Geographical Relationships*

Taro Tezuka, Ryong Lee, Yahiko Kambayashi
Kyoto University
Graduate School of Informatics
Yoshida-Honmachi, Sakyo-ku
Kyoto 606-8501 Japan
{tezuka, ryong, yahiko}@db.soc.i.kyoto-u.ac.jp

Hiroki Takakura
Kyoto University
Data Processing Center
Yoshida-Honmachi, Sakyo-ku
Kyoto 606-8501 Japan
takakura@rd.kudpc.kyoto-u.ac.jp

Abstract

Dealing with prepositions such as “near”, “between”, and “in front of” is very important in Geographic Information Systems (GISs). In most systems, real world distances are used to handle these prepositions.

One of the difficulties in processing these prepositions lies in the fact that their geographical range are distorted in people’s cognitive maps. For example, size of an area referred by preposition “near” gets narrowed when there exists more famous landmark right next to the base geographical objects. This is because users are likely to choose most famous landmark when referring to certain position. Also, area referred by “between” is not a straight line. It curves along the most commonly used pathway between base objects.

Difference in popularity of geographical objects is the main reason to cause such distortions in cognitive maps. Since there are large amount of data on the Web, we believe that such conceptual distortion can be calculated by analyzing web data. Popularity and co-occurrence rate are calculated through their frequency in web resources. Inference rules are set to restrict the target of conceptual prepositions using GIS and information obtained from the Web.

1. Introduction

Geographical information is very valuable source of knowledge about our world, and for decades Geographic Information Systems (GISs) have accumulated a huge amount of these information. However, those information are not easily accessible for everyone, due to the fact that it is often very difficult for untrained users to express their

intentions in GIS query language. In everyday conversation, people tend to use conceptual geographical prepositions such as “near”, “between”, and “in front of” very often. There are some systems that support queries with these prepositions, by using real world distances [10]. However, real world distances often differ from user’s intension. In this paper, we will discuss how we can handle such conceptual relationships by analyzing data from web pages.

Conceptual prepositions are based on cognitive models of geographical relationships users have in their minds, not the real world physical relationships. We will call this “cognitive map”. To process queries with these conceptual prepositions, a query resolver must have similar model within itself. We will show in later section that such model can be retrieved from the Web.

Figure 1 shows the motivation of the research. “Ginkakuji Temple” is one of the well-known sight-seeing spots in Kyoto, Japan. If we check web pages of shops which claims to be “near” Ginkakuji Temple, we can find the term “near” is used differently depending on the direction.

1. To the west: Kyoto University is located in the west of Ginkakuji Temple. Shops located in the middle may claim that they are “near Kyoto University” if they are shops for students, while those for tourists may claim to be “near Ginkakuji Temple”.
2. To the south: There is “the Philosopher’s Path” from Ginkakuji Temple to Nanzenji Temple. Shops near the path usually claim to be “along the Philosopher’s Path”, not “near Ginkakuji Temple”.
3. To the east: There is a mountain and there are no shops that may claim “near Ginkakuji Temple” in this direction.
4. To the north: Sight-seeings spots are scarce in the

¹This work was given the Best Paper Award in the First International Workshop on Web Geographical Information Systems (WGIS’01).

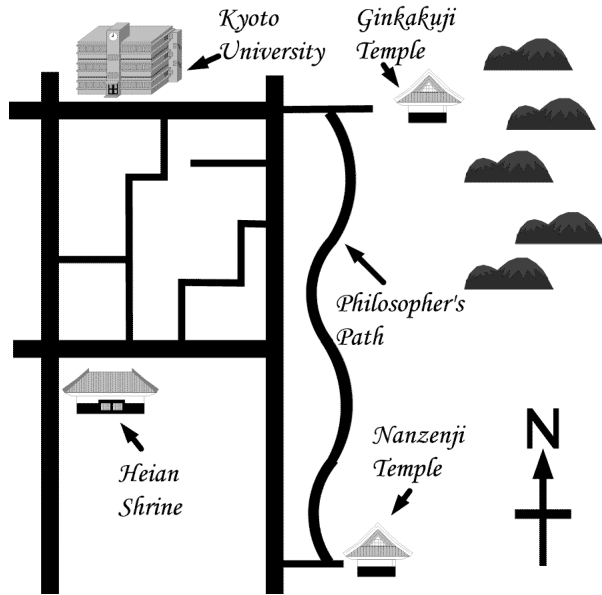


Figure 1. Main Streets and Narrow Roads

north, so shops claiming to be “near Ginkakuji Temple” stretches further than to the other directions.

