Volume 4 Issue IX, September 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Detection of Spatio-Textual Top-K Queries in Location-Based Services Using Reverse Keyword Search

U. L. Sindhu[1], F. Femila[2], J. Santhiya[3]
Department of Computer Science & Engineering, Sri Krishna College of Technology.

***Abstract—* A spatio-textual query retrieves the best objects with respect to a given location and a keyword set. A common solution is to employ a hybrid index which records the digest of the spatial and textual information. To efficiently process the new query, a novel hybrid index Keyword count R-tree (KcR-tree) to store and summarize the spatial and textual information of objects. To further improve the performance, three query optimization techniques, i.e., KcR*-tree, lazy upper-bound updating, and keyword set filtering were used. We also extend reverse keyword search for spatio top-k queries (RST Q) to allow the input location to be a spatial region instead of a point. A recently proposed query retrieves m objects within a minimum diameter that match the given keywords. More recently, a spatial approximate string query that is a range query augmented with a string similarity search predicate has been proposed in. The main focus of this paper is reverse spatio-textual query.**
Keywords— *Spatio-Textual Queries, Top-K Queries, Reverse Spatial Textual k nearest Neighbor, Reverse Keyword Search.*

I. INTRODUCTION

A. Three Tier Architecture

A three-tier architecture is a client-server architecture in which the functional process logic, data access, computer data storage and user interface are developed and maintained as independent modules on separate platforms.

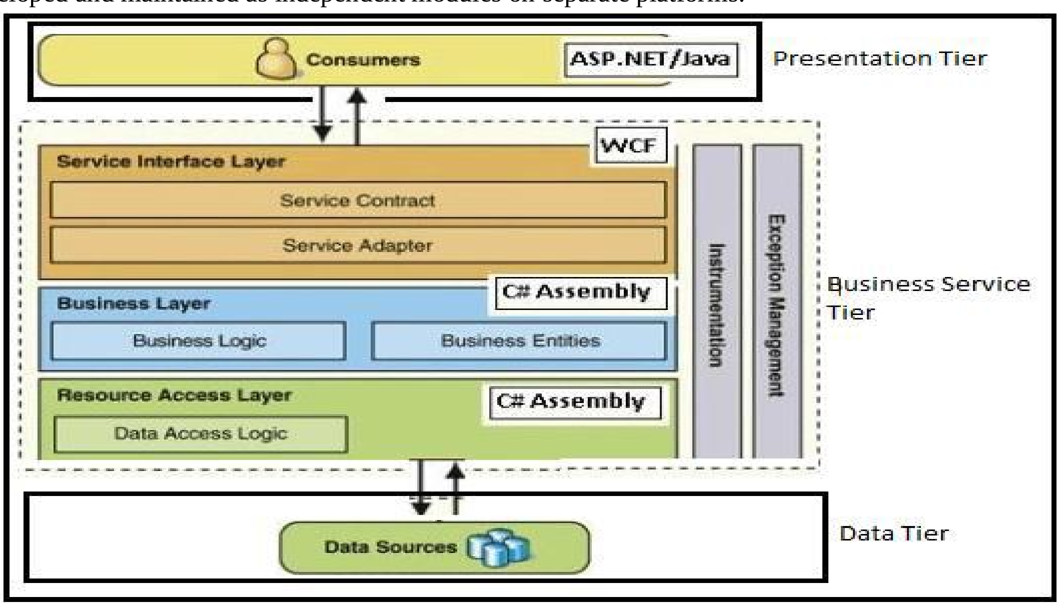

Fig 1. Three tier architecture

A 3-tier application is an application program that is organized into three major parts, each of which is distributed to a different place or places in a network. The three parts are:

- The workstation or presentation interface
- The business logic
- The database and programming related to managing it.

In a typical 3-tier application, the application user's workstation contains the programming that provides the graphical user interface (GUI) and application-specific entry forms or interactive windows. (Some data that is local or unique for the workstation user is also

Volume 4 Issue IX, September 2016
ISSN: 2321-9653

kept on the local hard disk.) Business logic is located on a local area network (LAN) server or other shared computer. The business logic acts as the server for client requests from workstations. In turn, it determines what data is needed (and where it is located) and acts as a client in relation to a third tier of programming that might be located on a mainframe computer. The third tier includes the database and a program to manage read and write access to it. While the organization of an application can be more complicated than this, the 3-tier view is a convenient way to think about the parts in a large-scale program.
A 3-tier application uses the client/server computing model. With three tiers or parts, each part can be developed concurrently by different team of programmers coding in different languages from the other tier developers. The 3-tier model makes it easier for an enterprise or software package to continually evolve an application as new needs and opportunities arise.
Existing applications or critical parts can be permanently or temporarily retained and encapsulated within the new tier of which it becomes a component.

