International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Routing Keyword Search Using KERG

Chaitali S. Chaudhari[1], M. M. Naoghare[2]

[1]Department of Computer Engineering, Sir Visvesvaraya Institute of Technology, Nashik, Maharashtra, India

Abstract: ***Searching a keyword on a vast web is somewhat easier, but the search over a enlarge number of structured and linked data creates a difficulty. Routing keywords only to applicable sources can reduce the high cost of searching of queries over all sources. It is difficult for web user to use this web data by means of SQL or SPARQL. We hire a keyword element relationship summary that compactly represents relationships between keywords and the data elements called the set-level keyword-element relationship graph (KERG).A multilevel scoring mechanism is suggested for computing the relevant of routing plans based on the level of keyword, data elements , element sets and subgraphs that connect these elements.***

Keywords: Keyword search, keyword query, Routing keywords, graph-structured data

1. Introduction

The world wide web is not only a collection of textual documents but also a web of linked data sources .It is difficult for the typical web users to use this web data by means of structured queries using languages like SQL or SPARQL. To this end, searching keyword has proven to be spontaneous. As inspite of structured queries, no special knowledge of the query language, the schema or the underlying data are needed. . Query processing over graph-structured data is enjoying a growing number of applications. A top-*k* keyword search query on a graph finds the top k answers, where each answer is a substructure of the graph containing all query keywords in database research, solutions have been given, which given a keyword search, retrieve the most applicable structured results or simply, select the single most relevant databases . However, these approaches are single-source solutions. They are not directly relevant to the web of Linked Data, where results are not binded by a single source but might complete several Linked Data sources. As opposed to the source selection problem, which is focusing on computing the most relevant sources, the problem here is to calculate the most relevant combinations of sources. The goal is to produce routing plans, which can be used to calculate results from multiple sources

To this end, we provide the following contributions:
We suggest examining the problem of routing keyword search for query search over a large number of structure and Linked Data sources. Routing keywords only to relevant sources can reduce the high cost of searching for structured results that span multiple sources. To the best of our knowledge, the work shown in this paper represents the first attempt to show the problem.

Existing work uses keyword relationships (KR) collected individually for single databases. We shows the relationships between keywords as well as those between data elements. They are constructed for the entire collection of linked sources, and then grouped as elements of a compact summary called the set-level keyword-element relationship graph (KERG).

IR-style ranking has been shown to involve relevance at the level of keywords. To cope with the increased keyword ambiguity in the web setting, we hire a multilevel relevance model, where elements to be considered are keywords, entities mentioning these keywords, corresponding sets of entities, relationships between elements of the same level, and inter-relationships between elements of different levels.

We implemented the approach and we show that when routing is applied to an existing searching keyword system to prune sources, substantial performance gain can be achieved.

2. Literature Survey

V. Hristidis, L. Gravano, and Y. Papakonstantinou [1], Proposed Keyword search is a familiar and potentially effective way to find information of interest that is “locked” inside relational databases. Current work has generally assumed that answers for a keyword query reside within a single database. Many practical settings, however, require that we combine tuples from multiple databases to obtain the desired answers. Such databases are often autonomous and heterogeneous in their schemas and data. This paper describes Kite, a solution to the keyword-search problem over heterogeneous relational databases. Kite combines schema matching and structure discovery techniques to find approximate foreign-key joins across heterogeneous databases. Such joins are critical for producing query results that span multiple databases and relations. Kite then exploits the joins – discovered automatically across the databases – to enable fast and effective querying over the distributed data. Our extensive experiments over real-world data sets show that (1) our query processing algorithms are efficient and (2) our approach manages to produce high-quality query results spanning multiple heterogeneous databases, with no need for human reconciliation of the different databases.

F. Liu, C.T. Yu, W. Meng, and A. Chowdhary [3], proposed with the amount of available text data in relational databases growing rapidly, the need for ordinary users to search such information is dramatically increasing. Even though the major RDBMSs have provided full-text search capabilities, they still require users to have knowledge of the database schemas and use a structured query language to search information. This search model is complicated for most ordinary users. Inspired by the big success of information retrieval (IR) style keyword search on the web, keyword

search in relational databases has recently emerged as a new research topic. The differences between text databases and relational databases result in three new challenges: (1) Answers needed by users are not limited to individual tuples, but results assembled from joining tuples from multiple tables are used to form answers in the form of tuple trees. (2) A single score for each answer (i.e. a tuple tree) is needed to estimate its relevance to a given query. These scores are used to rank the most relevant answers as high as possible. (3) Relational databases have much richer structures than text databases. Existing IR strategies are inadequate in ranking relational outputs. In this paper, we propose a novel IR ranking strategy for effective keyword search.

