ISSN:0975-9646

Efficient Keyword Query Routing for Search Engines

Ms. Pawar Prajakta Bhagwat[1]
ME Student, Dept of CE,
MCERC, Nashik, India.

Mr. Niranjan L. Bhale[2]
HOD, IT Dept,
MCERC, Nashik, India.

Abstract:A type of search that looks for matching documents that contain one or more words specified by the user is called keyword search. Find the info we need. It is for searching linked data sources on the web. There is a method for computing top-k routing plans based on their potentials to contain results for a given keyword query. Keywords and the data elements mentioning them are related by using a keyword-element relationship. A multilevel scoring mechanism is for computing the relevance of routing plans based on scores at the level of keywords, data elements, element sets, and subgraphs that connect these elements. Experiments carried out using 150 publicly available sources on the web showed that valid plans (precision@1 of 0.92) that are highly relevant (mean reciprocal rank of 0.89) can be computed in 1 second on average on a single PC. However, queries with more keywords could not be handled efficiently. For instance, queries with more than two keywords needed several seconds up to one minute. Thus, while this setting produced results of highest quality, it is not really affordable in a typical web scenario demanding high responsiveness. To produce results in minimum time, while not compromising too much on quality. The results suggest that keyword search without routing is especially problematic when the number of keywords is large. Thus the proposed system uses routing keyword search for queries having large number of keywords, which is for improving the performance of keyword search. This approach can greatly reduce time and space costs.

Keywords: Keyword search, keyword query, keyword query routing, graph-structured data.

1. INTRODUCTION

In recent years the Web has evolved from a global information space of linked documents to one where both documents and data are linked. Underpinning this evolution is a set of best practices for publishing and connecting structured data on the Web known as Linked Data. The adoption of the Linked Data best practices has lead to the extension of the Web with a global data space connecting data from diverse domains such as people, companies, books, scientific publications, films, music, television and radio programs, genes, proteins, drugs and clinical trials, online communities, statistical and scientific data, and reviews. The representation of the linked data on the web is shown in figure 1. This Web of Data enables new types of applications. There are generic Linked Data browsers which allow users to start browsing in one data source and then navigate along links into related data sources. There are Linked Data search engines that crawl the Web of Data by following links between data sources and provide expressive query capabilities over aggregated data, similar to how a local database is queried today. The Web of Data also opens up new possibilities for domain-specific applications. Unlike Web 2.0 mashups which work against a fixed set of data sources, Linked Data applications operate on top of an unbound, global data space. This enables them to deliver more complete answers as new data sources appear on the Web. We propose to investigate the problem of keyword query routing for keyword search over a large number of structured 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 presented in this paper represents the first attempt to address this problem.

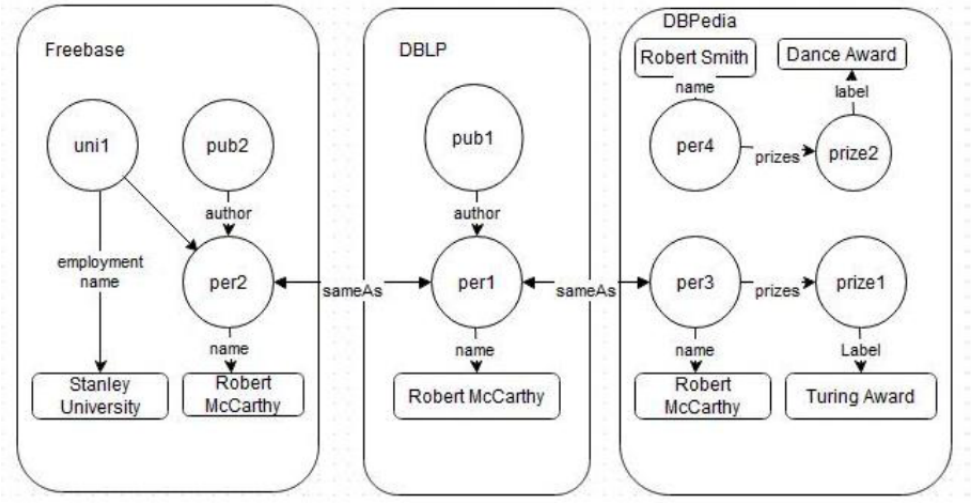

Figure 1: Example of Linked data on web

We use a graph-based data model to characterize individual data sources. In that model, we distinguish between an element-level data graph representing relationships between individual data elements, and a set-level data graph, which captures information about group of elements. This set-level graph essentially captures a part of the Linked Data schema on the web that is represented in RDFS, i.e., relations between classes. Often, a schema might be incomplete or simply does not exist for RDF data on the web. In such a case, a pseudo schema can be obtained by computing a structural summary such as a dataguide.