B. What is client/server?

The Client/Server computing model implies a form of processing when requests are submitted by a client or requests the server which processes them and returns the result to the client. The client and the server are two separate logical entities working together over a network to accomplish the task. Conceptually, the client server architecture can be defined as a special case of Co-operative processing where on entire application is shared between the client and a server system.

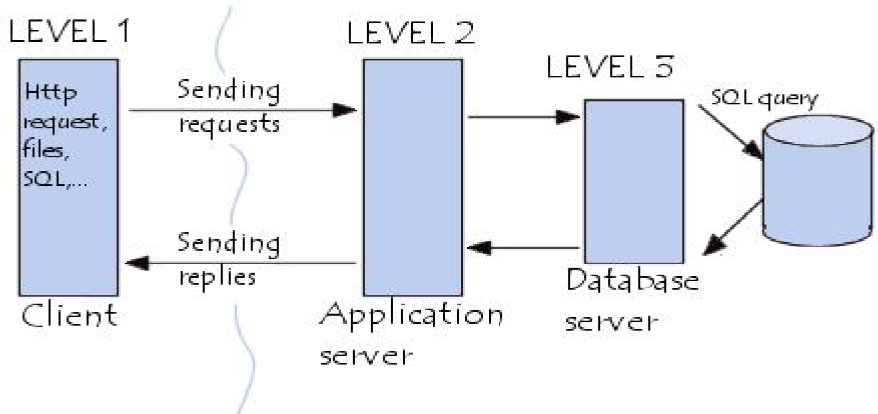

Fig 1.2 Client server architecture

C. Features Of Client/Server Computing

1) Improved access to information due to internet
2) Globalization of information
3) Easier maintenance of application and data
4) Graphically oriented, high interactive user interface
5) Increased developer productivity through ease of tools

In our paper we have divided core part into two parts. ASP pages, HTML pages are used as user interface (client). They gather the information from the user and process them. Sql 2000 is stored in IIS, which is used as server.

D. Development Approach

1) *Top down Approach:* The importance of new system is that it is user friendly and a better interface with users working on it. It can overcome the problems of manual system and the security problem. Top down approach of software development is the incremental approach to the construction of program structure. Modules are integrated by moving through the control hierarchy, beginning with the main control module. Modules subordinate to the main control modules are incorporate into the structure in either a depth first or breadth first manner. The top down approach is performed in a serious of five steps:

a) The main module that is the overall software is divided into five modules that are under the control of the main control module.
b) Depending on the top down approach selected subordinate stubs is replaced one at a time with actual components.
c) Tests are conducted as each component is integrated.
d) On completion of each test another stub is replaced with real time component.
e) Regression testing may be conducted to ensure that new errors have not been introduced.

II. RELATED WORKS

The manual work and the existing system of the application are very tedious. To overcome the disadvantage of existing system, proposed system is built .This deals with the Related works of the existing paper, their disadvantages and the related work of the proposed system are discussed.

A. Efficient Probabilistic Reverse Nearest Neighbour Query Processing On Uncertain Data

Given a query object q, a reverse nearest neighbour (RNN) query in a common certain database returns the objects having q as their nearest neighbour [3]. A new challenge for databases is dealing with uncertain objects. In this paper we consider probabilistic reverse nearest neighbour (PRNN) queries, which return the uncertain objects having the query object as nearest neighbour with a sufficiently high probability. I propose an algorithm for efficiently answering PRNN queries using new pruning mechanisms taking distance dependencies into account. We compare our algorithm to state-of-the-art approaches recently proposed. My experimental evaluation shows that the approach is able to significantly out perform previous approaches. In addition, we have shown how our approach can easily be extended to PRKNN (where k >1) query processing for which there is currently no efficient solution.

B. Retrieve Top-K Prestige Based Spatial Locations

The location-aware keyword query returns ranked objects that are near a query location and that have textual descriptions that match query keywords [7]. This query occurs inherently in many types of mobile and traditional web services and applications, e.g., Yellow Pages and Maps services. 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 [4]. We propose two algorithms that compute LKPT queries.

