Asian Journal of Science and Applied Technology (AJSAT)
Vol.2.No.1 2014pp1-6
available at: www.goniv.com
Paper Received :05-03-2014
Paper Published:28-03-2014
Paper Reviewed by: 1. John Arhter 2. Hendry Goyal
Editor : Prof. P.Muthukumar

# GEOGRAPHICAL APPROXIMATE STRING SEARCH FOR RETRIEVING ERRORIOUS DATA IN SPATIAL DATABASE

**P.Sudha, A.Kumaresan, K.Vijaya kumar, G.Nanda kumar,**
Dept of CSE,
S.K.P Engineering college
Tiruvannamalai,India,
sudhapalaniskp@gmail.com. kummaresan@gmail.com. vijayktvm@gmail.com.sivanesh09@gmail.com

## ABSTRACT

This work deals with approximate string search in large spatial database. Specifically focus on selectivity estimation for RSAS query in road networks. Selectivity estimation in road network is a union of string selectivity and spatial point selectivity. In this paper we propose a novel adaptive selection method, which is based on grouping technique. String selectivity is achieved by using q-grams and min-wise signature of strings. String similarity is measured by using edit distance metric technique, which is used to calculate threshold value between strings. Spatial point selected by using grouping technique called greedy algorithm. Space complexity of grouping method is based on neighborhood nodes. Effectiveness of this approach is high, when applying in large database.

***Keywords-***Approximate string search,Road network, Selectivity estimation, Spatial database.

## 1. INTRODUCTION

Recent years approximate string search is necessary for large databases and real world applications, when users submitting query with spelling errors, fuzzy search conditions, database contain some certain degree of errors in data. In road network R, we have nodes (V) and edges (E) that is R={V,E},each point $p_i$ resides on edge ($n_i$,$n_j$)belongs to E and V. Each node has unique id for identification.

Feifei li et al. uses range query to select search range r, which has lower and upper bounds of range. For geographical approximate string search range query is not suitable, because spatial database contains data about both spatial and string information. So we dub range query to RSAS (Road network Spatial Approximate String) query. Key issue in RSAS query is edit distance metric, between strings used to define the similarity and to find threshold. An edit operations are insertion, deletion, substitution of single character, swapping. For RSAS query, basic solution is based on dijkstra algorithm, which is degrades the performance quickly. Then another more popular solution is string matching index, which is only able to estimate strings only, not a spatial point. [1] Proposed RSASSOL, it's accurately provides answer to the query.

**Table.1.Frequently used notations**

| Symbol | Descraiption |
|---|---|
| P | Set of points with string |
| T | Edit distance threshold |
| B | Buckets |
| N | Set of nodes |
| R | Range query |
| S | Strings |
| G=(V,E) | Road network with vertex(edge) set V(E) |
| ρ (A,B) | Set resemblance of two sets A and B |
| $V_R$ | Set of reference nodes |
| ε(s1,s2) | Edit distance between strings s1 and s2 |
| d(q,p) | Distance between query point and point |

Road network data are stored at disk. Two separate files are maintained for storing adjacency list and points, each is separately indexed by using B+-tree. Grouped the nodes based on their connectivity and distance. For each node $n_i$ stored its distance of reference nodes in $V_R$, which is denoted as RDISTi and each one has id separately which is called as adjacency node id. From the one node each of the reference nodes and distance between nodes are stored at adjacency list file, its denoted as NDIST ($n_i$,$n_j$).Node id then distance between nodes and value is presented at pointer file. At the pointer file

length of the edge and node id is stored first, then offset distance and corresponding string is stored.

RSASSOL algorithm is proposed for query processing, that framework has five steps. First step is finding all sub graphs intersect with a query range r, it's achieved by using dijkstra algorithm, search ends when reach the starting node. Second step is using filter tree retrieving points with string it may be similar to query string. Third step is prune away the candidate points by calculating upper and lower bounds of their distance to query point, lower and upper bounds calculated by using MPALT algorithm and also used to compute shortest path. Fourth step is cut the candidate points using edit distance metric between query string and candidate string. Fifth step is check the exact distance of remaining candidate point to query point. At the end of query processing they got accurate result. Selectivity estimation is leaved as an open problem for future work [1].

