International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 06, June-2015

Keyword Augmented Nearest Neighbor Search

Anusha Rachel Mammen
P. G. Scholar,
Department of Computer Science and Engineering
Amal Jyothi College of Engineering , Kanjirapally,
Kottayam , Kerala, India

Elisabeth Thomas
Assistant Professor,
Department of Computer Science and Engineering
Amal Jyothi College of Engineering. Kanjirapally,
Kottayam, Kerala, India

***Abstract*— Keyword augmented nearest neighbor search operates on spatial database. Traditional nearest neighbor search focus on computing nearest neighbor using keywords . The proposed method investigate into computing nearest neighbor search using keywords by considering features as well as minimum interobjective distance. in a fast manner. It provides a framework for keyword search in spatial databases. It uses the concept of spatial inverted list to store location and keyword and R Tree for storing neighbours of a particular location. R Tree provides fast query performance and comes with algorithm that retrieve nearest neighbour with keywords.**

Keywords—Nearest neighbor search; features; spatial datbase

I. INTRODUCTION

Information retrieval is the process of retrieving information from a given set of documents. Information retrieval finds application in data mining, image processing etc. In the field of data mining, information retrieval can be performed in spatial data mining. Spatial data mining involve spatial data., a type of data involving spatial objects and its coordinates. Searching data is the most frequent activity among researchers. Keyword searching plays a significant role in spatial data mining. Keywords are terms that are associated with spatial data. For example, if users wants to go to an institution with facility 'basketball court', user can search for institution along with particular feature.

Keywords are preferred words that users search. Keywords helps to retrieve information faster. These are words that researchers look for when searching for information. Indexing provides fast keyword searching. Various approaches provide way for searching words. Searching problem in spatial data involve nearest neighbor search. Various nearest neighbor search algorithms has been proposed. Nearest neighbor search algorithm compute nearest point with respect to query point by considering minimum inter-objective distance.

Most relevant index used is inverted index , type of geo-textual index. Inverted index consist of coordinates involving location of spatial objects . Indexing of inverted index require scanning entire coordinate. This can lead to wastage of time . Bottleneck arise while scanning whole point. To avoid such problem , a variant of inverted index , Spatial Inverted List, is developed. Spatial Inverted List comes with algorithms to perform keyword augmented nearest neighbor search. Keyword related searches has sparked interest among researchers. Spatial keyword querying focus on querying information using keywords.

In most aspect, keywords are words in a sentence. To emulate in a better way the terms 'keywords' are modified in a such a way that keywords are not simply words but highlighted words that helps knowledge developers to retrieve information. Information comes in the form of web page, location based information or static documents. Location based information retrieval has gained interest among researchers as well as internet users. Several approaches has been proposed. This paper comes with the motivation of retrieving information based on keywords as well as considering minimum inter-distance. Inter-objective distance can be achieved suing nearest neighbor search algorithm.

Searching plays an important role in searching field. Keywords along with features are taken into consideration. This approach helps to improve speed performance of information retrieval and also save searching time. Today, knowledge plays an important role .This approach provides a way to retrieve knowledge regarding various join places or schools or hospitals. Features also aid in the information retrieval process. Querying spatial information using keywords is inspired by increasing trend of search engines. It is very important for knowledge developers to know information regarding location.

II. PROBLEM DEFI NITION

The goal of this paper is to compute nearest neighbor with respect to query point using keywords. In this context, nearest neighor query using keywords aims to retrieve information by considering both minimum inter-objective distance and related keywords. For example, consider five points p1,p2,p3,p4,p5 with keywords a,b,c,d,e. {p1,a},{p2,b},{p3,c},{p4,d},{p5,e}.
Query of the form food-a retrieve point p1 because it is the only point having keyword a.

