Volume 2 No. 5, May 2012 ISSN 2223-4985

International Journal of Information and Communication Technology Research

http://www.esjournals.org

Efficient Utilization of DBMS Potential in Spatial Data Mining Applications – Neighborhood Relation Modeling Approach

[1]Siddhi Nath Rajan, [2]Ashok K Sinha,[3]J B Singh

[1]IMS Engineering College, Ghaziabad, (MTU, Noida),INDIA,
Shobhit University,Meerut, India
[2]ABES Engineering College, Ghaziabad, INDIA
[3]Shobhit University, Meerut, India,

ABSTRACT

In the core concept of spatial data mining we need to investigate the neighbors of many objects in the single run of typical data mining algorithm. This means that in spatial data mining algorithm we have to efficiently process the neighborhood relations. An integration of spatial data mining algorithms and the potential of spatial database management system (SDBMS) will help efficiently providing general concept of neighborhood relation and its implementation. If properly integrated, this will speed up the industrial application development of spatial database and applying the traditional data mining concept on that[1][4][5][6].
Here in this paper **the neighborhood relation and neighbourhood index** has been discussed and the related small set of database primitives has been defined. The proposed basic operations reduce the search space and support well the typical data mining algorithms.
Such database primitives have been implemented on top of a commercially available Spatial Database Management System (SDBMS) Oracle10g [6][7].

Keywords: *Spatial Data Base,SDT*

1. INTRODUCTION

The explosive growth of spatial database has far outpaced the human ability to interpret this data. This creates an urgent need for new technology and tools that support the human in transforming the data into useful information and knowledge. Spatial Database Management System (SDBMS) is the database systems for the management of spatial data [2]. Spatial Data Mining (SDM) is the process to find the implicit regularities, rules or patterns hidden in large spatial database [1][4][5][6].

Spatial database system is a database system which offers spatial data types (SDT) in its data model and query language. It supports spatial data types in its implementation, providing at least spatial indexing and efficient algorithms for spatial joins. The spatial data types (POINT, LINE REGION etc.) provide us the fundamental abstraction for modeling the structure of geometric entities in space as well as their relationships. The spatial database management system must be able to retrieve from a large collection of objects in some space those lying within a particular area without scanning the whole set. For that spatial indexing is mandatory [8]. The SDBMS musty also have the feature to connect object from different classes through some spatial relationship in a better way than by filtering the Cartesian product [5][6][7].

In this paper a set of database primitives has been examined and introduced for mining the spatial database. Since the attributes of the neighbors of some object of interest may have an influence on the object itself so our database primitives are based on the concept of neighborhood relations. The techniques for efficiently supporting these primitives by DBMS, is presented here.

The paper is organized as follows. Section 2 describes the attributes of spatial data and spatial database system. Section 3 introduces modeling of spatial database, its geometry, data model and spatial query model. Section 4 describes Database Primitives for spatial data mining and shows how these can be expressed by using the proposed primitives. Section 5 presents the concept of neighbourhood indices. In section 6 the concluding issues and several issues for future research work has been discussed.

2. SPATIAL DATA AND SPATIAL DATABASE SYSTEM

In various fields there is a need to manage spatial data i.e. data related to space. One prominent example of

spatial data is the satellite images. To extract information from a satellite images it has to be processed with respect to a spatial frame of reference, possibly our Earth's surface. But the satellite images are not the only the spatial data and our Earth surface are not the only frame of reference. For example if we keep the PIN/ Zip code of an area then the super market transaction data is an example of spatial data. Since the advent of relational database system there have been attempts to manage such data in database. The requirements and techniques for dealing with objects in space that have identity and well defined extents, locations, and relationships are rather different from those for dealing with raster images. For dealing with the objects in space we should have ***Spatial Database System*** and for dealing with raster image we should have ***Image Database System.*** Image database system may include analysis techniques to extract objects in space from images, and offer some spatial database functionality, but are also prepared to store, manipulate and retrieve raster images as discrete entities [16]. Here we only discuss the spatial database systems in the restricted sense. The queries or command that we execute on spatial data is called spatial query. The query, for example, "What are the names of medical stores which keep more than five thousand types of medicines?" is an example of non-spatial query. But the query "What are the names of the medical stores within ten miles of Delhi Railway station" is an example of spatial query. We have already seen that our traditional DBMSs are capable of storing non spatial data and are capable of answering any kind of non-spatial queries. Despite their spectacular success, the prevalent view is that the majority of the existing DBMSs system are either incapable of managing spatial data or are not user friendly when doing so. Secondly if our database keeps the detail of a customers like number, name, address, product description (all non-spatial data) then the query like" List the top ten customers, in terms of sales, in the year 2005" is a non-spatial query and it will be very efficiently answered by our DBMS even if the DMBS has to scan the very large customer database. In fact the database will not scan through all the customers rather it will use index to narrow down the search. Now the query "List all the customers who reside within twenty miles of company headquarter" will confound the database. Now to process the query like this the database has to transform the company headquarters and customer address into a suitable reference system, possibly latitude and longitude, in which distances can be computed and compared. Here the database has to scan the entire database and compute the distance first and then compare the distance that the customer distance is within twenty miles or not. Here the DBMS will not be able to use index to narrow down the search because the traditional indices are incapable of ordering multi-dimensional coordinate data. Therefore a database management system that is tailored for handling spatial data and spatial query is necessarily required.