Y. Luo, X. Lin, W. Wang, and X. Zhou[4], proposed we study the effectiveness and the efficiency issues of answering top-k keyword query in relational database systems. We propose a new ranking formula by adapting existing IR techniques based on a natural notion of virtual document. Compared with previous approaches, our new ranking method is simple yet effective, and agrees with human perceptions. We also study efficient query processing methods for the new ranking method, and propose algorithms that have minimal accesses to the database. We have conducted extensive experiments on large-scale real databases using two popular RDBMSs. The experimental results demonstrate significant improvement to the alternative approaches in terms of retrieval effectiveness and efficiency.

M. Sayyadian, H. LeKhac [2], A. Doan, and L. Gravano, proposed Keyword search is a familiar and potentially effective way to find information of interest that is “locked” inside relational databases. Current work has generally assumed that answers for a keyword query reside within a single database. Many practical settings, however, require that we combine tuples from multiple databases to obtain the desired answers. Such databases are often autonomous and heterogeneous in their schemas and data. This paper describes Kite, a solution to the keyword-search problem over heterogeneous relational databases. Kite combines schema matching and structure discovery techniques to find approximate foreign-key joins across heterogeneous databases. Such joins are critical for producing query results that span multiple databases and relations. Kite then exploits the joins – discovered automatically across the databases – to enable fast and effective querying over the distributed data. Our extensive experiments over real-world data sets show that (1) our query processing algorithms are efficient and (2) our approach manages to produce high-quality query results spanning multiple heterogeneous databases, with no need for human reconciliation of the different databases.

B. Ding, J.X. Yu, S. Wang, L. Qin, X. Zhang, and X. Lin[6], proposed It is widely realized that the integration of database and information retrieval techniques will provide users with a wide range of high quality services. In this paper, we study processing an l keyword query, p1; p2…. pl, against a relational database which can be modeled as a weighted graph, G(V;E). Here V is a set of nodes (tuples) and E is a set of edges representing foreign key references between tuples. Let Vi _ V be a set of nodes that contain the keyword pi. We study finding top-k minimum cost connected trees that contain at least one node in every subset Vi, and denote our problem as GST-k. When k = 1, it is known as a minimum cost group Steiner tree problem which is NPComplete. We observe that the number of keywords, l, is small, and propose a novel parameterized solution, with l as a parameter, to find the optimal GST-1, in time complexity O(3ln + 2l((l + log n)n +m)), where n and m are the numbers of nodes and edges in graph G. Our solution can handle graphs with a large number of nodes. OurGST-1 solution can be easily extended to support GST-k, which outperforms the existing GST-k solutions over both weighted undirected/directed graphs. We conducted extensive experimental studies, and report our finding.

3. Proposed System

Proposed work can be categorized into two main categories:

3.1 Schema-Based

This approaches implemented on top of off-the-shelf databases. A input query is processed by mapping keywords to elements of the database (called keyword elements). Then, using the schema, valid join sequences are derived, which are then employed to join (“connect”) the computed keyword elements to form so-called candidate networks representing possible results to the keyword query.

3.2 Schema-Agnostic

This approach operates directly on the data. Structured results are computed by exploring the underlying data graph. The goal is to find structures in the data called Steiner trees (Steiner graphs in general), which connect keyword elements. Various kinds of algorithms have been proposed for the efficient exploration of query search results over data graphs, which might be very large.

The aim to identify data sources that contain results to a keyword search. In the Linked Data scenario, results might combine data from several sources.

1. **Keyword Routing Plan**
   The problem of keyword query routing is to find the top keyword routing plans based on their relevance to a query. A relevant plan should correspond to the information need as intended by the user.
2. **Multilevel Inter-Relationship Graph**
   To illustrate the search space of keyword query routing using a multilevel inter-relationship graph. At the lowest level individual data elements, and a set-level data graph, this captures information about group of elements.