C. Spatial Keyword Query Processing: An Experimental Evaluation

Geo-textual indices play an important role in spatial keyword querying. The existing geo-textual indices have not been compared systematically under the same experimental framework [9]. This makes it difficult to determine which indexing technique best supports specific functionality. We provide an all around survey of 12 state- of-the-art geo-textual indices. We propose a benchmark that enables the comparison of the spatial keyword query performance. We also report on the findings obtained when applying the bench-mark to the indices, thus uncovering new insights that may guide index selection as well as further research. In this chapter we have seen about the analysis of the system.

III. PROBLEM DEFINITION

UIDIS In GeoWeb, each object (e.g., POI web page) is associated with a geographical location and a textual description, which enables a wide range of location-based applications. A fundamental service in GeoWeb is “spatio-textual” query that takes a user location and a keyword set as inputs, and returns the most spatially and textually relevant objects. Users find it difficult to formulate their query keyword and instead prefer to choose from candidate keyword sets. The problem in this paper is, all the objects related to the search keyword is displayed without any priority.

IV. PROPOSED WORK

The spatio-textual query retrieves a group of spatial web objects such that the group’s keywords cover the query keywords and the objects are the nearest to the query location. Fan et al studied the spatio-textual similarity search on regions of interest (ROIs) that contain region based spatial information and textual descriptions. The measurement takes into account the personalized deviation of Web services QoS and user’s QoS experiences, in order to improve the accuracy of similarity computation. Although several CF based Web service QoS prediction methods have been proposed in recent years, the performance still needs significant improvement. We propose a location-aware personalized CF method for Web service recommendation. The proposed method leverages both locations of users and Web services when selecting similar neighbors for the target user or service. To evaluate the performance of our proposed method, we conduct a set of comprehensive experiments using a real-world Web service dataset. Based on the above enhanced similarity measurement, we proposed a location-aware CF-based Web service QoS prediction method for service recommendation.

A. Proposed System Algorithms

The algorithm proposed was the Efficient Query Processing Algorithm .We first formally define notations for the convenience of describing our method and algorithms. The Top-K similar neighbour selection algorithm can be employed to select K Web services that are most similar to the target Web service. The algorithm first searches local users for similar users.

This algorithm has a high probability of finding users similar to the active user in his/her local region. Prediction coverage is also an important metric for evaluating a QoS prediction algorithm. The proposed system benefits in addition to the prediction accuracy, another advantage of our method is its high efficiency of QoS prediction. This indicates that our method is more scalable than traditional CF methods when applied to large-scale service recommender systems. The reason is that, in most cases we can limit similar neighbour searching to a small subset of users (or Web services), especially when K is small.

V. EXPERIMENTAL RESULTS

System design is the process of planning a new system to complement or altogether replace the old system. The purpose of the design phase is the first step in moving from the problem domain to the solution domain. System design is also called top-level design. The design phase translates the logical aspects of the system into physical aspects of the system.