Figure 2 is a model that shows relationships among real world, cognitive map, and the web resources. Dotted lines indicate relationships between entities of different layers, although they are not one-to-one mappings.

The model asserts that the web resources are not direct projection of the real world. They are created based on cognitive map in contents creators’ minds. User queries with conceptual geographical prepositions also comes from cognitive maps. In order to express these prepositions, our proposed query resolver must comprehend cognitive maps. This task will be accomplished by obtaining information of the real world from GIS, and complementing it with the web resources. Our strategy can intuitively be understood as a two-side attack toward the cognitive map by means of GIS and the Web.

Many GISs has tried to define conceptual geographical prepositions in terms of topology or geometry, and failed. We will introduce concepts of popularity, co-occurrences, and other semantic properties, in order to give more appropriate means of expressing those prepositions.

The rest of the paper is organized as follows. Section 2 will discuss what kind of problems occur when one tries to handle geographical prepositions using topology and geometry only. In Section 3, methods for defining those prepositions will be given. Section 4 describes some related work. Section 5 concludes this paper.

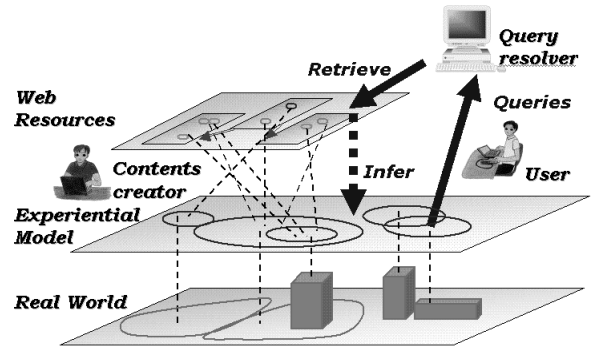


Figure 2. The Web, cognitive map, and the reality

2. Problems with Geometric Calculation

Conventional spatial query languages have mainly focused on resolving geometrical and topological queries [5][12][14]. In this section we will point out that geometrical translation of geographical queries are not omnipotent.

In Figure 1, we have shown that conceptual meanings of “near” are different from real physical meanings. Other prepositions like “around”, “between”, “behind”, “in front of” have similar problems. In this section we will describe those distortions in more details.

2.1. Semantic Knowledge Based Distortion

There are difference between the real world and people’s cognitive maps regarding geographical relationship “near”. We did a primitive experiment using search engine Google[22]. We compared number of pages containing combinations of “near” or “in front of” with various place names. The experiment was done in Japanese. Place names were chosen from Kyoto, the previous capital of Japan. “Kyoto Station” and “Kyoto Tower” were included in these place names. They are very close to each other, within 5 minutes walking distance.

We have found out that in most place names, the number of “near PLACENAME” overwhelms the number of “in front of PLACENAME”. For example, “near Kyoto Station” scored 152 hits, while “in front of Kyoto Station” scored 39. Out of 457 place names we have checked, 91 had this order, 32 had the reversed order, and 11 had same number of “near PLACENAME” and “in front of PLACENAME”. (For the rest, no combination with either “near” or “in front of” were found.) One notable example of the reversed order was the Kyoto Tower. “In front of Kyoto Tower” scored 17, while “near Kyoto Tower” only scored 4.

We assumed following explanation to this phenomenon. Kyoto Station and Kyoto Tower are right next to each other.

Kyoto Station is much more important landmark than Kyoto Tower. Therefore, people are more likely to say "near Kyoto Station" than saying "near Kyoto Tower". Expression "near Kyoto Tower" is mainly used to refer to a small area, which is much closer from Kyoto Tower than from Kyoto Station. Here, a cognitive map is distorted from the real world so that the area referred by "near Kyoto Tower" was strongly narrowed by a landmark "Kyoto Station".

