International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 05, May-2015

An Empirical Study of Effective and Versatile Keyword Query Search

Tejashree R. Shinde
Computer Engineering
JSPM's BSIOT, Wagholi
Pune, India

Prof. Sanchika A. Bajpai
Computer Engineering
JSPM's BSIOT, Wagholi
Pune, India

***Abstract*— In today's world, the huge amount of information is maintained and stored on World Wide Web. In day to day life every person need some information which can be extracted from web through the various search engine. It is the simple and easiest way to gain knowledge of any unknown field which user eager to know about. User expects to get relevant information according to its query. The information should be relevant as well as valid. To fulfill the user requirements number of techniques are used to provide the best and expected results. In this information searching is done with the help of keyword which is known as query keyword and searching is called as keyword searching.**

A huge amount of research work focusing on the keyword searching, retrieval and query processing has been done in the relational database. The overall work of the respective field is in scattered and diverse form which needs to collect and organize it in well manner so that it can be helpful for further research. In this paper, a survey of work on keyword querying in databases is presented. Relating to the explained context, this paper gives a brief description of various keyword searching, retrieval of the relevant keywords and query processing techniques with their limitations.

Keywords*— *Keyword Search, Relational Database, Schema-Less, Schema Based, XML, Heterogeneous Data, Linked Data, Information Retrieval(IR), Keyword Query Routing.

I. INTRODUCTION

Querying using keyword is simply the most popular form of querying today. Keyword searching is mostly used to search related documents on the web. Querying of databases is currently based on complex query languages which are inappropriate for the casual user, since they are complex and difficult to understand. To the classic SQL in querying relational databases with large, often unknown schema and instances, keyword queries offer an alternative. The challenge in answering such queries is to discover their semantics, construct the SQL queries and explore to retrieve the expected tuples. The discovered structure is the semantic interpretation of keyword query. Existing approaches typically rely on the database content. As the relational data complexity increases the user move towards the less technical skilled approach. The keyword search is popular due to its simplicity and user-friendly nature with the end user who may be less comfortable with the existing techniques. One key problem in web keyword search techniques to databases is that information related to a single answer to a query keyword may be split across multiple tuples in different relation.

Numerous studies and techniques have been found in the computer science research literature. The existing techniques consider the database as a network of interconnected tuples, through the network they find the keywords in the query and connected components are derived based on association of tuples and return these connected tuples as an answer to the keyword query. To do so, specialized indexing techniques are applied over it, which indexes the database content. Using these indexing techniques, the tuples of interest may directly retrieve or they may instead construct the queries expressions. This is the general idea followed by the modern commercial database management systems.

Unfortunately, the existing system suffers from low precise, less relevant query results. Some of the existing system satisfies the efficiency factor but failed to fulfil the effectiveness of the keyword query. IR techniques allow users to search unstructured information using keywords based on scoring and ranking methods. In this case user does not need to know about database schema.

II. MOTIVATION

Searching for information is an essential component of human lives. Web search engines are widely used for searching textual documents, images and videos. Traditionally, to access these resources, users should have knowledge about structured query languages, like SQL and XQuery. They also need to access data schemas of individual application domain, out of all these most of them are fast evolving, complex and even unavailable on web. A general question to ask is whether the current technology can allow users to effectively access structured data using keyword queries. The result of a keyword search over structured data will automatically collect relevant data that are in distinct locations but are interrelated and collectively relevant to the query.

The popular mode of search is through the use of keywords, which are nothing but a small number of highly discriminating terms. Keyword search offers a straight forward, goal oriented and flexible method of retrieving information. The success of keyword search on the web has generated interest in keyword search interfaces to relational databases and similar structured data sources. Keyword search interfaces offer a simple and flexible alternative. Because of rapid information growth in the information era, many real-time applications need integrating both DB and IR technologies in one system. The sophisticated DB techniques provide users with effective and efficient ways to process structured data maintained and managed by RDBMS.

The advanced IR techniques allow users to use keywords to access unstructured data with scoring and ranking mechanism. Most of previous works of keyword search over textual documents (e.g., HTML documents) have been proposed. The existing web search engines like Google, Yahoo, etc used to produce a list of pages and these pages do not give integrated information from multiple interrelated pages to answer with meaningful keyword query. In the event, there are no pages that contain all the keywords, resulted pages will be with some of the input keywords ranked by relevancy. Even if two or more interrelated pages contain all the keywords, the existing technique cannot integrate the pages into one relevant and meaningful answer.