III. RELATED WORK

Spatial keyword search is a general category that has sparked interest among researchers. Jao B. Rocha Junior [1] considered a kind of geo-textual index named Spatial Inverted Index. This method maps each keyword or term to a distinct aggregated tree or to a block that stores the spatial textual objects that contain term. The most frequent terms are stored in tree. The less frequent terms are stored in blocks in a file, one block per term. In our context, the concept of Spatial Inverted Index has been considered with slight variation, maps each term or keyword to a block. Dingming Wu [2] proposed a join algorithm named ITERATE to

compute join queries efficiently, generic algorithm applicable to spatial inverted index . This algorithm provides way to execute queries efficiently.

Glsi R. Hjaltson [3] aims to obtain data objects in their order of distance from a query point. Searching in a database on the basis of distance is called distance browsing. The method in [3] focus on distance browsing algorithms , plays a significant role in retrieving nearest neigbour. Marius [4] provides method to improve performance of retrieval of information. The method in [4] cluster the data points using the full distance across all dimensions. Clustering done recursively to the points in each region. Recursion is stopped when the number of points in a region is smaller than required clusters. The method in [5] helps to improve query processing. Given a number of keywords and one or more locations that a user is interested in , a location-based web search retrieves and ranks the most textually and spatially relevant information. Ali Khodaei [6] consider a form of index to store keywords. Space is partitioned into number of cells. Each cell is treated similar to a textual document.

Lisi Chen [7] surveys various indices to improve query performance. Various nearest neighbor search algorithms in [3], [8] provide ways to compute nearest neighbor. The method in [9] is referred to create R Tree from space filling curve. Space utilization is very high. X. Cao [10] provide ways to achieve spatial keyword querying functionality , useful and relevant to users.

IV. SPATIAL INVERTED LIST

Spatial inverted list is compressed version of inverted index. Inverted Index is an index to store keywords and coordinates. This list is a variant of inverted index to store required keywords associated with query point. It acts as a method to prune irrelevant keywords. Fast nearest neighbor search using keywords is performed using this list. In this context, retrieval of nearest neighbor involves two process: keyword searching and nearest neighbor searching. Both activities are combined to perform keyword augmented nearest neighbor search. Each point in spatial inverted list consist of multidimensional points. These points are converted to one dimensional point. For computing nearest neighbor search, data structure such as R Tree are used. There are methods to retrieve information using keywords. This paper design a method to retrieve information using keywords along with its features. Features are terms that describe a keyword. Keyword searching provides a way to extract meaningful information from a set of words. Spatial Inverted List helps to achieve this.

Keyword augmented nearest neighbor search augment keyword along with distance to retrieve nearest neighbor . Spatial inverted list comes under the category of textual index. Textual index is the most common index used for keyword query performance. Keyword augmented nearest neighbor search operates on spatial database. Spatial inverted list focus on investigating spatial database which consist of spatial objects. Since it involve coordinates of spatial objects, common elements can retrieved by performing intersection operation. Intersected list can be retrieved. Spatial keyword querying performed in spatial inverted list. It is actually a database or data storage for storing information regarding spatial objects. Spatial objects are spatial data type that provide spatial related information. Various applications like mobile apps, google maps facilitates to work on spatial related information. In this context, google map has been considered for portraying information

keywords	Inverted list
a	P1, P4
b	P1,P2,P7
c	P5,P6,P8
d	P2,P3,P6,P8
e	P4,P5,P6,P7
a, b	P1,P2,P4,P7
b, c	P1,P2,P5,P6,P7,P8

Fig. 1 Spatial Inverted List

Fig 1 is an example of spatial inverted list where P1,P2..P8 indicate points and a, c,…d represent keywords. First tuple of spatial inverted list indicate keyword ‘a’ can be found in points P1 and P4 respectively. Consider a query where user wants to retrieve points having keyword ‘b’ only, Points P1,P2 and P7 will be retrieved. This list gives an overview of available points along with associated keywords. To prune irrelevant keywords, we go for keyword search indexing. This list will be compressed to get smaller version of index in order to perform keyword augmented nearest neighbor search . Inverted list is simply points along with spatial related information. Each point consists of multidimensional points. These points are converted to single dimensional points to plot in R Tree, for fast query performance. R Tree is a spatial access method to perform spatial data mining. Access methods provide ways to access data. In this context, data refer to location. Importance given to location dependent search.