3.3 Block diagram of the proposed system:

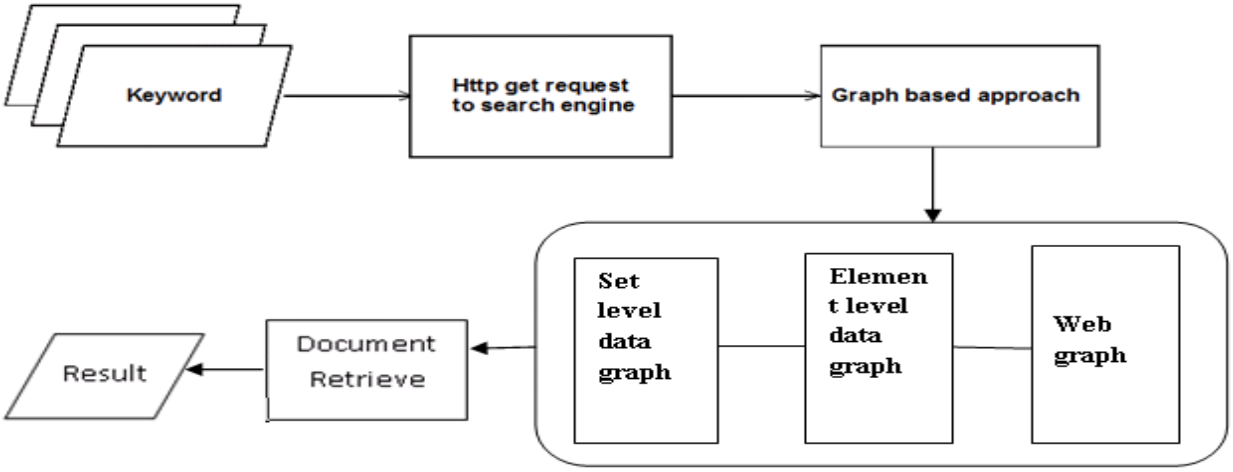

Figure 1: Block diagram of the proposed system

4. Conclusion

We have presented a solution to the problem of Routing Keyword Search using KERG. Based on modeling the search space as a multilevel inter-relationship graph, we proposed a summary model that groups keyword and element relationships at the level of sets, and developed a multilevel ranking scheme to incorporate relevance at different dimensions.

References

[1] 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.
[2] M. Sayyadian, H. LeKhac, A. Doan, and L. Gravano, "Efficient Keyword Search across Heterogeneous Relational Databases," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), pp. 346-355, 2007.
[3] F. Liu, C.T. Yu, W. Meng, and A. Chowdhury, "Effective Keyword Search in Relational Databases," Proc. ACM SIGMOD Conf., pp. 563-574, 2006.
[4] 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.
[5] G. Ladwig and T. Tran, "Index Structures and Top-K Join Algorithms for Native Keyword Search Databases," Proc. 20th ACM Int'l Conf. Information and Knowledge Management (CIKM),pp. 1505-1514, 2011.
[6] 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.
[7] S. Chaudhuri and G. Das, "Keyword Querying and Ranking in Databases," Proceedings of the VLDB Endowment, vol. 2, pp. 1658–1659, August 2009. [Online]. Available: http://dl.acm.org/citation.cfm?id=1687553. 1687622
[8] Y. Chen, W. Wang, Z. Liu, and X. Lin, "Keyword Search on Structured and Semi-Structured Data," in Proceedings of the 35th SIGMOD International Conference on Management of Data, ser. SIGMOD '09, June 2009, pp. 1005–1010.
[9] J. Coffman and A. C. Weaver, "A Framework for Evaluating Database Keyword Search Strategies," in Proceedings of the 19th ACM International Conference on Information and Knowledge Management, ser. CIKM '10, October 2010,pp. 729–738. [Online]. Available: http://doi.acm.org/10.1145/1871437.1871531
[10] H. He, H. Wang, J. Yang, and P. S. Yu, "BLINKS: Ranked Keyword Searches on Graphs," in Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, ser. SIGMOD '07, June 2007, pp. 305–316.
[11] 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.
[12] C. D. Manning, P. Raghavan, and H. Sch¨utze," Introduction to Information Retrieval." New York, NY: Cambridge University Press, 2008.
[13] Tan P-N, Steinbach M. and Kumar V., Introduction to Data Mining, Addison Wesley, 2006.
[14] Soumen Chakrabarti, "Mining the Web: Discovering Knowledge from Hypertext Data" Indian Institute of Technology, Bombay, ISBN: 1-55860-754-4
[15] Usama Fayyad, Georges G. Grinstein, and Andreas Wierse ,"Information Visualization in Data Mining and Knowledge Discovery "

Author Profile

Ms. Chaitali S. Chaudhari has completed her B.E in Computer Engineering from Pune University and currently pursuing Master of Engineering from SVIT Chincholi, Nashik, India

Prof. M. M. Naoghare has completed her B.E in Computer Engineering from College of Engineering, Badnera, Amravati University and M.E in Computer Science & Engineering from P.R.M.I.T & R, Badnera, Amravati. She is presently working as an Associate Professor in SVIT Chincholi, Nashik, India