The next-generation web search engines require link awareness, information items that are linked through hyperlinks. The efficiency of keyword search on structured and semi-structured data remains a challenging problem. The traditional approaches have always employed the inverted index to process keyword queries, which is important for unstructured data but inefficient for semi-structured and structured data. Very few existing works could be universally applied to unstructured data (e.g., textual documents), semi-structured data (e.g., XML documents), structured data (e.g., relational databases) and graph data. Providing both effective and efficient serviceability over such heterogeneous collections within a single search engine remains a big challenge.

In this paper, a review study of the different techniques for keyword searching on relational data, heterogeneous data, linked data and different approaches of keyword searching is taken in brief. The III section describes the related work of various existing systems, section IV concludes with the effective results.

III. RELATED WORK

The relational database management system [RDBMS] was first created in the 1970s. Then its popularity has sky touching and it has become a primary data storage structure in both academic and commercial fields. Relational databases ranges from small, personal databases like Microsoft Access to large-scale database servers like Oracle, Microsoft SQL Server, and MySQL. In relational databases, information needed to answer a keyword query is often split across the tables (tuples), due to normalization. Basically, the work is divided into two directions A. Keyword Search Approach B. Database Selection. Further there are two basic approaches of keyword search based on computing most relevant structured result are as follows ;

A. Keyword Search Approach

Further there are two basic approaches of keyword search based on computing most relevant structured result are as follows ;

1) Schema Based Keyword Search Approach

The nature of set operations used in SQL and the underneath relational algebra, a data graph GD is considered as an undirected graph by avoiding the direction of references between tuples, the resulted structure is undirected in nature. Schema-based approaches support keyword search over relational databases by direct execution of SQL commands. These techniques process design of the relational schema as a graph where edges denote relationships between tuples. The database's full text indices recognize all tuples that contain search terms, a join expression is created for each possible relationship between these tuples [1],[2],[3].

DISCOVER [1] operates on relational databases. Provide functionality of information discovery on the relational database by allowing its user to submit keyword queries without any knowledge of the database schema or of SQL. DISCOVER resulted into qualified joining networks of tuples which are associated as they join on their primary and foreign keys and contain all the keywords of the query. DISCOVER proceeds in two different steps, first the candidate network generation is there and second is candidate network evaluation.

The basic goal of DISCOVER is to find relevant candidate network without redundancy whose size can be data bound without exploiting the schema structure. The important property of DISCOVER is the selection of optimal execution plan is NP complete. It uses greedy algorithm. DISCOVER introduce minimal joining networks, that are trees of tuples where any two adjacent tuples join through a primary key to foreign key relationship.

In [2], text and structured data are often stored side by side within standard relational database management systems (RDBMSs). Commercial RDBMSs generally provide querying capabilities for text attributes that incorporate state-of-the art information retrieval (IR) relevance ranking strategies. This search functionality requires that queries specify the exact column or columns. The requirement that queries specify the exact columns to match can be cumbersome and inflexible from a user perspective: good answers to a keyword query might need to be "assembled".

This observation has motivated recent research on free-form keyword search over relational DBMS. In this paper, IR-style document relevance ranking strategies are adapted to the problem of processing free-form keyword queries over relational DBMS. The noticeable thing is, this approach can handle queries with both AND and OR semantics and exploits the sophisticated single-column text-search functionality often available in commercial relational DBMS. IR-style keyword searching returns few most relevant matches; efficiency is achieved as techniques focus on the top-k matches for the query, for moderate values of k. The given approaches are pipe-lined, in the sense that execution can efficiently resume to compute the "next-k" matches if the user so desires. The key contribution is, it produces high quality results of query keywords. This approach does not require any semantic knowledge about the database. Sparse algorithm is used in the given approach.

In SPARK [3], efficiency and effectiveness are the main aspect of the top-k keyword query. In this approach a new ranking formula by adapting existing IR techniques based on natural notion of virtual document is proposed. As compared with previous approaches, new ranking method is simple, effective and agrees with human perception. It supports both AND and OR semantics and also solve the problem of non monotonic ranking. SPARK proposes skyline sweeping algorithm. It improves the performance of the system.