A. Keyword Search Indexing

The objective of keyword search indexing is to generate documents that contain one or more keywords. Keywords are highlighted words that help users for information retrieval. Searching make use of an index called inverted index to retrieve corresponding keywords Searching in inverted index .involves scanning all points in the inverted list. Intersected points are stored in the index to retrieve points. Large number of places that are highly relevant can be retrieved. Keywords are special features associated with keywords. Inverted index consist of points and respective keywords. Update operation can be done on inverted index. One can easily find required keyword by scanning each point in the index. Thus it acts as pruning method to retrieve relevant items. Inverted index provides way to get keywords in a fast manner.

Indexing provide fast retrieval of information using keywords in a fast manner. When users search for keywords, query keyword is compared with matching keywords in inverted index and corresponding keywords are retrieved. Keyword search is a pruning method to retrieve relevant keywords. Keyword refer to features related to users point of interest. Point of interest can be school, hospitals or restaurants.

Features involve facilities in a school, departments in hospital or menu of restaurant. This paper is motivated by increasing observation of various search engines. Search engine finds interesting enthusiasm among knowledge developers. Indexing using keywords on inverted first checks for matching documents, later retrieve required keywords. It is a form of text mining. In other words, extracting meaningful information from a data structure which can used for further analysis. Search operation takes log n time . Each time, it search n items. A keyword is defined as a word . This indexing finds application in single query processing. For further research processing, join query processing can be considered. Join query processing is simply joining multiple queries. Keyword relationship plays a significant role in searching keywords. Terms associated with keywords are generated. The presence of more coordinates in spatial inverted list motivated to create a tree structure to optimize speed of query performance.

B. R Tree

The document or keywords after pruning are stored in R Tree along with its distance for fast query performance. R Tree considers not only keywords but also minimum inter-objective distance. Building R Tree provides way to store accurate terms. The basic idea of R Tree is to group nearby objects having same keywords. The basic construction of R Tree comes with creation of node. Node consist of either query object or rectangle. Rectangle or minimum bounding rectangle is defined by four coordinates (x-min,y-min,x-max,y-max) . In this context, it refers to latitude and longitude of two location.

R Tree algorithm starts with initialization of properties to set the properties of R Tree. Maximum number of nodes and minimum number of nodes are sent as parameters in R Tree. Then add function called to add value of rectangle that is location of spatial object followed by calling of chooseNode function to determine node to place the rectangle depending on the area of rectangle. The value will be stored in node having minimum area. AdjustTree function called to adjust the position of tree to place new node. Its entries are added and stored in R Tree.

Searching point require accessing R Tree several times. Due to fast query performance of tree structure, searching does not require much time. R Tree store information in blocks individually. Information is accessed block by block. Each block consists of several points having particular keyword. R Tree comes with nearest neighbor search algorithm to perform nearest neighbor. R Tree are used for indexing multidimensional points. The minimum bounding rectangle of an object serves as the index of search. R Tree provides way to filter irrelevant information to get accurate outcome. Location with respect to particular location fetched and stored in R Tree along with spatial features. Building R Tree efficiently balances information retrieval.

R Tree with child leaves are build on same level. If more than one point have same keywords, they are grouped into one box which shelve all points with respective to query points, providing way to perform search in a fast manner. Performs in a better way if tree structure contains few entries. Each node contain points in list with distance. The grouping method reduce space complexity. Keywords relevant to query stored in R Tree stored. Time complexity may vary. Each node indexing require log m time where m denote m nodes. Root node must have maximum capacity than its siblings. It must be given maximum priority. Each node is a combination with distance . When number of keywords increases, it performs in a better manner. Conventional index types do not efficiently handle spatial related information. Data insertion in R Tree involving groping data based on nearest location.

