OPTIMIZED RESULTS USING FAST NEAREST NEIGHBOR KEYWORD COVER SEARCH

B.BINDU,
M.Tech (Cse) From Jagruti Institute of Engineering and Technology

Mr. P. GIRIDHAR
Assistant Professor, Department Of Computer Science and Engineering, Jagruti Institute of engineering And Technology, Telangana State, India.

ABSTRACT:

Now days there are many modern android applications that aim to find out objects satisfying both spatial predicate and a predicate on their associated texts. In this paper, for finding nearest ATM CENTERS a simple solution is introduced which relies on IR2 tree [1].IR2 tree includes few deficiencies that affect its efficiency [2]. To increase the efficiency a new method called spatial inverted index [1] is introduced that extends the standard inverted index to address multidimensional information. This new SI(selection index) method comes with algorithms which will answer nearest neighbor queries with keywords in real time. As verified by experiments, this projected technique performs the IR2 tree in query time reaction very significantly.

Keywords: Keyword Search, Spatial Inverted Index. Information Retrieval Tree, selection index method .

1. INTRODUCTION

Present system searches location on the basis of their geometric location from the client location. In this paper, we are going to add to an application which will search the nearest location with given Keywords.

In this paper, we considered only Bank as a location and Keyword as a Bank ATMs. For example, if a client wants search nearest Bank then he can discover it with its Administrations. Means if a client wants "SBI" only then he can enter SBI as the Keywords then it will return nearest Banks which has SBI ATMs. As of now, the best solution to such questions is based on the IR2-tree, which is utilized as a part of this paper. This algorithm is exceptionally proficient to search location with given Keywords [1]. Also, there is a system called spatially transformed file that is utilized with multidimensional data and that accompanies nearest neighbor search with given Keywords.

Related Work: A spatial database manages multidimensional objects (such as points, rectangles, etc.), and provides fast access to those objects based on different selection criteria. The importance of spatial databases is reflected by the convenience of modeling entities of reality in a geometric manner [3]. For example, locations of Branches, Banks, hospitals and so on are often represented as points in a map, while larger extents such as parks, lakes, and landscapes often as a combination of rectangles [4]. Many functionalities of a spatial database are useful in various ways in specific contexts [5].

Existing System: In the previous system, real nearest neighbor lies quite far away from the query point, while all the closer neighbors are missing at least one of the

query keywords. As its fail to find the nearer location of Branches having with all the keywords or available ATMs of its Branches. Existing system mainly focus on finding the nearest top neighbor where each node have to match whole query keywords. All these method are low efficient for incremental query.

Proposed System: To overcome the drawbacks of previous applications, we proposed an application for android users. In our system we are mainly dealing with searching and nearer location issues and database manage multidimensional objects which resulted in failure of previous systems [2]. To deal with spatial index as searching the entered keyword and from that find the nearest location having that keyword available and showing the location of Branches having ATMs available in map. So easier to find the location of nearer Branches in map having the available keyword.

II. System functions

Client Registration: In this module, the user will have to register first. Once the user does the registration then he/she can access the application. For registration user have to enter the basic information about him. Users also have to set the username and password. This all registration information is get stored into database. The IMEI number is automatically get stored into database once user do the registration.

Client Login: In this module, after the registration Client can login through mentioned username and password.

Bank Registration: In this module, Admin register the Bank with its Services. Bank owner have to do the registration then only the Bank get search through application. Also Bank owner have to add the ATMs which is available in the Bank so that user can search the Bank through keyword. Only registered Banks will be displayed in the application. This Bank's location will be seen in the map with distance. Each Bank owner will get the separate username and password for login. Module 3.1: Bank Login / Admin: In this model once Bank Owner login into application then he can insert the ATMs or update the ATMs.

Searching Keyword: In this module, the user will enter the keyword searching for ATMs available in Places which will nearer from its position. Whenever user will enter keyword (ATM name) it will match data with the bank database server and find the nearest branch with the available entered ATM by Client. For nearest Branch we are using IR2tree & compression. The IR2-Tree is a combination of an R-Tree and signature files. In particular, each node of an IR2-Tree contains both spatial and keyword information; the former in the form of a minimum bounding area and the latter in the form of a signature. An IR2-Tree facilitates both top-k spatial queries and top-k spatial keyword queries as we explain below. More formally, an IR2-Tree R is a height-balanced tree data structure, where each leaf node has entries of the form (Obj Ptr, A, S). Obj Ptr and A are defined as in the R-Tree while S is the signature of the object referred by Obj Ptr. Anon-leaf node has entries of the form (Node Ptr, A, S). Node Ptr and A are defined as in the R-Tree while

S is the signature of the node. The signature of a node is the superimposition (OR-ing) of all the signatures of its entries. Thus a signature of a node is equivalent to a signature for all the documents in its sub tree.