Fig.1 shows the synthetic real dataset PN (Pondicherry), Points are randomly selected from corresponding dataset. The RSAS query result is based on range and threshold value. Selection result is p5, p6, p13, exact result using RSASSOL [1] is p13.This is achieved by using edit distance pruning and min wise signature.

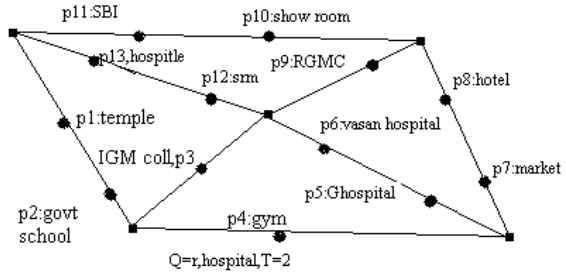


Fig.1.example for RSAS query.

So in this paper we propose a novel selection method which is based on grouping method. The main valuable point is, selected both string and spatial point in road network. Section 2 is about related work, section3 explain about selectivity estimation. Section 3.1 has preliminaries which contain related basic techniques.section3.2 contains proposed method.Section4 is explains about experimental evaluation and section5 is about conclusion of this work.

## 2. RELATED WORK

### 2.1. Selectivity estimation in spatial databases:

Acharya et al [2]. Several grouping techniques are proposed, which are uniformity assumption for spatial data, approximating spatial data, equi-partitioning of spatial data, R-tree indexing. Also proposed a new advanced technique called Min-skew partitioning. Min-skew is constructed based on BSP (Binary Space Partitioning).BSP based method called Min-skew partitioning is outperforms comparing to other techniques in large range of query workload and dataset. Main key point is low memory consumption during construction. These techniques are proposed basically using spatial indices, histograms, binary space partitioning, spatial skew.

### 2.2. A Primitive operator for similarity joins in data cleansing:

This paper [3] introduced new primitive operator called SSJoin operator, which is used to implement similarity joins for find string similarity functions between strings. It supports various similarity functions, which are edit similarity, generalized edit similarity, jaccard similarity, hamming distance. SSJoin operator is popular for set similarity join, which is its works accurately at large datasets.

### 2.3. Estimating the selectivity of approximate string queries:

This paper [4] proposed VSol (vertical solution) estimator for estimating selectivity of approximate short string queries. This technique is based on inverse strings, which is used to time taken to estimate the selectivity of query string is not depend on database string. Next computing substrings length of q, which is called as q-grams[1],[8],[9]. Based on q-grams min-wise signature is computed, again signature is clustered using k-means clustering algorithm. Sum of min-wise signature and their q-gram is dataset for VSol. This estimator used L-M similarity method to estimate the string.

L is a number of matching q-grams between two strings and their edit distance is smaller than threshold,
L=|s|-1-(T-1)*q. M is number of q-gram selected by VSol, which is match with query string. The VSol solution is intersection of L and M similarity for P data set. Set of string ids that appear in all L lists in the ith combination with

$$L_i, 1 \le i \le \binom{L}{M} \quad (1)$$

The L-M similarity is defined as:

$$\rho LM = |\cup L_i| \quad (2)$$

Estimate the selectivity as $\frac{\rho LM}{|P|}$

Examining ρLM is costly operation because computing sorted inverted list for all q-grams in database and specifying all M choose L combination of lists. So the straight forward solution is based on min-wise signature computing ρLM.