3. MODELING THE SPATIAL DATABASE

Spatial database supports the **object-relational** model for representing geometries. The object-relational model uses a table with a single column of MDSYS.SDO_GEOMETRY and a single row per geometry instance.

The benefits provided by the object-relational model include:

- Support for many geometry types, including arcs, circles, compound polygons, compound line strings, and optimized rectangles
- Ease of this is used in creating and maintaining indexes and in performing spatial queries[14]
- Index maintenance by the Oracle9*i/Oracle 10g* database server
- Geometries modeled in a single row and single column
- Optimal performance

Once this data (spatial & non-spatial attributes) is stored in an Oracle database, it can be easily manipulated, retrieved, and related to all the other data stored in the database.

Here the database that I have modeled is having non-spatial attributes like state code, state name, state population, GDP, HDI (Human Development Index), GDM (Gross Decadal Migration), Male / Female Literacy rate etc.

The spatial attribute is the coordinate geometry, or vector-based representation of the shape of the feature. One example record for the spatial attribute of state (state code 8) is as follows:

Table 1 : India table (Spatial)

SCODE [NUMBER(2)]	STATE VARCHAR2(15)	GEOM(SDO_GTYPE, SDO_SRID, SDO_POINT(X, Y, Z), SDO_ELEM_INFO, SDO_ORDINATES)
8	HIMACHAL PRADESH	SDO_GEOMETRY(3, NULL, NULL, SDO_ELEM_INFO_ARRAY(1, 3, 1), SDO_ORDINATE_ARRAY(76.74781, 33.13081, 76.79898, 33.17299, 76.80225, 33.19224, 76.87304, 33.16656, 76. 87627, 33.04643, 76.96978, 32.95371, 77.30756, 32.7907, 77.48283, 32.81619, 77.73751, 32.92343, 77.80364, 32.8901, 77.95713, 32.59996, 78.08154, 32.54597, 78.38139, 32.70317, 78.33295, 32.51224, 78.38515, 32.41067, 78.44552, 32.48663, 78.54114, 32.3985, 78.49571, 32.19438, 78.64144, 32.15363, 78.70949, 32.04522, 78.81904, 31.8893, 78.76118, 31.83996, 78.73883, 31.68039, 78.86332, 31.53328, 78.7380 2, 31.42156, 78.82472, 31.37178, 78.93773, 31.19118, 78.98969, 31.04711, 78.92285, 31.03384, 78.77012, 31.12323, 78.48527, 31.14775, 78.33103, 31.2257, 78.09207, 31.08548, 77.94696, 31.09831, 77.84177, 30.95703, 77.77457, 30.85812, 77.80212 , 30.79307, 77.74058, 30.72932, 77.78062, 30.5992, 77.73944, 30.52043, 77.81848, 30.47179, 77.58863, 30.35748, 77.60455, 30.32675, 77.39915, 30.37851, 77.16259, 30.4516, 77.15977, 30.64745, 77.09982, 30.719, 76.90885, 30.83886, 76.78426, 30 .87783, 76.61871, 30.97418, 76.64863, 31.15116, 76.62164, 31.22522, 76.44391, 31.2787, 76.42535, 31.38883, 76.35649, 31.35165, 76.29746, 31.28027, 76.23729, 31.24496, 76.14407, 31.40202, 76.07506, 31.65327, 75.99037, 31.75643, 76.00291, 31.84049, 75.83363, 31.96626, 75.64168, 32.04745, 75.73992, 32.24332, 75.94176, 32.37217, 75.92956, 32.42459, 75.86211, 32.46481, 75.95469, 32.59579, 75.93014, 32.72071, 75.83159, 32.85352, 75.86507, 32.89038, 75.97394, 32.86325, 76.11696, 32.97295, 76.26595, 32.98344, 76.4364, 33.13328, 76.74781, 33.13081))
--	-----	------