This is also found in a larger scale too. Although Otsu City in central Japan has more population than Morioka City (289,601 to 288,844), expression "near Otsu" has only 5 hits, while "near Morioka" scores 21 hits. It also comes out that "Morioka" by itself scores 174,000 hits, while "Otsu" scores 150,000. The percentage of "near PLACENAME" to "PLACENAME" alone, is in case of Otsu 0.003%, while for Morioka it is 0.012%.

We can explain this difference using the fact that Otsu City is surrounded by other large cities including Kyoto and Kusatsu, while population around Morioka City is scarce.

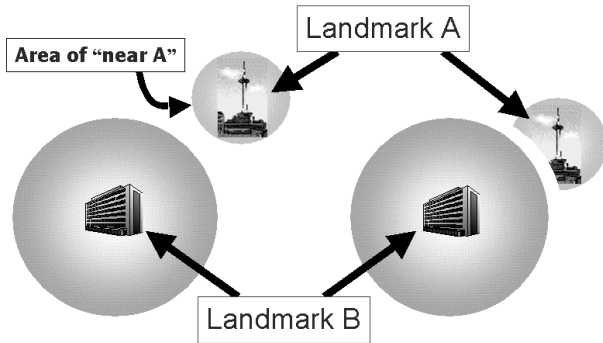


Figure 3. The mechanism of semantic knowledge based distortion

What we have mentioned already was dealing with popularity of base geographical objects. Detailed consideration shows that popularity knowledge is not enough to recover the area intended.

Preposition "near" does not state any distinct size or shape of area by itself. Neither does the phrase "near Kyoto Station". The complete form "cafes near Kyoto Station" gives a set of cafes, and through this set the query resolver can determine the area intended by a user, in that particular context.

It is obvious that simple geometric calculation can not obtain appropriate size and shape referred by "near". We will call the difference between simple geometric model and our experiential understanding "semantic knowledge based distortions".

To calculate the area asserted by "near", we must know the importance values of other geographical objects surrounding it. In case of cities, populations might be one good

way to measure their importance. This is available from conventional GISs. However, for Kyoto Station and Kyoto Tower, no data in GIS indicates their "popularity". Therefore, we must seek for different data sources. We consider the Web to be one answer. Popularity on the Web can be used to identify landmarks and their values of importance. Other semantic knowledge can also be obtained from the Web, although this will be a more difficult task, involving information extraction technique from natural language.

2.2. Path Popularity Based Distortion

Let's think of a query asking for "the most inexpensive hotel between Kyoto City and Tokyo Metropolitan Area".

If a query resolver receives this constraint "between Kyoto and Tokyo" literally, or in a simple geometrical sense, search range will include Chausuyama Mountain and Minobu City, which are in mountainous area where no major roads that connect Kyoto and Tokyo passes. Hotels in this area are less likely to be what the user wanted. This situation is indicated in Figure 4.

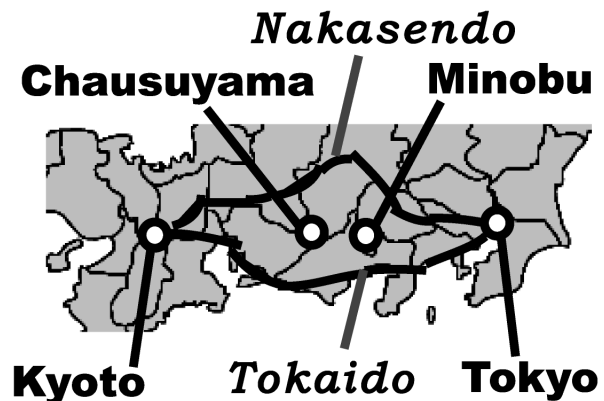


Figure 4. "Between" means "along path"

To give a better answer, the query resolver must know that "between Kyoto and Tokyo" are more likely to mean narrow area along Eastern Coastal Road ("Tokaido Road") or along Central Mountainous Road ("Nakasendo Road"). This kind of distortion results from people's activity of movement. More people uses Tokaido Road and Nakasendo Road as their route when they go from Tokyo to Kyoto and vice versa. Areas along these roads are likely to come to their minds when they say "between Kyoto and Tokyo". Although we can go from Tokyo to Kyoto using local roads passing through Chausuyama Mountain and Minobu City, these roads are often ignored.