The web is no longer a collection of textual data but also a web of interlinked data sources. One project that largely contributes to this development is Linking Open Data. Through this, a vast amount of structured information was made publicly available. Querying that huge amount of data in an intuitive way is challenging. Collectively, Linked Data comprise hundreds of sources containing billions of RDF triples, which are connected by millions of links. While different kinds of links can be established, the ones frequently published are sameAs links, which denote that two RDF resources represent the same real-world object. The representation of the linked data on the web.

The linked data Web already contains valuable data in diverse areas, such as e-government, ecommerce, and the biosciences. Additionally, the number of available datasets has grown solidly since its inception [2]. In order to search such data we use keyword search techniques which employ keyword query routing. To decrease the high cost incurred in searching structured results that span multiple sources, we propose routing of the keywords to the relevant databases. As opposed to the source selection problem [3], which is focusing on computing the most relevant sources, the problem here is to compute the most relevant combinations of sources. The goal is to produce routing plans, which can be used to compute results from multiple sources. For selecting the correct routing plan, we use graphs that are developed based on the relationships between the keywords present in the keyword query. This relationship is considered at the various levels such as keyword level, element level, set level etc..

Existing system investigates the problem of keyword query routing for keyword search over a large number of structured and Linked Data sources. Based on modeling the search space as a multilevel inter-relationship graph, a summary model is used for grouping keyword and element relationships at the level of sets. It uses a multilevel ranking scheme to incorporate relevance at different dimensions. This system does not compute all but uses several mechanisms to prune some answers. It could not handle queries with multiple keywords efficiently.

The rest of paper is organized as follows. Section 2 provides the brief outline on the existing work. The proposed system in the section 3 before we conclude in the section 4.

2. RELATED WORK

Keyword Query Search can be divided into two directions of work. They are:

1) Keyword search approaches compute the most relevant structured results.
2) Solutions for source selection compute the most relevant sources.

In the keyword searching, we mainly follow two approaches. They are schema-based approaches and schema-agnostic approaches.

Schema-based approaches are implemented on top of off-the-shelf databases. A keyword is processed by mapping keywords to the elements of the databases, called keyword elements. Then, using the schema, valid join sequences are derived and are employed to join the computed keyword elements to form the candidate-networks that represent the possible results to the keyword query.

Schema-agnostic approaches operate directly on the data. By exploring the underlying graphs the structured results are computed in these approaches. Keywords and elements which are connected are represented using Steiner trees/graphs. The goal of this approach is to find structures in the Steiner trees. Various kinds of algorithms have been proposed for the efficient exploration of keyword search results over data graphs, which might be very large. Examples are bidirectional search [4] and dynamic programming [5]. Recently, a system called Kite extends schema based techniques to find candidate networks in the multi source setting [6]. It employs schema matching techniques to discover links between sources and uses structure discovery techniques to find foreign key joins across sources. Also based on pre computed links, Hermes [7] translates keywords to structured queries.

In order to get the efficient results for keyword search, the selection of the relevant data sources plays a major role. The main idea is based on modeling databases using keyword relationships. A keyword relationship is a pair of keywords that can be connected via a sequence of join operations. A database is considered relevant if its keyword relationship model covers all pairs of query keywords.

M-KS [3] considers only binary relationships between keywords. It incurs a large number of false positives for queries with more than two keywords. This is the case when all query keywords are pair wise related but there is no combined join sequence which connects all of them. G-KS [8] 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. Each edge between two nodes corresponding to the keywords.

For routing the keywords to the relevant data sources and searching the given keyword query, we propose four different approaches. They are: 1) Keyword level model 2) Element level model, 3) Set level model, and 4) Query expansion using linguistic and semantic features. We compute the keyword query result and keyword routing plan [1] which is the two important factors of keyword routing. In keyword level, we mainly consider the relationship between the keywords in the keyword query. This relationship can be represented using Keyword Relationship Graph (KRG) [8]. It captures relationships at

the keyword level. As opposed to keyword search solutions, relationships captured by a KRG are not direct edges between tuples but stand for paths between keywords.

For database selection, KRG relationships are retrieved for all pairs of query keywords to construct a sub graph. Based on these keyword relationships alone, it is not possible to guarantee that such a sub graph is also a Steiner graph (i.e., to guarantee that the database is relevant). To address this, sub graphs are validated by finding those that contain Steiner graphs. This is a filtering step, which makes use of information in the KRG as well as additional information about which keywords are contained in which tuples in the database. It is similar to the exploration of Steiner graph in keyword search, where the goal is to ensure that not only keywords but also tuples mentioning them are connected. However, since KRG focuses on database selection, it only needs to know whether two keywords are connected by some join sequences or not. This information is stored as relationships in the KRG and can be retrieved directly. For keyword search, paths between data elements have to be retrieved and explored. Retrieving and exploring paths that might be composed of several edges are clearly more expensive than retrieving relationships between keywords.