A. User And Admin Login

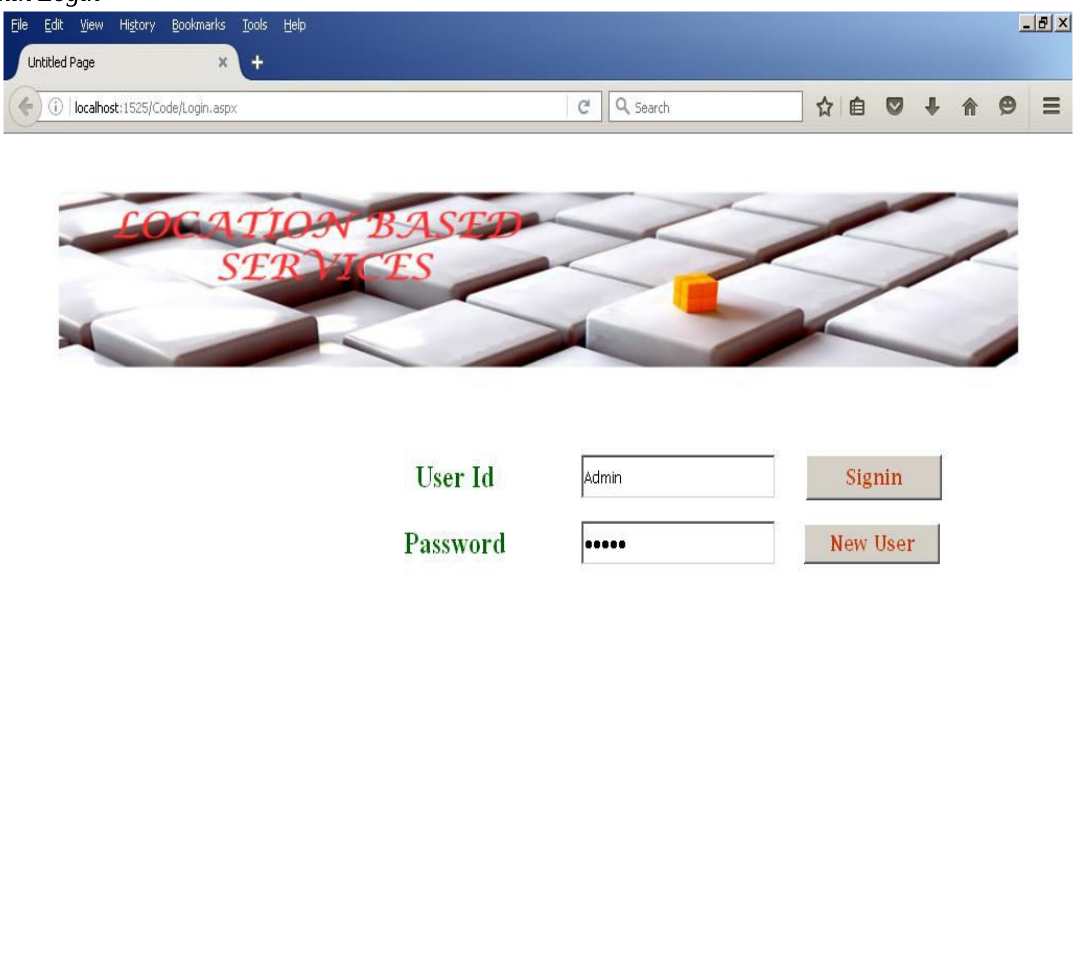

Fig 5.1 User and admin login

Volume 4 Issue IX, September 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

