Keyword Query Diversification Based On Context: A Survey

[1] **Mrudul Arkadi,** [2] **P. B. Mali**

[1, 2] Department of Computer Engineering,
Smt Kashibai Navale College of Engineering, Maharashtra Pune, India

Abstract - Keyword search has received a lot of awareness in the database section as it is constructive approach for querying a database preferably no need of knowing its underlying schema. Though, keyword search queries frequently return multiple results. One merit outcome is to rank results such that the one best result will appear first. Still, this approach can suffer from redundancy and ambiguity problem. So there is need of automatic diversification of search results based on context or surroundings. There are two algorithms which are efficient, namely baseline and anchor pruning analysis results into top-k efficient search result collaborating feature selection & keyword diversification model. Efficiency measures are listed that evaluate effectiveness of search result.

Keywords - *XML, Mutual Information, Baseline Solution, Anchor, n-DCG.*

1. Introduction

Well Proven and most probably used popular mechanism for querying document systems and the WWW (World Wide Web) is Keyword search. It is commonly applied to extract useful and relevant data through Internet. [1] Now a days Searching for the information is quite necessary. There are giant groupings of structured & semi-structured data, on the web & in the in enterprises, like relational databases, XML data extracted from text documents, workflows etc.[2] User need to acquire knowledge about structured query languages like SQL & XQuery to have access to these resources . Making database searchable will increase the data size that a user can have access and to give search results with improved quality, which is analyzed by the keyword search on textual documents, and thus increase the database usability. That is why keyword search on structured data is emerging attracted topic recently [2]. If more keywords are present in query it becomes easy to guess user intension behind their search. There is not even a single efficient exposition of a query is available which can satisfy each and every user. Hence, multiple interpretations may produce overlapping results. [11] Sometimes user involvement helps to get correct path of search but it takes lots of time as per result size. [4] Demanding situation is when user enters imprecise unclear small number of keywords as query. To deal with this, paper gives a method of providing diverse query suggestions to users based on context of given keywords in information to be searched. So, user may choose exact query or change it. [10]

2. Literature Survey

In Xrank [3] it is able to accept both XML & HTML documents and by using ranking scheme similar to pagerank scheme of google [2] they evaluate ranked results. XRANK takes into account .XML structure like hierarchy & hyperlink and a 2-D (two-dimensional) notion of keyword proximity, while computing the ranking for a keyword search XML queries.

XRANK a generalizing system that derives the basic keywords from the Hyperlinks searched in search results of search engines XRANK can query over a mix of HTML & XML documents. But in paper have currently taken a document-centric view, where they assume that query results are strictly hierarchical. We have to focus some parts like computing cost, unclear and repetitive result due to ranking scheme.

In diversifying search results [11] a tabular format is used to provide the data for these relationships. A order of parent/child tables contributes to its taxonomy. The external tables can store the actual values. Whenever, taxonomy of information is present in the document, the problem like unclear queries may arrive. They present a approach to give diversifying results that points to minimize the risk of un-satisfaction of the user. However, it is hard to get this useful taxonomy and query logs. In addition, the differentiated outcomes in IR are often formed at document levels. [10][11]

Table 1: Survey Table

Sr. No	Paper Title	Techniques and methods used	Findings
1	Xrank: Rank keyword search over xml documents	ElemRank , Hybrid Dewey Inverted List	Accept both XML and HTML documents, Only Document Centric Assumption of query results being strictly hierarchical
2	DivQ: Diversification for keyword search over structured databases	Greedy algorithm, taxonomy	Intent aware metrics, Greedy algorithm for approximation of specifically ambiguous queries, Hard to get query log & taxonomy
3	Processing xml keyword search by constructing effective structured queries	Probabilistic model	Considered only structured database, Used scheme to balance the relevance and novelty of keyword search, Probabilistic model of reranking
4	Diversifying search results	Scoring function	Adaptive XML keyword search, Derive semantics of keyword query, Scoring function

DivQ presents Construction of structured queries candidates relates the problem of intent based keyword query diversification. [6]. Their brief idea is to first map each keyword to a group of attributes (metadata), and then construct a giant number of structured query applicants by merging the attribute-keyword pairs. They assume that each structured query applicant represents a type of search motive i.e. a query interpretation. However, these works hard to be applied in real application due to the following limitations It may generate & evaluate a giant number of the structured XML queries and there is not even a assurance that this structured queries to be evaluated can find matched results due to the structural constraints. With the reference of [11] XBridge is approach for adaptive keyword search over XML. XBridge can acquire the explanations for a keyword query & it can give group of structured queries. This is done by analyzing keyword query & XML schemas. Limitation of this is, metadata information of XML is responsible for constructing structured queries procedure & in XBridge giant number of queries get generated & evaluated so overhead of query logs and it work only on structural constraints.

