ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 6, Special Issue 5, March 2017

National Conference on Advanced Computing, Communication and Electrical Systems - (NCACCES'17)

24th - 25th March 2017

Organized by

C. H. Mohammed Koya
KMEA Engineering College, Kerala- 683561, India

A Survey on Keyword Diversification Over XML Data

Migi K [1], Jasila E. K. [2]

M. Tech. Student, Department of Computer Science and Engineering, MES College, Kuttippuram, Kerala, India[1]

Assistant Professor, Department of Computer Science and Engineering, MES College, Kuttippuram, Kerala, India[2]

ABSTRACT*:* Keyword queries are those terms that users enter and use to retrieve documents that have all or any of those terms. They are the most familiar and popular method used by ordinary users to search data. Keyword queries are highly ambiguous. Keyword search querying has emerged as one of the most effective way for information discovery, especially over HTML documents in the World Wide Web. Because of its simplicity keyword queries are one of the most effective ways for information discovery in World Wide Web. When using Keyword queries users do not have to learn a complex query language, nor needs him to have any prior knowledge about the structure of the underlying data. When considering the keyword query interpretations, a single keyword query interpretation will not be sufficient to satisfy the user. Many interpretations may yield unnecessary results. This will lead to user's dissatisfaction. Diversification is mainly meant to minimize user's dissatisfaction. The keyword query should return the major possible interpretations for the keywords in the underlying database. This will enable the user to easily select the intended interpretation. In the case of information retrieval (IR) keyword queries always retrieve a list of relevant documents and it needs to be analyzed manually one by one. But keyword queries over structured data give a more direct and effective way of diversification. In the case of keyword search over structured or semi-structured database if a keyword comes as value of more than one attribute, then each occurrence can be taken as different interpretations. Each interpretation will yield different results. Keyword Query Diversification can be defined as for a given keyword query over the XML dataset, the user should get a result set of top-k results, where each result should be relevant to the given keyword query and they should be maximum different to each other [1]. Diversification is done based on the underlying data that is to be searched. There are many approaches for keyword query diversification over XML data. These approaches derive different search intentions for each keyword in the query. This paper presents a comparative study on these different approaches. And also identifies context-based keyword diversification is the most effective method for keyword query diversification which will automatically diversifies the given keyword query based on different contexts of the keywords in the XML data. The method is to extract feature terms for each keyword in the given keyword query. And to generate query candidates of high relevance and novelty. Then search results are retrieved for those query candidates. This method can efficiently retrieve relevant search results, the method works effectively with purely text based data set. The existing method cannot retrieve images and text formats like italics and bold as such. The proposed system will overcome this drawback by considering images and text formats while diversification. It can also provide effective query suggestions to users.

KEYWORDS: XML, SLCA, Keyword search, Context-based diversification, Mutual Information.

I. INTRODUCTION

A. *XML*: XML is Extensible Markup Language that defines a set of rules for encoding documents in a format that is both human and machine readable. Extensible means users can define their own tags, the order in which the tags occur, processed and displayed. XML is a data description language. XML documents are semi-structured. Tags in XML separate semantic elements and enforce hierarchies with in data. Large repositories of XML data exist over the web and are mainly accessed by using XML query languages such as Xpath and Xquery [2]. Such queries return large answer set making it quiet challenging for the user to select the best result. Keyword query over XML data does not result in the entire document. It concentrates more on specific information contents.

B. *SLCA*: Results of XML keyword search can be deeply nested nodes that contain the keywords. These nodes will have specific information contents. Returning these nodes can give more context information. For example keyword query over XML data can return fragments rooted at the smallest lowest common ancestor, SLCA, nodes. SLCA is a widely adopted semantics to define meaningful results for keyword queries [3]. A keyword search using the SLCA semantics returns nodes in the XML data that satisfy the following two conditions [4]:

1. The subtrees rooted at the nodes contain all the keywords.
2. The nodes do not have any proper descendant node that satisfies condition 1.

Thus specific results in XML keyword search is represented by SLCA semantics with minimal information content.

II. LITERATURE SURVEY

Ambiguity is an inherent property of keyword queries. It will be very difficult to derive the search intentions of a keyword query if it contains a small number of keywords. To address this challenging problem the most effective way is to provide diverse keyword query suggestions. This is done based on the keywords in the query and the data to be searched. There are many approaches for keyword query diversification over XML data.