Algorithm: Input:-ATMs list, Input query

Algorithm IR2TreeGeneration ()

{

1. Let Hl be the list of Banks along with their latitude and longitude

Let Hm be the ATMs available in respective Bank

2. Let Sm be the signature set for each ATMs, For each (ATMs in Hl)

{ //signature generation Hashing (ATMs);

Add hash ATMs in Sm;

}

Let Sh be the signature for Bank list For each (Hl in List)

{

//

Generation of signature for each Bank Hashing (ATMs (i) OR ATMs (i+1)); Add hashing to Sh

}

3. Declare HLc is list of Bank having all ATMs which satisfies query For each (val in Sh)

{

Check if (q = val) Add Bank in HLc

}

4. Declare SHl sorted list of Banks from HLcS Declare Dmin is min distance from list of sorted Banks //List along with distances Hl=Sorted list (HLc); Dmin=min(Hl)

}

7. Sorting User can search the Bank by sorting the names of Banks.

Map view: all the names of BANKs will appear in the list which came from database and find the position in map (google_play_service_library) is required for showing position of BANKs in map and which will be more easier to Client to get the nearer BANKs from its current position.

Distance Search: Client can find the distance from source to destination. So that it can be easier to find the distance and reached the destination. It will give the distance of the Bank from the current location. Logic for calculating distance between user and Bank:

double dLat = Math.toRadians(lat1-lat2);

double dLng = Math.toRadians(lng1-lng2);

double a = Math.sin(dLat/2) * Math.sin(dLat/2)
+Math.cos(Math.toRadians(lat2)) * Math.cos(Math.toRadians(lat1)) * Math.sin(dLng/2) * Math.sin(dLng/2);
double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));

double dist = earthRadius * c;

Solutions based on inverted indexes:

Inverted indexes (I-index) have proved to be an effective access method for keyword-based document retrieval [1]. In the spatial context, nothing prevents us from treating the text description Wp of a point p as a document, and then, building an I-index. Each word in the vocabulary has an inverted list, enumerating the ids of the points that have the word in their documents. According to the experiments of [2], when Wq has only a single word, the performance of I-index is very bad, which is expected because everything in the inverted list of that word must be verified. Interestingly, as the size of Wq increases, the performance gap between Iindex and IR2-tree keeps narrowing such that I-index even starts to outperform IR2-tree at |Wq| = 4. This is not as surprising as it may seem. As |Wq| grows large, not many objects need to be verified because the number of objects carrying all the query keywords drops rapidly. On the other hand, at this point an advantage of Iindex starts to pay off. That is, scanning an inverted list is relatively cheap because it involves only sequentialI/Os1, as opposed to the random nature of accessing the nodes of an IR2- tree.

3. RESULT AND DISCUSSION

The SI-index, accompanied by the proposed query algorithms [1], has presented itself as an excellent tradeoff between space and query efficiency. Compared to IFR, it consumes significantly less space, and yet, answers queries much faster. Compared to IR2-tree, its superiority is overwhelming since its query time is typically lower by a factor of orders of magnitude.

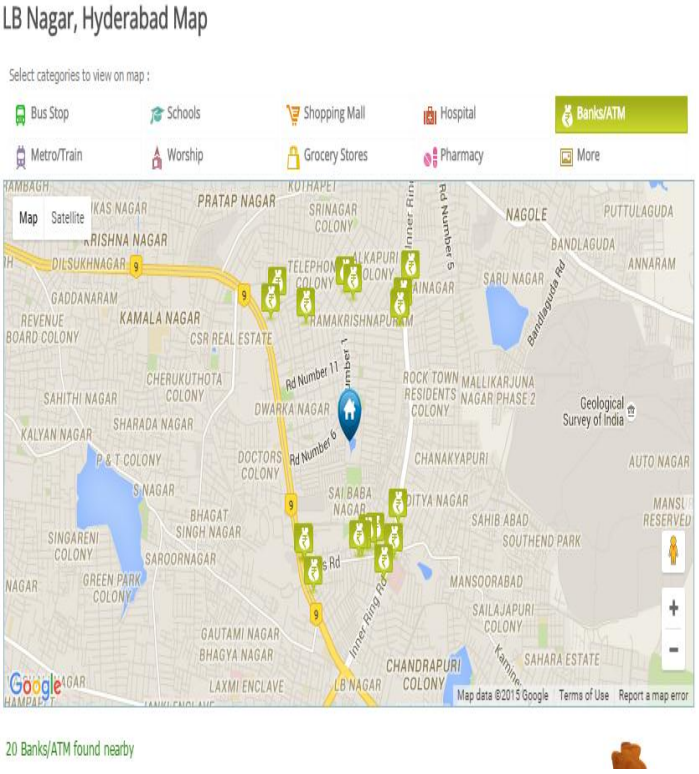

Fig 1. List of banks available at LB Nagar (Keyword)