We call this difference between the cognitive map and the real world as "path popularity based distortions". Paths are narrow geographical objects which connects two or more geographical objects. Roads, river, railways are examples of paths. Abstract model for this distortion is shown

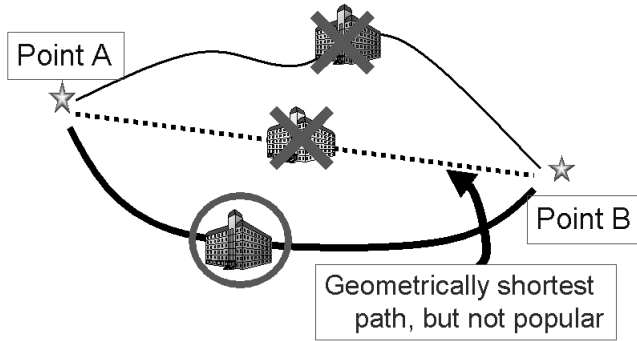


Figure 5. The mechanism of path popularity based distortion

in Figure fig:between. To make our query resolver recognize this type of distortion and answer queries correctly, we must measure the degree of importance for each path. This is not available from GIS. However, we can find them in the web resources.

2.3. Object Identification Problems

There is a case where one river changes its name as it flows. An example is found in Kyoto, where one river changes its name from Hozu River to Ooi River, and then to Katsura River.

Suppose a user asked for "a map of Arashiyama with Katsura River". Arashiyama in Kyoto is a famous sightseeing spot along Ooi River, which is the upper stream of Katsura River. A block-headed system gives a large scale map that includes both Arashiyama and Katsura River. However, it is less likely that this is what the user wanted. There are two other possibilities. The user has mistakenly thought of Ooi River as Katsura River, or she is a resident along Katsura River and has conventionally called the whole river by this name.

Preferable response to this query is to give a small scale map of Arashiyama with Ooi River. The query resolver can deduce this answer if it knows two facts: Ooi River is the upper stream of Katsura River, and that people tends to refer to the whole river by using the most familiar name.

Many rivers change their names as they flow. However, this relationship can not be expressed using simple "connectivity" and "identity" only, since popularities of each name must be considered. This is another example where topological and geometrical definition of geographical relationships fail. Concept of popularity is necessary in defining geographical relationship of one river changing to another river.

We can find same problem in many other path-shaped geographical objects, so we will name this "object identification problems" in general.

3. Expressing Conceptual Geographical Prepositions

We now present a method to express conceptual geographical prepositions, based on knowledge acquired from GIS and the Web.

3.1. Obtaining Information from the Web

Popularity and co-occurrence rates play key roles in our query resolver. We get this information mainly from the Web. Other possible sources are encyclopedias and newspaper articles. However, those sources will be of less importance compared to the Web, because they reflect cognitive maps of fewer people, thus are more likely to differ from the average geographical recognition of the whole population.

To obtain popularity and co-occurrence rates, conventional web crawling algorithms are considered to be powerful enough [3][4]. First, we gather a set of web pages by crawling through the Web.

To calculate popularity of a geographical object, the number of web pages containing the name for this geographical object is counted. To calculate co-occurrence rate, the number of web pages containing both words in question are counted first, then this number is divided by a product of the numbers of web pages containing each of the two words. In the following formula, co-occurrence rate of word *a* and word *b* is expressed as $c(a, b)$. $p(a, b)$ indicates the number of web pages containing both words, while $p(a)$ is the number of web pages containing word *a*, and $p(b)$ means the number of web pages containing word *b*.

$$c(a, b) = \frac{p(a, b)}{p(a) \times p(b)}$$

Although we used web pages as units of counting geographical words, there are other possibilities. A sentence, a paragraph, a group of words separated by HTML tags, and a Web site are among other possibilities. Some units might work better on determining certain relationships, while others work well for other relationships. This is an interesting field to explore, and will be included in our future work.

3.2. Expressions for Semantic Knowledge Based Distortion

In this section, inference rules are proposed to express the distortion of experiential world from real world, regarding preposition "near".

As we have mentioned earlier, preposition "near" can have its meaning defined as a function that returns a set of target objects, when given noun phrases before and after it.

Thus the size and shape referred by “near” is context dependent, and must be calculated each time it is used. This subsection defines preposition “near” as a series of inference rules.

