International Journal of Engineering & Technology, 7 (2.19) (2018) 17-20

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Efficient NN Spatial Keyword Search Using Spatial Inverted (SI) Index

B.A.Vishnupriya[1]*, N.Senthamarai[2], S.Bharathi[3]

*[1,2,3]Assistant Professor, Department of Computer Science and Engineering, School of Computing,
Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology,Avadi, Chennai-62, TamilNadu, India.
Email: vishnupriya@veltech.edu.in

Abstract

"Spatial information mining", or learning revelation in spatial database, alludes to the illustration out of concealed information, spatial relations, or different examples that are not unequivocally put away in spatial databases. To get to the spatial database alongside the catchphrase another kind of inquiry called spatial watchword question is utilized. A spatial watchword inquiry get client area and client given catchphrases as contentions and gives web protests that are spatially and literarily material to these information. The current answers for such inquiries depend on IR2-tree that has a couple of inadequacies as space utilization and event of false hit is extremely huge when the question of the last outcome is far from the inquiry point .To beat this issue a novel file structure called Spatial Inverted file is proposed. Presently a-days use of portable is expanding enormously .In the versatile system an intermediary is set between base station and Location Based Server (LBS).This intermediary utilizes the Spatial modified file procedure to answer the SK inquiry by utilizing spatial data from the base station and printed data from the client question. The outcome from the SI record is given to two file structure in the intermediary called EVR Tree and Grid list. The Estimated Valid Region (EVR) for the present area of the client and required spatial articles are produced and come back to the client. On the off chance that the EVR is absent in the two file structure of intermediary it offer question to LBS. In the event that the client given inquiry is miss written or miss spelled it can be oversee by SI record utilizing n gram/2L Approximation file.

Keywords: *Inverted index: Multidimensional objects: Nearest neighbor Search: Spatial keyword query: spatial query processing*

1. Introduction

The raising prominence of administrations, for example, Google Earth and Yahoo Maps, and also other geographic applications, questions in spatial databases has turned out to be slowly imperative as of late. Ebb and flow investigate on inquiries goes well past unadulterated spatial questions, for example, closest neighbor questions, spatial range inquiries, and spatial joins.

1. **Nearest neighbor query:** Find the nearest question the question protest Q. The issue may likewise be stretched out to k closest neighbor (kNN) query that is to discover k nearest questions the query protest. For instance, discover the 10 urban communities closest to Paris.

2. **Spatial range query:** Find all the objects that lie within a given distance from the query object Q. For example, find all the restaurant that are within 10 km from current location.

3. **Spatial join query:** Give R and S a chance to be two arrangements of articles, a spatial join query is to discover protest sets from these sets with the goal that they fulfill some predetermined join predicate. The case of conceivable join predicates incorporate INTERSECTS, DISJOINT, DISTANCE and CONTAINS operations. A case of spatial join query is to discover all inn film combines so separate between them is less that 1km.

Questions on spatial articles related with printed data rep-loathed by watchwords are starting to get huge consideration from the spatial database explore group and the business. Most mobile users require location specific information while searching on web. For example, if the user enters keyword as “cardio care” the spatial database has to provide the nearest cardio care hospital by considering the current user location. This type of query is called spatial keyword queries [2] i.e. the result has to satisfy the spatial predicate along with the textual information. These SK queries use two types of indexing scheme i.e. using proximity indexing to handle spatial data and textual indexing to handle with textual data. Obtaining the exact location of mobile user is a difficult process. It is difficult for server to answer for query given by user.

The rest of this paper is organized as follows. In Section 2, the study of spatial keyword query is introduced. Section 3 reviews about the existing system. In section 4 the proposed system is presented and section 5 concludes this paper.

2. Related works

2.1 Hybrid Index Structures for Location-Based Web Search

Prevalence of area construct administrations with respect to the web, offering approach to new issues, for example, discovering site pages whose substance is identified with a specific place or locale, and how to speak to the spatial piece of the web content in order to effectively process spatial catchphrase queries on the web indexes. The majority of the website pages speak to geo-graphical extension as place names. Zhou et al [13] propose three approaches utilizing a cross breed file structure that joins rearranged records with R*-Trees [1]. Three distinctive joining record plans are considered:

(a) Inverted file and R*-tree double index,
(b) First inverted file then R*-tree and
(c) First R*-tree then inverted file.

Area based web search tool work process contains disconnected preparing and internet handling. The previous incorporates extricating geographic degrees and ordering site pages as per their extensions. The last incorporates recovering area mindful data, positioning and exhibiting the recovered outcome. These models in light of proposed mixture file structures comprise of four sections:

1) Extractor

Extracts geographical variety of pages and translates them to MBRs. A gazetteer converts location name into MBRs.