3. Proposed Architecture

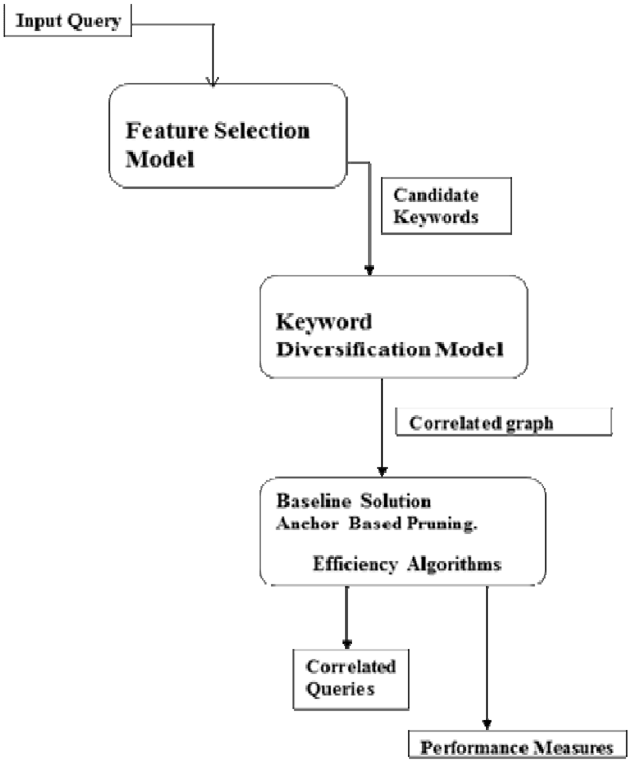

Figure 1: System Architecture

3.1. Model of Feature Selection

As we need high relevance and maximal diversification top-k queries from XML data our first model is selection of feature. The main problem for the pattern classification system is the feature selection. Mutual information used with feature variables reduces the redundancy and improve the classification accuracy. [7] Relevance based term-pair dictionary is prepared based on mutual information. If terms x & y co-occurring in result set and if terms x & y are not dependent i.e. Knowing result x will not be useful for knowing any information about result y & vice versa. Hence total collective information will be zero. But, if the results x,y are same, then it is trouble free to find out value of x by only knowing value of the y & vice versa. Hence, the simple measure used to evaluate how much the observed word co-occurrences magnify the dependency of feature terms while reduce the redundancy of feature terms.[11]We get matrix of features for query keywords using the term-pairs in dictionary. Using the popularly accepted mutual information model as follows: [10]

$$MI(x,y,T)=Prob(x,y,T)*Log\,({}^{prob(x,y,T)}/{}_{Prob(x,T)*Prob(y,T)})\ldots(1)$$

3.2. Keyword Diversification Model

We get candidate queries from model of feature selection & that transferred score of a mutual information based candidate queries we use in keyword diversification model. According to Bayes theorem:

$$Prob(q_{new}|q,T)= {}^{Prob(q|q_{new},T)} * {}^{Prob+(q_{new}|T)} / {}_{Prob(q|T)})\ldots.(2)$$

Where Prob(q_{new}|q,T) models the likelihood of generating the observed query q while the intended query is actually q_{new} and ${}^{Prob(q_{new}/T)}$ is the probability of query generation. To avoid overlapped result sets novelty which we take into consideration. SLCA i.e. Smallest and Lowest Common Ancestors) for a tree, if a Node is taken as an SLCA result, then its ancestor Nodes cannot become the SLCA results. It allows purifying the diversified results into more specific ones when we incrementally deal with more query candidates. [9, 4, 8] Through extraction of feature terms we first measure correlation of each pair of terms using model of a mutual information in equation (1) then correlation values are arranged and maintained by graph based arrangement before processing of queries that means correlated graph is pre-computed.