To express inference rules, we use Prolog language with negation [19]. Prolog language is mainly used in studying knowledge-bases. It uses set of inference rules and facts to deduce whether given query is true or false. If it is true, proper values are given to variables found in the query. Prolog with negation differs from its original that it has an operator “not”. This operator represents oppositeness.

A difficulty in using Prolog with negation lies in handling of recursive definition. However, our proposed inference rules does not use any of this, so this will not be a problem.

In Prolog expression, we write a query “SELECT cafe NEAR Kyoto-Tower” as

- *?near(X,'Kyoto-Tower'),type(X,'cafe')*.

Here, “Kyoto Tower” will be called a base object, and whatever retrieved as a result of query will be retrieved objects. “Cafe” is a type constraint for retrieved objects. In its simplest form, our proposed query resolver has a knowledge:

- *near(X,Y) :- distance(X,Y,D),D < reach(Y).*

Sign “:-” indicates that whenever right side of the sign is true, left side must be true too. Function “*reach(Y)*” returns maximum reference distance of preposition “near” around geographical object Y. It is set to be a function of base object’s type or popularity.

Due to the distortion mentioned in Section 2.1, we want to restrict referred object of “near” only to those that are closer to the base object, compared to any other object which has same type as the base object. We introduce new predicate “nonear”, and changes rule about “near” so that it has an opposite of “nonear” in its body.

- *near(X,Y) :- distance(X,Y,D),X̄,Y,D < reach(Y),not nonear(X,Y).*
- *nonear(X,Y) :- distance(X,Y,D),distance(X,W,E), type(Y,T),type(W,T),X̄,Y,X̄W,YW,D > E.*

This eliminates from the query result all cafes that are farther from Kyoto Tower, compared to other geographical object of the same type. If we set “landmark” to be the type for Kyoto Tower, cafes closer to other landmark will be eliminated from a query result. If Kyoto Station is also set to be a landmark, all cafes closer to Kyoto Station than to Kyoto Tower will not be included in a query result. This corresponds to the tendency that these cafes are rather referred to be “near Kyoto Station” than “near Kyoto Tower”.

Similarly, we can set “sight-seeing spot”, “tower”, or “building” to be types for Kyoto Tower. Types must be related to each other, so that the query resolver can make inference regarding types. In the example above, types such as “tower” and “station” must be related with type “landmark”. This knowledge reduces cost of assigning so many types to each geographical object.

Type restriction plays very important role in our system, since landmarks are subjective concept and varies among different group of users.

Geographical objects and their types create knowledge structure. This structure is created in a following way. Relationships between abstract keywords can be obtained from encyclopedia databases. Each geographical object is inserted into this structure using either encyclopedia or the Web. Co-occurrence rate will play key role here. Co-occurrence rate is a frequency of two words appearing together in a same unit of resource. The unit can either be a sentence, a paragraph, or a web page. Co-occurrence rate for a pair of words is calculated by counting the number of units containing both words, then dividing it by a product of the numbers of units containing each word. Each geographical object is tied to an abstract type when co-occurrence rate of the two are high enough. This way, a knowledge structure of geographical objects and abstract types are created. Such structure is mentioned in [8].

The inference rule above did not consider popularity of each landmark. Although Kyoto Station is much more famous and must have wider area of influence, it is not reflected in our inference rules. If we want to add this fact to the rule, we must change it as follows.

- *near(X,Y) :- distance(X,Y,D),D < reach(Y),not nonear(X,Y).*
- *nonear(X,Y) :- distance(X,Y,D),distance(X,W,E),type(Y,T),type(W,T), X̄,Y,X̄W,YW,popularity(Y,P),popularity(W,Q),iv(P,D) < iv(Q,E).*

Function “*iv(P,D)*” gives influence value of a geographical object with popularity P, at a point D distant from it. One way is to define it like gravity, so that it is proportional to P and inversely proportional to a square of D. This part is not difficult to implement.

Distance between geographical objects is calculated using GIS. Popularity can be estimated using the Web, as mentioned in Section 3.1. Although type knowledge will be more difficult to obtain, the Web will be the important source of these knowledge.

3.3. Expressions for Path Popularity Based Distortion

Here we challenge the query “the most inexpensive hotel between Kyoto and Tokyo”. The part “the most inexpensive” can be handled using regular SQL, so we concentrate on solving “hotel between Kyoto and Tokyo” part.

