Optimized Prediction of Hard Keyword Queries Over Databases

Mr. Sameer P Mamadapure
Department of Computer Engineering,
JSPM's JSCOE Pune, India
Email:sam16.mamadapure@gmail.com

Prof P D Lambhate
Department of Computer Engineering
JSPM's JSCOE Pune, India
Email:lambhatepoonam9@gmail.com

Abstract— Keyword Query Interface on databases gives easy access to data, but undergo from low ranking quality, i.e., low precision and/or recall. It would be constructive to recognize queries that are likely to have low ranking quality to improve the user satisfaction. For example, the system may suggest to the user alternative queries for such difficult queries. Goal of this paper is to predict the characteristics of hard queries and propose a novel framework to measure the degree of difficulty for a keyword query over a database, allowing for both the structure and the content of the database and the results of query. There are query difficulty prediction model but results indicate that even with structured data, finding the desired answers to keyword queries is still a hard task.

Further, we will use linguistic features Such as morphological features, syntactical features, and semantic features for effective prediction of difficult keyword queries over database. Due to this, Time required for predicting the difficult keywords over large dataset is minimized and process becomes robust and accurate.

Keywords—*Query performance, query effectiveness, keyword query, robustness,Prediction,Database*

*****

I. INTRODUCTION

KEYWORD query interfaces (KQIs) for databases paying much attention in the last decade due to ease of use in searching and exploring the data. [2], [4] keyword queries characteristically have many possible answers. KQIs must recognize the information needs behind keyword queries and rank the answers so that the desired answers appear at the top of the 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 element(s) for each query term. For e.g. keyword Godfather 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 Workshop where KQIs are evaluated over the well-known IMDB data set which contains structured information about movies and people. [9] One more effort is the series of Semantic Search Challenges (Research) at Semantic Search Workshop, where the data set is the Billion Triple Challenge.[10] 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 KQI 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 predicting or analyzing the difficulties of queries over databases. Researchers have proposed some methods to detect difficult queries over plain text document collections. But, these techniques are not applicable to our problem since they ignore the schema of the database. In particular, as mentioned earlier, a KQI 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 user satisfaction.

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 generally 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 relatively similar, the query 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. However, 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.

Ranking-score-based: The ranking score of a document returned by the retrieval systems 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 [3], documents [11] 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.
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. [12] Forecast query difficulty based on linguistic features, using TreeTagger and other natural language processing tools. Topic features include morphological features (number of words, average of proper nouns, and average number of numeral 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.[13]

III. LITERATURE REVIEW:

References	Techniques used	Limitations
Learning to Estimate Query Difficulty - Including Applications to Missing Content Detection and Distributed Information	Machine learning	query length, restricted amount of training data
Back to the Roots: A Probabilistic Framework for Query-Performance Prediction	Post-retrieval (Clarity)	depends mainly on the domain knowledge and understanding user's preferences.
A Unified Framework for Post-Retrieval Query-Performance Prediction	Post-retrieval	It outperforms on large dataset

IV. PROPOSED SYSTEM

A) Problem Definition:

From our study few research studies presented on predicting or analyzing the difficulties of queries over databases. There are different methods presented for identifying difficult queries over plain text document collections recently. However such methods 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:
1) Pre-retrieval methods are having less prediction accuracies.
2) Post-retrieval methods are having better prediction 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 aim is to present new improve method for difficult keyword prediction by overcoming the limitations of Scalability, dataset flexibility, and string approximation. 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 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. We will be use linguistic features Such as morphological features, syntactical features, semantic features for effective prediction of difficult keyword queries over database.

C) Mathematical Model:

Structured Robustness:-
Let V be the number of distinct terms in database DB. Each attribute value $Aa \in$ A, $1 \le a \le |A|$, can be modeled using a V-dimensional multivariate distribution $X_a = (X_a,1, \ldots, X_a,V)$, where $X_a,j \in X_A$ is a random variable that represents the frequency of term wj in Aa. The probability mass function of Xa is:

$$F_{Xa}(\overrightarrow{X_a}) = Pr(X_a,1 = X_a,1, \ldots, X_a,V = X_a,V),$$

where $\overrightarrow{X_a} = xa,1, \ldots, X_a,V$ and X_a,j , $\overrightarrow{X_a}$ are non-negative integers.