A. STRUCTURED SEARCH RESULT DIFFERENTIATION: This work measures the difference of XML keyword search results by comparing their feature sets. This approach proposes techniques for comparing and differentiating structured search results [5]. And also implements a system which differentiates search results on XML data, XRed, XML Result Differentiation. Associated with each search result there will be a set of features. The selection of feature set is according to metadata information in XML data. A Differentiation Feature Set, DFS, will be a subset of features that highlights the differences between search results. DFS helps users to compare search results. DFS should be of small size and contain reasonable summary that captures the specific characteristics of the result. In DFSs generation maximal differentiation is the optimal goal. If the DFSs of two results have common features types then they are comparable by their DFSs. If the DFSs of two results have different characteristics for those common feature types then two results are differentiable. The number of feature terms on which the DFSs are differentiable is called Degree of Differentiation, DoD. DoD of a set of DFSs is used for differentiating a set of results. DFS of each result should be chosen to maximize the total DoD. The approach proves that the problem of generating optimal DFSs for a set of query results is NP-hard. It also put forwards two efficient optimal algorithms namely single-swap and multi-swap algorithms.

B. PROCESSING XML KEYWORD SEARCH BY CONSTRUCTING EFFECTIVE STRUCTURED SEARCH QUERIES: This approach designs an adaptive XML keyword search method, called XBridge [6]. The first step in this approach is to generate a set of structured queries by analyzing the given keyword query and the schema of XML data. This is done based on the labels and terms in the keyword query. A query can have a format of *l : k*, where *l* is the label and *k* is the term.

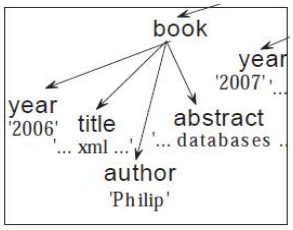

Figure 1: Example Master Entity [6]

```
For $b in bibliography/books/book
Where $b/year = '2006'
        and contains($b/title, 'xml')
        and contains($b/author, 'Philip')
Return $b
```

Figure 2: XQuery [6]

Consider a keyword query *q : (year : 2006; title : xml; author : Philip)*. For the given keyword query retrieve all the Master Entities [6]. Given a set of labels $\{l_i \mid 1 \le i \le n\}$ and XML schema tree *T*, the master entity is defined as the root node of the subtree T_{sub} of *T* such that T_{sub} contains at least one schema node labeled as $l_1 \ldots l_n$. Figure 1 shows a master entity for the given keyword query. Structured queries are produced corresponding to each master entity. Figure 2 shows the structured query in XQuery for the example Master Entity in Figure 1. As such let *QS* be the set of structured queries. Sort *QS* according to Context Score. Context Score is computed based on the distance from v_i to v_m. Where v_m be the master entity, v_i be the node having the matching label l_i. Evaluate structured queries $q_i \in QS$ with respect to its context score over the XML data and retrieve all the results. Compute the score function for each result and select those with highest score. The score is evaluated using the weight of individual keyword. As the size of the data set increases efficiency of this approach decreases [6]. No change in processing speed when size of the data set is less than 50M. Processing speed decreases when size of the data set increases above 50M. Processing speed decreases by 80% when data set size is 70M.

C. DIVQ, DIVERSIFICATION FOR KEYWORD SEARCH OVER STRUCTURED DATABASE: This approach also translates the given keyword query to a set of structured queries [7]. Each structured query is called a query interpretation. Given a keyword query, a data set can offer a broad range of query interpretations with various semantics. For example, consider a keyword query, *K: "CONSIDERATION CHRISTOPHER GUEST"* [7]. First step is

to translated it into a set of keyword interpretations such as *"director: CHRISTOPHER"*, *"director: GUEST"* and *"movie: CONSIDERATION"*. A keyword interpretation is a combination of attribute and keyword, $A_i : k_i$. Then join the keyword interpretations using a predefined query template which is a structural pattern that is frequently used to query the database [7]. Thus create a set of query interpretations. Then compute relevance score of each query interpretation *Q* with respect to the keyword query *K*. Relevance score can be computed by using the probability of generating keyword interpretation *I* and query template *T* for the given keyword query *K*. Select the query interpretation with maximum relevance score and add it to a set *QI*. The following query interpretations are added to *QI* based on both its relevance and similarity to the query interpretations already in *QI*. This is by computing the Score function which uses the average similarity between *Q* and all the interpretations in *QI*. This approach defines a parameter λ to represent tradeoff between relevance and novelty of query interpretation. This approach uses nDCG, normalized Discounted Cumulative Gain for evaluation.

D. ON QUERY RESULT DIVERSIFICATION: Several existing methods for diversifying query results are evaluated in this approach. This approach presents a thorough experimental evaluation of the various diversification techniques implemented in a common framework. It also put forwards two new methods, namely the Greedy with Marginal Contribution (GMC) and the Greedy Randomized with Neighborhood Expansion (GNE). This approach states the result diversification problem as a tradeoff between finding relevant, i.e., similar, elements to the query and diverse elements in the result set [8]. Both methods use maximum marginal contribution (mmc) to measure diversification. The GMC method selects the element with the highest maximum marginal contribution (mmc) and builds the result set R incrementally by considering the solution constructed so far. GNE is a randomized approach proposed for the diversification problem. The GMC algorithm always selects an element with highest mmc value to be added to the solution set. But GNE is a randomized approach that always chooses an element randomly among the top ranked ones to be included in the solution.

