

INCLUSION OF ROAD NETWORK IN THE SPATIAL DATABASE FOR FEATURES SEARCHING USING DYNAMIC INDEX

¹Dr. S. Sivasubramanian, M.Tech., Ph.D, Professor: sivamdu2001@yahoo.com

²Ms.M.Samundeeswari, B.Tech, (M.Tech): mssamu.me@gmail.com

Department of Computer Science and Engineering
Bharath University Chennai-73

ABSTRACT

Spatial database systems manage large collections of geographic entities, which apart from spatial attributes contain spatial information and non spatial information (e.g., name, size, type, price, etc.). An attractive type of preference queries, which select the best spatial location with respect to the quality of facilities in its spatial area. Given a set D of interesting objects (e.g., candidate locations), a top-k spatial preference query retrieves the k objects in D with the highest scores. The featured score of a given object is derived from the quality of features (e.g., location and nearby features) in its spatial neighborhood. For example, using a landed property agency database of flats for Sale, a customer may want to rank the flats with respect to the appropriateness of their location, defined after aggregating the qualities of other features (e.g., restaurants, bus stop, hospital, market, school, etc.) within their spatial neighborhood. This neighborhood concept can be defined by different functions by the user. It can be an explicit circular region within a given distance from the flat. Another sensitive definition is to assign higher rates to the features based on their proximity to the land. In this paper, we formally define spatial preference queries and propose suitable dynamic index techniques and searching algorithms for them. We extend [1] results with dynamic index structure in order to accommodate time - variant changes in the spatial data. In my current work is the top-k spatial preference query on road network, in which the distance between object and road is defined by their shortest path distance.

Keywords — *spatial information, spatial location*

1. INTRODUCTION

Data Mining (the analysis step of the Knowledge Discovery in Databases process, or KDD), a relatively young and interdisciplinary field in astronomy, business, computer science, economics, physics, social sciences and others is the process of discovering new patterns from large data sets involving methods from statistics and artificial intelligence but also database management. Spatial database systems manage large collections of geographic entity, which apart from spatial attributes contain spatial information and non-spatial information (e.g., name, size, type, price, location etc.). In this paper, an interesting type of partiality query, which select the best spatial location with respect to the excellence of conveniences in its spatial area. Given a set D of interesting objects (e.g., candidate locations), a top-k spatial preference query retrieves the k objects in D with the highest scores. The score of a location is defined by the kind of quality in its spatial locality. A customer may want to rank the contents of this database with respect to the quality of their locations, quantified by aggregating non-spatial characteristics of

other features (e.g., restaurants, cafes, hospital, market, etc.) in the spatial neighbourhood of the flat (defined by a spatial range around it). Quality may be subjective and query-parametric.

Fig. 1a illustrates the locations of an object data set D (hotels) in white, and two feature data sets: the set F1 (restaurants) in gray, and the set F2 (cafes) in black. Quality points are labelled by excellence values that can be obtained from rating providers (e.g., <http://www.zagat.com/>). For the ease of argument, the qualities are normalized to values in [0, 1]. The score $T(p)$ of a hotel p is defined in terms of: 1) the maximum quality for each feature in the neighbourhood region of p , and 2) the aggregation of those qualities.

- (a) Range score
- (b) Influence score

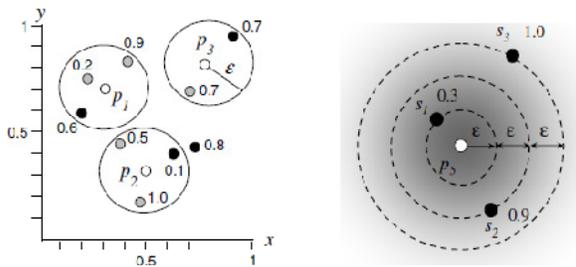


Fig 1. Examples of top-k spatial preference queries

A simple score instance, called the range score, binds the neighbourhood region to a circular region at p with radius (shown as a circle), and the aggregate function to SUM. For instance, the maximum quality of gray and black points within the circle of p_1 are 0.9 and 0.6, respectively, so the score of p_1 is $T(p_1) = 0.9 + 0.6 = 1.5$. Similarly, we obtain $T(p_2) = 1.0 + 0.1 = 1.1$ and $T(p_3) = 0.7 + 0.7 = 1.4$. Hence, the hotel p_1 is returned as the top result. In detail, the semantics of the combined function is relevant to the Customer's query. The SUM function challenges to set of scales the overall qualities of all features. For the MIN function, the top result becomes p_3 , with the score $T(p_3) = \min \{0.7, 0.7\} = 0.7$. It ensures that the top result has reasonably high qualities in all features. For the MAX function, the top result is p_2 , with $T(p_2) = \max \{1.0, 0.1\} = 1.0$. It is used to optimize the superiority in exacting feature, but not necessarily all of them. The neighbourhood region in the above spatial preference query can also be defined by other score functions. An important score is the influence score. As opposed to the crisp radius ϵ constraint in the range score, the influence score smoothens the effect of ϵ and assigns higher weights to cafes that are closer to the hotel. Fig. 1b shows a hotel p_5 and three cafes s_1, s_2, s_3 (with their quality values). The circles have their radius as multiples of ϵ . Now, the score of a cafe s_j is computed by multiplying its quality with the weight 2^{-j} , where j is the order of the smallest circle containing s_j . For example, the scores of s_1, s_2 and s_3 are $0.3/2^1 = 0.15$, $0.9/2^2 = 0.225$, and $1.0/2^3 = 0.125$, respectively. The influence score of p_5 is taken as the highest value (0.225). Traditionally, there are two types of ranking lands: First one spatial ranking, which orders the objects based on their distance and score from a reference feature, and second Non-spatial ranking, which orders the objects by an combined method on their non-spatial values. Our top-k spatial preference query integrates these two types of ranking in an spontaneous way. A brute-force approach for evaluating it is to calculate the scores of all objects in D and select

the top-k land. This technique, however, is expected to be very costly for large input data sets. In this paper, we propose alternative techniques that aim at minimizing the I/O accesses to the object and feature data sets, while being also computationally efficient. Our techniques apply on spatial-separation access functions and work out score bounds for the objects indexed by them, which are used to effectively trim the try to find space. Specifically, we contribute the branch-and-bound (BB) algorithm and the feature join (FJ) algorithm for efficiently processing the top-k spatial preference query.

2. PROBLEM IDENTIFICATION

A Spatial partiality query ranks lands based on the qualities of features in their spatial zone. Feature refers to a class of objects in a spatial plan such as specific conveniences or services. A customer may want to rank the contents of this database with respect to the quality of their locations, quantified by aggregating non-spatial characteristics of other features in the spatial neighborhood of the flat. Quality may be subjective and query-parametric. For example, using a landed property agency database of flats for Sale, a customer may want to rank the flats with respect to the appropriateness of their location, top-10 flats with the largest sizes and lowest prices, defined after aggregating the qualities of other features (e.g., restaurants, bus stop, hospital, market, school, etc.) within their spatial neighbourhood. In spatial databases, ranking is often associated to nearest neighbor (NN) recovery. Given a query location, we are interested in retrieving the set of nearest objects to it that qualify a condition (e.g., restaurants). Assuming that the set of interesting objects is dynamic indexed by an R-tree, we can apply distance score bounds and traverse the index in a branch-and-bound approach to obtain the answer.

Therefore, we propose extend results with dynamic index structure in order to accommodate time-variant changes in the spatial data using searching algorithms. Spatial ranking, which orders the objects based on their distance and score from a reference feature, and second Non-spatial ranking, which orders the objects by an combined method on their non-spatial values.

3 METHOD ANALYSIS

3.1 Feature-join new Algorithm

Input:

Wk = New Min-heap

k- Size

(f1, f2, fm)- Features Tree

D- Data Objects

O_i-Road Proximity

T (e)-Upper Bound Score For R-Tree Entry

γ - Existing Score in database

N_c – Child Node

e - Entry

H- Tree

Output:

T(e)-Upper Bound Score result

Algorithm Feature Join (Tree D, Trees F1, F2,, Fm)

1. H:=new max-heap (combination score as the key);
2. insert (F1.root,F2.root, ,Fm. root) into H;
3. while H is not empty do
4. deheap (f1; f2;. . . ; fm)from H;
5. if for any $c \in [1,m]$, f_c points to a leaf node then
6. for $c:=1$ to m do
7. read the child node L_c pointed by f_c ;
8. Find Result(D. root, L1; ;Lm);

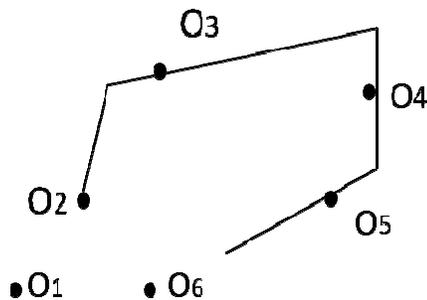
Algorithm Find Result (Node N, Nodes L1; ;Lm)