B. Services Of Admin

Fig 5.2 Services of administrator

C. Adding Base Stations

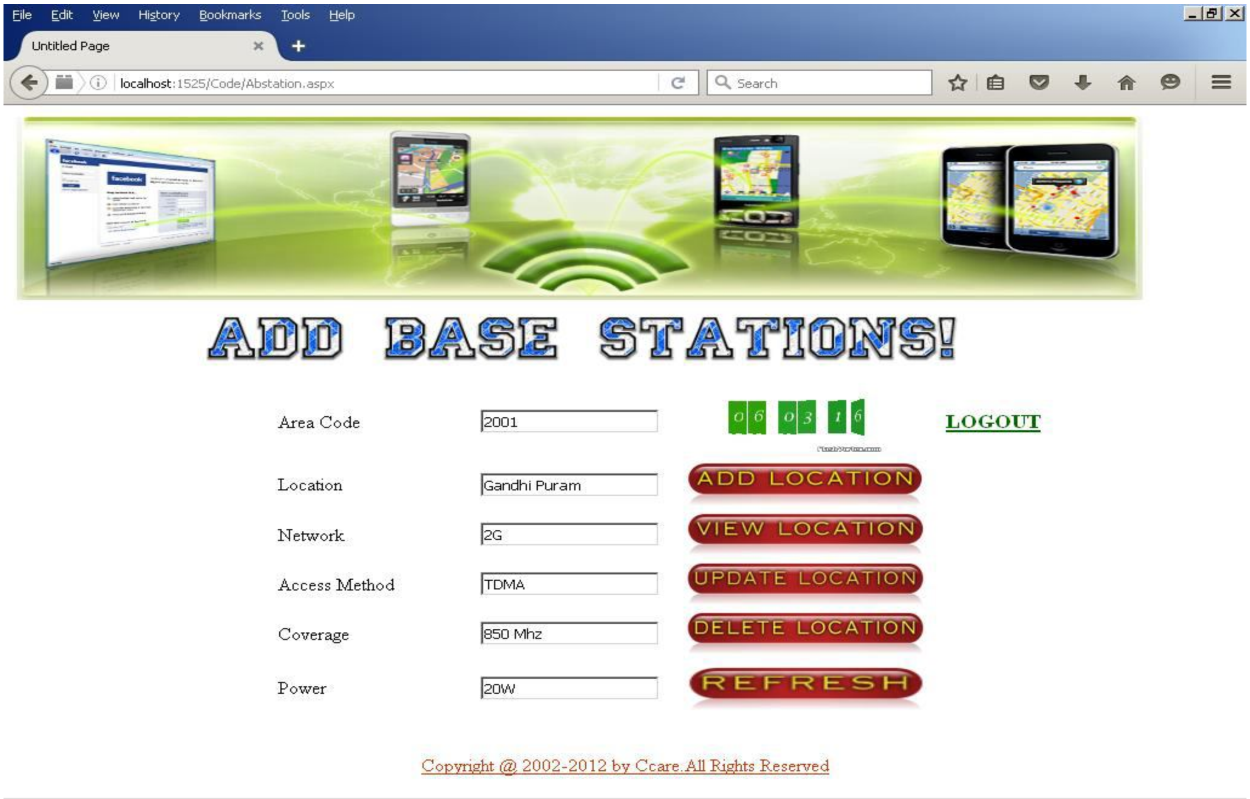

Fig 5.3 Adding base stations

Volume 4 Issue IX, September 2016
ISSN: 2321-9653

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

D. User Registration

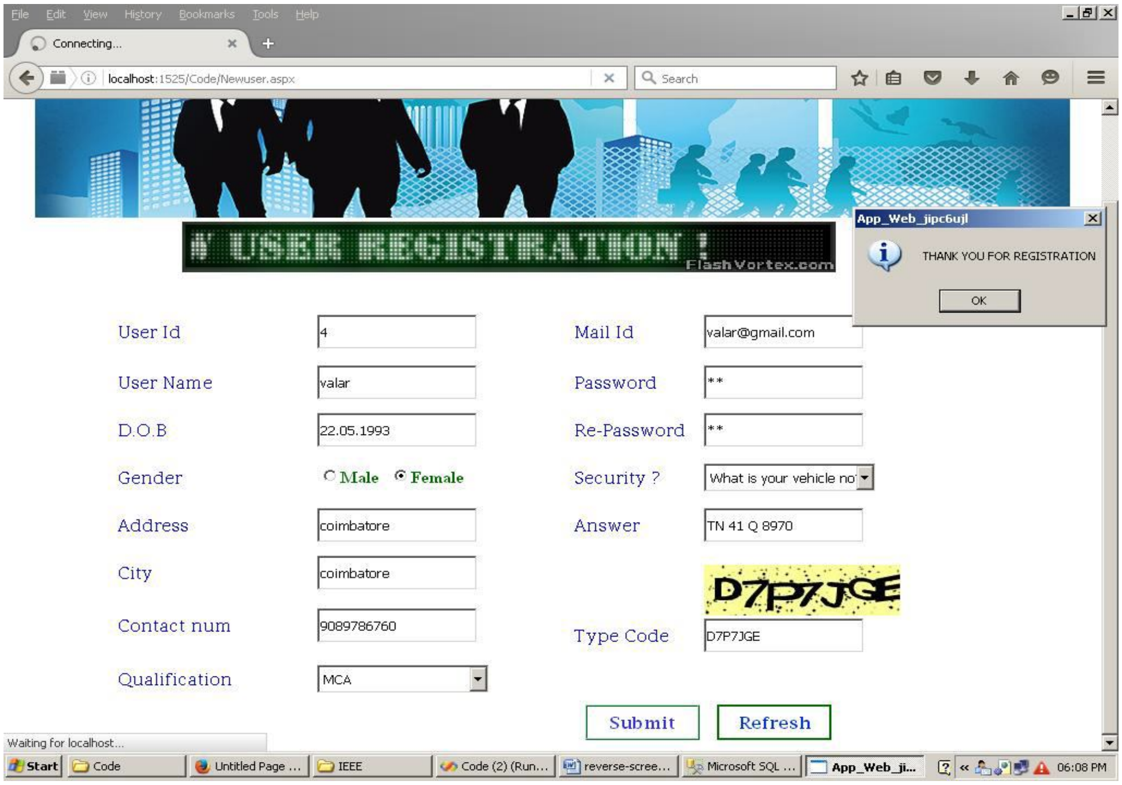

Fig 5.4 User registration

VI. PERFORMANCE EVALUATION

A. Experimental Setup

1) *Implemented algorithms:* For point-based RSTQ, we implement the KcR-tree-base algorithm with all optimizations. For comparison, we also implement two baseline algorithms. The first one is a naive method based on the IR-tree.

