An Efficient Query Search using Cluster Based Approach in Data Mining

Potnuru Santosh[1], Mula.Sudhakar[2],

[1]*Final M.Tech Student,* [2]Asst.professor

[1,2]Dept of CSE, *Sarada Institute of Science, Technology and Management (SISTAM), Srikakulam, Andhra Pradesh*

Abstract:

To the best of our knowledge, there has not been any work on predicting or analyzing the difficulties of queries over databases. Researchers have proposed some methods to detect difficult queries over plain text document collections. However, these techniques are not applicable to our problem since they ignore the structure of the database. In particular, as mentioned earlier, a Keyword query interface must assign each query term to a schema element in the database. It must also distinguish the desired result type. We empirically show that direct adaptations of these techniques are ineffective for structured data. In this paper we are propose topic based cluster search algorithm for search of keyword in the database. By implementing this technique we can improve more efficiency of query oriented keyword search

Keywords: *Query performance, query effectiveness, keyword query, robustness, cosine similarity.*

I. INTRODUCTION

The classical query processing is easy to manipulating disadvantages of search of keywords in the documents. By implementing classical query processing we can provide efficient query related searching over the documents. In the keyword query interface is face the much attention in the last decade for due to the flexibility of searching and exploring data in the documents. By performing keyword query interface we can identify information related to query. By performing searching operation we can get the query related documents appears at the top of list[1][2]. Since any entity in the data set that contain keyword is potential answer, that key word related documents will be appear and perform the efficient query searching operation. For performing searching of queries some of the difficulties for answering queries are as follows. Unlike the queries in languages like sql, so that users do not specify desired schema elements for each query term. Second schema is users don't get specified output because users don't give enough information for performing searching operation.

Recently so many collaborative efforts to provide standard efforts and benchmark of keyword search methods are proposed. In this paper we are propose an efficient query searching operation for getting related documents over query. By implementing this concept we can get efficient search result and also reduce time complexity for performing searching operation. In the proposed system we are initial performing text preprocessing for reduce the tags from html documents. By performing text preprocessing we can get the only text format data. after getting text format data we can calculate local frequency and global frequency of each documents. By calculating local frequency and global frequency of each document we can easily identify of number of word containing in the document. After finding the we can build mvs matrix for finding the distance all documents. After that we can perform clusterization for grouping all related document into single group.

Cluster is a group of objects that belongs to the same class. In other words, similar objects are grouped in one cluster and dissimilar objects are grouped in another cluster. Clustering is the process of making a group of abstract objects into classes of similar objects. A cluster of data objects can be treated as one group. While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups. The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups. Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing. Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns. IN the field of biology, it can be used to derive plant and animal taxonomies, categorize genes with similar functionalities and gain insight into structures inherent to populations. Clustering also helps in identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location. Clustering also helps in classifying

documents on the web for information discovery. Clustering is also used in outlier detection applications such as detection of credit card fraud. As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster.

II. RELATED WORK

Researchers have been proposed different types of hard quires over unstructured documents [3][4][5]. Those methods are broadly categorized into two types i.e. pre retrieval and post retrieval. Pre-retrieval methods [6], [7] predict the difficulty of a query without computing its results. These methods usually use the statistical properties of the terms in the query to measure *specificity*, *ambiguity*, or *term-relatedness* of the query to predict its difficulty [8]. Examples of these statistical characteristics are average inverse document frequency of the query terms or the number of documents that contain at least one query term [6]. These methods generally assume that the more discriminative the query terms are, the easier the query will be. Empirical studies indicate that these methods have limited prediction accuracies [9], [10]. Post-retrieval methods utilize the results of a query to predict its difficulty and generally fall into one of the following categories.