C. Incremental Nearest Neighbor Search

The algorithm for fast nearest neighbor search is based on incremental nearest neighbor search, plays a significant role in information retrieval using keywords. Tradition al algorithm for nearest neighbor search problem is K nearest neighbor search algorithm whose number of neighbors to retrieve is fixed in advance. This algorithm when applied on R Tree resulted in slow retrieval of information. This motivated to focus on variant of nearest neighbor search algorithm. Incremental nearest neighbor search algorithm aims to retrieve N number of neighbors in a fast manner. The algorithm query points by computing distance between user location and query location. Value of N is not fixed in advance. Thus it retrives neighbors incrementally , one by one, as per user's requirement.

The objects based on distance are ranked in incremental nearest neighbor search algorithm. Priority queue is the data structure used in incremental nearest neighbor search algorithm. The idea is to obtain small result when number of neighbors is not known in advance. Nearest neighbor queries finds algorithm very interesting. Using this algorithm helps to retrieve data in a queue manner. The algorithm works in a top down manner. Elements are removed from queue and checked for matching information . The corresponding information with its distance is stored in queue with distance as priority. The cost of each operation in priority queue is O(log t) where t is the size of the priority queue. Incremental nearest neighbor search algorithm can be applied on nodes of R Tree.

Nodes of the tree are visited one by one. Queue is used to keep track of nodes visited. The key idea of priority queue is to store nodes as well as objects based on distance. Nearest neighbor search compute distance by comparing required distance with default distance. It works on any number of dimensions. Here data objects are uniformly distributed in space and mainly focus on two dimensional space. Priority queue size plays an important role nearest neighbor search. As size increases, more and more information can be stored. Theoretically, cost of priority queue operation increases when queue size grows larger. Its content stored in disk based structure. Incremental nearest neighbor search algorithm falls under the category of nearest neighbor search. It helps to perform search based on distance of objects in a fast manner by combining inverted index .

D. Experiment

The experiment was implemented on real dataset which consists of spatial objects and its latitude and longitude.

Google map dynamically loads required information. The dimensionality has always been considered 2 with each axis consisting of integers from 0 to 10. The value of integer can vary upto infinity. Vocabulary of real dataset consists of several keywords. For experimental purpose, we use only seven words with keywords of varying length (W1,W2….W9). Each word appears in the text document of several location. Consider nearest neighbor search. Query parameter involve length of different words. Several keyword associated with information is generated as input for fast nearest neighbor search with keywords. Keyword length upto 10 has been taken into consideration. Its execution time is measured for analysis. Three nearby location has been taken into consideration for keyword search. For various keywords , execution time varies.

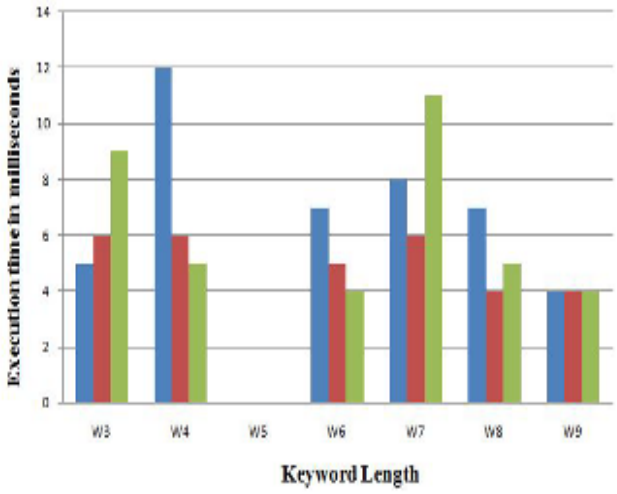

Fig. 2 Execution time versus keyword length

Fig 2 shows bar-chart indicating variation of execution time. The horizontal axis of bar-chart indicate length of various keywords. The vertical axis indicate execution time in milliseconds of various keywords. Fig 2 shows keyword length of 8 has constant execution time. Execution time depends on start time and end time of various keywords used for searching. It keeps on changing. W8 indicate keyword of length 8. When keyword of length 3 is used for searching, location 1 gives an execution time of 5 milliseconds, location 2 gives an execution time of 6 millisecond and location 3 gives an execution time of 9 milliseconds. Its execution time increases.

