Volume 6, Issue 4, April 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Efficient Estimation of Complex Queries

Prof. Anand Bhosle
Department of Information Technology, International Institute of Information Technology,
Pune, Maharashtra, India

Abstract- complex Keyword query on databases give easy access to data, but undergo poor ranking quality, i.e., low precision and/or recall. It would be very easy if we identify queries that are likely to have low ranking quality.so we may suggest some alternative queries for such complex queries so the the desired results may appear at the top. aim of this paper is to Estimate characteristics of complex keyword queries and propose a to count the degree of hardness for a complex keyword query over a database, allowing for both the design and the content of the database and the results of query. There are query difficulty pre-diction model but results indicate that even with structured data, finding the desired answers to keyword queries is still a hard task. Further, paper present a new approach is nothing but extended outline in which before calculating Structured Robustness score we are applying the K-means clustering to divide input dataset into number of clusters those having legitimate is. As well, we will use Semantic and Syntactic structure of keyword Query such as syntactical features, and semantic features for smart prediction of keyword queries over database which will minimize Time required for predicting the complex keywords over large databases and process becomes Efficient and accurate.

Keywords- KQI: Keyword Query Interfaces, SR: Structured Robustness, QAO-Approx: Query Specific Attribute Approximation, SGS-Approx: Static Global Stats Approximation, CR: Clarity Score, URM: Unstructured Robustness Method, WIG: Weighted Information Gain, NQC: Normalized Query commitment

I. INTRODUCTION

Keyword queries for databases are getting much popularity in the last decade due to ease of use in searching and exploring the data. [2], [4] keyword queries characteristically have many possible answers. Keyword Query Interfaces must recognize the information needs behind queries and rank the answers so that the desired answers appear at the top of result list. Databases contain entity and attributes and values. Some of the problems of answering a query are likely to have users do not specify the preferred schema for each query term. For e.g. keyword God Father on the movie database does not state that user is interested in title or Distributor Company. So, a KQI must find the desired attributes associated with each term in the query and users do not give enough information about their desired entities. For example; keyword may return movies or actors or producers. Recently, there have been joint efforts are taken for giving standard benchmarks and evaluation platforms for keyword search methods over databases. One effort is the data-centric track of INEX Work-shop where KQIs are evaluated over the well-known IMDB data set which contains structured information about movies and people. [2] One more effort is the series of Semantic Search Challenges at Semantic Search Workshop, where the data set is the Billion Triple Challenge.[1] It is Extracted from Wikipedia.

The queries are used from Yahoo! Keyword query log. Users have provided relevance judgments for both benchmarks. These results indicate that even with structured data, finding the preferred answers to keyword queries is still a hard task. Ranking quality of the methods used in both workshops, observed that they performing very poorly on a subset of queries. For example, consider the query ancient Rome era over the IMDB data set. Users would like to see information about movies that talk about ancient Rome. For this query, the XML search methods which we implemented return rankings of considerably lower quality than their average ranking quality over all queries. Therefore, some queries are more difficult than others. Furthermore, no matter which ranking method is used, we cannot deliver a reasonable ranking for these queries.

It is important for a Keyword Query Interface to recognize such queries and warn the user or employ alternative techniques like query reformulation or query suggestions. It may also use techniques such as query results diversification. There has not been any work on Estimating the difficulties of queries over databases. Researchers have proposed some methods to predict difficult queries over plain text document. But, these techniques are not applicable to our problem statement since they ignore the schema of the database. In particular, as mentioned earlier, a Keyword Query Interfaces must assign each query term to a schema element(s) in the database In this paper, analyzing the characteristics of difficult queries over databases and propose a novel method to detect or identify such queries that are likely to improve Estimation of Difficult Keyword Queries.