**2.4. Node and edge selectivity estimation for range queries in spatial networks:**

This paper [5] proposed four methods to estimate number of nodes and edges intersect with the range. First one is estimation based on MDS (Multi-Dimensional Scaling) for comparison purpose, which only gives efficient estimation for small graphs with large time and space. Second method is global parameter estimation method, which is based on two global parameters, average edge width(w) and average node degree(deg).It performs efficient estimation regular and almost regular uniform graphs only. Third estimation method called local density estimation method which has two methods is used to examine density, which are local counting density estimator and kernel density estimator. For local counting density method need to compute shortest path of full region, in case of kernel density method computing shortest path between nodes. This method gives efficiency in both uniform and non-uniform graphs also. Fourth method, Binary encoding method assigns label to each node based on label distance is calculated directly by using labels. Network distance between two nodes is approximated by using hamming distance of binary codes. Binary codes are encoded form of labels. This method gives most accurate estimation in all graphs.

**2.5. CCAM: A Connectivity-Clustered Access Method for Networks and Network computations:**

CCAM [6] is an access method for networks, which uses cluster connectivity. It supports insert, delete, find, create, get-A-successor, get successor operations. This is used to find connectivity among clustered nodes. It's also explains about connectivity among cluster.

## 3. SELECTIVITY ESTIMATION

### 3.1. Preliminaries:

#### 3.1.1. SAS query

Spatial Approximate String query q is combination of spatial predicate $q_r$and string predicate $q_s$ .Range query r is used as a spatial predicate and which is defined as query point q and radius. String predicate $q_s$defined as string s and an edit threshold T. Point in road network d (q,p) is a distance between two points q and p.

$A_r=\{p_x \in p \wedge d(q, p)\leq r\}$ (3)

$A_s=\{p_x \in p \wedge \acute{\epsilon}(s,s_x) \leq r\}$ (4)

(3),(4)used in[1],[10].SAS query q= ($q_r$,$q_s$) retrieves set of points A= $A_r \cap A_s$.

#### 3.1.2. Edit Distance Pruning

The edit distance between two strings s1 and s2 is the minimum number of edit operations of one letter that are have to transform s1 into s2.The edit operations are may be insert, delete, substitution of single character, swapping. Edit distance between two characters denoted as ɛ́(s1,s2).At the beginning and end of string s, that is fewer than q or s, Special characters # and $ are used generally. For example,(hospital, hospitle)=2,that is q-gram of length of string hospital(s1) is 2 {#h, ho, os, sp, pi, it, ta, al, l$}.The q-gram of length 2 for the string hospitle are{#h, ho, os, sp, pi, it, tl, le, e$}.When using this function, our problem becomes finding all $s1,s2 \in S$ that is $\acute{\epsilon}(s1,s2) \leq 2(T)$.This metric is used in many papers[7], [8].

#### 3.1.3. The min-wise signature

The min-wise independent permutation is first introduced at [13],[14].Min-wise independent permutations *f* must satisfy(1).The large of elements U̇ for any set X that is $X \subseteq U̇$, For any $x \in X$. π is chosen at random in *f* .π(X) is produces permutations of X.

Pr (min {π(X)} = π(x) = 1/|X|. (5)

π(x) is the location value of x in the resulted permutations, and min {π(X)} =min {π(x) |$x \in X$}.Its useful for estimating set closeness.

The set closeness of two sets A and B is

$$\rho(A, B) = \frac{|A \cap B|}{|A \cup B|}. \quad (6)$$

### 3.2. Proposed mechanism

Several selectivity estimation methods have been proposed in [2], [5] but no one is combined with string and spatial data. General method to achieve selectivity estimation for both string and spatial points is partition the network as many buckets B. Buckets may be independent but within bucket points are must be related or uniform. Each bucket is intersect with given query range r. Uniformity of points is assumed based on [1], [11]. Each bucket is defined by using MBR(Minimum Bounding Rectangle). That is Estimated number of points from B that also in r it must be directly proportional to total number of points P in bucket B and total area of B intersected with r(range query),which is successfully applied in[2],[11],[12].