When keyword of length 4 is used for keyword search, location 1 gives an execution time of 12 milliseconds, location 2 gives an execution time of 6 milliseconds and location 3 gives an execution time of 4 milliseconds. Its execution time decrease in various location. When keyword of length 6 is used for search, location 1 gives execution time of 7milliseconds ,location 2 gives execution time of 5 milliseconds and location 3 gives execution time of 4 milliseconds. Its execution time decrease. When keyword length of 7 is used for keyword searching, location 1 gives an execution time of 8 milliseconds, location 2 gives execution time of 6 milliseconds and location 3 gives execution time of 11 milliseconds. Execution time either increase or decrease. When keyword of length 8 is used for searching, location 1 gives execution time of 7 milliseconds, location 2 gives execution time of 4 milliseconds and location 3 gives execution time of 5 milliseconds. From this analysis, it is found that execution time changes with varying keyword length and when length is very high, execution time remains constant.

V. CONCLUSION

Keyword searching plays a significant role in search engine. Searching mainly focus on keywords .This paper comes with method to retrieve information based on keyword searching along with its features by considering minimum inter-objective distance. New variant of inverted index is developed which has the ability to perform nearest neighbor search using keywords in a fast manner. This method concentrates only on single query. For future work, join processing of queries can be taken into consideration .Another extension is to compare various nearest neighbor search algorithms.

ACKNOWLEDGMENT

We thank computer science department of amal Jyothi college of Engineering for providing us with relevant data. This work was supported as part of innovative project.

REFERENCES

[1] Yufei Tao and A.C Kot, " Fast Nearest neighbour Search with Keywords", IEEE Trans. Inf. Forensics Security, vol 26, no 4, April 2014.

[2] DingMing Wu, Man Ling Yu, Gao Cong and Christian S Jensen, " Joint Top-K Spatial Keyword Query Processing", IEEE Trans. Knowledge and Data Engineering, vol 24, no 10, October 2012.

[3] Gilsi R Hjaltsen and Hansen Samet, " Distance Browsing in Spatial Databases", ACM Transactions on Database Systems, pp 265-318, June 1999

[4] Marius Miya and David G, " Scalable Nearest Neighbor Algorithms for High Dimensional Data", IEEE Trans. On Pattern Analysis and Machine Intelligence, vol 36, no 11, November 2014.

[5] Joao B. Rocha Junior, Orestis Ckorkhas, Simon Jonassen and Kjetil Norvag , " Efficient Processing of Top-K Spatial Keyword Querying", Springer 2011.

[6] Ali Khodai, Cyrus Shahabi, Chen Li, " SKIF-P: a point-based indexing and ranking of web documents for spatial keyword search", Springer 2011.

[7] Lisi Chen, Gao Cong, Christian S Jensen, Dingming W, "Spatial Keyword Query Processing : An Experimental Evaluation", VLDB Endowment, 2013.

[8] Ying Zhang, Wenj Zhang, Quianlu Lin, Xuemin Lin and Hung Tao, " Effectively indexing the Multidimensional Uncertain Object", IEEE Transaction, March 2014.

[9] Ibrahim Kamal and Christian Faloutsos, " Hilbert R Tree: An improved R Tree Using Fractals", VLDB Conference, 1994.

[10] Xin Cao, Lisi Chen, Christian S Length, " Spatial Keyword Querying", Springer 2012.