2) Indexer

Assemble half breed file structures to incorporate content and loca-tion data of website pages. For literarily list site pages, inverted records are utilized and to get to spatially, two-dimensional spatial files are utilized, for instance, R-tree family, quad-tree and matrix structure. R-tree [1] utilizes the base jumping rectangle (MBR) as an estimation to a spatial question, which is like our approximation of the geological degree. R*-tree is a variation of R tree that can additionally propel seek execution. Among the three record structure initially altered document then R*-tree give better query time.

3) Ranker

It positions comes about by land significance and also non-topographical importance. The objective of the ranker is to restore those imperative pages which are most important to content catchphrases as well as most significant to question areas.

4) Interface

User friendly to give input for location-based search queries and to obtain geographical and textual applicable results.

2.2 Processing Spatial Keyword (SK) Queries In Geographic Information Retrieval (GIR) Systems

Hariharan et al [8] proposed an indexing structure called kR* tree based on the third method of Zhou et al [13] i.e. First R* tree then inverted file. Applications such as catastrophe reaction, critical infrastructure protection and liability analysis need more information than that of location based information contained in GIS database. GIS system that provides the retrieval of both the spatial and textual information is Geographic Information Retrieval (GIR) systems. A hybrid index structure is proposed that reduce performance bottleneck by combining both the spatial and textual information in a meaningful way.

2.2.1 Construction of GIR System

The system mainly consist of four main components namely GIR database, Indexer, Ranker and Interface.

GIR Applications:

The class of applications that take *SK* queries as input, process them and output information sources that are relevant to the queries are called as GIR applications. Geographic (or local) search engines, location-based services (LBS) and GIS search engines are various examples of GIR applications. This system can abuse both the unequivocal and verifiable geographic references found in the data sources. For instance, in un-organized sources, for example, Web pages, geographic references are certainly found as place names, phone numbers, addresses, and so forth alongside the other content issue. These certain references are ex-tracted and changed over to metric estimations. In organized sources, for example, GIS databases, the geographic references are unequivocally characterized as spatial characteristics alongside non-spatial content properties.

GIR Database:

The contribution to a GIR database can originate from unstructured and organized information sources. There are various pre-handling steps engaged with changing over an unstructured source to a GIR database, for example, geo extraction, geo coordinating and geo engendering.

Indexer:

The essential scan measurements for preparing SK queries are spatial and content properties. Consequently, the indexer system fabricates information structures on spatial and literary properties. Every watchword in the content having a spatial circulation. The spatial appropriation of different catchphrases might be associated in space. The ordering system utilizes this spatial relationship of watchwords, subsequently catching their joint appropriation in space. Catching this joint dissemination fundamentally enhances the recovery of answers to SK queries.

Ranker:

It combines the ranking functions of spatial and textual attributes. The target of the this function is to assign score to objects in GIR database based on its relevance to the *SK* query. Given a *SK* query $Q = \{q_r, q_t\}$ the overall ranking function is

$$F_{sk} = \alpha_1 . F_r(q_r, o_r) + \alpha_2 . F_t(q_t, o_t), \quad (1)$$

Where $F_r(q_r, o_r)$ is a geographic ranking function, and $F_r(q_t, o_t)$ is a Keyword-based ranking function. α_1 and α_2 are suitable weights where $\alpha_1 + \alpha_2 = 1$.

Interface:

The interface enables the client to enter SK queries utilizing a guide and a content interface. Each SK queries comprises of an arrangement of printed catchphrases and a geographic locale of intrigue determined as an inquiry rectangle. The interface displays a positioned result to every question.

2.2.2 KR*-Tree: An Indexing Mechanism

Effectiveness of indexing strategies in answering SK queries with respect to the following two criteria:

- Pruning text and space.
- Managing queries with multiple keywords.

The KR*-tree is constructed in a way similar to how an R*-tree is constructed, but with minimal overhead in handling the keywords. There are two distinct steps involved in the construction of the tree.

Step1: First the set of keywords corresponding to each node of the tree is determined.

Step2: Next the set of keywords in each node is converted to a KR*-tree List.

3. Existing system

Ivan et al [6] proposed a new indexing technique called IR^2 tree (Information Retrieval R-tree) to answer spatial keyword queries. The IR^2 tree consist of R-Tree data structure to handle spatial information and Signature file in which the keywords are stored as hash-based bit strings called superimposed coding (SC).To find the query keyword in the document SC carry out membership tests.SC is conservative that if it result as "no" then the keyword is certainly not in the document. If it result returns "yes" then it search the entire document. The spatial proximity is given by the R-Tree.