III. OBSERVATION AND ANALYSIS

Table 1: Advantages and disadvantages of different diversification approaches

Approaches	Query structuring	Result formulation	Context based	Depends on
Structured search result differentiation	Not needed	Post processing	No	XML metadata
Xbridge	Needed	Post processing	No	XML metadata
DivQ	Needed	Incremental way	No	XML metadata
GMC and GNE	Not needed	Incremental way	No	XML matadata

For the purpose of performance analysis, different methods for keyword query diversification are compared in Table 1. Need for query structuring, result formulation method and metadata dependency are taken as the parameters for comparison. The main disadvantage of the approach, Structured Search Result Differentiation, is that feature selection is limited by available metadata information of the XML data. It is a method of post-process search result analysis. That is, retrieving all the results and then eliminating the unqualified ones. XBridge includes generation and evaluation of a large number of structured queries. There is no guarantee that all the evaluated structured queries' results will be useful. The process of constructing structured queries has to rely on the metadata information in XML data. These are the main drawbacks of XBridge. In the case of DivQ, one of the main advantage is probability distribution function is used for score calculation. DivQ also considers similarity between the query interpretations while result formulation. The disadvantage with DivQ is, a large number of structured XML queries have to be generated and evaluated. The process of structured query construction relies on the metadata information of XML data. In DivQ two generated structured queries with slightly different structures may be considered as different search intentions. This will hurt the effectiveness of diversification. While with GMC and GNE, the main advantage is users can set the value for λ, the tradeoff between finding the most relevant and diverse elements. Both GMC and GNE construct the result set in an incremental way by using mmc. But the main disadvantage is that in most of the cases it will be difficult to set a threshold for the tradeoff value.

IV. CONTEXT-BASED DIVERSIFICATION FOR KEYWORD QUERIES OVER XML DATA

With this approach diversification is based on different contexts of the keywords, in the given keyword query, with respect to the XML data. For the given keyword query q, and the XML data to be searched, this approach consists of two steps [9].
1. Feature selection model: Derive the keyword search candidate for the query.
2. Keyword search diversification model: Computes top- k diversified query candidates of high relevance and novelty.
The corresponding SLCA results of top-k query candidates with high relevance and novelty retrieves optimal results of the keyword query.

A. FEATURE SELECTION MODEL: Feature selection model includes deriving the co-related feature terms for each keyword in the given keyword query, from XML data [9]. A search intention is a combination of the feature term and the original query keyword. The derived search intentions give the diversified context of the query keyword. Let *T* be the XML data and *W* be its relevance-based term-pair dictionary. In this approach, a term correlated graph, *G*, is precomputed before query evaluation. During the XML data tree traversal, extract the meaningful text information from the entity nodes in XML data. Produce a set of term-pairs by scanning the extracted text. After that, all the generated term-pairs will be recorded in the term correlated graph. Also record the count of each term-pair from different entity nodes. After completing the XML data tree traversal, the mutual information score for each term-pair can be computed. To reduce the size of correlation graph, the term-pairs with their correlation lower than a threshold can be filtered out. Mutual information is the quantity used to measure how much one random variable tells about other random variable [10]. It can be calculated by using probability distribution. The next step is to evaluate relevance of each derived search intentions to the original keyword query and the novelty of its produced results.

B. KEYWORD SEARCH DIVERSIFICATION MODEL: For a given keyword query and a set of derived search intentions, keyword search diversification model is to find top-k diversified query candidates. Each query candidates should be relevant and novel [9]. Relevance: The generated query q_{new} has the maximal probability of interpreting the contexts of original query q regarding to the data to be searched. Novelty: The generated query q_{new} has a maximal difference from the previously generated query set Q. After evaluating relevance and novelty, a scoring function can be computed as a product of relevance and novelty.

C. KEYWORD SEARCH DIVERSIFICATION: Approach: Anchor-based pruning method is designed to improve the efficiency of keyword search diversification by utilizing the intermediate results. The approach is capable of producing new and distinct SLCA result in each iteration.