II. LITERATURE SURVEY

Some Research studies have presented different methods to predict hard queries over unstructured documents or plain text collection. It can classify into two groups: pre-retrieval and post-retrieval methods.Pre-retrieval methods 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. Examples are average inverse document frequency of the query terms or the number of documents that contain at least one query term. These methods normally assume that the more discriminative the query terms are, the easier the query will be. Post retrieval methods make use of the results of a query to forecast its difficulty and generally fall into one of the following categories. Clarity-score-based: It is based on the concept of clarity score assume that users are concerned in a very few topics. Thus, sufficiently noticeable from other

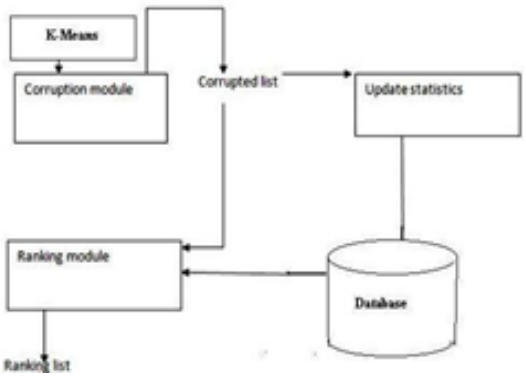
documents in the collection. It is efficient than pre-retrieval based methods for text documents. Some systems compute the distinguish ability of the queries results from the documents in the collection by comparing the probability distribution of terms in the results with the probability distribution of terms in the whole collection. If these probability distributions are comparatively similar, the results contain information about almost as many topics as the whole collection, thus, the query is considered difficult. Several successors propose methods to improve the efficiency and effectiveness of clarity score. Though, one requires domain knowledge about the data sets to extend idea of clarity score. Each topic in a database contains the entities that are about a similar subject. Ranking-score-based: The ranking score of a document returned by the system for an input query may estimate the similarity of the query and the document. Some recent methods measure the difficulty of a query based on the score distribution of its results. Robustness-based: Another group of post-retrieval methods argue that the results of an easy query are relatively stable against the perturbation of queries [5], documents [6] or ranking algorithms.

Our proposed query difficulty prediction model falls in this category. Some methods use machine learning techniques to study the properties of difficult queries and predict their hardness. They have similar limitations as the other approaches when applied to structured data. Moreover, their success depends on the amount and quality of their available training data. Enough and high quality training data is not available for many databases. Some researchers propose frameworks that theoretically explain existing Predictors and combine them to achieve higher prediction accuracy. It present novel learning methods for estimating the quality of results returned by a search engine in response to a query.

Estimation is based on the agreement between the top results of the full query and the top results of its sub-queries. It express the usefulness of quality estimation for several applications, among them improvement of retrieval, detecting queries for which no relevant content exists in the document collection, and distributed information retrieval. It describes two methods for learning an estimator of query difficulty. The learned estimator predicts the expected precision of the query by analyzing the overlap between the results of the full query and the results of its sub-queries. [8] Limitations: 1)The quality of query prediction strongly depends on the query length. 2)The restricted amount of training data. The query-performance prediction task is estimating the effectiveness of a search per-formed in response to a query in lack of relevance judgments. Post-retrieval predictors analyze the result list of top-retrieved documents.

Framework is based on using a pseudo effective and/or ineffective ranking as reference comparisons to the ranking at hand, the quality of which it want to predict. [4] limitations: It outperforms on large dataset. Keyword queries over structured databases are disreputably ambiguous. No single understanding of a keyword query can satisfy all users, and multiple interpretations may yield overlapping results. It proposes a scheme to balance the relevance and novelty of keyword search results over structured databases.

Firstly, it presents a probabilistic model which effectively ranks the possible interpretations of a keyword query over structured data. [7] Forecast query difficulty based on linguistic fea-tures, using TreeTagger and other natural language processing tools. Topic features include morphological features (number of words, average of proper nouns, and average number of numeic values), syntactical features (average conjunctions and prepositions, average syntactic depth and link span) or semantic features (average polysemy value). They found that the only positively correlated feature is the number of proper nouns. It use some morphological or syntactic features in our topic prediction algorithm.[4]



System Architecture

III. PROPOSED SYSTEM

A. Problem Definition:

Some of the researchers have designed systems for Predicting or analyzing the difficulties of queries over databases. There are different ways presented for identifying difficult queries over plain text document collections recently. However such methods are not applicable to our problem statement since they do not consider 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: 1) Pre-retrieval methods are having less prediction accuracies. 2) Post-

