IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 17-22
www.iosrjournals.org

Efficient Filtering Algorithms for Location-Aware Publish/Subscribe With R-Tree Model

Ms.K.Siya[1], MCA., Mrs.P.Selvi[2], M.Sc., M.Phil.,
Research Scholar, Computer Science, Vivekanandha college for women, India[1].
Assistant Professor, Computer Science, Vivekanandha college of Arts and Sciences for women, India[2].

Abstract: *In this paper LBS systems employ 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 addition, thesis on Location-Based Service (LBS) have been rising in recent years due to a wide range of potential applications. One of the active topics is the mining and prediction of mobile movements and associated transactions. Most of existing studies focus on discovering mobile patterns from the whole logs. However, this kind of patterns may not be precise enough for predictions since the differentiated mobile behaviour among users and temporal periods are not considered. The paper proposes a novel algorithm, namely, Cluster-based Temporal Mode Sequential Pattern Mine (CTMSP-Mine), to discover the Cluster-based Temporal Mode Sequential Patterns (CTMSPs). Moreover, a prediction strategy is proposed to predict the subsequent mobile behaviours. In ECTMSP-Mine, user clusters are constructed by a novel algorithm named Enhanced Cluster Affinity Search Technique (ECAST) and similarities between users are evaluated by the proposed measure, Alternative Location-Based Service Alignment (ALBS-Alignment). 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 behaviours with considerations of user relations and temporal property simultaneously. Through experimental evaluation under various simulated conditions, the proposed methods are shown to deliver excellent performance.*

Keyword: *Web Dataset, Filtering, Prediction Strategy, R^- Tree, ALBS.*

I. Introduction

Data mining is the process of extract patterns from data. Data mining is seen as an increasingly important tool by modern business to transform data into an informational advantage. It is currently used in a wide range of profiling practices, such as marketing, surveillance, fraud detection, and scientific discovery.

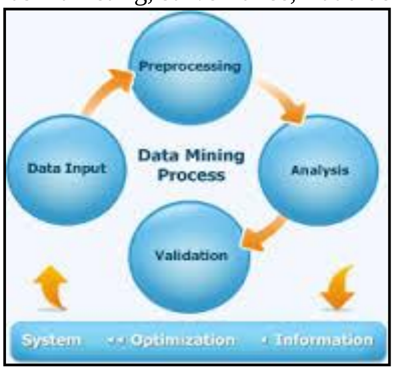

Fig 1.1 Data Mining

Data mining commonly involves four classes of tasks:

- Clustering - is the task of discovering groups and structures in the data that are in some way or another "similar", without using known structures in the data.
- Classification - is the task of generalizing known structure to apply to new data. For example, an email program might attempt to classify an email as legitimate or spam. Frequent algorithms include decision tree learning, nearest neighbour, naive Bayesian classification, neural networks and support vector machines.
- Regression - Attempts to find a function which models the data with the least error.

Association rule learning - Searches for relationships between variables. For example supermarket might gather data on customer purchasing habits. Using association rule learning, the supermarket can determine which products are frequently bought together and use this information for marketing purposes. This is sometimes referred to as market basket analysis. An essential task in our proposed framework is to determine the frequent item set with the problem may be solved by using store and item category ontology. However, the store or item ontology may not match with the mobile transaction dataset. The finding problem is to automatically compute the store and item similarities from the mobile transaction dataset, which captures mobile users' moving and transactional behaviours (in terms of movement among stores and purchased items). From the database are following information available:

1) for a given store, know which items are available for sale;
2) for a given item, know which stores sell this item.

The information can help us to infer which stores or items are similar. As observe that people usually purchase similar items in certain stores, these stores may be considered as similar. The propose system is a parameter-less data mining model, named Similarity Inference Model, to tackle this task of computing store and item similarities.In proposed system plan to discover more efficient mobile commerce pattern mining algorithm, design more efficient similarity inference models, and develop profound prediction strategies to further enhance the MCE framework. In addition, we plan to apply the MCE framework to other application, such as object tracking sensor networks and location-based services, aiming to achieve high precision in predicting object behaviours. Therefore, proposed system develops a Enhanced Cluster-based Temporal Mobile Sequential Pattern Mine (ECTMSP-Mine) algorithm.

II. Literature Survey

Xin Caoy et al [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. This query occurs inherently in many types of mobile and traditional web services and applications, e.g., Yellow Pages and Maps services. Previous work considers the potential results of such a query as being independent when ranking them. However, a relevant result object with nearby objects that are also relevant to the query is likely to be preferable over a relevant object without relevant nearby objects. 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.

