Perspective Based Diversification for Keyword Queries' Over XML data

Mr.T.Ravi Kumar[1], V.Chandana Swetha[2,] K.Harika[3], Ch.Gayathri Sravya[4]

[1]Assistant professor, Andhra Loyola Institute Of Engineering & Technology, Vijayawada.
[2,3,4] B.Tech, