Improvement of Top-k query algorithm for moving objects in road networks

Zhen Wang

The Information Technology Institute, Liaodong University, China

ldxywz@163.com

Keywords: Road network; Top-k query algorithm; LBS; Security section;

Abstract. The road network environment, Top-k query algorithm of spatial object query efficiency is a measure of the important indicators of query performance. Because in the practical application of query location is often mobile, using the traditional snapshot query to deal with the efficiency of mobile query point Top-k query is poor. In this paper, the improved algorithm combines the extended tree and the Safe Zone, which reduces the number of edges of the traverse network, and improves the efficiency of the query and the communication cost.

Finally, the experimental results on real road network data set show that the algorithm proposed in this paper is efficient in different parameter conditions.

Introduction

At present, the information query system with huge amount of data processing, a query can return hundreds of millions of results, but the user may only be the top results concern. Therefore, in order to improve the user experience, how to quickly and accurately before returning K results become the focus of research.

Related research

In 2013, R. Zhong [1] K proposed nearest neighbor query method for spatial object in road network environment. A balanced tree index structure G tree is proposed. Most of the existing research based on road network is to query the nearest neighbor problem, only to return to the nearest object based on distance, this kind of problem can be attributed to the Internet Top-k query. Y.-K. Huang [2] proposed a method of continuous monitoring of mobile objects in the road network. Rocha-Junior[3]proposed a static Top-k query method in road network environment.

Due to the expansion of the scale of spatial data, the characteristics of diversity, how to effectively deal with the spatial data query has become a hot spot. Kolahdouza [4] proposed to solve the spatial network of continuous k nearest neighbor query Upper Bound Algorithm algorithm, According to the method, this method returns $k+1$ spatial object according to the query location, then calculates an upper bound for the k nearest neighbor query which is used to exclude the position. Wu and others [5] proposed a Top-k based space keyword continuous query method, using a Safe Zone concept to ensure the correctness of the results of the query. In this paper, combined with the expansion of the tree and Safe Zone, the way to reduce the number of times the edge of the network to improve the efficiency of the query.

Top-k continuous query optimization algorithm for road network space

Algorithm. The algorithm uses the expansion tree traversal network edge, the expansion tree in continuous query efficient processing problems which play an important role.The specific process is snapshot algorithm:

(a) First, update the value of the first k objects in the results set R_q. Positioning an end node n' of e side , Set s as the root node of tree expansion T , Set n' to its child node. Put s and n' insert into the queue N and mark it as already accessed. In the edge of the *e(s,n')* query the candidate that value is lower than ε, and updates R_q and ε. From the priority queue N and dequeue s node n recently, And to handle the adjacent nodes which have not been visited. The termination condition of the algorithm is that all the network has been extended, or the minimum network distance from the rest of the object generated by the lower score is higher than or equal to the current k of the value of the object has been found. The lower limit value of the node n can be obtained by using s and n of the network distanceand the maximum text correlation value($q = 1$).Therefore, the lower limit value of a node is greater than or equal to ε, Shows that even if there is an object that has not been accessed, the o and the query keywords have the greatest correlation with the text. Its ranking can not be higher than the first k objects in the R_q, That is $\delta(s,o.l) \geq \delta(s,n)$.

(b) The maximum use of each text $t \in q.d$ mapping module stored value $\lambda_{\max}$ to calculate the maximum text correlation $\theta_{\max}$, The lower bound of the value of the minimum network distance and the maximum maximum text correlation value $\theta_{\max}$ is calculated by using the shortest distance between the edge and the query node $\tau_{\min}$. When the lower bound value $\tau_{\min}$ is less than e, the exact value of the object on the edge is calculated and the candidate set C with the value below e is returned.

The (a) part of the algorithm has completed the detailed steps to obtain the results of the snapshot query. The (b) part of the algorithm completes the process of finding candidate sets.

Extended tree. The previous algorithm can obtain the Top-k result of one end node on the edge. However, after obtaining an endpoint of the Top-k result, If the n' from the other end of the node to re traverse the entire space network needs to carry out a lot of unnecessary operations. Due to the calculation result of n' nodes set, n for the root node of the tree tree (n' for the root node) is still effective. Therefore, the expansion tree to avoid traversing some space of the edges of the network.

Safety section. Solve the problem of continuous monitoring of space network query can be used to call the snapshot query algorithm, but too much time. As a result, the security section technique is used, which is an effective part of the Top-k result of a client side. According to the above theory to carry out the calculation of the safety section.

First, consider the object in O_{d1} and O_{r1}, O_{r2} object. For each object o_{d1} that belongs to O_{d1} and each object o_r belongs to O_{r1} or O_{d1}. The distance between the new query location and the object o_{d1} will become larger, And the distance with o_r will become smaller. Can use the formula 3 to calculate the candidate safety section D.

$$a \cdot \left[d\left(q.l, o_{d1}.l\right) + \frac{D}{d_{\max}} \right] + b \cdot [1 - j\left(o_r\right)] + (1 - a - b) \cdot \left[1 - q\left(q.d, o_{d1}.d\right) \right]$$

$$= a \cdot \left[d\left(q.l, o_r.l\right) - \frac{D}{d_{\max}} \right] + b \cdot [1 - j\left(o_r\right)] + (1 - a - b) \cdot \left[1 - q\left(q.d, o_r.d\right) \right] \quad (3)$$

Secondly, consider the object in O_{d2} and the object in O_{r1}, O_{r2}. For each object o_{d1} that belongs to O_{d1} and each object o_r belongs to O_{rl} or O_{d1}. o_{d2} can be replaced only if the client moves to the right side of the o_{d2}, Because the distance from the o_{d2} to the new query location will become larger.

Therefore, according to the formula 4, when the o_{d2} is replaced, the first calculation of an offset segment D', which indicates that the client side to the right side of the o_{d2} mobile how far. After that, the candidate security section is obtained by the formula 3. This process only needs to add the D' to the distance between q and o_{d2}.

$$a \cdot D' + b \cdot [1 - j\,(o_r)] + (1 - a - b) \cdot \left[1 - q\,(q.d, o_{d2}.d)\right]$$
$$= a \cdot \left[d\,(q.l, o_r.l) - d\,(q.l, o_{d2}.l) - D'\right] + b \cdot [1 - j\,(o_r)] + (1 - a - b) \cdot \left[1 - q(q.d, o_r.d)\right]$$

$$D = \left[d\,(q.l, o_{d2}.l) + D'\right] \cdot d_{\max} \tag{4}$$

Simulation Results

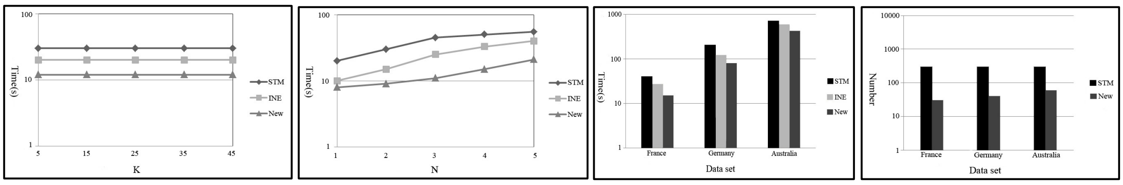
In the experiment, we compare the algorithm with Network Expansion Incremental (INE) algorithm, and test whether the query efficiency is improved by using the extended tree; In addition, compared with Method Straightforward (STM) algorithm, this paper used to test the communication and execution efficiency of the algorithm after the introduction of the security section.Experiments using the French, German, Australian network as a real data set, this paper uses the Brinkhoff generator to generate the trajectory of the mobile query point.

This experiment uses three different scale real road network data sets, and analyzes the effect of the number of *k* on the performance of the algorithm, Effect of *n* number of keywords on the performance of the algorithm, different data sets affect the performance of the new algorithm. The results of experiments are as follows:

(a) As shown in Fig.1(a), the number of queries have three kinds of algorithm execution time effect, this is because the snapshot query execution process to a lot of related objects are not considered in the final results. Therefore, although the value of sustained growth, the number of objects of the three algorithms to consider is also very similar, and no extra calculation more.

(b) As shown in Fig.1(b), when the number of keywords increases the execution time of the algorithm also increased, but the degree is not very obvious. This is because the number of more keywords means more object related to the query. For the above three algorithms, when using the snapshot query algorithm, the expansion stage need to consider more objects.

(c) Fig.1(c) and Fig.1(d) show, with the increase of the size of the data set, the performance of the three algorithms was decreased in different degree. But in this case, the efficiency of the algorithm proposed in this paper in any data set are among the two kinds of algorithm is superior to the other. This is because it uses the expansion tree to effectively avoid the repeated traversal process.



(a) The effect of *K* on the execution time of the algorithm(b) The effect of *N* on the execution time of the algorithm

(c) The effect of data set on execution time of algorithm (d) The effect of data set on communication consumption

Fig.1 Experiment results

Conclusion

The network for moving query point of continuous Top-k query method based on, through the expansion of the tree structure of the INE algorithm and naive STM algorithms were compared, respectively from the keyword query number, query results in different environment, and the number of data sets under analysis query system performance. Experimental results show that after the introduction of extended tree, this algorithm in query execution time, and so on, are better than INE algorithm; after the introduction of the security section of the algorithm in this paper on communication consumption and frequency to lower than STM algorithm.

References

[1] Zhong R, Li G, Tan K L, et al. G-tree: an efficient index for KNN search on road networks[C]//Proceedings of the 22nd ACM international conference on Conference on information & knowledge management. ACM, 2013: 39-48.

[2] Y.-K. Huang, Z.W. Chen and C. Lee, Continuous K-nearest neighbor query over moving objects in road networks, APWeb/WAIM2009, LNCS5446, 2009, pp.28–38.

[3]Rocha-Junior J B, Nørvåg K. Top-k spatial keyword queries on road networks[C]//Proceedings of the 15th international conference on extending database technology. ACM, 2012: 168-179.

[4]Kolahdouzan M R, Shahabi C. Continuous K-Nearest Neighbor Queries in Spatial Network Databases[C]//STDBM. 2004: 33-40.

[5]Huang W, Li G, Tan K L, et al. Efficient safe-region construction for moving top-k spatial keyword queries[C]//Proceedings of the 21st ACM international conference on Information and knowledge management. ACM, 2012: 932-941.