The example record of non-spatial attribute for the same state is as follows:

Table 2: India table nonspatial

Scode	Population (In lacks)	HDI	GDM	GDI	Lit_ratio (F/M)
8	6455	0.667	1.23	0.664	0.789

Similar 35 records are entered for different 35 states & Union Teritories of India for both spatial and non spatial tables.

Query Model

Spatial database uses a *two-tier* query model to resolve spatial queries and spatial joins. The term is used to indicate that two distinct operations are performed to resolve queries. The output of the two combined operations yields the exact result set.

The two operations are referred to as *primary* and *secondary* filter operations.

- The **primary filter** permits fast selection of candidate records to pass along to the secondary filter. The primary filter compares geometry approximations to reduce computation complexity and is considered a lower-cost filter [11]. Because the primary filter compares geometric approximations, it returns a superset of the exact result set.
- The **secondary filter** applies exact computations to geometries that result from the primary filter. The secondary filter yields an accurate answer to a spatial query. The secondary filter operation is computationally expensive, but it is only applied to the primary filter results, not the entire data set.

The query based on the table data (table- 1 & Table-1) would be simply Relational-SQL and/or Spatial-SQL. Some of the outputs of the queries are as follows:

Query 1: SELECT * FROM INDIA
Query 2: SELECT * FROM INDIA WHERE STATE='RAJASTHAN' OR STATE='BIHAR'

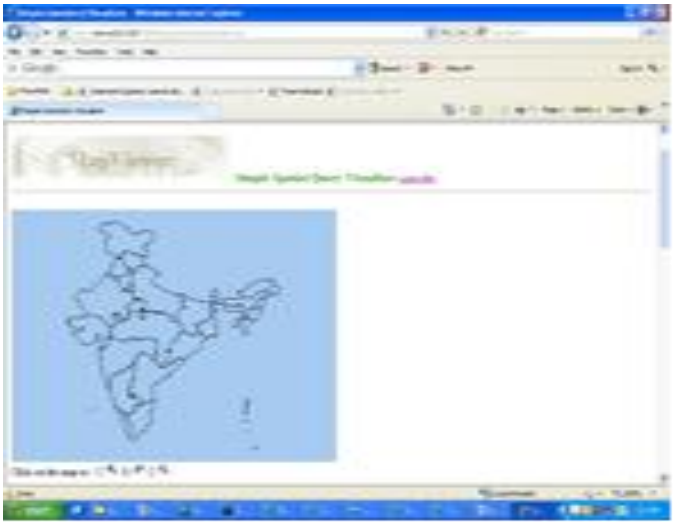

Fig 1: Output of Query 1

Fig 2: Output of Query 2

4. DATABASE PRIMITIVES FOR SPATIAL DATA MINING

General purpose data mining tools, such as Clementine, See5/C5.0, and Enterprise Miner, are designed to analyze large commercial databases. Although these tools were primarily designed to identify customer-buying patterns in market basket data[19]. They have also been used in analyzing scientific and engineering data, astronomical data, multi-media data, genomic data, and web data. Extracting interesting and useful patterns from spatial data sets is more difficult than extracting corresponding patterns from traditional numeric and categorical data. It is due to the complexity of spatial data types, spatial relationships, and spatial autocorrelation[13][17][18].

As it has been stated that the major difference between mining in relational database and mining in spatial database is that the attributes of the neighbors of some object of interest may have an influence on object itself[20]. Thus our database primitives are based on the concept of spatial neighborhood relations.

4.1 Neighborhood Relation

The neighborhood relation says that the mutual influence between two objects depends on factors such as the topology, the distance or the direction between the objects. For example a polluted pond can cause different degree and different type of disease in the neighborhood location or a a unit of industry can cause different degree of pollution in the neighborhood location. The **topological**, **distance** and **directional** relation are the binary relation i.e. they are the relations between pairs of objects [12]. We use set of points as a generic representation of spatial objects [3].