1. for each entry $e \in N$ do
2. if N is non-leaf then
3. compute T (e) by entries in L1,,Lm;
4. if $T(e) > \theta$ then
5. read the child node N' pointed by e;
6. Find Result(N' , L1, ,Lm);

3.2 Method Description

- An alternative method for evaluating a top-k spatial preference query is to perform a multi way spatial join on the feature trees F1;F2; . . . ;Fm to obtain combinations of feature points which can be in the neighborhood of some object from D.
- The score of a combination $f_1; f_2; . . . ; f_m$ as defined upper bound of the scores of all combinations $s_1; s_2; . . . ; s_m$ of feature points, such that s_c is located in the sub-tree of f_c
- We progressively deheap the combination with the largest score. If all its entries point to leaf nodes, then we load these nodes L1; . . . ; Lm and call Find Result to traverse the object R-tree D and find potential results

4. CURRENT RESEARCH WORK



Insertion of road network can be used in features database.

Given threshold T, Road Proximity is define as follows

$$\begin{aligned} \text{Road Proximity (O}_i\text{)} &= 1 \text{ if } d_i < T \\ &= 0, \text{ otherwise} \\ &= \sphericalangle \text{ if } d_i < T/2 \end{aligned}$$

O1, O2, O3, O4, O5, O6 are objects near in road.

Theme is

(o1,1),
(o2,1),
(o3,1),
(o4,1),
(o5,0),
(o6,0)

1 Means nearest distance

0 means long distance

We can get the shortest path from the reference of road network using the road proximity.

5. CONCLUSION

In this paper, we studied top-k spatial preference queries, which provide a novel type of ranking for spatial objects based on qualities of features in their neighbourhood. The neighbourhood of an object p is captured by the scoring function: 1) the range score restricts the neighbourhood to a crisp region centred at p, whereas 2) the influence score relaxes the neighbourhood to the whole space and assigns higher weights to locations closer to p. We presented five algorithms for processing top-k spatial preference queries. The baseline algorithm computes the scores of every object by querying on feature data sets. The algorithm GP is a variant of SP that reduces I/O cost by computing scores of objects in the same leaf node concurrently. The algorithm BB derives upper bound scores for nonleaf entries in the object tree, and prunes those that cannot lead to better results. The algorithm BB* is a variant of BB that utilizes an optimized method for computing the scores of objects (and upper bound scores of nonleaf entries). The algorithm FJ performs a multi way join on feature trees to obtain qualified combinations of feature points and then search for their relevant objects in the object tree. Based on our experimental findings, BB* is scalable to large data sets and it is the most robust algorithm with respect to various parameters. However, FJ is the best algorithm in cases where the number m of feature data sets is low and each feature data set is small. In my future work is the top-k spatial preference query on road network, in which the distance between object and road is defined by their shortest path distance.

6. REFERENCES

- [1] Man Lung Yiu, Hua Lu, Nikos Mamoulis, and Michail Vaitis, "Ranking Spatial Data by Quality Preferences" 2011.
- [2] M. L. Yiu, X. Dai, N. Mamoulis, and M. Vaitis, "Top-k Spatial Preference Queries," in ICDE, 2007.
- [3] G. R. Hjaltason and H. Samet, "Distance Browsing in Spatial Databases," TODS, vol. 24(2), pp. 265–318, 1999.
- [4] K.S. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When is 'Nearest Neighbor' Meaningful?" Proc. Seventh Int'l Conf. Database Theory (ICDT), 1999.
- [5] 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.
- [6] N. Mamoulis, M.L. Yiu, K.H. Cheng, and D.W. Cheung, "Efficient Top-k Aggregation of Ranked Inputs," ACM Trans. Database Systems, vol. 32, no. 3, p. 19, 2007.

Author profile

1) Mr.S.Sivasubramanain, M.Tech(CSE)., pursuing Ph.d. as an Asst.professor from Department of CSE of Bharath University, Chennai, Tamil Nadu. He has more than 10 years of teaching and research experience and his areas of specialization are mobile computing, Database Management System, Computer Networks, Networks Security and Data Mining.



2) Ms. M. Samundeeswari received her Bachelor of Technology in Information Technology from Anna University and Pursuing Master of Technology in Computer Science and engineering from Bharath University, Chennai.