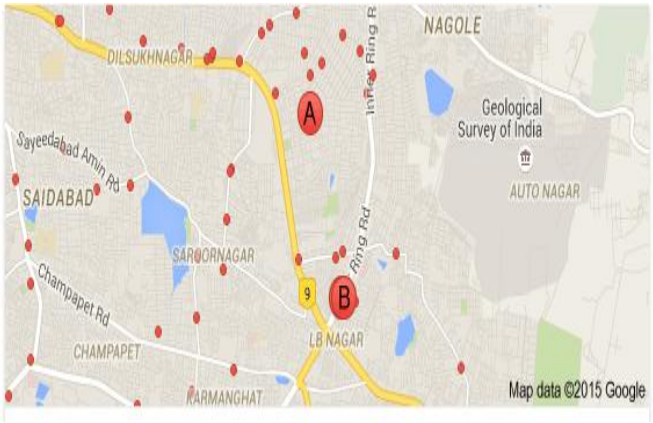

Fig 2. List of SBI ATMs Available at LB Nagar

4. CONCLUSION

In This paper focus on Fast Nearest Neighbor Search with Keywords is extremely effective for searching nearest BANKs from user with expected ATMs. It does this by IR2 tree algorithm-Compression, Merging and Distance Browsing, and GPS System. In this we demonstrate this application using android application we have so many applications that can be used as search engine which is able to efficiently support novel forms of spatial queries that are integrated with keyword search. In this Paper we have developed an access method called the Spatial Inverted Index. This method is fairly space economical and it has ability to perform keyword augmented nearest neighbor search in real time. This method is based on conventional technology of Inverted Index. It is readily incorporable in a commercial search engine.

V. REFERENCES

1) Yufei Tao and Cheng Sheng In proc of Chinese University "Fast Nearest Neighbor Search with Keywords"

2) I. D. Felipe, V. Hristidis, and N. Rishe. Keyword search on spatial databases. In Proc. of International Conference on Data Engineering (ICDE), pages 656–665, 2008.

3) N. Beckmann, H. Kriegel, R. Schneider, and B. Seeger. The R*-tree: An efficient and robust access method for points and rectangles. In Proc. of ACM Management of Data (SIGMOD), pages 322–331, 1990.

4) G. Bhalotia, A. Hulgeri, C. Nakhe, S. Chakrabarti, and S. Sudarshan. Keyword searching and browsing in databases using banks.In Proc. of International Conference on Data Engineering (ICDE), Pages 431–440, 2002.

5) C. Faloutsos and S. Christodoulakis. Signature files: An access method for documents and its analytical performance evaluation. ACM Transactions on Information Systems (TOIS), 2(4):267–288, 1984.

[6]. X. Cao, L. Chen, G. Cong, C. S. Jensen, Q. Qu, A. Skovsgaard, D. Wu, and M. L. Yiu. Spatial keyword querying. In ER, pages 16–29, 2012.

[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]. X. Cao, G. Cong, C. S. Jensen, and B. C. Ooi.Collective spatial keyword querying. In Proc. of ACM Management of Data (SIG-MOD), pages 373– 384, 2011.

[9]. B. Chazelle, J. Kilian, R. Rubinfeld, and A. Tal. The bloomier filter: an efficient data structure for static support lookup tables. In Proc. of the Annual ACM- SIAM

Symposium on Discrete Algorithms(SODA), pages 30–39, 2004.

[10]. Y.-Y. Chen, T. Suel, and A. Markowetz. Efficient query processing in geographic web search engines. In Proc. of ACM Managementof Data (SIGMOD), pages 277–288.

, 2006

B.BINDU is pursuing M.Tech degree in, Computer Science and Engineering from Jagruti Institute of Engineering and Technology, Telangana State, India.

Mr. P.GIRIDHAR is presently working as Assistant Professor in, Department of computer science and engineering, Telangana State,India.