IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

Survey of Efficient Filtering Algorithms for Location-Aware Subscribe with R-Tree Model

Ms. K. Siya, MCA.[1], Mrs. P. Selvi M.Sc, M.Phil[2]

M. Phil Full Time Scholar, Vivekanandha College for Women, Tiruchengode[1]

Assistant Professor, Vivekanandha College of Arts and Science for Women (Autonomous), Tiruchengode[2]

Abstract: In this paper survey LBS systems employ a pull model or user-initiated model, where a user issues a query to a server which responds with location aware answers. To provide users with instant replies, a push model or server-initiated model is becoming an inevitable computing model in next-generation location-based services. In the push model, subscribers register spatio-textual subscription to capture their interests, and publishers post spatio textual messages. These calls for a high-performance location-aware publish/subscribe system to deliver messages to relevant subscribers. This computing model bring new user experiences to mobile users, and can help users recover information without explicitly issue a query. The publish/subscribe system should support tens of millions of subscribers and deliver messages to relevant subscribers in milliseconds. while messages and subscriptions contain both location information and textual description, it is rather costly to deliver messages to relevant subscribers. These calls for an efficient filtering technique to support location-aware publish/subscribe services. Moreover, a prediction strategy is proposed to predict the subsequent mobile behaviors. Mean while, a time segmentation approach is presented to find segmenting time intervals where similar mobile characteristics exist. To the best knowledge, this is the first work on mining and prediction of mobile behaviors with considerations of user relations and temporal property simultaneously. Through survey under various the proposed methods are shown to deliver excellent performance.

Keywords: LBS, MBR Filter, Pull Model, Push Model.

I. INTRODUCTION

Data mining is a widely used technique for discover valuable information in a complex data set and a number of studies have discussed the issue of mobile behavior mining. The main difference between these literatures is the involved information of proposed patterns. Previous studied by Tseng and Tsui addressed the problem of mining associated service patterns in mobile web networks. Tseng and Lin also proposed SMAP-Mine to efficiently mine users' sequential mobile access patterns, based on the FP-Tree.

Yun and Chen proposed a novel method of mining mobile sequential patterns. To increase the accuracy of predictions, the moving path was taken into consideration in the previous studies. However, mobile behaviors vary among different user clusters or at various time intervals. The prediction of mobile behavior will be more precise if we can discover the corresponding mobile patterns in each user cluster and time interval. To provide accurate location-based services for users, effective mobile behavior mining systems are required pressingly.

Clustering mobile transaction data help in the discovery of social groups, which are used in applications such as targeted advertising, shared data allocation, and personalization of content services. In previous studies, users are typically cluster according to their personal profiles (e.g., age, sex, and occupation). In the real applications of mobile environments, it is often difficult to obtain users' profiles. That is, we may only have access to users' mobile transaction data.

To achieve the goal of user clustering without user profiles, the evaluation of the similarities of mobile transaction sequences (MTSs) is needed. while a number of clustering algorithms have been studied in the rich literature, they are not appropriate in the LBS scenario in consideration of the following issues: 1) Most clustering can only process data with spatial similarity measures, while clustering methods with non-spatial similarity measures are required for LBS environments. 2) Most clustering methods request the users to set up some parameters.

However, in real application, it is difficult to find out the right parameters manually for the clustering tasks. Hence, an automatic clustering method is required. Although there exist many non-spatial similarity measures, most of them are used to measure the string similarity. However, the mobile transaction sequence discussed in this paper include multiple and heterogeneous information such as time, location, and services. Therefore, the existing measures are not appropriate directly for measuring the similarity of mobile transaction sequences.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

II. LITRATURE SURVEY

Xin Caoy [1] describe the location-aware keyword query returns ranked objects that are near a query location and that have textual descriptions that match query keywords. The paper proposes the concept of prestige-based relevance to capture both the textual relevance of an object to a query and the effects of nearby objects. Based on this, a new type of query, the Location-aware top-k Prestige-based Text retrieval (LkPT) query, is proposed that retrieves the top-k spatial web objects ranked according to both prestige-based relevance and location proximity. In this paper proposed two algorithms that compute LkPT queries. Empirical studies with real-world spatial data demonstrate that LkPT queries are more effective in retrieving web objects than a previous approach that does not consider the effects of nearby objects; and they show that the proposed algorithms are scalable and outperform a baseline approach significantly.

Xin Cao, Christian S. Jensen and Beng Chin Ooi [2] define the problem of retrieving a group of spatial web objects such that the group's keywords cover the query's keywords and such that objects are nearest to the query location and have the lowest inter-object distances. Specifically, we study two variants of this problem, both of which are NP-complete. We devise exact solutions as well as approximate solutions with provable approximation bounds to the problems. User needs may exist that are not easily satisfied by a single object, but where groups of objects may combine to meet the user needs. Put differently, the objects in a group collectively meet the user needs.