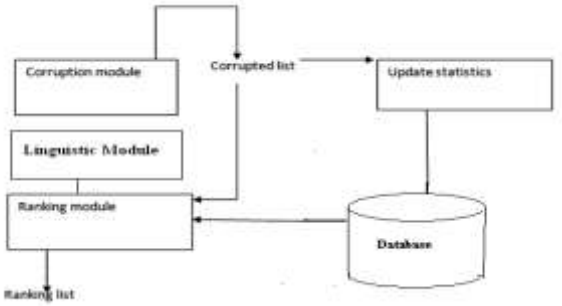
$$fX_A(\vec{X}) = f\overrightarrow{X_A}(\overrightarrow{X_1}, \ldots, \overrightarrow{X_{|A|}})$$
$$= Pr(\overrightarrow{X_1} = \overrightarrow{X_1}, \ldots, X_{|A|} = \overrightarrow{X_{|A|}}),$$

where $\overrightarrow{X_a} \in \vec{X}$ are vectors of size V that contain non-negative integers. The domain of $\vec{X}$ is all $|A| \times V$ matrices that contain non-negative integers, i.e. $M(|A| \times V)$.

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, respectively.

$SR(Q, g, DB, XDB)$
$= E\{Sim(L(Q, g, DB), L(Q, g, XDB))\}$
$= \sum_x Sim(L(Q, g, DB), L(Q, g, \_x)) fXDB(\_x),$
Where X , M $(|A| \times V)$ and Sim denotes the Spearman rank correlation between the ranked answer lists.

Noise Generation in Databases:-



we have:
$fX_A(\vec{X}) = \prod_{X_A \in \overrightarrow{X_a}} fX_A(\overrightarrow{X_a})$
and
$fX_A(\vec{X}) = \prod_{X_{A,j} \in \overrightarrow{X_a}} fX_{A,j}(X_{A,j})$

where $X_J \in \overrightarrow{X_J}$ depicts the number of times W_J appears in a noisy version of attribute value Ai and $fX_{i,J}(X_J)$ computes the probability of term w_J to appear in A_J X_J times.

The noise generation models attribute value A_i whose attribute is T_t and entity set is S_s

$$\hat{f}X_{a,j}(X_{a,j}) = \gamma A fX_{a,j}(X_{a,j}) + \gamma r f\gamma_{i,j}(X_{t,j}) + \gamma s f z_{s,j}(x_{s,j})$$

where $0 \le \gamma A, \gamma T, \gamma S \le 1$ and $\gamma A + \gamma T + \gamma S = 1$. $fX_{a,j}$, $fY_{t,j}$, and $fY_{s,t}$ model the noise in attribute value, attribute, and entity set levels, respectively. Parameters γA, γT and γS have the same values for all terms w $Q \cap V$ and are set empirically.

Since each attribute value A_a is a small document, we model $fX_{i,j}$ as a Poisson distribution:

$$fX_{a,j}(X_{a,j}) = \frac{e^{-\lambda_{a,j}} \lambda_{a,j}^{x_{a,j}}}{x_{a,j}}$$

Similarly, we model each attribute T_t, $1 \le t \le |T|$, as a bag of words and use Poisson distribution to model the noise generation in the attribute level:

$$f\gamma_{t,j}(X_{t,j}) = \frac{e^{-\lambda_{t,j}} \lambda_{t,j}^{x_{t,j}}}{x_{t,j}}$$

Using similar assumptions, we model the changes in the frequencies of the terms in entity set S_s, $1 \le s \le |S|$, using Poisson distribution:

 ISSN: 2321-8169
4792 - 4796

$$fZ_{s,j}(X_{s,j}) = \frac{e^{-\lambda_{s,j}} \lambda_{s,j}^{x_{s,j}}}{x_{s,j}}$$

D) Implementation Methodology:

Proposed algorithm:

I) Efficient Computation of SR Score

Structured Robustness 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 g, Metadata M, Inverted indexes I, Number of corruption iteration N.

1: SR ← 0; C ← {}
2: FOR i =1 ← N DO
3: I' ← I; M' ← M; L' ← L;
4: FOR each result R in L DO
5: FOR each attribute value A in R DO
6: A' ← A
7: FOR each keywords w in Q DO
8: Compute of w in A'
9: IF of w varies in A'and A THEN
10: update A'; M' and entry of w in I';
11: Add A' to R';
12: Add R' to L';
13: Rank L' using g, which returns L, based onI'; M';
14: SR+ = Sim (L; L');
15: RETURN Sr ← SR N;

II) Approximation Algorithms:

Approximation algorithms is used to improve the efficiency of SR Algorithm

a) Query-specific Attribute values Only Approximation (QAO-Approx):

Corrupts only the attribute values that match at least one query term.

b) Static Global Stats Approximation (SGS-Approx)

