

Implementation of X-Tree Structure for Multidimensional Data Ranking

Kavitha Raju
Ph.D Scholar,
St. Peter's University
Avadi, Chennai,
Tamilnadu, India

S.K.Srivatsa
Senior Professor,
St. Joseph's College of Engineering,
Old Mahabalipuram Road,
Chennai, India

ABSTRACT

In this modern world, travelling has become as normal routine for most of the people in the cities. At the same time, frequency of happening of accidents also increasing alarmingly. Also one cannot avoid health care and should take proper health checks in hospitals periodically. At the time of accidents, to save precious lives, we must know better hospitals nearby. In that context, in this paper X-tree ^[1] structure has been implemented for multidimensional data ranking. Problem definition is explained in section1, importance of X-tree is explained in section2, Proposed System explained in section3, Advantages of Proposed system explained in Section 4, Implementation and sample data are explained in sections 5 and 6, module description and other constraints have been explained in the remaining sections.

General Terms

Data structures, Algorithm, Multidimensional data

Keywords

X-tree, data ranking, quality ranking.

1. INTRODUCTION

A spatial preference query ranks ^[2]objects based on the qualities of features in their spatial. A user may want to rank the hospital with respect to the appropriateness of their location, defined after aggregating the qualities of other features (for example service type, hospital quality etc..) within the spatial neighborhood. Such a neighborhood concept can be specified by the user via different functions. It can be an explicit circular region within a given distance from the hospital. Another intuitive definition is to assign higher values based on their proximity to the hospital. This paper defines spatial preference queries and proposed appropriate indexing techniques and search algorithms for them. An extensive evaluation of these methods on both real and synthetic data reveals that an optimized branch and bound solution are efficient and robust with respect to different parameters incorporating the applicable criteria that follow.

2. PROBLEM DEFINITIONS

A spatial preference query ranks objects based on the qualities of features in their spatial neighborhood. This project is used to identify the nearby hospital from their location. For example, using a hospital database, a customer may want to rank the hospitals with respect to the appropriateness of their

location, defined after aggregating the qualities of other features within their spatial neighborhood. Such a neighborhood concept can be specified by the user via different functions. It can be an explicit circular region within a given distance from the hospitals. Another intuitive definition is to assign higher weights to the features based on their proximity to the hospital. In this paper, we formally define spatial preference queries and propose appropriate ranking techniques. The X-tree indexing mechanism enables users to add their own desirable feature in future. It will expand the scope of this project.