Xin Cao et al [2] describe a the proliferation of geo-positioning and geo-tagging, spatial web objects that possess both a geographical location and a textual description are gaining in prevalence, and spatial keyword queries that exploit both location and textual description are gaining in prominence. However, the queries studied so far generally focus on finding individual objects that each satisfy a query rather than finding groups of objects where the objects in a group collectively satisfy a query. 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.

Ju Fan et al [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, we find the similar ROIs by considering spatial overlap and textual similarity.

III. R-Tree Methodology

R-trees are tree data structures used for spatial access methods, i.e., for indexing multi-dimensional information such as geographical coordinates, rectangles or polygons. The R-tree was proposed has found significant use in both theoretical and applied contexts. A common real-world usage for an R-tree might be to store spatial objects such as restaurant locations or the polygons that typical maps are made of: streets, buildings, outlines of lakes, coastlines, etc. and then find answers quickly to queries such as "Find all museums within 2 km of my current location", "retrieve all road segments within 2 km of my location" (to display them in a navigation system) or "find the nearest gas station" (although not taking roads into account). The R-tree can also accelerate nearest neighbour search for various distance metrics, including great-circle distance.

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 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.

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 Range Search (TypeNode RN, Type Region 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 Range Search(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
Insert (TypeEntry E, TypeNode RN)
/* Inserts a new entry E in an R-tree with root node RN */
```

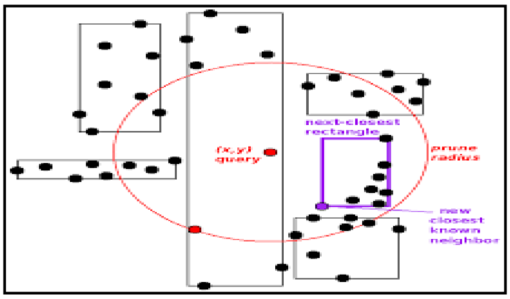

Fig. 3.1 Data with Corresponding R-tree

IV. Enhanced Clustering Of Mobile Transaction Sequences

The PMCP-Mine algorithm is performed in a bottom-up manner. We first discover frequent transaction behaviors in a single store, eventually, the complete mobile commerce patterns can be obtained by the PMCP-Mine algorithm. The PMCP-Mine algorithm is divided into three main phases: In order to mine the cluster-based temporal mobile sequential patterns efficiently, we proposed a novel method named CTMSP-Mine to achieve this mining procedure. In ECTMSP-Mine, both factors of user cluster and time interval are taken into account such that the complete mobile sequential patterns can be discovered.

A mobile transaction sequence can be viewed as a sequence string, where each element in the string indicates a mobile transaction. The major challenge to tackle is to measure the content similarity between mobile transactions. The ALBS-Alignment algorithm is proposed, which can obtain the similarity. ALBS-Alignment is based on the consideration that two mobile transaction sequences are more similar, when the orders and timestamps of their mobile transactions are more similar. ECAST algorithm is used to cluster the users. In a mobile transaction database, users in the different user groups may have different mobile transaction behaviors. The first task to tackle is to cluster mobile transaction sequences. In this module, a parameter-less clustering algorithm called ECAST is proposed.

LBS-Alignment Algorithm

The LBS-Alignment algorithm is proposed, which can obtain the similarity. LBS-Alignment is based on the consideration that two mobile transaction sequences are more similar, when the orders and timestamps of their mobile transactions are more similar.

Based on this concept, it is designed specifically the time penalty (TP) and the service reward (SR) in the ALBS-Alignment. The base similarity score is set as 0.5. Two mobile transactions can be aligned if their locations are the same. Otherwise, a location penalty is generated to decrease their similarity score. The location penalty is defined as 0.5/ ($|s_1| + |s_2|$) where $|s_1|$ and $|s_2|$ are the lengths of sequences s_1 and s_2 , respectively. Notice that the maximal number of location penalties is $|s_1| + |s_2|$. When two sequences are totally different, their similarity score is 0.

/* ALSB Algorithm */

Input: Two mobile transaction sequences s and s'

Output: The similarity between s and s'