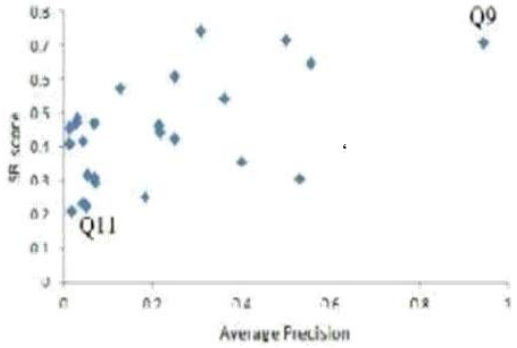
retrieval methods are having better pre-diction accuracies but it requires domain knowledge about the data Sets to extend idea of clarity score for queries over databases. 3) Each topic in a database contains the entities that are about a similar subject. 4) Some Post-retrieval methods success only depends on the amount and quality of their available data. Above problems were mitigated by recently presented efficient method for prediction of difficult keywords over databases. This method efficiently solving the problem of predicting the effectiveness of keyword queries over DBs as compared to existing methods with highest level of accuracy. This method takes less time and having relatively low errors for predicting difficulty of queries. This method suffered from limitations like not evaluated with large datasets As well as string approximation is not taken under considerations.

B. Proposed Method:

In this paper our main focus is to improve process for difficult keyword Estimation by Eliminating the of Scalability, dataset flexibility, and string approxi-mation limitations in existing system. This devised approach is nothing but extended framework in which before going to calculate Structured Robustness score we are applying the K-means clustering to divide input dataset into number of clusters those having legitimate information.Because of this, Time required for Estimating the difficult keywords over large dataset is minimized and process becomes robust and accurate. In addition to this, spatial approximate string Query is presented. We are going to use edit distance as the similarity Measurement for the string predicate and focus on the range queries as the spatial predicate. As well we will be use semantic and syntactic features for effective prediction of difficult keyword queries over database.

C. Proposed Method:

In this paper our main aim is to present new improve method for difficult keyword prediction by overcoming the limitations of Scalability, dataset flexibility, and string approxi-mation. This New approach is nothing but extended framework in which before going to calculate SR score we are applying the K-means clustering to divide input dataset into number of clusters those having legitimate informations. As well .Due to this, Time required for predicting the difficult keywords over large dataset is minimized and process becomes robust and accurate. In addition to this, spatial approximate string.



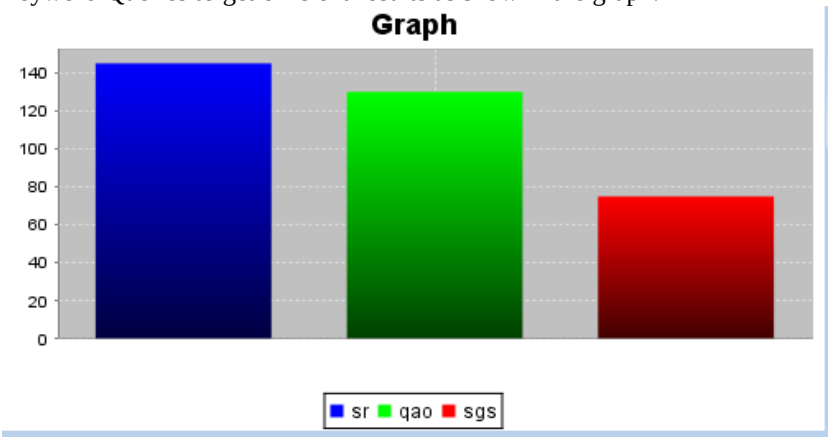
Mathematical Model:

Let V be the number of distinct terms in database
DB. Each attribute Value Aa A; 1 <=a<=
kAj can be modeled using a V-dimensional multivariate
distributionXa = (Xa; 1::::Xa;vv); whereXa;ja Xa
is a random variable that represents the frequency of term
wjinAa:
The probability mass function of Xais :
fxa(~xa) = Pr(Xa;1 = Xa;1 ::Xa;v = Xa;v):::::::::(1)
where ~xa = xa;1 :::xa;v and xa;j are non-negative integers.
fXA(~x) = fXA(~x1:::::::X~jAj)=Pr(X1=~x1::::::::XjAj=xjA~j
Structured Robustness calculation: It ranges between 1 and 1, where 1, 1, and 0 indicate perfect positive correlation, perfect negative correlation, and almost no correlation, respec-tively.
SR(Q,g,DB,XDB) = EfSim(L(Q; g; DB); L(Q; g; XDB))g
where M(jAjXV)and Sim denotes the Spearman rankcorrelation between the ranked answer lists.
1) Semantic and Syntactic Structure of Query can be efficiently used to Estimate Difficult Keyword Queries
2) Linguistic features: We will compute correlation scoresof linguistic features and the average recall and precision scores for the difficult keyword queries. Correlation is a simplestatistical measure, ranging from -1 to +1.
3) Efficient Computation of Structured Robustness Score: Structured Robust-ness Algorithm: Which computes the exact SR score based on the top K result entities.
Input Query Q,Top-K result list L of Q by ranking function ,Metadata M,Inverted indexes I,Number of corruption itera-tion N.
output: SR score for Q
1: SR 0; C Fg
2 : FOR i = 1 N DO