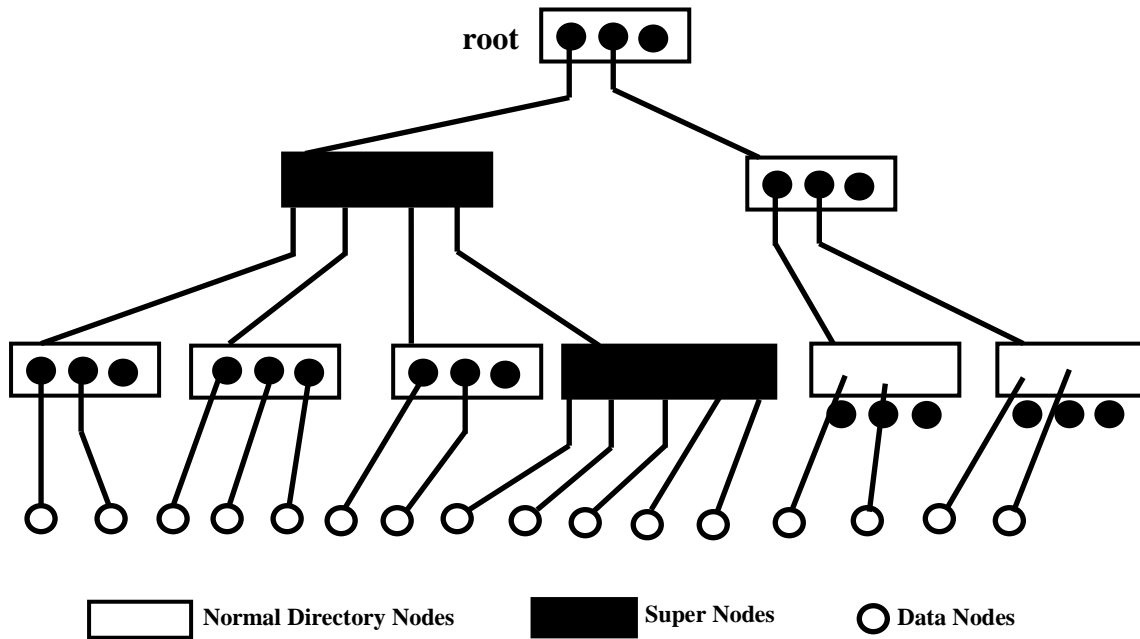
3. PROPOSED SYSTEM

- Spatial ranking, which orders the objects according to their distance from a reference point.
- Quality ranking, these orders the objects of an aggregate function on their stated values. State values as indicated by examples, this new query have a wide range of applications in service recommendation and decision support systems. To our knowledge, there is no efficient solution for processing the state values preference query existing.
- Neighbor ranking, these orders the objects of an aggregate function on their stated values. State value passed by nearest neighbor in that hospital location.
- This will consider about the hospital based on the suggestion or comment passed by neighbor entries such as fair, service etc., this module is used to find best and good hospital based on neighbor comment.

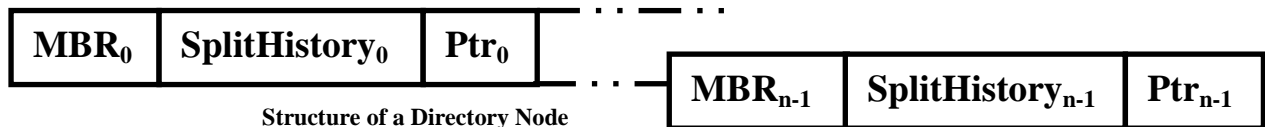
4. ADVANTAGES OF PROPOSED SYSTEM

- Existing methods rank the data for the given single key but proposed method allows additional user key.
- In existing system it is very expensive for large datasets. In proposed system, collection of database is very easy.
- It is easy to access the information; it also reduces the manpower consumption.
- It reduces the time for searching the required information

5. X-TREE STRUCTURE



Structure of the X-Tree



Structure of a Directory Node

Fig 1: Structure of the X-Tree

6. FEATURES OF X-TREE

When segregating the statistics details by area or by group some points may be similar or same. It will be useful to avoid such overlaps. So the computer memory and time will be saved. Another specialty of this feature is the super node can make the extendable directory node whenever required. Also instead of scanning multiple paths, scanning linear path will be easy. Figure 1 shows an architecture design of X-Tree^[1]. Figure 2 explains the node generation of X-Tree^[1].

```
int X_DirectoryNode::insert(DataObject obj, X_Node **new_node)
{
    SET_OF_MBR *s1, *s2;
    X_Node *follow, *new_son;
    int return_value;

    follow = choose_subtree(obj); // choose a son node to insert obj into
    return_value = follow->insert(obj, &new_son); // insert obj into subtree
    update_mbr(follow->calc_mbr()); // update MBR of old son node
    if (return_value == SPLIT) {
        add_mbr(new_son->calc_mbr()); // insert mbr of new son node into current node
        if (num_of_mbrs() > CAPACITY) { // overflow occurs
            if (split(mbrs, s1, s2) == TRUE) {
                // topological or overlap-minimal split was successful
                set_mbrs(s1);
                *new_node = new X_DirectoryNode(s2);
                return SPLIT;
            }
            else // there is no good split
            {
                *new_node = new X_SuperNode();
                (*new_node)->set_mbrs(mbrs);
                return SUPERNODE;
            }
        }
    }
    else if (return_value == SUPERNODE) { // node 'follow' becomes a supernode
        remove_son(follow);
        insert_son(new_son);
    }
    return NO_SPLIT;
}
```

X-tree Insertion Algorithm for Directory Nodes.