Topological Relation: The topological relations [9] between two objects A and B are derived from the nine intersections of the interiors, the boundaries and the complements of A and B with each other. The relations could be A *disjoint* B, A *meets* B, A *overlaps* B, A *equals* B, A *covers* B, A *covered-by* B, A *contains* B, A *inside* B.

Distance relation: Let *dist* be a distance function, let σ be one of the arithmetic predicate <,>, or =, let c be a real number and let O_1 and O_2 be spatial objects i.e. $O_1, O_2 \in 2^{Points}$. Then a distance relation A $distance_{\sigma\ c}$ B holds iff $dist(O_1,O_2)\ \sigma\ c$.

Direction Relation: To define direction relations O_2RO_1 we distinguish between the source object O_1 and destination object O_2 of the direction relation R. There are several possibilities to define direction relations depending upon the number of points they consider in the source and destination. We define the direction relation of two spatially extended objects using one representative point $rep(O_1)$ of the source object O_1 and all points of the destination objects O_2. The representative point of a source object may e.g. be the *centre* of the object. This representative point is used as the ordering of a virtual coordinate system and its quadrants define the directions.

Furthermore the topological, distance and direction relations may be combined by the logical operator ∧ (and) as well as ∨ (or) to express a complex neighborhood relation.

5. DBMS SUPPORT

The commercial RDBMS (e.g. Oracle) can be used to integrate the basic operations for spatial data mining [10]. All the potentials of these systems can be effectively used in spatial data mining applications. The DBMS supports many spatial index structure e.g. **R-tree** [8]. They are used in speeding up the processing of spatial queries or nearest queries [7]. If the spatial objects are fairly complex, however, retrieving the neighbours of some object this way is still very time consuming. It is due to the complexity of the evaluation of neighbourhood relations on such objects. Furthermore when creating all neighbourhood paths with a given source objects, a very large number of *neighbours* operations have to be performed. Finally many spatial databases are static since there are not many updates on objects[15]. Therefore, **materializing** the relevant

neighbourhood graphs by using the concept of **neighborhood indices** would be quite useful in executing spatial query.

5.1 Creating Neighborhood Index

To create a neighbourhood index I_{max}^{DB}, a spatial join on DB with respect to the neighbourhood relation is performed. . For each pair of objects returned by the spatial join we then have to determine the exact **distance**, the **direction** relation and the **topological** relation[12]. The resulting tuples of the form (O_1, O_2, Distance, Direction, Topological) are stored in a relation (A separate relational table) which is indexed by a R-tree on the attribute Object-ID. The newly created relation (Table 3) would look like this:

Table 3: Neighborhood Relation Table

Object-ID	Neighbour	Distance	Direction	Topology
O_1	O_2	3.4	Northeast	disjoint
O_1	O_3	0	Northwest	disjoint

When an update occurs in the database it is not required to rebuilt a neighbourhood index from scratch. Instead, entries can be "incrementally" inserted or deleted from a neighbourhood index.

The database primitive is used on top of the commercial database Oracle 10g and it is used for experimental performance evaluation. If the average number of neighbours is very large for a spatial neighbourhood relation, we also have to consider the size of the neighbourhood index.

6. CONCLUSION

Here I have presented the approach of materializing the geographic information and using the potential of commercial database for storing spatial information and using neighbourhood graphs and paths. A small set of basic operations on these graphs and paths were defined as database primitives which can be used for spatial data mining applications (spatial clustering, spatial characterization, spatial trend detection, and spatial classification).

6.1 Issues for Future Research

As discussed earlier the database primitives were implemented on top of the commercial DBMS. It is true that the system overhead imposed on this is large but the applicability and efficiency of data mining task would increase. Different thematic layer of application attributes can be built on the neighborhood relational table In my further research on state wise spread of epidemiology like AIDS and its trend in India, various thematic layers of the factors like population, human development index, gross decadal migration, literacy ratio etc were considered. Further in some spatial database the dimension of time plays an important role so data mining in such spatio-temporal database is promising area of further research. For example analyst may be interest in learning the spatial-temporal trend of the spread of any epidemiology in country.