Corrupt only the top-K result entities

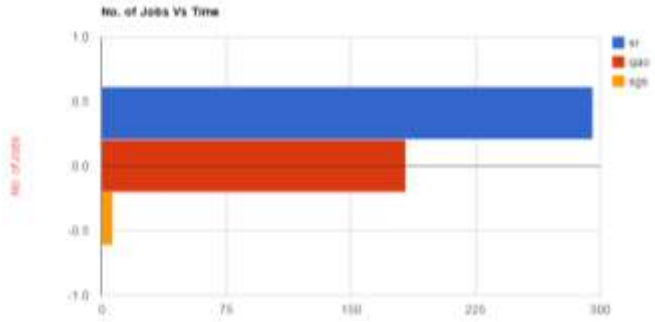
III) Ranking Algorithm: PRMS (Probabilistic Retrieval Model for Semi structured Data)

Incorporating mapping probabilities, improves retrieval effectiveness significantly over Semi structured Data

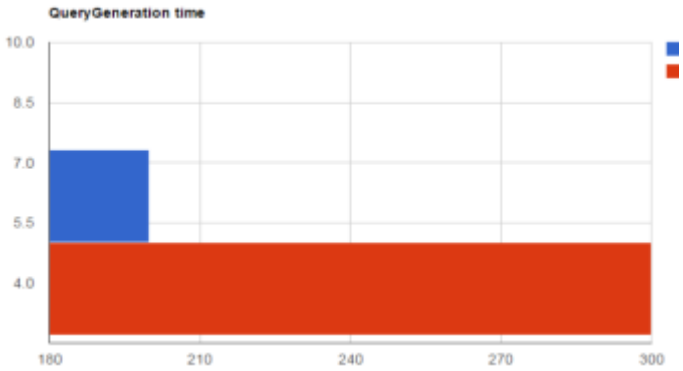
IV) We will compute correlation scores between linguistic features and the average recall and precision scores for the difficult keyword queries. Correlation is a simple statistical measure, ranging from +1 to -1.

IV .Experimental Analysis and Result:

Data Set of movie database is used for implementation. Data set is gathered through (http://www.inex.otago.ac.nz/tracks/strong/strong.asp)



Comparison Graph



Query generation Time in miliseconds

IV .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 unstructured 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 we are going apply linguistic features by this query complexity can be analyze. As well it will measure the degree of the difficulty of a keyword query over a database, using the ranking robustness principle. 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 Prediction of Difficult Keyword Queries over Databases," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 6, JUNE 2014

[2] Hristidis, L. Gravano, and Y. Papakonstantinou.“Efficient IR style keyword search over relational databases,” in Proc. 29th VLDB Conf., Berlin, Germany, 2003, pp. 850861.

[3] E. Yom-Tov, S. Fine, D. Carmel, and A. Darlow,“Learning to estimate query difficulty:Including applications to missing content detection and distributed information retrieval,” in Proc.28th Annu. Int. ACM SIGIR Conf. Research Development Information Retrieval, Salvador, Brazil, 2005, pp. 512519.

[4] O. Kurland, A. Shtok, S. Hummel, F. Raiber, D. Carmel, and O. Rom,“”Back to the roots: A probabilistic framework for query performance prediction,,”” in Proc. 21st Int. CIKM, Maui, HI, USA,2012, pp. 823832..

[5] O. Kurland, A. Shtok, D. Carmel, and S. Hummel,“”A Unified framework for postretrieval query-performance prediction,””in Proc. 3rd Int. ICTIR, Bertinoro, Italy,2011, pp. 1526.

[6] S. C. Townsend, Y. Zhou, and B. Croft,“”Predicting query performance,””in Proc.SIGIR 02, Tampere, Finland, pp. 299306.

[7] J. Han, M. Kamber, and J. Pei,“Data Mining: Concepts and Techniques,”San Francisco,CA: Morgan Kaufmann, 2011.

[8] 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.

[9] Overview of the INEX 2011 Data-Centric Track Qiuyue Wang1,2, Georgina Ramírez3, Maarten Marx4, Martin Theobald5, Jaap Kamps6

[10] T. Tran, P. Mika, H. Wang, and M. Grobelnik, “Semsearch ´S10,”in *Proc. 3rd Int. WWW Conf.*, Raleigh, NC, USA, 2010

[11] 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.

[12] E. Demidova, P. Fankhauser, X. Zhou, and W. Nejdl, “DivQ: Diversification for keyword search over structured databases,” in *Proc. SIGIR’ 10*, Geneva, Switzerland, pp. 331–338

[13] Josiane Mothe , Ludovic Tanguy “Linguistic features to predict query difficulty - a case study on previous TREC campaigns”