DATASET	DP	CD
Total # of objects	121,0822	2,249,727
Total # of unique keywords	62,382	292,087

Table .1. Dataset Description

2) *Datasets:* The experiments are conducted on two datasets: DianPing (DP) and CaliforniaDBpedia (CD). The DP dataset was crawled from dianping.com, the largest chinese restaurant review site. The keywords of the objects were extracted from the comments associated to them by using a popular Chinese word processing tool ICTCLAS.3 The average number of keywords per object is 31. CD is a synthesized dataset which combines the spatial data in California and a real collection of article categories from DBpedia. Table 1 summarizes the statistics of each dataset. Due to space limitations, in the sequel the results of DP dataset are presented unless otherwise stated.

PARAMETER	VALUE RANGE	DEFAULT
Dataset cardinality	30,000-1,000,000	121,082
l	2-4	2
Keyword number of target object	10-40	31
ω_s	0.1,0.5,0.9	0.5
ω_t	0.9,0.5,0.1	0.5
k	10-50	10
t (the target object is the *t*th- NN object to the query point in the spatial dimension)	0.1k-2k	0.5k
Edge length of query region	100-3,000 m	1,000 m

Table .2. Parameter Settings

All objects are assumed to be located in a 100,000 m * 100,000 m space. For performance metrics, the CPU time is measured and the I/O cost is measured. The default settings and value ranges of the system parameters are given in Table 2.

B. Experiments For Point-BASED RSTQ

1) *Effect Of Candidate Keyword Set Size*: The evaluation of the effect of varying the parameter l, the maximum number of query keywords was done, that result in different sizes of candidate keyword sets. In our DP dataset, when l is set to 2, 3, and 4, the average size of candidate keyword sets is 450, 4,500, and 32,000, respectively. As shown in Fig. 6.2.1, the CPU time of all algorithms grows proportionally to the number of candidate keyword sets. This is because the ranking bounds are estimated independently for each keyword set.

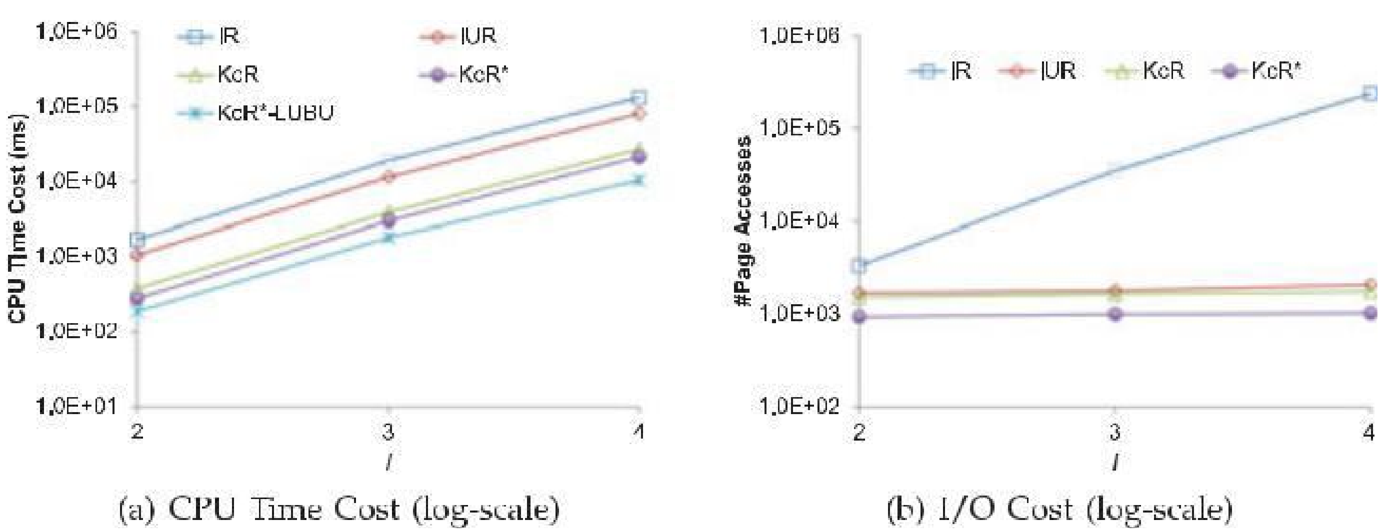

Fig. 6.2.1. Effect of candidate keyword set size (DP)