A query predicted to perform poorly, may not necessarily be ambiguous but may just not be covered in the body to which it is submitted. Also, identifying difficult queries related to a particular topic can be a valuable asset for collection keepers who can determine what kind of documents are expected by users and missing in the collection. Another important factor for collection keepers is the find ability of documents, that is how easy is it for searchers to retrieve documents of interest Predictions are also important in the case of well-performing queries.[11] When deriving search results from different search engines and corpora, the predictions of the query with respect to each body can be used to select the best body or to merge the results across all corpora with weights according to the predicted query effectiveness score. Also, consider that the cost of searching can be decreased given a multiple partitioned body, as is common practice for very large corpora. If the documents are partitioned by, for instance, language or by topic, predicting to which partition to send the query saves time and bandwidth, as not all partitions need to be searched. Moreover, should the performance of a query appear to be sufficiently good, the query can be improved by some affirmative action such as automatic query expansion with pseudo-relevance feedback, In pseudo-relevance feedback it is assumed that the top K retrieved documents are relevant and so for a query with low effectiveness most or all of the top K documents would be irrelevant. Notably, expanding a poorly performing query leads to query drift and possibly to an even lower effectiveness while expanding queries with a reasonable performance and thus a number of relevant documents among the top K retrieved documents is more likely to lead to a gain in effectiveness. Another recently proposed application of prediction methods is to shorten long queries by filtering out predicted extraneous terms, in view of improving their effectiveness.

III. EXISTING SYSTEM

As per our recent literature reviews, there has not been any work on predicting or analyzing the difficulties of queries over databases. Researchers have proposed some methods to detect difficult queries over plain text document collections recently. But techniques are not applicable to our problem since they ignore the structure of the database. There are two categories of existing methods, pre-retrieval and post-retrieval for predicting the difficulties of query.

But below are limitations of this method:

- Pre-retrieval methods are having less prediction accuracies
- Post-retrieval methods are having better prediction accuracies but one requires domain knowledge about the data sets to extend idea of clarity score for queries over databases.
- Each topic in a database contains the entities that are about a similar subject.
- Some Post-retrieval methods success only depends on the amount and quality of their available training data

IV. PROPOSED SYSTEM

The main objective of proposed system is to perform the efficient query search and reduce the time complexity of in the searching process. In this paper we are proposed an efficient query searching process i.e. topic based cluster search algorithm. By implementing this algorithm we can get efficient search result and also reduce time for searching the query. Before performing the search the query we can take sample document and search query in that documents. The implementation procedure of topic based cluster algorithm is as follows.

Text Pre-processing:

In the text pre-processing we can get only text formatted data for searching query. Before performing search operations we can get all documents and reduce all tag in that document. After

getting each document text we can find out relative frequency (R_{freq}) of each document. Before finding relative frequency we also find local and global frequency of each word in the document. The local frequency (L_{freq})of each can be calculated by number of occurrence of each word in the document. After finding local frequency of each word in the document we can find out global frequency(G_{freq}) . Using both frequencies we can find out relative frequency of each document by using following formula.

$$R_{freq} = L_{freq} + G_{freq} / 2.0$$

After finding relative frequency we can calculate document weight of each document by using following formula.

N=size of each document

$L_{freq=}$ Local frequency of each word in the document

G_{freq} =Global Freqency of each document

$$Weight(W) = L_{freq} * Math.Log(N/G_{freq}) + 0.01$$

By using that formula we can calculate each document weight. After we can create MVS Matrix of each document to other documents.

Build MVS Matrix:

In the generation of MVS matrix we can calculate cosine similarity each document to other document. Based on MVS matrix we can perform the clusterization of documents. The cosine similarity of any two document can be find by using following equation.

d1= Total number of words in first document

d2= total number of words in second document

d_{prd}= d1*d2

$d1_{sqr}$ = d1*d1

$d2_{sqr}$ d2*d2

$$d_{sqrprd} = d1_{sqr} * d2_{sqr}$$

$$sim = d_{prd}/d_{sqrprd}$$

By using those formulas we find out each document cosine similarity and also we generate matrix formatted data. likewise we can calculate cosine similarity of each document to other document and arranged in the form matrix.

k means clustering algorithm for grouping related documents:

By calculating of MVS matrix we can perform the clusterization process. By performing clusterization process we can grouping all relating document into single group. Before performing clusterization we get all cosine similarity of each document to other document. Based on cosine similarity of each document we can perform clusterization process. The step of clusterization process is as follows.

1. Enter the number of cluster for performing clustering of document.
2. After that finding number of documents are available in the database.
3. Randomly choose the centroid of document based on number of clusters we want.
4. After finding centroid document we can get cosine similarity of each centroid document.
5. After that we can also get remaining document of cosine similarity.
6. Find out distance of each centroid to other document based on cosine similarity by using following formula

```
for(int i=0;i<docs.size();i++)
   {

    int minInd =0;
    double mindis=0;

    for(int j=0;j<k;j++)
    {
     Double dis =
cosSim(docs.get(i),getCentriod(clusturs[j]));

      if(j==0 || mindis>dis)
      {
       minInd=j;
       mindis=dis;
      }
     }

     clusturs[minInd].add(docs.get(i));

    }.
```

By using that code we can find out related documents in a group. After grouping all related document into group perfrom the searching operation in those groups and get only query matched document.