2) Schema Free Keyword Search Approach

This approach is also called as graph based approach. Graph based search techniques are more general than schema based approaches for relational database, linked database and XML. By observing and studying the underlying graphs, the structured results are computed. The Keywords which are connected and the elements are represented using Steiner trees. The main goal of this approach is to find out structure in the Steiner trees. To find the optimal group Steiner tree is NP complete problem, there are so many efficient and effective algorithms to find the optimal tree for a fixed number of terminals is a dynamic programming algorithm for the optimal solution but remains exponential in the number of search terms. The algorithm recites additional results in approximate sequence. Several kinds of algorithms have been proposed for the efficient exploration of keyword search results. The resulted query keywords are very large in size.

a) Steiner Tree-Based Keyword Search Approach

The Steiner tree-based keyword search approach show two categories of algorithm under Steiner tree-based semantic search approach. First is backward search approach, second is dynamic programming approach.

- *Backsword Search Approach*

The BANKS [4] system enables keyword-based search on database, with both data and schema browsing. This approach enables users to extract information in a simple manner without any knowledge of the schema or any need for writing complex queries. In relational database, information required to answer a keyword query is often divided in the form of tables. Due to such fact a solution to a keyword query may consist of multiple linked tuples. One possible approach to keyword search on databases is to create artificial documents that collect related information. This results in duplication of data, and it is not feasible to create documents corresponding to every meaningful combination of data. It is best to provide support for keyword querying directly on databases.

BANKS provides a rich interface to browse data, with automatic generation of hyperlinks. The BANKS system is developed in Java using servlets and JDBC and can be run on any database without any programming. By avoiding directionality would cause pr problems because of "hubs" that are connected to a large numbers of nodes. Many nodes would be within a short distance of many other nodes, reducing the effectiveness of tree-weight based scoring mechanism. This problem can be solved by creating for each edge (u, v) a backward edge (v, u).

- *Dynamic Programming*

It is widely realized that the integration of database and information retrieval techniques will provide users with a wide range of high quality services. This approach will give brief introduction of processing an l-keyword query, p1; p2; _ _ _; pl, against a relational database which can be modelled as a weighted graph, G(V,E). Here V is a set of nodes and E is a set of edges representing foreign key references between tuples [5]. With a keyword query, users can find the connections among the tuples stored in relations without the needs of knowing the relational schema imposed by RDBMS. In this approach Minimum group Steiner tree problem (GST-1) is explained which is useful for finding top-k minimum cost connected trees in database [5]. This is a NP complete problem.

3) Bidirectional Expansion

Relational, XML and HTML data can be represented as graphs with entities as nodes and relationships as edges. Text is associated with nodes and edges. Keyword search on such graphs has received much attention after many years. A noticeable problem in this approach is to efficiently extract from the data graph a small number of the "best" answer trees.

A backward expanding search is commonly used for predominantly text-driven queries [6]. But its performance goes low if one keyword match with nodes or some node has very large degree. To resolve the problem a new search algorithm Bidirectional Search is proposed which improves on Backward Expanding search by allowing forward search from potential roots towards leaves.

4) Keyword-Search over XML Documents

This approach considers the problem of efficiently producing ranked results for keyword search queries over hyperlinked XML documents. Analyzing ande evaluating keyword search queries over hierarchical XML documents as opposed to flat HTML documents introduces many new challenges [7]. XML keyword search queries do not always return entire documents but can return deeply nested XML elements that contain the desired keywords. The nested structure of XML implies that the notion of ranking is no longer at the depth of coarseness of a document, but depends on the coarseness of an XML element.

The notion of keyword proximity is more complex in the hierarchical XML data model. The XRANK [7] system that is designed to handle these novel features of XML keyword search. The experimental results show that XRANK offers both space and performance benefits when compared with existing approaches. An attractive feature of XRANK is that it naturally generalizes a hyperlink based HTML search engine such as Google, Yahoo, Safari, etc. Therefore this approach can be applied to query a mix of HTML and XML documents. Dewey Inverted List (DIL) query processing algorithm is derived for common longest prefix. XRANK considers element to element links in addition to document to document links.