Ju Fan [3] describe a location-based services (LBS) have become more and more ubiquitous recently. Existing methods focus on finding relevant points-of-interest (POIs) based on users' locations and query keywords. Nowadays, modern LBS applications generate a new kind of spatio-textual data, regions-of-interest (ROIs), containing region-based spatial information and textual description, e.g., mobile user profiles with active regions and interest tags. To satisfy search requirements on ROIs, we study a new research problem, called spatio-textual similarity search: Given a set of ROIs and a query ROI, to find the similar ROIs by considering spatial overlap and textual similarity. Spatio-textual similarity search has many important applications, e.g., social marketing in location-aware social networks. It calls for an efficient search method to support large scales of spatio-textual data in LBS systems. In this paper, we formalize the problem of spatio-textual similarity search, and study the research challenges that naturally arise in this problem. A challenge is how to evaluate the similarity between two ROIs. Another challenge is how to achieve high search efficiency as LBS systems are required to support millions of users and respond to queries in milliseconds. Given a query ROI, there may be a huge amount of ROIs having significant overlaps with the query, thus it is rather expensive to find similar answers.

Jiaheng Lu et al [4]descrbe a geographic objects associated with descriptive texts are becoming prevalent. This gives prominence to spatial keyword queries that take into account both the locations and textual descriptions of content. Specifically, the relevance of an object to a query is measured by spatial-textual similarity that is based on both spatial proximity and textual similarity. In this paper, we define Reverse Spatial Textual k Nearest Neighbor (RSTkNN) query, i.e., finding objects that take the query object as one of their k most spatial-textual similar objects. Existing works on reverse kNN queries focus solely on spatial locations but ignore text relevance. To answer RSTkNN queries efficiently, we propose a hybrid index tree called IUR-tree (Intersection-Union R-Tree) that effectively combines location proximity with textual similarity. Based on the IUR-tree, we design a branch-and-bound search algorithm. To further accelerate the query processing, we propose an enhanced variant of the IUR-tree called clustered IUR-tree and two corresponding optimization algorithms. Empirical studies show that the proposed algorithms offer scalability and are capable of excellent performance.

Dingming Wu et al [5] describe a moving top-k spatial keyword (MkSK) query, which takes into account a continuously moving query location, enables a mobile client to be continuously aware of the top-k spatial web objects that best match a query with respect to location and text relevance. The increasing mobile use of the web and the proliferation of geo-positioning render it of interest to consider a scenario where spatial keyword search is outsourced to a separate service provider capable at handling the voluminous spatial web objects available from various sources. A key challenge is that the service provider may return inaccurate or incorrect query results (intentionally or not), e.g., due to cost considerations or invasion of hackers. Therefore, it is attractive to be able to authenticate the query results at the client side. Existing authentication techniques are either inefficient or inapplicable for the kind of query we consider. We propose new authentication data structures, the MIR-tree and MIR-tree, that enable the authentication of MkSK queries at low computation and communication costs. We design a verification object for authenticating MkSK queries, and we provide algorithms for constructing verification objects and using these for verifying query results.

III. R-TREE STRUCTURE

R-trees are hierarchical data structures based on B+ trees. They are used for the dynamic organization of a set of d-dimensional geometric objects representing them by the minimum bounding d-dimensional rectangles (MBR). Each

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

node of the R-tree corresponds to the MBR that bounds its children. The leaves of the tree contain pointers to the database objects instead of pointers to children nodes. The nodes are implemented as disk pages.

An MBR can be included (in the geometrical sense) in many nodes, but it can be associated to only one of them. This means that a spatial search may visit many nodes before confirming the existence of a given MBR. Also, it is easy to see that the representation of geometric objects through their MBRs may result in false alarms. To resolve false alarms, the candidate objects must be examined. For instance, Figure 4.1 and 4.2 illustrates the case where two polygons do not intersect each other, but their MBRs do. Therefore, the R-tree plays the role of a filtering mechanism to reduce the costly direct examination of geometric objects.