3 : I0 I; M0 M; L0 L;
4 : FOR each result R in L DO
5 : FOR each attribute value A in R DO
6 : A0 A
7 : FOR each keywords w in Q DO
8 : Compute of w in A0
model the noise in attribute value, attribute, and entity set levels. 9 : IF of w varies in A0and A THEN
10 : update A0; M0 and entry of w in I0;
11 : Add A0 to R0;
12 : Add R0 to L0;
13 : Rank L0 using g,which returns L,based onI0; M0;
14 : SR+ = Sim(L; L0);
15 : RETURN Sr SR N;

IV. RESULTS

Data sets: The INEX data set is from the INEX 2010 Data Centric Track. The INEX data set contains two entity sets: movie and person. Each entity in the movie entity set represents one movie with attributes like title, keywords, and year. The person entity set contains attributes like name, nickname, and biography ,we have considered the Semantic and Syntactic structure of Keyword Queries to get efficient results as show in the graph.



V. CONCLUSION

In this paper, It introduces the novel problem of predicting the effectiveness of keyword queries over databases. It shows that the current prediction methods for queries over unstruc-tured data sources cannot be effectively used to solve this problem. It present new improve method for difficult keyword prediction by overcoming the limitations of Scalability, dataset flexibility, and string approximation. This New approach is nothing but extended framework in which before going to calculate SR score we are applying the K-means clustering to divide input dataset into number of clusters those having legitimate informations. As well it will measure the degree of the difficulty of a keyword query over a database, using the ranking robustness principle. Additionally, we will be use linguistic features Such as morphological features, syntactical features, and semantic features for effective prediction of diffi-cult keyword queries over database. The algorithms predict the difficulty of a query with relatively low errors and negligible time overheads

REFERENCES

[1] Shiwen Cheng, Arash Termehchy, and Vagelis Hristidis "Efficient Pre-diction of Difficult Keyword Queries over Databases," IEEE TRANS-ACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 6, JUNE 2014
[2] Hristidis, L. Gravano, and Y. Papakonstantinou."Efficient IR style key-word search over relational databases," in Proc. 29th VLDB Conf., Berlin, Germany, 2003, pp. 850861.
[3] O. Kurland, A. Shtok, S. Hummel, F. Raiber, D. Carmel, and O. Rom,""Back to the roots: A probabilistic framework for queryperfor-mance prediction,,"" in Proc. 21st Int. CIKM, Maui, HI, USA,2012, pp. 823832..
[4] S. C. Townsend, Y. Zhou, and B. Croft,""Predicting query perfor-mance,""in Proc. SIGIR 02, Tampere, Finland, pp. 299306.
[5] J. Han, M. Kamber, and J. Pei,"Data Mining: Concepts and Tech- niques,"San Francisco, CA: Morgan Kaufmann, 2011.
[6] S. Cheng, A. Termehchy, and V. Hristidis,"Predicting the effectiveness of keyword queries on databases,"in Proc. 21st ACM Int. CIKM, Maui, HI, 2012, pp. 1213-1222.
[7] Y. Zhou and B. Croft,"Ranking robustness: A novel framework to predict query performance,"in Proc. 15th ACM Int. CIKM, Geneva, Switzerland, 2006, pp. 567574.
[8] E. Demidova, P. Fankhauser, X. Zhou, and W. Nejdl,"DivQ: Diversifi-cation for keyword search over structured databases,"in Proc. SIGIR 10, Geneva, Switzerland, pp. 331338.