P	Wp
p1	{A,B}
p2	{B,D}
p3	{D}
p4	{A,E}
p5	{C,E}
p6	{C,D,E}
p7	{B,E,}
p8	{C,D}

Figure 3.1 Spatial data objects along with texts

Word	Hashed bit String
A	00101
B	01001
C	00011
D	00110
E	10010

Figure 3.2 Hashed Bit string with length=5 & m=2

Consider the spatial objects along with their related keywords are shown in the Figure 3.1.The bit string for each word is computed using two parameters: l and m where l represents the length of the bit string and m represents the presence of number of ones in the bit string. Figure 3.2 shows the computed hashed bit string for each words with l=5 and m=2. For the sequence of words the bit string calculation is performed by OR operation of two or more keywords .For instant, to calculate the bit string for set of keywords {A, B} compute OR operation between the hashed bit string of A and B the result string for {A, B} is 01101.Likewise hashed string for two or more keywords is calculated.

From the given spatial objects and bit sting corresponding IR^2 tree is generated as shown in Figure 3.3.For a given query point q with keyword set {X, Y}, the algorithm call the entries of the IR^2-tree in increasing order of their MBRs to q and prune the entries whose signature does not satisfies the given keyword set. For Example 2^{nd}bit string of B contains one so B cannot be search from R_4 and R_6 since it does not contains one in the 2^{nd} bit. So it prune the R_4 and R_6.

These IR^2 tree for spatial keyword query has some limitations given below:

- False hit occurs i.e. direct the search to some objects even though they do not have all the required keywords.
- Loading of full text description is required.
- Need more space to store R-Tree.
- Not suitable for mobile user since the location changes continuously.

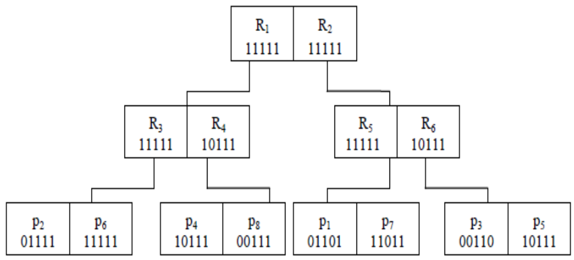

Figure 3.3 Generation of IR^2 tree

4. Proposed System

To overcome these limitations a new technique called Spatial Inverted (SI) index is introduced for spatial keyword queries. Figure 4.1 shows the system architecture.

A mobile user can give keyword as query to search the nearest restaurant, hospital, entertainment places etc. Keyword and user location is processed in the spatial inverted index. If the query given by the user is misspelled or with typo error, this type of fuzzy keyword is manage by n gram/2L Approximate inverted index. Construction of this indexing algorithm involves 4 steps:

Step 1: Extracting m-subsequence's from documents.
Step 2: Constructing the back end index.
Step 3: Extracting n-grams from m-subsequence's.
Step 4: Building the front end index.

First search the m-sub sequences by comparing n-gram of query string with n-grams of front end index. Then find the relevant document for the corresponding m-subsequences.SI index uses this subsequence and n-grams in compressed form.

SI index converts two dimensional co-ordinates of location into one dimensional Z-value. Then gap- keeping technique is applied to Z-value and ids of inverted list to compress the spatial objects. This Space Filling curves preserves the special proximity and creates the block using d-gap of Z-value and pseudo ids.Then blocks are added to leaf of R-Trees.

The index structure in proxy provides the Estimated Valid Region (EVR) of user .Two index structures are used namely EVR Tree and Grid index along with cache. Proxy searches the query location in any EVR of EVR tree. If it contains return the EVR region with spatial objects else it search the grid index.

If the result does not present in two indexes proxy requests kNN query to LBS and provide the result by constructing EVR for the new region. Finally query result and EVR is returned to the mobile client.

5. Conclusion

A spatial keyword query in general satisfies spatial and textual predicates. But it does not applicable for mobile environment.Because spatial information of mobile client is difficult to find. To overcome this problem Spatial Inverted (SI) index technique is introduced in proxy. In addition to this fuzzy keyword is also managed in the proposed system by using n-gram/2L Approximate indexing algorithm. It also overcomes the limitations of existing technique such as false hit and reduces the storage space by using compression technique. Blocks in R-Tree help us to pick the data much faster than searching from beginning. Furthermore, two index structures in proxy provide the Estimated Valid Region (EVR) and relevant query result to the mobile user. This paper extends the concept of spatial keyword query from mobile user to mobile spatial objects.