5) Keyword Search over Linked Data

Existing work on keyword search relies on an element-level model (data graphs) to compute keyword query results [9], [10]. Elements mentioning keywords are retrieved from this model and paths between them are explored to compute Steiner graphs. Keyword Relationship Graph (KRG) captures relationships at the keyword level. Relationships captured by a KRG are not direct edges between tuples but stand for paths between keywords.

To route keywords only to relevant sources to reduce the high cost of processing keyword search queries over all sources is the new trade in keyword searching. In this concept a multilevel scoring mechanism is proposed for computing

the relevance of routing plans based on scores at different levels of keywords, data elements, element sets and sub graphs that connect these elements. The Keyword Element Relationship Graph (KERG) algorithm is proposed in this approach [8].

B. Database Selection

The goal is to identify the most relevant databases. The main idea is based on modelling databases using keyword relationships. A keyword relationship is a pair of keywords that can be connected via a sequence of join operations [8]. A database is relevant if its keyword relationship model covers all pairs of query keywords. M-KS [9] captures relationships using a matrix. It considers only binary relationships between keywords.

G-KS [10] addresses this problem by considering more complex relationships between keywords using a keyword relationship graph (KRG). Each node in the graph corresponds to a keyword. Compared to M-KS, G-KS computes more relevant sources, G-KS adopts IR-style ranking to compute TF-IDF for keywords and for keyword relationships. It provides an additional level of filtering, validating connections between keywords based on complex relationships and distance information in the KRG.

IV. CONCLUSION

In this paper, the review is conducted for comparing the performance of different approaches used for keyword search in different dynamic environment. Keyword searching in relational database shows the versatility in their behaviour. This survey based on the basic approaches and how they are efficient, effective and diversity in algorithms proposed accordingly. The paper also gives the idea about ranking and scoring mechanism. Keyword query search is very popular approach for retrieving linked data in an efficient and effective manner which reduces the high cost of searching.

ACKNOWLEDGEMENT

Every work is source which requires support from many people and areas. It gives me proud privilege to publish my sincere work on the respective topic under the valuable guidance of Prof. S. A. Bajpai. I would like to thank my organization for timely help and inspiration and also to all the unseen authors of various articles on the internet, helping me to become aware of the ongoing research in this field and all my colleagues for providing help and support in my work.

REFERENCES

(1) V. Hristidis and Y. Papakonstantinou, "Discover: Keyword earch in Relational Databases", Proc. 28th Int'l Conf. Very Large Data Bases(VLDB), pp. 670-681, 2002.
(2) V. Hristidis, L. Gravano, and Y. Papakonstantinou, "Efficient IR-Style Keyword Search over Relational Databases", Proc. 29th Int'l Conf. Very Large Data Bases (VLDB), pp. 850-861, 2003.
(3) Y. Luo, X. Lin, W. Wang, and X. Zhou, "Spark: Top-K Keyword Query in Relational Databases", Proc. ACM SIGMOD Conf.,pp. 115-126 , 2007.
(4) G. Bhalotia, A. Hulgeri, C. Nakhey, S. Chakrabarti, and S. Sudarshan, "Keyword searching and browsing in databases using BANKS", ICDE , 2002.
(5) B. Ding, J.X. Yu, S. Wang, L. Qin, X. Zhang, and X. Lin, "Finding Top-K Min-Cost Connected Trees in Databases", Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), pp. 836-845, 2007.
(6) V. Kacholia, S. Pandit, S. Chakrabarti, S. Sudarshan, R. Desai, and H. Karambelkar, "Bidirectional Expansion for Keyword Search on Graph Databases", Proc. 31st Int'l Conf. Very Large Data Bases (VLDB), pp. 505-516, 2005.
(7) L. Guo, F. Shao, C. Botev, and J. Shanmugasundaram, " XRANK: Ranked keyword search over XML documents", SIGMOD 2003.
(8) Thanh Tran and Lei Zhang, "Keyword Query Routing", IEEE Transactions, VOL.26, NO.2, February 2014.
(9) B. Yu, G. Li, K.R. Sollins, and A.K.H. Tung, "Effective Keyword-Based Selection of Relational Databases", Proc. ACM SIGMOD Conf., pp. 139-150, 2007.
(10) Q. H. Vu, B.C. Ooi, D. Papadias, and A.K.H. Tung, "A Graph Method for Keyword-Based Selection of the Top-K Databases," Proc. ACM SIGMOD Conf., pp. 915-926, 2008.