Fig 2: X-Tree Algorithm

7. IMPLEMENTATION OF X-TREE FOR HOSPITAL RANKING

Automated tree structure has been raised the directory nodes as per keys, each individual directory has been considered as a linear path and its nodes are referred by its search key node in a linear manner. The following picture Figure 3 shows the sample data.

Hospital Name	Hospital Contact	Hospital Services	Hospital Rank	Hospital Years of Operation	Specialized Hospital Location	Hospital Address
Apollo Hospital	902044132	24 hrs	12th rank in TN	25 years	ENT, CARD	Chennai, India
Fortis Hospital	902044132	24 hrs	12th rank in TN	25 years	ENT, CARD	Chennai, India
Medanta Hospital	902044132	24 hrs	12th rank in TN	25 years	ENT, CARD	Chennai, India
...

Fig 3: Hospital Data Generation using X-Tree

8. MODULES

- Spatial Ranking
- Quality Ranking
- Neighborhood Ranking

9. EXPERIMENTAL RESULTS

9.1 Spatial Ranking

Spatial ranking is a method which orders the objects according to their distance from a reference point. This module will give a spatial ranking of the given starting point to ending point: Here one can find a route map of the two locations and suggestion list to take travel based on Google map coordinates like left, right, north and South Pole direction from the source coordinates. This module is the important task of this project. Based on this module a patient or his well wisher can find the address and location of the required hospital easily and quickly.

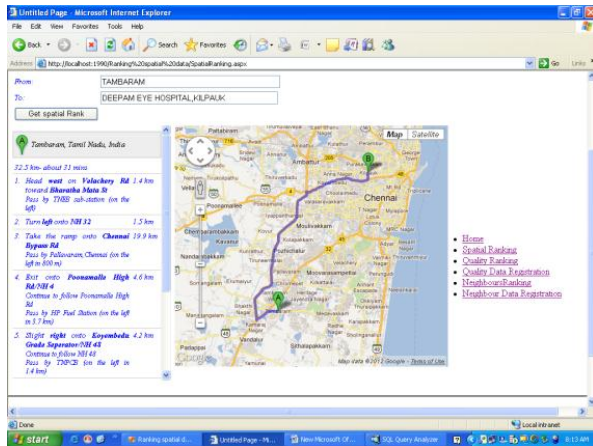


Fig 4: Location preference using a map

Figure 4 shows the spatial ranking for the hospitals. It is easy to find location of the particular hospital which will be searched by the user. It is not only displayed the hospital names, but also shows the direction to the hospital which can be searched by the user.

9.2 Quality Ranking

Quality ranking is one which orders the objects by an aggregate function on their stated values. State values as indicated by examples, have a wide range of applications in service recommendation and decision support systems query. To our knowledge, there is no efficient solution for processing the state values preference query existing.

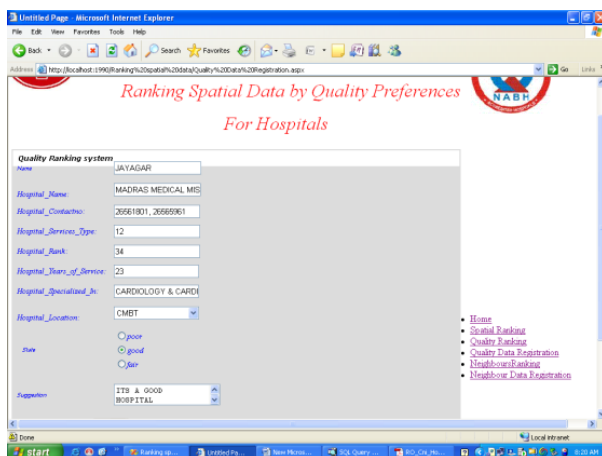


Fig 5: Quality Ranking of Hospitals

Figure 5 shows Quality of the hospitals. Here user also comment the quality of the hospital and can give rank for it.