3.3. Keyword Diversification Model

3.3.1 Baseline Solution

a. User entered query
b. Feature extraction
c. remove stop words
d. Measure correlation of each pair (information recorded in form of metric)
e. Correlated graph of values is maintained Offline
f. Calculate score of a mutual information
g. Evaluate all the relevant & irrelevant data to get To k relevant queries
h. compare results[10]

Retrieve the relevant feature terms with high mutual scores from the term correlated graph of the XML data. Then generate query candidates list, that are sorted in order of descending total mutual scores and finally compute the SLCAs of all the relevant & irrelevant data also we have to find out and remove the same or ancestor SLCA results as keyword search. These results for each query candidate and measure its diversification score. As such, the top-k diversified query candidates and their co-related results chosen and returned.

3.3.2 Anchor-Based Pruning Solution

Main cost of Base Line Solution Algorithm is spent on computing SLCA and discarding unqualified SLCA results from result sets. To reduce this computational cost we are proposing Anchor Based Pruning Solution. In selection of anchor nodes, a group of query candidate's Q and a new query candidate q_{new} is given. The generated SLCA results considered as the Anchor nodes. Necessary steps for algorithm to work are as follows, we are having anchor Node (Va) and a new query candidate. Its Candidate Keyword node lists which will be sub-divided into four areas to be anchored by Va.

- Nodes of keyword
- which are the ancestors of Va, denoted as LVa-anc;
- Nodes of keyword which are the previous siblings of Va, denoted as LVa-pre.
- Nodes of keyword which are the descendants of Va, denoted as LVa-des;
- Nodes of keyword which are the next siblings of Va, denoted as LVa-next.

We have that LVa-anc will not generate any new result. Each of the other three areas may generate new and distinct SLCA results individually. No new and distinct SLCA results generated across the areas. Consider two SLCA results X1 and X2 (assume X1 precedes X2) for the current query set Q. For the next query q_{new} = {s1; s2; s3} and its keyword instance lists L = { ls1; ls2; ls3}, the keyword instances in L will be separated into four areas by the anchor X1:

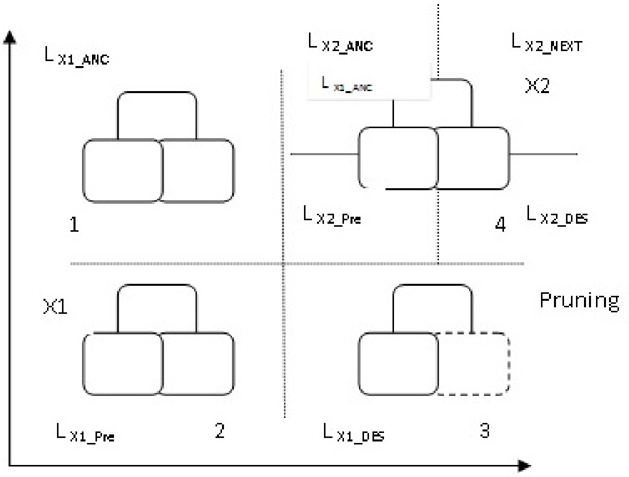

Figure 2: Anchor Nodes

For the next query q_{new} = {s1; s2; s3} and its keyword instance lists L = {ls1; ls2; ls3}, the keyword instances in L will be separated into four areas by the anchor X1:

1) LX1anc in which all nodes of keyword are the ancestor of X1, so LX1anc cannot generate new and distinct SLCA results.

2) LX1 pre in which all nodes of keyword are the previous siblings of X1 so we may generate new SLCA results if the results are still bounded in the area.