The query can be expressed as follows.

- *?between('Kyoto','Tokyo',X),type(X,'hotel')*.

It is absurd to store all facts regarding predicate “between” in the database. The query resolver must have inference rules so that it can deduce predicate “between”. The rule about “between” can be expressed as follows.

- *between(X,Y,Z) :- path(X,Y,P),along(Z,P).*

Predicate “path(X,Y,P)” indicates that path P connects object X and object Y. Predicate “along(Z,P)” means that object Z is located on the path P. However, we have another option to define “along(Z,P)” in a way that it becomes true when object Z is located within certain distance from path P.

We must add the importance of paths to the inference rules.

- *between(X,Y,Z) :-
path(X,Y,P),along(Z,P),not nobetween(X,Y,Z).*
- *nobetween(X,Y,Z) :-
path(X,Y,P),along(Z,P),importance(P,I),
path(X,Y,Q),importance(Q,J),P.Q,I < J.*

With these rules, queries can be interpreted to retrieve hotels along the most important path connecting Kyoto and Tokyo. The path will most likely be the Eastern Coastal Road (Tokaido Road).

If the user is not satisfied with the response, she must set another query indicating what she wants in more details. If she wanted to find hotels in Chausuyama or Minobu City, she must add that request to the query. In this case, the rules mentioned above are not used as a constraint.

Topological relationships between geographical objects and paths can be gained from GIS. Many algorithms are being proposed for this task [15]. Importance value of each path is obtained from the number each path appears in the Web. This is done just like we did with landmarks. However, there are more aspects in choosing which path to use. Different groups of users use different paths. Paths popular to tourists may differ from that of local inhabitants. Drivers’ view may differ from pedestrians’ view.

To support different view points, paths can be associated with keywords. One path might be related with “tour”, while other might be related with “drive”. The query resolver must also have knowledge structure to relate keywords. This way, a “wide” road can be considered appropriate road to “drive”, while a “historical” road is preferred for “tour” occasion. This knowledge structure can be obtained from encyclopedia or the Web, using the same method mentioned in Section 3.2.

In order to distinguish which view point a user is based on, query resolver can either directly ask the user, or infer it from the words contained in the query. If word “hotel”

was found in the query, the user is likely to be a tourist. If “Gas Station” was in the query, the user is most likely a driver. Knowledge structure of geographical object types can be used here.

We now look at a concrete example, a query asking for “a cafe between Ginkakuji Temple and Nanzenji Temple”. Two temples are both very popular among tourists. They are shown in Figure fig:ginkakuji. One possible answer to this query is to give cafes along the main streets connecting Ginkakuji Temple and Nanzenji Temple. However, better answer is to give cafes along a path popular among tourists. The fact that Ginkakuji Temple and Nanzenji Temple are both sightseeing spots indicates that the query was most likely made by a tourist.

The query resolver checks for each paths connecting Ginkakuji Temple and Nanzenji Temple, co-occurrence rates with the word “tour”, using the web resources. Then the resolver can find the Philosopher’s Path, which is very popular among tourists.

3.4. Expressions for Object Identification Problem

To solve the problem with the query “a map of Arashiyama with Katsura River”, we must set a knowledge that indicates “paths are likely to be referred using the name for its part”.

In a simplest form, the given query is expressed as follows:

- *?include(M,'Arashiyama'),include(M,'Katsura-River'),
type(M,'map')*.

According to the tendency with paths having many names, we can set following knowledge to the query resolver.

- *include(M,X) :-
physicalinclude(M,Y),equals(X,Y).*
- *include(M,X) :- physicalinclude(M,X).*
- *equals(X,Y) :-
changesitsname(X,Y),type(X,'line'),type(Y,'line').*
- *type(X,'line') :- type(X,'river').*

Here, predicate “include(M,X)” is defined to be a conceptual one, while predicate “physicalinclude(M,X)” is that of the physical world, calculated from GIS data. This way, the query resolver gives the answer which are more likely to be what the user wanted.

Candidates for “changesitsname” relation can be obtained from topological relationship expressed in GIS. However, we can narrow them down using co-occurrence rates of candidate objects on the Web. If two adjacent path names score high co-occurrence rate with same geographical object, it is likely that these two names have “changesitsname” relationship.