9.3 Neighbor Ranking

Neighbor ranking is which orders the objects of an aggregate function on their state values. State value passed by nearest neighbor to the hospital area. This will consider the suggestion or comments by a neighbor about the hospital like fair, good and bad. This module will be used to find a better hospital based on neighbor comment.

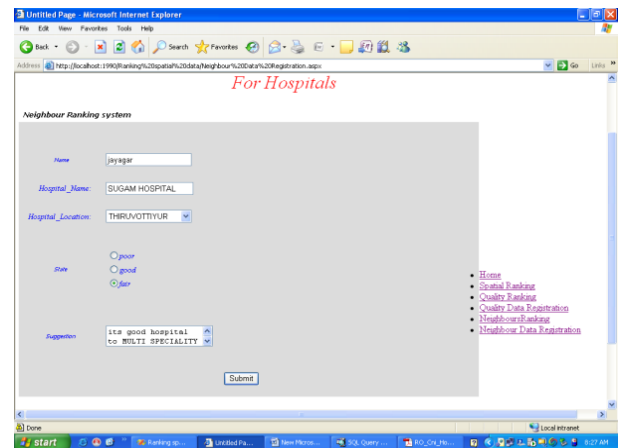


Fig 6: Neighbor ranking of hospitals

Figure 6 shows the consideration about the hospital based on the suggestions or comments by neighbor entries.

10. CONCLUSIONS

In this paper, we are explaining our experimental result which was carried out for the hospital data to implement X-tree structure. Spatial preference queries provide a ranking for spatial objects based on qualities of features in their neighborhood. The neighborhood of an object p is captured by the scoring function:

- (i) The range score restricts the neighborhood to a crisp region centered at p , whereas
- (ii) The influence score relaxes the neighborhood to the whole space and assigns higher values to locations closer to p . Spatial data performance analysis was already explained in.

Thus the X-tree structure is enabled multi-dimensional searching for hospital data and very useful for the persons who are in search of hospitals as per their requirement. This X-Tree concept can be extended for virtual databases in future.

11. ACKNOWLEDGMENTS

Our sincere thanks to MurugaPoopathiand Jayagar who have helped towards development of thispaper and data collection of the project.

12. REFERENCES

- [1] A. Stefan Berchtold Daniel, The X-tree: An Index Structure for High-Dimensional Data, Keim Hans-Peter Kriegei Institute for Computer Science, University of Munich, Oettingenstr. 67, D-80538 Munich, Germany.
- [2] M. L. Yiu, X. Dai, N. Mamoulis, and M. Vaitis, “Top-k Spatial Preference Queries,” Proc. IEEE Int’l Conf. Data Eng. (ICDE), 2007.
- [3] G. R. Hjaltason and H. Samet, “Distance Browsing in Spatial Databases,” ACM Trans. Database Systems, vol. 24, no. 2, pp. 265-318, 1999.
- [4] R. Weber, H. J. Schek, and S. Blott, “A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces,” Proc. Int’l Conf. Very Large Data Bases (VLDB), 1998.
- [5] K. S. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, “When is ‘Nearest Neighbor’ Meaningful?” Proc. Seventh International Conference Database Theory (ICDT), 1999.
- [6] I. F. Ilyas, W.G. Aref, and A. Elmagarmid, “Supporting Top-k Join Queries in Relational Databases,” Proc. 29th Int’l Conf. Very Large Data Bases (VLDB), 2003.
- [7] D. Papadias, P. Kalnis, J. Zhang, and Y. Tao, “Efficient OLAP Operations in Spatial Data Warehouses,” Proc. Int’l Symp. Spatial and Temporal Databases (SSTD), 2001.