Basic idea of selectivity estimator is to build buckets $B_1,B_2,...,B_n$ for n buckets. Number of points in *i*th bucket be $n_i$ , and its area is Θ($B_i$).For each bucket $B_i$,we have to build Vsol estimator $Vsol_i$ based on min-wise signature of q-gram inverted list of string in bucket. Selectivity estimation of query Q={r,(s,T)},Which is achieved for each bucket $B_i$ intersects with range r and intersection area is Θ($B_i$, *r*),then L-M similarity is $\rho^i$LM for strings s in bucket $B_i$.Then based on [1] $A_{Bi}$ is

$$|A_{Bi}| = n_i \frac{\Theta(B_i r)}{\Theta(B_i)} \frac{\rho^i LM}{n_i} = \frac{\Theta(B_i r)}{\Theta(B_i)} \rho^i LM \quad (7)$$

### 3.2.1. Adaptive selection algorithm for RSAS query

First our idea simply applied using greedy algorithm, which proceeds in n number of iterations.For each iteration one bucket is produced.At ith iteration, randomly selecting unallocated points and keep adding to bucket $B_i$.Until no depletion to overall uncertainty of current configuration. Directly applying greedy algorithm to large spatial database is expensive and time consuming.

So apply that idea in B+- tree index such as adjacency list file and points file. It will add metrics to this work. First apply divide and conquer algorithm to an adjacency list file. Divide tree index as two partitions, that is left most tree and right most tree separately. Solve the dividends using greedy algorithm. That is for each iteration one bucket is generated based on geometric related points. Generated bucket contains geometrically grouped points, already neibourhood points are stored in adjacency list.

Adjacency list contents are node id, corresponding reference node distance, neibourhood node offset distance and points name both are stored in separate points file. Assume root and leftmost tree is LT and rightmost tree is RT. Apply greedy algorithm in both LT and RT parallel. Generate n buckets based on their geometrical distance, which is group the points based on geometrically closest distance. Both dividends solutions (buckets) are independent. For adding points to bucket, start at root node then leftmost tree keep adding closest points in bucket, for each iteration one node is added in bucket. After ith iteration remaining unallocated points are locally grouped n-1th bucket that points are uniform and geometrically independent. When n-1 buckets are constructed end the iteration, group all remaining nodes in same bucket. Then merge both dividends to get global solution. Finally after construction of n-1 buckets have to build Vsol estimator for each bucket.

For given RSAS query, we have to find range r based on range query. Then select the buckets which are intersect within range r. Apply Vsol estimator to each bucket based on [4] to estimate the selectivity. Basically Vsol estimator is based on L-M similarity.

**Algorithm:**

1./*Select the corresponding B+-tree adjacency list index*/
2./*Using divide and conquer algorithm compute LT and RT*/
3.if node=root& leftmost tree then
4. add to LT
5.else
6. add to RT
7./*Apply greedy algorithm for both LT and RT separately and generate the buckets*/
8. for(i=0;i<n-1;i++)
9. if $P_i$ =related then
10. add to corresponding $B_i$
11. end if
12. else
13. ignore
14. end for
15./* Make the remaining nodes as bucket and which are independent and uniform*/
16. if i=n-1
17. group unallocated points to $B_i$
18. end for
19. /* Conquer the buckets for global solution. Then apply Vsol estimator for buckets for String selection*/