In our query resolver, defining various geographical prepositions are required whenever necessary. People's geographical understandings are far more complex than that expressed in terms of topology or geometry. Because there are infinite possibilities in geographical relationships, it is not a good strategy to prepare predefined set of relationships. Rather, we want this set to be dynamic, so that we can extend it whenever we want to declare new type of geographical relationship. This is why we use inference rules as a method for defining geographical prepositions.

Defining new geographical relationships is discussed in [16]. Here, an object-oriented model regarding conceptual space and time is being proposed. This modeling allows users to extend geographical relationships using inheritance. It provides another possible way of defining new conceptual geographical relationship.

3.5. Applications in Searching the Web

Determining the extent of nearness enables a system to extend the area of searching, when there are only few results obtained from an original query. This is a spatial version of searching by similar words. Suppose a query was given as "a parking lot near Kyoto Tower". Even if the system could not find a parking lot near Kyoto Tower, it can extend the area of searching to "near Kyoto Station", if it knows that Kyoto Station is near Kyoto Tower. The same method can also be performed on other geographical prepositions, such as "a restaurant on Shijo Street". Because most web users are unacquainted with complex searching techniques, an automatic support by the system will be of great help to users.

4. Related Work

This paper proposed a way to express conceptual geographical prepositions, by constructing predicate style knowledge of people's cognitive map, using GIS and information retrieved from the Web.

Till now, two strategies exist in formalizing abstract geographical knowledge: spatial logics and model-based approaches.

Various spatial logics has been proposed, including sorted logic to differentiate regions from points [17], and modal logic to express possible object locations as possible worlds [2][11]. Most spatial logic deals only with topological and directional relationships. Formalization of conceptual geographical prepositions such as "near" and "between" are yet to be accomplished.

In model-based approaches, various models are being presented, including intersection matrices, 2D-G strings, symbolic arrays, and spatial indexes [15]. Most widely

used model is an array representation. Arrays are capable of expressing topological and directional relationships [7][14][21]. However, expressing conceptual geographical prepositions are not performed in model-based approaches either.

One reason why both strategies have not dealt with conceptual geographical prepositions is because they tried to formalize spatial relationships in general. Instead, our research has focused on geography in specific, thus facing rich features of conceptual geographical prepositions. Artificial Intelligence researches have also challenged spatial recognitions. However, they cover spatial recognitions in general, and do not focus on special features of geographical prepositions. Artificial Intelligence studies have strong tie with robot navigation. Therefore, relative spatial relationships from one view point, such as "right", "left", and "behind", are more likely to be discussed [9][20]. Geographical relationships are different from these relationships because they do not specify one view point. They are constructed considering all possible view points. They are based on a sum of subjective spatial relationships; in our word, cognitive map.

Making contrast with those attempts to formalize spatial relationships using axioms and models, remarkable new area of naive geography is beginning to rise. These researches came under the influenced of naive physics, an attempt to understand people's recognition about physical laws [6][13]. Although naive geography studies propose the use of cognitive map to handle conceptual geographical queries, methods for handling these queries are yet to be given.

Our paper discussed one such method to express conceptual geographical prepositions. We gave examples to show geometrical and topological calculation are not enough, and proposed concrete way to express them. Our method is based on predicate logic, and has strong capability to be extended.

5. Conclusion

We proposed a method to handle queries with conceptual geographical prepositions, using GIS and information retrieved from the Web. Conceptual geographical prepositions are difficult to be resolved, due to its distortion created by surrounding geographical objects. These distortions can be taken into account if relationships between geographical objects were expressed as predicates. We proposed inference rule to define conceptual geographical prepositions, based on people's cognitive maps of external world. We showed several examples of conceptual geographical prepositions and relationships, and proposed concrete method of expressing them.

Future research will set inference rules on other concep-

tual prepositions which were not mentioned in this paper. This will result in further formalization of these prepositions. More information from the Web will be required to determine increased variety of conceptual geographical relationships. Other methods to retrieve information from the Web must also be sought for.

Acknowledgement

This work has been supported in part by CREST of JST (Japan Science and Technology Corporation).