Algorithm 1 Anchor-Based Pruning Algorithm

Input: q, T, G
Output: Q, top-k diverse query candidates and Φ, its SLCA results

```
for each keyword in q do
    retrieve feature-terms form G
end for
Generate a new query candidate q_new
Obtain the node list of q_new and compute the relevance of q_new
for each newly generated q_new do
    if there exist some previously generated SLCA results then
        Take it as anchors to partition the keyword node list of q_new
        Process the SLCA results of q_new
    else if
        thenCompute SLCA from Original node list of q_new
    end if
end for
Compute the Score() for q_new
if |Q| < k then
    Record q_new and Score(q_new) in to Q
    Record SLCA results of q_new in to Φ
else if
    thenCompare the score of q_new and previously generated query candidates
in Q and eliminate the unqualified ones
end if
Return Q and Φ
```

Anchor based pruning algorithm avoids the unnecessary computational cost of unqualified SLCA results, i.e., it avoids duplicates and ancestors. This method uses the notion of anchor nodes. Anchor nodes can be defined as, given a set of

query candidates Q that have been already processed and a new query candidate q_{new}, the generated SLCA results Φ of Q can be taken as the anchors for efficiently computing the SLCA results of q_{new} by partitioning the keyword nodes of q_{new}. The main advantages of this algorithm are the number of keyword nodes to be skipped increases with the increase in the number of anchor nodes. Each iteration will produce distinct SLCAs and there is no need for further comparison. This algorithm also reduces the computational cost by avoiding the computation of unqualified SLCAs.

V. PROBLEM IDENTIFIED WITH THE EXISTING SYSTEM

All the approaches works with purely text based XML data set. Anchor Based Pruning Algorithm provides diversification of keyword queries by considering different contexts of the keywords in the given XML data set. The data set used in all the methods does not include images and text formats like italics, bold etc. In XML images are represented by using the tag *<image> pathname </image>*, bold text is represented between ** and ** tags, and italics is represented between ** and ** tags. If data set contains these tags all the existing approaches are incapable of considering them as images and text formats. With slight modifications the approach can treat images, text formats and URLs differently. So that it can be retrieved as such, as search results. The context based method can be modified so that it can provide effective query suggestions to the user. The purpose of keyword query diversification is to provide the most relevant and diverse results to the user. User's satisfaction is guaranteed when diversification is based on different contexts of the keyword in XML dataset. These different contexts can be used to provide query suggestions.

VI. FUTURE SCOPE

The context based method can be modified that it can operate on datasets that are not purely text based. Developing a method that can extract styled contents and images from an image based XML dataset and retrieve them as such.

VII. CONCLUSION

Different approaches for keyword search diversification are described. Through comparative study identified context based diversification can deliver best query suggestion that it considers correlated feature terms. These correlated feature terms are selected based on their mutual information value. In this approach diversification is based on the context of the query keywords in the data. Diversification is measured with respect to their relevance to the original query and the novelty of the results. This approach can return top-k diversified query candidates and its corresponding SLCA results to user.

REFERENCES

[1] M. Hasan, A. Mueen, V. J. Tsotras, and E. J. Keogh, "Diversifying query results on semi-structured data," in Proc. 21st ACM Int. Conf. Inf. Knowl. Manag., pp. 20992103, 2012.
[2] Zhou, Junfeng, "Fast SLCA and ELCA computation for XML keyword queries based on set intersection," in 2012 IEEE 28th International Conference on Data Engineering, pp. 905-916. IEEE, 2012.
[3] Sun, Chong, Chee-Yong Chan, and Amit K. Goenka. "Multiway slca-based keyword search in xml data," in Proceedings of the 16th international conference on World Wide Web, pp. 1043-1052. ACM, 2007.
[4] L. Guo, F. Shao, C. Botev, and J. Shanmugasundaram, "Xrank: Ranked keyword search over xml documents," in Proc. SIGMOD Conf., pp. 1627, 2003.
[5] Z. Liu, P. Sun, and Y. Chen, "Structured search result differentiation," J. Proc. VLDB Endowment, vol. 2, no. 1, pp. 313324, 2009.
[6] J. Li, C. Liu, R. Zhou, and B. Ning, "Processing xml keyword search by constructing effective structured queries," in Advances in Data and Web Management. New York, NY, USA: Springer, pp. 8899, 2009.
[7] E. Demidova, P. Fankhauser, X. Zhou, and W. Nejdl, "DivQ: Diversification for keyword search over structured databases," in Proc. SIGIR, pp. 331338, 2010.
[8] M. R. Vieira, H. L. Razente, M. C. N. Barioni, M. Hadjieleftheriou, D. Srivastava, C. Traina J., and V. J. Tsotras, "On query result diversification," in Proc. IEEE 27th Int. Conf. Data Eng., pp. 11631174, 2011.
[9] Jianxin Li, Chengfei Liu, and Jeffrey Xu Yu, "Context-based diversification for keyword queries over XML data," IEEE Transactions on Knowledge and Data Engineering, VOL. 27, NO. 3 , MARCH 2015.
[10] C. O. Sakar and O. Kursum, "A hybrid method for feature selection based on mutual information" in Proc. 20th Int. Conf. Pattern Recognit., pp. 4360-4363, 2010.