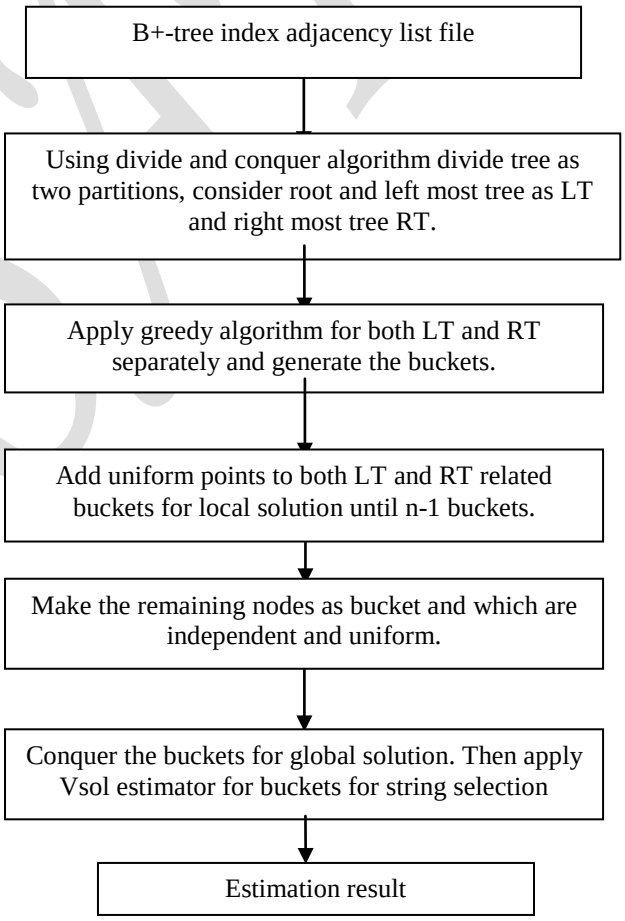


Fig.2.Selectivity estimation system architecture

Selectivity estimation algorithm is presented as a flowchart in figuare2.Line one is instruction about step one in flowchart, we have to select corresponding B+- tree adjacency list file. B+- tree is suitable disk based storage data structure for

indexing in ordered way. Key advantage in this tree is all the information's are stored in leaves. Second line is about using divide and conquer algorithm divided as LT and RT. Third line is if condition, condition is true means it execute line4,if it false executed line6. Next line is applying corresponding greedy algorithm to LT and RT. Greedy algorithm is used for to get optimal solution at each stage. Its providing also best optimal local solution and also hoping global solution. Line 8 is having For loop, used for execute repeatedly until condition specified in loop. Next line is if condition used for if points in list file are related means go to next line otherwise ignores the point.

Line 15 is after completion of grouping adds remaining points to one bucket called n bucket. Then conquer the buckets and apply Vsol estimator for all buckets to estimate the string selectivity.

## 4. EXPERIMENTAL EVALUATION

We use one synthetic road network dataset PN obtained from open street network ,it contain 7513 nodes and 7879 edges, strings are assigned based on coordinates and we randomly selected 10000 points. The default min wise signature length is 50 and hash values is 200bytes.Default range setup is r=500 and threshold value is 2.In selectivity estimation of RSAS query measured when to calculate accuracy of estimation and its relative error μ,its measured by using estimated result and corresponding answer. Selectivity estimator denoted as f. $\mu = \frac{|\tilde{f} - |A||}{|A|}$, the accuracy of estimator is high when size of buckets is large. Selectivity estimation relative error is nearly 0.2 to 0.5, when applied in large dataset. Its experimentally explained in figure 3, shows the result of μ.

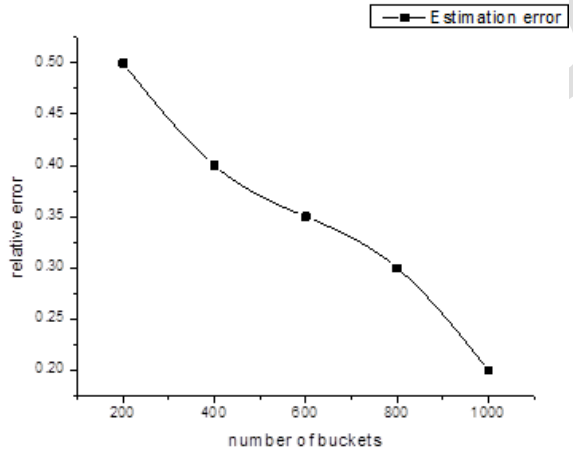


Fig.3.selectivity estimator error for RSAS query

## 5. CONCLUSION

This paper addresses solution to the selectivity estimation for RSAS query. Selectivity estimation is based on grouping technique called greedy algorithm. This technique is very effective when applied in large dataset and also bucket size. Future work includes designing method which is more update friendly.