Keyword search over relational databases finds the answers of tuples in the databases which are connected through primary/foreign keys and contain query keywords. As there are usually large numbers of tuples in the databases, these methods are rather expensive to find answers by on-the-fly enumerating the connections. To address this problem, proposed tuple units [9] to efficiently answer keyword queries. A tuple unit is a set of highly relevant tuples which contain query keywords.

3. THE PROPOSED SYSTEM

To route keywords only to relevant sources to reduce the high cost of processing keyword search queries over all sources. A novel method was used for computing top-k routing plans based on their potentials to contain results for a given keyword query. It employs a keyword-element relationship summary that compactly represents relationships between keywords and the data elements mentioning them. A multilevel scoring mechanism was proposed for computing the relevance of routing plans based on scores at the level of keywords, data elements, element sets, and subgraphs that connect these elements. Also to investigate the problem of keyword query routing for keyword search over a large number of structured and Linked Data sources. This system was having more advantages: 1) Routing keywords only to relevant sources can reduce the high cost of searching for structured results that span multiple sources. 2) The routing plans, produced can be used to compute results from multiple sources.

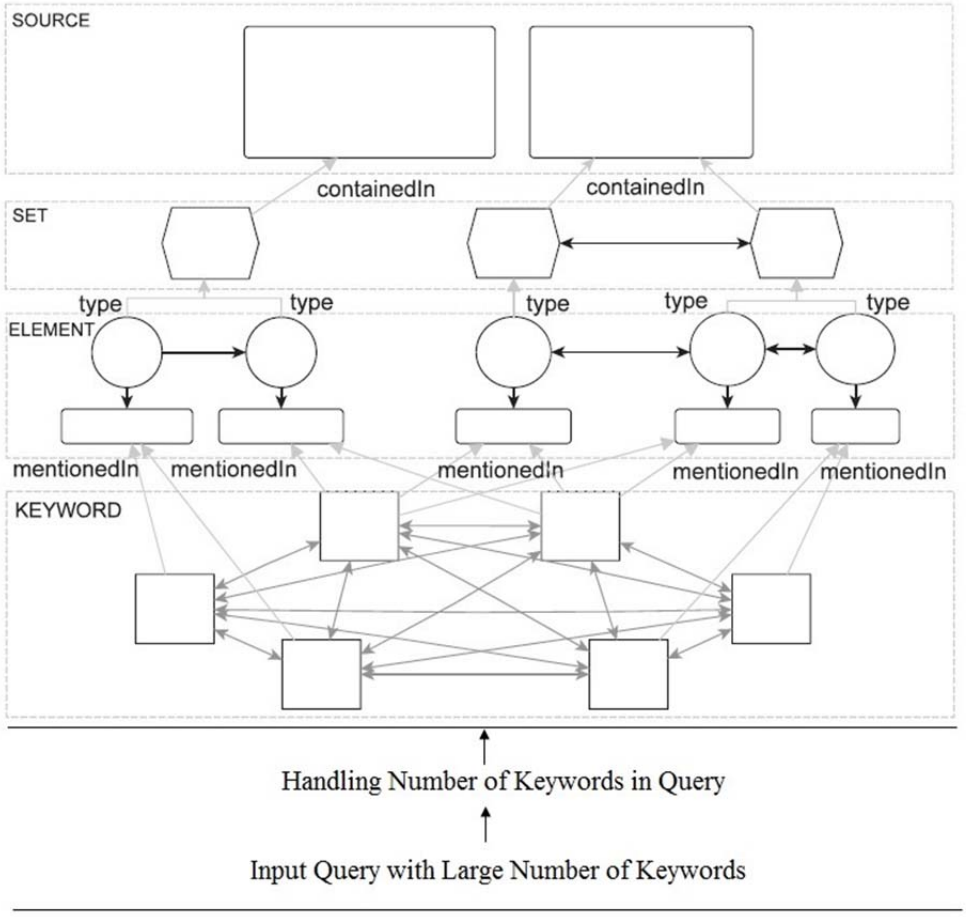

Figure 2: Inter Relationship between Elements.

However, queries with more keywords could not be handled efficiently. For instance, queries with more than two keywords needed several seconds up to one minute. Thus, while this setting produced results of highest quality, it is not really affordable in a typical web scenario demanding high responsiveness. To produce results in minimum time, while not compromising too much on quality. The results suggest that keyword search without routing is especially problematic when the number of keywords is large. Thus the proposed system uses routing keyword search for the queries having large number of keywords.