3) LX1 des in which all nodes of keyword are the descendant of X1 so it may produce new SLCA results that will replace X1;

4) LX1 next in which all nodes of keyword are the next siblings of X1 so it may produce new results, but it may be further sub-divided by the anchor X2. If there will be no intersection of all keyword node lists in an area, then the nodes in this area can be pruned directly e.g., l3s1 and l3s2 can be pruned without computation if l3s3 is empty in Lx1 des. Similarly, we can process LX2 pre, LX2 des and LX2 next. After that, a group of new and distinct SLCA results obtained with regards to the new query set Q S $q_{new.}$

3.3.3 Performance Measures

Performance is all about to calculate how much relevant each result is w.r.t. search query. Also search time is important measure. For that use of Discounted cumulative gain (DCG) is a ranking quality measure. In information repossession, it is used measure correctness of web search engine algorithms or related applications. Depending on the search query, the list of result varies in length. The first step in the nDCG computation is formation of a gain vector G. The gainG[k] at rank k measured as the relevant result. The gain may be lessen with increasing rank, to penalize documents that rank low, reflecting the additional user effort. The discounted gain is collected over k to obtain the DCG (Discounted Cumulative Gain) value and normalized using the ideal gain at rank k to finally obtain the nDCG value.[11]**.**

Differentiation of Diversified queries is evaluated through method like if the specific keyword query occurs at the top 1-5 positions in its corresponding suggestion list, then the functionality of the diversification model is marked by 1.If its placed at the top 6-10 positions, then the usefulness is marked by 0.5. If it's placed at the position range 11-20, then the usefulness is marked as 0.25. Otherwise, the suggestion is unusual for users because it is seldom for users to check above 20 suggestions in a sequence, i.e., the usefulness is labeled as zero. [10]

4. Conclusion

This Paper provides survey based on keyword search diversification, when ordinary user search for keyword queries it is difficult to find appropriate search solution, especially when search query is short and vague, it becomes ambivalent. Model of feature selection extracts features i.e. candidate keywords or term-pair. Diversification model works on the basis of Bayes theorem collaborates correlation graph of correlation values of term-pair based on mutual information model and Baseline and anchor pruning solutions leads to top-k search result. Search query is nothing but to find user intension human involvement sometimes help but it is time-consuming so methods like baseline solution and anchor pruning solution automatically diversifies search while considering performance measures. Context based classification makes user task easier as someone gets diversified result all related to someone's search intensions. Baseline and anchor based pruning algorithms diversify XML data & effectively measured through relevance measure and relations between different query results.

References

[1] G. Li, B. C. Ooi, J. Feng, J. Wang, and L. Zhou."EASE: an e®ective 3-in-1 keyword search method for unstructured, semi-structured and structured data." In SIGMOD, 2008.

[2] Y. Chen, W. Wang, Z. Liu, and X. Lin, "Keyword search on structured and semi-structured data," in Proc. SIGMOD Conf., 2009, pp. 1005–1010.

[3] L. Guo, F. Shao, C. Botev, and J. Shanmugasundaram, "Xrank: Rank keyword search over xml documents," in Proc. SIGMOD Conf., 2003, pp. 16–27.

[4] C. Sun, C. Y. Chan, and A. K. Goenka, "Multiway SLCA-based keyword search in xml data," in Proc. 16th Int. Conf. World Wide Web, 2007, pp. 1043–1052.

[5] Angel and N. Koudas, "Efficient diversity-aware search," in Proc. SIGMOD Conf., 2011, pp. 781–792.

[6] E. Demidova, P. Fankhauser, X. Zhou, and W. Nejdl, "DivQ: Diversification for keyword search over structured databases," in Proc. SIGIR, 2010, pp. 331–338

[7] 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, 2009, pp. 88–99.

[8] H. Peng, F. Long, and C. H. Q. Ding, "Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy," IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 8, pp. 1226–1238, Aug. 2005.

[9] C. Sun, C. Y. Chan, and A. K. Goenka, "Multiway SLCA-based keyword search in xml data," in Proc. 16th Int. Conf. World Wide Web, 2007, pp. 1043–1052.

[10] Jianxin Li, Chengfei Liu, Member, IEEE, and Jeffrey Xu Yu, Senior Member, IEEE," Context-Based Diversification for Keyword Queries Over XML Data" IEEE transactions on knowledge and data engineering, VOL. 27, NO. 3, MARCH 2015

[11] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong, "Diversifying search results," in Proc. 2nd ACM Int. Conf. Web Search Data Mining, 2009, pp. 5–14.