## REFERENCES

[1].Feife iLi,Bin Yao, Mingwang Tang, and Marios Hadjieleftheriou"Spatial approximate string search" *IEEE transaction on knowledge and data engineering*,vol.25,no 6,pp.1394-1409,JUNE 2013.

[2].Swarup Acharya, Viswanath Poosala, and Sridhar Ramaswamy, "Selectivity Estimation in Spatial Databases," *Proc. ACM SIGMOD'99 International Conference on Management of Data*, pp. 13-24, Volume 28 Issue 2, June 1999.

[3].Surajit Chaudhuri, Venkateesh Ganti, and Raghav Kaushik, "A Primitive Operator for Similarity Joins in Data Cleaning," *Proc. 22 International Conference on Data Eng.(ICDE)*, pp. 5-16, 2006,April 2006.

[4].Arturas Mazeika, Mickel.H. Bo¨hlen, Nick Koudas, and Divesh Srivastava,"Estimating the Selectivity of Approximate String Queries,"*ACM Trans. Database Systems*, vol. 32, issue. 2, pp. 12-52, June 2007.

[5]. E. Tiakas, A.N. Papadopoulos, A. Nanopoulos, and Y. Manolopoulos,"Node and Edge Selectivity Estimation for Range Queries in Spatial Networks," *Information Systems*, vol. 34, pp. 328-352, May2009.

[6].Shashi Shekhar and Duen Ren Liu, "CCAM: A Connectivity-Clustered Access Method for Networks and Network Computations," *IEEE Trans. Knowledge and Data Engineering*, vol. 9, Issue 1, pp. 102-119, January 1997.

[7].Arvind Arasu, Surajit Chaudhuri, Kris Ganjam, and Raghav Kaushik, "Incorporating String Transformations in Record Matching," Proc. *ACM SIGMOD'08 International Conf. Management of Data*, pp. 1231-1234, June2008.

[8].Surajit Chaudhuri, Kris Ganjam, Venkatesh Ganti, and Rajeev Motwani, "Robust and Efficient Fuzzy Match for Online Data Cleaning," *Proc. ACM SIGMOD03 Int'l Conf. Management of Data*, pp. 313-324, June 2003

[9].Xiaochun Yang, Bin Wang, and Chen Li, "Cost-Based Variable-Length-Gram Selection for String Collections to Support Approximate Queries Efficiently," *Proc. ACM SIGMOD'08 International Conf. Management of Data*,pp. 353-364, June 2008.

[10].Dimitris Papadias, Jun Zhang, Nikos Mamoulis, and Yufei Tao, "Query Processing in Spatial Network Databases," *Proc. 29Int'l Conf. Very Large Data Bases VLDB'03*, pp. 802-813, September 2003.

[11].Dimitrios Gunopulos, George Kollios, Vassilis J. Tsotras, and Carlotta Domeniconi," Selectivity Estimators for Multidimensional Range Queries Over Real Attributes," *Springer,The VLDB J.*, vol. 14, no. 2, pp 137-154, April 2005.

[12].Liang Jin and Chen Li, "Selectivity Estimation for Fuzzy String Predicates in Large Data Sets," *Proc. 31Int'l Conf. Very Large Data Bases (VLDB03)*,pp. 397-408,August 2005.

[13].Andrei Z. Broder, Moses Charikar, Alan M. Frieze, and Micheal Mitzenmacher, "Min-Wise Independent Permutations (Extended Abstract)," Proc.*ACM 30th Symp. Theory of Computing*,STOC'98, pp. 327-336, May1998.

[14].Edith Cohen, "Size-Estimation Framework with Applications to Transitive Closure and Reachability," J. *Computer and System Sciences*, vol. 55, no. 3, pp. 441-453, December 1997.