Topic based searching process:

In this module we perform the searching operation of query in the document. In this we can get each cluster document and convert into text format. After that we can search each word in that cluster and find out the word id existing in that group or not. So that if the word is existing in the cluster we display that document in the cluster. Likewise we can search all cluster documents and get only the query related cluster document. By implementing those concepts we can get more

effective search result and also time complexity for performing search operation.

V. CONCLUSIONS

In this paper we are proposed a novel problem for performing effective searching operation in documents. By implementing this concept we can improve more efficiency of searching operation and also reduce time complexity. In this paper we are proposed topic based cluster searching algorithm for finding related document of query search. In this algorithm we can find out each document cosine similarity and also find out distance of centroid document to other documents. After finding distance we can perform clusterization process by using k means clustering algorithm. After performing clustering process we can perform the searching process for query. By performing query search we can get all query related documents of clusters can be display. By implementing those concepts we can improve efficiency in the searching operation.

VI. REFERENCES

[1]. V. Hristidis, L. Gravano, and Y Papakonstantinou, "Efficient IRstyle keyword search over relational databases," in *Proc. 29th VLDB Conf.*, Berlin, Germany, 2003, pp. 850–861.

[2]. G. Bhalotia, A. Hulgeri, C. Nakhe, S. Chakrabarti, and S. Sudarshan, "Keyword searching and browsing in databases using BANKS," in *Proc. 18th ICDE*, San Jose, CA, USA, 2002, pp. 431–440.

[3]. S. C. Townsend, Y. Zhou, and B. Croft, "Predicting query performance," in *Proc. SIGIR '02*, Tampere, Finland, pp. 299–306.

[4]. Y. Zhou and B. Croft, "Ranking robustness: A novel framework to predict query performance," in *Proc. 15th ACM Int. CIKM*, Geneva, Switzerland, 2006, pp. 567–574.

[5]. Y. Zhou and W. B. Croft, "Query performance prediction in web search environments," in *Proc. 30th Annu. Int. ACM SIGIR*, New York, NY, USA, 2007, pp. 543–550.

[6]. B. He and I. Ounis, "Query performance prediction," *Inf. Syst.*, vol. 31, no. 7, pp. 585–594, Nov. 2006.

[7]. Y. Zhao, F. Scholer, and Y. Tsegay, "Effective pre-retrieval query performance prediction using similarity and variability evidence," in *Proc. 30th ECIR*, Berlin, Germany, 2008, pp. 52–64.

[8] C. Hauff, L. Azzopardi, and D. Hiemstra, "The combination and evaluation of query performance prediction methods," in *Proc. 31st ECIR*, Toulouse, France, 2009, pp. 301–312.

[9]. S. C. Townsend, Y. Zhou, and B. Croft, "Predicting query performance," in *Proc. SIGIR '02*, Tampere, Finland, pp. 299–306.

[10]. C. Hauff, V. Murdock, and R. Baeza-Yates, "Improved query difficulty prediction for the Web," in *Proc. 17th CIKM*, Napa Valley, CA, USA, 2008, pp. 439–448.

[11]. Claudia Hauff,"Predicting The Effectiveness Of queries And Retrieval Systems", January 29, 2010

[12]. A. Shtok, O. Kurland, and D. Carmel, "Predicting query performance by query-drift estimation," in *Proc. 2nd ICTIR*, Heidelberg, Germany, 2009, pp. 305–312.

BIOGRAPHIES:

Potnuru Santosh is a Student in M.Tech (CSE) in Sarada Institute of science Technology and management, Srikakulam. He received her B.Tech (CSE) from Sivani Institute of technology, chilakapalem, Srikakulam. His interesting areas are Data ware housing and network security.

Mula.Sudhakar is working as a Asst.professor in Sarada Institute of Science, Technology And Management, Srikakulam, Andhra Pradesh. He received his M.Tech (SE) from Sarada Institute of Science, Technology And Management, Srikakulam. JNTU Kakinada Andhra Pradesh. His research areas include Cloud Computing,Dataminig,Network Security.