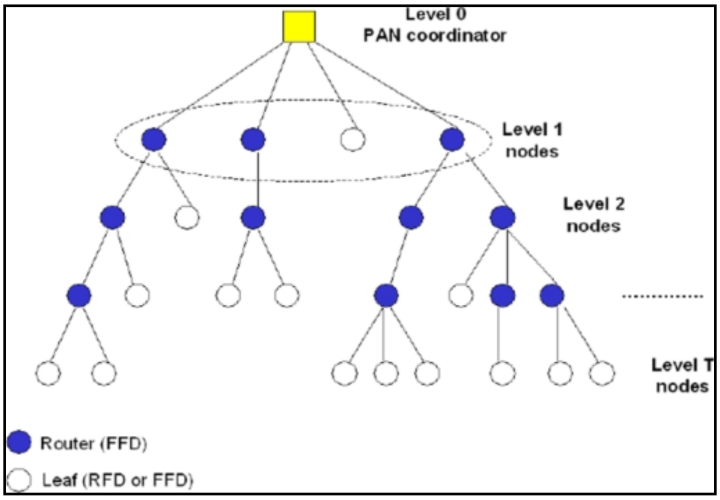

FIG 3.1 R-Trees Topology

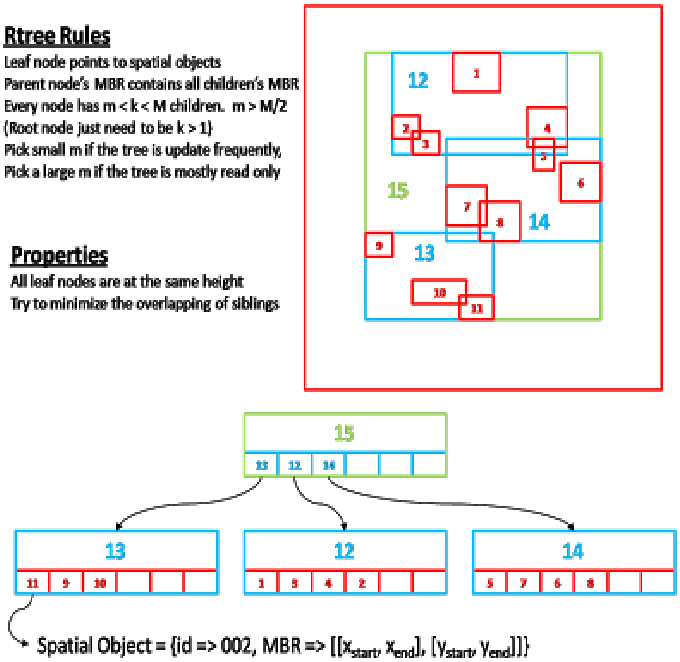

Fig 3.2 MBR TREE

An R-tree of order (m,M) has the following characteristics:

- Each leaf node (unless it is the root) can host up to M entries, whereas the minimum allowed number of entries is m ! M/2. Each entry is of the form (mbr, oid), such that mbr is the MBR that spatially contains the object and oid is the object's identifier.
- The number of entries that each internal node can store is again between m ! M/2 and M. Each entry is of the form (mbr, p), where p is a pointer to a child of the node and mbr is the MBR that spatially contains the MBRs contained in this child.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

- The minimum allowed number of entries in the root node is 2, unless it is a leaf (in this case, it may contain zero or a single entry).
- All leaves of the R-tree are at the same level.

From the definition of the R-tree, it follows that it is a height-balanced tree. As mentioned, it comprises a generalization of the B+-tree structure for many dimensions. R-trees are dynamic data structures, i.e., global reorganization is not required to handle insertions or deletions. Figure 4.3 shows a set of the MBRs of some data geometric objects (not shown). These MBRs are D,E, F,G,H, I, J,K,L,M, and N, which will be stored at the leaf level of the R-tree. The same figure demonstrates the three MBRs (A,B, and C) that organize the aforementioned rectangles into an internal node of the R-tree. Assuming that M = 4 and m = 2, Figure 4.4 depicts the corresponding MBR. It is evident that several R-trees can represent the same set of data rectangles. Each time, the resulting R-tree is determined by the insertion (and/or deletion) order of its entries Fig 4.5.

Algorithm RangeSearch (TypeNode RN, TypeRegion Q)
/* Finds all rectangles that are stored in an R-tree with root node RN, which are intersected by a query rectangle Q. Answers are stored in the set A */
if RN is not a leaf node
examine each entry e of RN to find those e.mbr that intersect Q
foreach such entry e call RangeSearch(e.ptr,Q)
else // RN is a leaf node
examine all entries e and find those for which e.mbr intersects Q
add these entries to the answer set A endif