The search space of keyword query routing using a multilevel inter-relationship graph. At the lowest level, it models relationships between keywords. In the upper most levels, there are W (N, ε) and the source-level web graph, which contains sources as nodes. The inter-relationships between elements at different levels are illustrated in Figure 2. A keyword is mentioned in some entity descriptions at the element level. Entities at the element level are associated with a set-level element via type. A set-level element is contained in a source. There is an edge between two keywords if two elements at the element level mentioning these keywords are connected via a path. Fig. represents a holistic view of the search space. Based on this view, we propose a ranking scheme that deals with relevance at many levels. Further, Fig. provides different perspectives on the search space. Based on this representation of the search space, existing work on keyword search and database selection can be extended to solve the problem of keyword query routing.

For selecting the correct routing plan, we use graphs that are developed based on the relationships between the keywords present in the keyword query. This relationship is considered at the various levels such as keyword level, element level, set level e.t.c. The goal is to produce routing plans, which can be used to compute results from multiple sources. However, queries with more keywords could not be handled efficiently. For instance, queries with more than two keywords needed several seconds up to one minute. Thus proposed system tries to handle such queries with number of keywords and tries to minimize the computing time.

CONCLUSION

This paper helps to improve the performance of keyword search, without compromising its result quality. Investigate the problem of keyword query routing for keyword search over a large number of structured and Linked Data sources. Routing keywords only to relevant sources can reduce the high cost of searching for structured results that span multiple sources. We use a graph-based data model to characterize individual data sources. For selecting the correct routing plan, we use graphs that are developed based on the relationships between the keywords present in the keyword query. This relationship is considered at the various levels such as keyword level, element level, set level e.t.c. In the existing system, Routing keywords return all the source which may or may not be the relevant sources.

However, queries with more keywords could not be handled efficiently. For instance, queries with more than two keywords needed several seconds up to one minute. Thus, while this setting produced results of highest quality, it is not really affordable in a typical web scenario demanding high responsiveness. To produce results in minimum time, while not compromising too much on quality. The results suggest that keyword search without routing is especially problematic when the number of keywords is large. Thus the proposed system uses routing keyword search for the queries having large number of keywords.

REFERENCES

[1] Thanh Tran and Lei Zhang, "Keyword Query Routing", IEEE Transactions, VOL.26, NO.2, February 2014.

[2] T. Berners-Lee, Linked Data Design Issues, 2009;www.w3.org/DesignIssues / LinkedData.html

[3] 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.

[4] V. Kacholia, S. Pandit, S. Chakrabarti, S. Sudarshan, R. Desai, and H. Karam-belkar, "Bidirectional Expansion for Keyword Search on Graph Databases", Proc. 31st Intl Conf. Very Large Data Bases (VLDB), pp. 505-516, 2005.

[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 Intl Conf. Data Eng. (ICDE), pp. 836845, 2007.

[6] M. Sayyadian, H. LeKhac, A. Doan, and L. Gravano, "Efficient Keyword Search Across Heterogeneous Relational Databases", Proc. IEEE 23rd Intl Conf. Data Eng. (ICDE), pp. 346-355, 2007.

[7] T. Tran, H. Wang, and P. Haase, "Hermes: Data Web Search on a Pay-as-You-Go Integration Infrastructure", J. Web Semantics, vol. 7, no. 3, pp. 189-203, 2009.

[8] 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.

[9] Jianhua Feng, Guoliang Li and Jianyong Wang, "Finding Top-k answers in keyword search over relational databases using tuple units", IEEE transactions, VOL. 23 NO. 12, December 2011.

[10] G. Li, B.C. Ooi, J. Feng, J. Wang, and L. Zhou, "Ease: An Effective 3-in-1 Keyword Search Method for Unstructured, SemiStructured and Structured Data", Proc. ACM SIGMOD Conf., pp. 903-914, 2008.

[11] R. Goldman and J. Widom, "DataGuides: Enabling Query Formulation and Optimization in Semistructured Databases", Proc. 23rd Intl Conf. Very Large Data Bases (VLDB), pp. 436-445, 1997.

[12] K. Collins- Thompson, "Reducing the risk of query expansion via robust con-strained optimization". In CIKM. ACM, 2009.

[13] H. Deng, G. C. Runger, and E. Tuv. "Bias of importance measures for multi-valued attributes and solutions". In ICANN (2), volume 6792, pages 293300. Springer, 2011.

[14] D. Mladenic, J. Brank, M. Grobelnik, and N. Milic-Frayling. "Feature selection using linear classiffier weights: interaction with classiffication models". In Pro-ceedings of the 27th Annual International ACM SIGIRConference SIGIR2004. ACM, 2004.

[15] Saeedeh Shekarpour, Jens Lehmann, and Sren Auer, "Keyword Query Expan-sion on Linked Data Using Linguistic and Semantic Features", IEEE Seventh International Conference on Semantic Computing, 2013.