References

- [1] H. Aonuma, H. Imai and Y. Kambayashi, "A Visual System of Placing Characters Appropriately in Multimedia Map Databases", Proc. of the IFIP TC2/WG 2.6 Working Conference on Visual Database Systems, Tokyo, 1989
- [2] B. Bennet, "Modal Logics for Qualitative Spatial Reasoning", Bulletin of the IGPL 3, 1996
- [3] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine", Proc. of 7th Int. World Wide Web Conf., 1998
- [4] J. Cho, H. Garcia-Molina and L. Page, "Efficient Crawling Through URL Ordering", Proc. of 7th Int. Web Conference (WWW98), Brisbane, 1998
- [5] M. J. Egenhofer and A. U. Frank, "Towards a Spatial Query Language: User Interface Considerations", Proc. of the 14th VLDB Conf., pp. 124-133, Los Angeles, 1988
- [6] M. J. Egenhofer and D. M. Mark, "Naive Geography", In A. U. Frank and W. Kuhn (Eds.), Spatial Information Theory: A Theoretical Basis for GIS, LNCS 988, pp.1-15, Springer, Berlin, 1995
- [7] J. Glasgow and A. Malton, "A Semantics for Model-Based Spatial Reasoning", Mental Models in Discourse Processing and Reasoning, G. Rickheit and C. Habel (Eds.), pp. 259-297, Elsevier Science, 1999
- [8] K. Horikawa, M. Arikawa, H. Takakura and Y. Kambayashi, "Dynamic Map Synthesis Utilizing Extended Thesauruses and Reuse of Query Generation Process", Proc. of the 5th Int. Workshop on Advances in Geographic Information Systems, Las Vegas, 1997
- [9] K. Hiraki, T. Nishizawa and Y. Anzai, "Acquisition and Use of Constraint Expressions for Spatial Knowledge", Computer Software, Vol.8, No.4, pp.30-44, Tokyo, 1991
- [10] R. Kasturi, R. Fernandez, M. L. Amlani and W. C. Feng, "Map Data Processing in Geographic Information Systems", Computer, Vol.22, No.12, pp. 10-21, 1989
- [11] O. Lemon, "Semantical Foundation of Spatial Logics", 5th Int. Conf. on Principles of Knowledge Representation and Reasoning, Morgan Kaufmann, 1996
- [12] D. J. Maguire, M. F. Goodchild and D. W. Rhind (Eds.), "Geographical Information Systems: Principles and Applications", Longman Scientific & Technical, Harlow, 1991
- [13] D. M. Mark and A. U. Frank, "Experiential and Formal Models of Geographic Space", Environment and Planning B 23, pp. 3-24, 1995
- [14] J. Orenstein, "Spatial query processing in an object-oriented database system", Proc. of the 5th ACM-SIGMOD Conference, pp. 326-336, 1986.
- [15] D. Papadias and T. Sellis, "On the Qualitative Representation of Spatial Knowledge in 2D Space", VLDB Journal, Special Issue on Spatial Databases, Vol3(4), pp. 479-516, 1994
- [16] C. Parent, S. Spaccapietra and E. Zimányi, "Spatio-temporal conceptual models: data structures + space + time", Proc. of the 7th Int. Symp. on Advances in Geographic Information Systems, pp. 26 - 33, Kansas City, 1999
- [17] D. A. Randell, Z. Cui and A. G. Cohn, "A Spatial Logic based on Regions and Connection", 3rd Int. Conf. on Knowledge Representation and Reasoning, Morgan Kaufmann, 1992
- [18] A. K. H. Tung, R. T. Ng, L. V. S. Lakshmanan and J. Han, "Geo-spatial Clustering with User-Specified Constraints", Proc. of the Int. Workshop on Multimedia Data Mining, in conjunction with ACM SIGKDD Conference, Boston, 2000
- [19] J. D. Ullman, "Principles of Database and Knowledge-Base Systems", "Computer Science Press, 1988-1989
- [20] A. Yamada and T. Nishida, "Use of Diagrammatic Representations in Natural Language Processing", Journal of Japanese Society for Artificial Intelligence, Vol.9, No.2, pp. 201-204, Tokyo, 1994
- [21] X. Zhou, Y. Zhang, S. Lu and G. Chen, "On Spatial Information Retrieval and Database Generalization", Proc. of Kyoto Int. Conf. on Digital Libraries, pp. 380-386, Kyoto, 2000
- [22] Google, <http://www.google.com>