REFERENCES

[1] Agrawal, R., Imielinski, T., and Swami, A. 1993. Database mining: A performance perspective. IEEE Transactions on Knowledge and Data Engineering, 5(6):914–925.

[2] Bill, F. 1991. Fundamentals of Geographical Information Systems: Hardware, Software and Data (in German). Heidelberg, Germany: Wichmann Publishing.

[3] Egenhofer, M.J. 1991. Reasoning about binary topological relations. In Proc. 2nd Int. Symp. on Large Spatial Databases, Zurich, Switzerland, pp. 143–160.

[4] Ester, M., Kriegel, H.-P., and Sander, J. 1997. Spatial data mining: A database approach. In Proc. 5th Int. Symp. on Large Spatial Databases, Berlin, Germany, pp. 47–66.

[5] Ester, M., Frommelt, A., Kriegel, H.-P., and Sander, J. 1998. Algorithms for characterization and trend detection in spatial databases. In Proc. 4th Int. Conf. on Knowledge Discovery and Data Mining, New York City, NY, pp. 44–50.

[6] Fayyad, U.M.J., Piatetsky-Shapiro, G., and Smyth, P. 1996. From data mining to knowledge discovery: An overview. In Advances in Knowledge Discovery and Data Mining. Menlo Park: AAAI Press, pp. 1–34.

[7] Gueting, R.H. 1994. An introduction to spatial database systems. VLDB Journal Special Issue on Spatial

Database Systems, 3(4).

[8] Guttman, A. 1984. R-trees: A dynamic index structure for spatial searching. In Proc. ACM SIGMOD Int. Conf. on Management of Data, pp. 47–54.

[9] Koperski, K. and Han, J. 1995. Discovery of spatial association rules in geographic information databases. In Proc. 4th Int. Symp. on Large Spatial Databases (SSD '95), Portland, ME, pp. 47–66.

[10] Koperski, K., Adhikary, J., and Han, J. 1996. Knowledge discovery in spatial databases: Progress and challenges. In Proc. SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery. Technical Report 96-08, University of British Columbia, Vancouver, Canada.

[11] Koperski, K., Han, J., and Stefanovic, N. 1998. An efficient two-step method for classification of spatial data. In Proc. Symposium on Spatial Data Handling (SDH '98),

[12] Lu, W. and Han, J. 1992. Distance-associated join indices for spatial range search. In Proc. 8th Int. Conf. on Data Engineering, Phoenix, AZ, pp. 284–292.

[13] Ng, R.T. and Han, J. 1994. Efficient and effective clustering methods for spatial data mining. In Proc. 20th Int. Conf. on Very Large Data Bases, Santiago, Chile, pp. 144–155.

[14] Rotem, D. 1991. Spatial join indices. In Proc. 7th Int. Conf. on Data Engineering, Kobe, Japan, pp. 500–509. Sander, J., Ester, M., Kriegel, H.-P., and Xu, X. 1998. Density-based clustering in spatial databases: A new algorithm and its applications. In Data Mining and Knowledge Discovery, 2(2). Valduriez, P. 1987. Join indices. ACM Transactions on Database Systems, 12(2):218–246.

[15] Ester Martin,Hans-Peter Kriegel,Sander Jorg, 2001 Algorithms and Applications for Spatial data Mining, Published in Geographic Data Mining and Knowledge Discovery, Research Monographs in GIS, Taylor and Fransis.

[16] Lianying Sun ,Suping Peng,Dezheng Zhang .Spatial data mining based on supermap model[J]. Computer engineering and applications, 2002,(11).30- 32.

[17] Haiyan Zhou,Jiayao Wang,Sheng Wu .Spatial data mining technique and application [J]. Bulletin of surveying and mapping, 2002,(2).11- 13.

[18] Deren Li,Shuliang Wang,Deyi Li et al. Theories and Technologies of Spatial data mining and knowledge Discovery[J]. Geomatics and Information Science of Wuhan Universtity, 2002,6.221- 233.

[19] Ladnee R,Petry F E ,Cobb M A .Fuzzy Set Approaches to Spatial Data Mining of Association Rules[J].Transin GIS, 2003,7(1).123-138.

[20] LiuYu, QuBo, ZhuZhongying, ShiSongjiao. A Study on Theory and Methodology of Spatial Data Mining [J]. MicrocomputerApplications, 2000,8.15 - 18.