2) *Effect of dataset cardinality:* This experiment tests the scalability of the proposed algorithms in terms of the dataset cardinality. Given a cardinality setting, the objects are randomly selected from the synthesized CD dataset. As shown in Fig. 6.2.2, even if the number of the objects grows up to 1 million, the CPU time of the proposed algorithms is still well below 2s. This can be explained as follows. Although the total number of objects is increased, the accessed nodes are limited to those which are close to the query point in spatial and textual dimensions. In addition, many candidate keyword sets are still filtered as the high-level index nodes are accessed.\

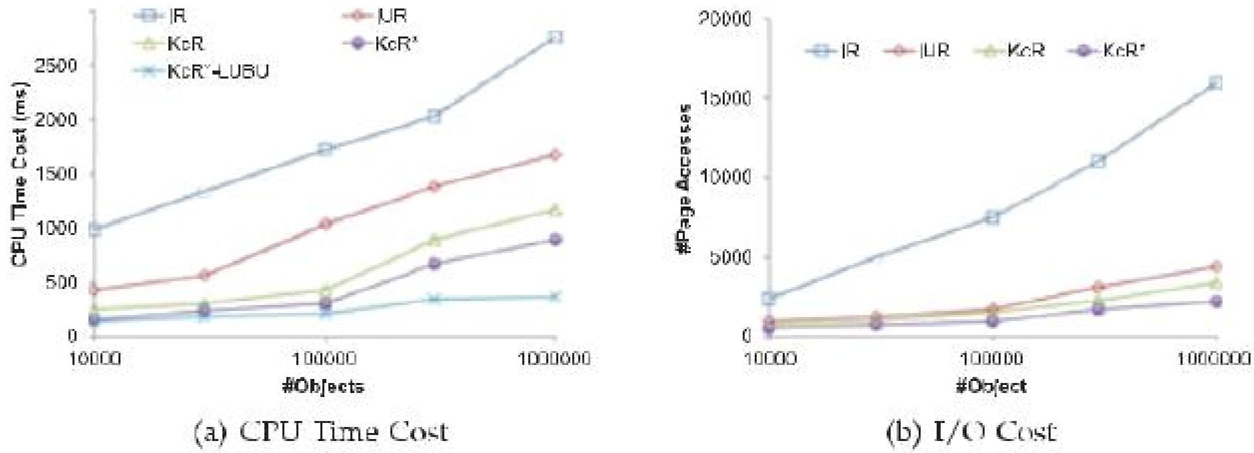

Fig. 6.2.2. Effect of dataset cardinality (CD)

3) *Effect Of* ***ωs***: In this experiment, the setting of **ωs** in the scoring function is varied. As shown in Figure 6.2.3, as **ωs** decreases, the performance of IUR-tree, KcR-tree, KcR*-tree, and KcR*-LUBU is improved, while that of IR-tree is degraded. Note that the gap between KcR-tree and KcR*-tree becomes even larger with a smaller value of **ωs**, since the clustering of KcR*- tree considering the textual dimension plays a more important role in this case.