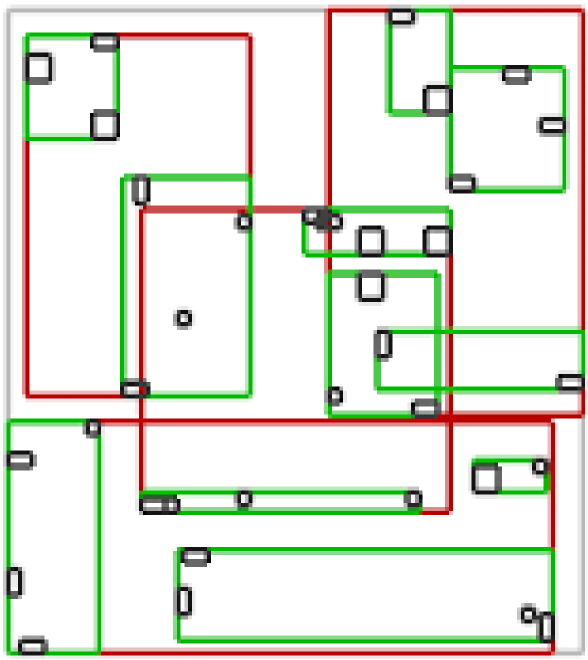

Fig. 3.3 Data MBRs and their MBRs

R-TREE ALGORITHM:
Insert (TypeEntry E, TypeNode RN)
/* Inserts a new entry E in an R-tree with root node RN */
Step 1:
Traverse the tree from root RN to the appropriate leaf. At each level, select the node, L, whose MBR will require the minimum area enlargement to cover E.mbr
Step 2:
In case of ties, select the node whose MBR has the minimum area
if the selected leaf L can accommodate E
Insert E into L
Update all MBRs in the path from the root to L,so that all of them cover E.mbr
Else // L is already full

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

Step 3:

Let E be the set consisting of all L's entries and the new entry E Select as seeds two entries e1, e2 ! E, where the distance between e1 and e2 is the maximum among all other pairs of entries from E Form two nodes, L1 and L2, where the first contains e1 and the second e2

Step 4:

Examine the remaining members of E one by one and assign them to L1 or L2, depending on which of the MBRs of these nodes will require the minimum area enlargement so as to cover this entry

if a tie occurs
Assign the entry to the node whose MBR has the smaller area
endif
if a tie occurs again
Assign the entry to the node that contains the smaller number of entries
endif

Step 5:

if during the assignment of entries, there remain " entries to be assigned and the one node contains m " " entries

Step 6: Assign all the remaining entries to this node without considering the aforementioned criteria /* so that the node will contain at least m entries */ endif

Step 7:

Update the MBRs of nodes that are in the path from root to L, so as to cover L1 and accommodate L2

Step 8: Perform splits at the upper levels if necessary

In case the root has to be split, create a new root
Increase the height of the tree by one
endif

V. CONCLUSION AND FUTURE ENHANCEMENT

CONCLUSION

The R^t-tree only selects a single representative token, it does not take full advantage of the textual information to do textual pruning. To address this issue, propose selecting multiple representative tokens and devise an efficient filtering algorithm to directly find answers without the verification step. The new system develop into useful if the below enhancements are made in future.

- In future work, the method can be applied to real data sets. In addition, the CTMSP-Mine can be applied to other applications, such as GPS navigations, with the aim to enhance precision for predicting user behaviors.
- The application if industrial as web site, can be use as of anywhere.
- The new system is designed such that those enhancements can be integrated with current modules easily with less integration work.

ACKNOWLEDGMENT

My heartfelt gratitude goes to my beloved guide **Mrs. Selvi,** Assistant Professor, Department of Computer Science, Vivekanandha College of Arts and Sciences for Women (Autonomous), Tiruchengode, India for dedication and patience in assigning me her valuable advice and efforts during the course of my studies.

REFERENCES

[1] X. Cao, G. Cong, and C. S. Jensen, "Retrieving top-k prestige-based relevant spatial web objects," Proc. VLDB Endowment, vol. 3, no. 1, pp. 373–384, 2010.
[2] X. Cao, G. Cong, C. S. Jensen, and B. C. Ooi, "Collective spatial keyword querying", in Proc. ACM SIGMOD Int. Conf. Manage.Data, 2011, pp. 373–384.
[3] J. Fan, G. Li, L. Zhou, S. Chen, and J. Hu, "Seal: Spatio-textual similarity search," Proc. VLDB Endowment, vol. 5, no. 9, pp. 824–835, 2012.
[4] J. Lu, Y. Lu, and G. Cong, "Reverse spatial and textual k nearest neighbor search," in Proc. ACM SIGMOD Int. Conf. Manage. Data, 2011, pp. 349–360
[5] D. Wu, M. L. Yiu, C. S. Jensen, and G. Cong, "Efficient continuously moving top-k spatial keyword query processing," in Proc. Int. Conf. Data Eng., 2011, pp. 541–552.
[6] B. Yao, F. Li, M. Hadjieleftheriou, and K. Hou, "Approximate string search in spatial databases," in Proc. Int. Conf. Data Eng., 2010, pp. 545–556.
[7] D. Zhang, B. C. Ooi, and A. K. H. Tung, "Locating mapped resources in web 2.0," in Proc. Int. Conf. Data Eng., 2010, pp. 521–532.
[8] R. Zhong, J. Fan, G. Li, K.-L. Tan, and L. Zhou, "Location-aware instant search," in Proc. 21st ACM Int. Conf. Inf. knowl. Manage., 2012, pp. 385–394.