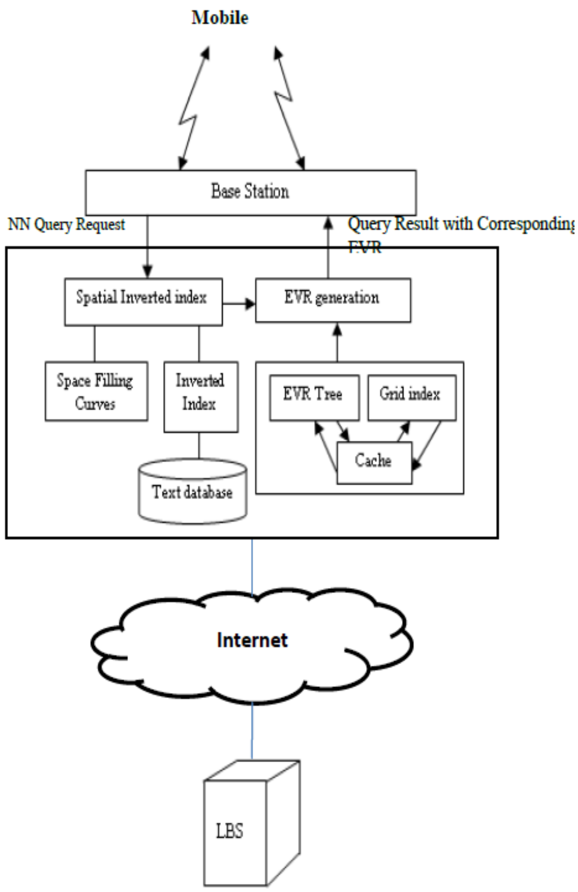

Figure 4.1 System Architecture

Acknowledgement

Sincerely we thank to lord for his blessings and guided for throughout the way. However, it would not been possible without his kind support and help of many individuals and organizations. We also express our gratitude to friends and parents for their manual support.

References

[1] Beckmann .N, Kriegel .H, Schneider .R, and Seeger.B, "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.

[2] Cao .X, Chen .L, Cong .G, Jensen .C. S, Qu .Q,

Skovsgaard .A, Wu .D, and Yiu .M. L,"SPATIAL KEYWORD QUERYING", In ER 16–29, 2012.

[3] Cao .X, Cong . G, and Jensen .C. S, "RETRIEVING TOP-K

PRESTIGE - BASED RELEVANT SPATIAL WEB OBJECT", PVLDB 3, 1, 373–384, 2010.

[4] Chen .Y.-Y, Suel .T, and Markowetz .A, "EFFICIENT

QUERY PROCESSING IN GEOGRAPHIC WEB SEARCH ENGINES", In SIGMOD Conference 277–288,2006.

[5] Cong .G, Jensen .C. S, and Wu .D, "EFFICIENT

RETRIEVAL OF THE TOP-K MOST RELEVANT SPATIAL WEB OBJECTS". PVLDB 2, 1, 337–348, 2009.

[6] Felipe .I. D, Hristidis .V, and Rishe . N, "KEYWORD SEARCH ON SPATIAL DATABASES",In ICDE 656–665, 2008.

[7] Guttman .A, "R-TREES: A DYNAMIC INDEX STRUCTURE FOR SPATIAL SEARCHING", In SIGMOD Conference 47–57, 1984.

[8] Hariharan . R, Hore . B, Li . C, and Mehrotra .S, " PROCESSING SPATIALKEYWORD (SK) QUERIES IN GEOGRAPHIC INFORMATION RETRIEVAL (GIR) SYSTEMS", In Proc of Scientific and Statistical Database Management (SSDBM), 2007.

[9] Lu .J, Lu .Y, and Cong .G, "REVERSE SPATIAL AND TEXTUAL K NEAREST NEIGHBOR SEARCH", In SIGMOD Conference 349–360, 2011.

[10] Papadias .D, Kalnis .P, Zhang .J, and Tao .Y,"EFFICIENT OLAP OPERATIONS IN SPATIAL DATA WAREHOUSES" In *SSTD*, pages 443–459.

[11] Song .Z, and Roussopoulos .N, "K-NEAREST NEIGHBOR SEARCH FOR MOVING QUERY POINT", In SSTD 79–96, 2001.

[12] Yufei Tao, Cheng Sheng, "FAST NEAREST NEIGHBOR SEARCH WITH KEYWORDS" IEEE Transactions on Knowledge and Data Engineering, VOL: PP NO: 99 YEAR 2013.

[13] Zhou .Y, Xie .X, Wang .C, Gong .Y, and Ma .W.-Y,"HYBRID INDEX STRUCTURES FOR LOCATION-BASED WEB SEARCH",In Proc of Conference on Information and Knowledge Management (CIKM), pages 155–162,2005.

[14] Zhang .D, Chee .Y.M, Mondal .A, Tung .A. K. H, and Kitsuregawa .M, " KEYWORD SEARCH IN SPATIAL DATABASES: TOWARDS SEARCHING BY DOCUMENT", In ICDE 688–699, 2009.

[15] Zobel .J, and Moffat .A, "INVERTED FILES FOR TEXT SEARCH ENGINES" ACM Computer Survey, 38(2), 2006.

[16] Zobel .J, and Moffat .A, "INVERTED FILES FOR TEXT SEARCH ENGINES" ACM Computer Survey, 38(2), 2006.