```
ALBS-Alignment (s,S')
p ← 0.5 / (s.length + s'.length)   /*p is the location penalty */
M0,0 ← 0.5
Mi,0 ← Mi-1,0 - p□ □ □i = {1,2,....,s.length)
M0,j ←M0,j-1 - p□ □ □j = {1,2,....,s'.length)
For i ←1 to s.length
For j ←1 to s'.length
For j ←1 to s'.length
If si.location = sj'.location
TP ←p * | si.time = sj'.time| / len        /* time penalty */
SR ←p * ( si.service ∩□ sj.'service/   si.service U□ sj.'service)
/* service reward */
Mi,j ←Max(Mi-1,j-1 – TP + SR, Mi+1,j – p, Mi,j-1 – p)
Else
Mi,j ←Max(Mi-1,j– p, Mi,j-1 – p)
End If
End For
End For
Return Ms.length,s'.length
```

V. Results And Discussion

The following **fig 5.1** describes experimental result for R-Tree and CTMSP Tree error rate analysis. The table contains number of transaction datasets with count and error rate analysis for R-Tree and CTMSP Tree rate details are shown.

$$\text{Error Rate} = ([R1-R2]/R1) * 100$$

The following Fig 5.1 describes experimental result for R-Tree and CTMSP Tree Transaction rate analysis. The figure contains number of transaction rate for R-Tree and CTMSP Tree details are shown. The following Fig 5.2 describes experimental result for R-Tree and CTMSP Tree error rate analysis. The figure contains AVG error rate for R-Tree and CTMSP Tree details are shown.

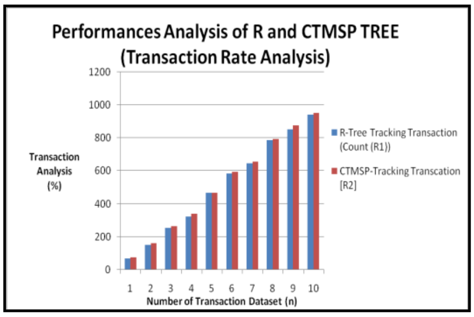

Fig 5.1 Performances Transaction Analysis of R-CTMSP Tree

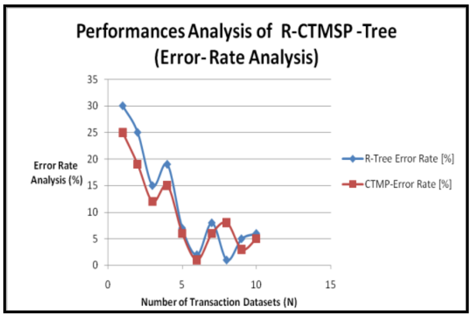

Fig 5.2 Performances Error Rate Analysis –R-Tree and CTMSP-Tree Topology

This paper is used to eliminate time complexity rate while finding high frequent item sets in a transaction database. In this proposed research, tree construction process using two strategies, namely E-MTS (Enhanced Mobile Transaction Set items) and LSB-CMSTP. It also used to reduce number scans to the database and time reduces.

VI. Conclusion

In this paper, a novel method named ECTMSP-Mine is proposed, for discovering CTMSPs in LBS environments. Furthermore, novel prediction strategies are proposed to predict the subsequent user mobile behaviours using the discovered ECTMSPs. In ECTMSP-Mine, first a transaction clustering algorithm is proposed named ECO-Smart-CAST to form user clusters based on the mobile transactions using the proposed ELBS-Alignment similarity measurement. Then, the time segmentation algorithm is utilized to generate the most suitable time intervals. To our best of mobile behaviours associated with user clusters and temporal relations.

A series of experiment were conducted for evaluating the performance of the proposed methods. The experimental results show that ECO-Smart-CAST method achieves high-quality clustering results and the proposed ECBSS strategy obtains highly precise results for user classification. Meanwhile, the algorithms obtain the most proper and correct time intervals. For behaviour prediction, ECTMSP is shown to outperform other prediction methods in terms of precision and F-measure. The experimental results demonstrate that the proposed methods are efficient and accurate under various conditions.The application works well for given tasks in windows environment. Any node with .Net framework installed can execute the application and identifies the best site. The underlying mechanism can be extended to any / all kind of web servers and even in multi-platform like Linux, Solaris and more. The system is planned to expand the services can be given as input to IBM architecture also. The system eliminates the difficulties in the existing system. It is developed in a user-friendly manner. The system is very fast in applying algorithm. This software is very particular in predict the subsequent mobile behaviours.

- In future work, the method can be apply 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 applications if developed as web site can be used from anywhere.
- The new system is designed such that those enhancements can be integrated with current modules easily with less integration work.

Acknowledgements

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.