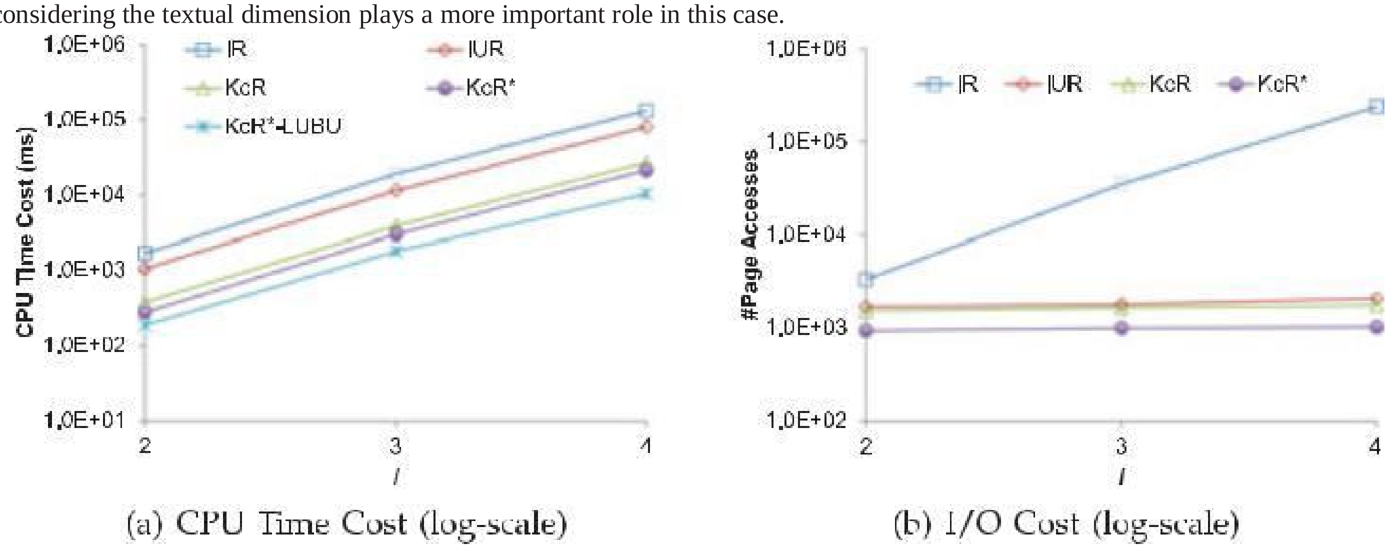

Fig. 6.2.3 Effect of **ωs** (DP)

VII. CONCLUSIONS AND FUTURE WORK

In this paper, we have studied the problem of reverse keyword search for spatio-textual top-k queries (RST Q). We have devised a hybrid index KcR-tree to store the spatial and textual information of objects to accelerate the processing of RST Q. Also, we have proposed three query optimization techniques, i.e. KcR*-tree, lazy upper bound updating, and keyword set filtering to further optimize the performance. For region-based RST Q, we have proposed a reduction-based technique to avoid enumerating an infinite number of query points. Extensive experimental results demonstrate the efficiency of the proposed methods and algorithms under various system settings. In particular, my proposed method out performs the baseline query processing method by up to 97% in

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

terms of the CPU time and up to 99% in terms of the I/O cost.

As for future work, we will extend this work to the why-not problem, which aims at promoting the ranking of the target object with the least query modification. Moreover, we are also interested in studying RST Q in a dynamic environment where the keywords of the objects are changing. We believe a dynamic indexing scheme needs to be designed for this purpose.

REFERENCES

[1] L. Chen, G. Cong, C. S. Jensen, and D. Wu. Spatial Keyword Query Processing: An Experimental Evaluation. PVLDB, 6(3): 217-228, 2013.
[2] J. Fan, G. Li, L. Zhou, S. Chen, and J. Hu. SEAL: spatio-textual similarity search. PVLDB, 5(9):824-835, 2012.
[3] M. A. Cheema, W. Zhang, X. Lin, and Y. Zhang. Efficiently processing snapshot and continuous reverse k nearest neighbour's queries. VLDB J., 21(5):703-728, 2012.
[4] T. Bernecker, T. Emrich, H.-P. Kriegel, M. Renz, S. Zankl, and A. Zufle. Efficient probabilistic reverse nearest neighbour query processing on uncertain data. PVLDB, 4(10):669-680, 2011.
[5] X. Cao, G. Cong, C. S. Jensen, and B. C. Ooi. Collective spatial keyword querying. In SIGMOD, 2011.
[6] M. A. Cheema, X. Lin,W.Wang,W. Zhang, and J. Pei. Probabilistic reverse nearest neighbour queries on uncertain data. TKDE , 22(4):550-564, 2010.
[7] X. Cao, G. Cong, and C. S. Jensen. Retrieving top-k prestige-based relevant spatial web objects. PVLDB 3(1): 373-384, 2010.
[8] G. Cong, C. S. Jensen, D. Wu. Efficient retrieval of the top-k most relevant spatial web objects. PVLDB, 2(1):337-348, 2009.
[9] I. De Felipe, V. Hristidis, and N. Rishe. Keyword search on spatial databases. In ICDE, pp. 656-665, 2008.
[10] Y.-Y. Chen, T. Suel, and A. Markowetz. Efficient query processing in geographic web search engines. In SIGMOD, pp. 277-288, 2006.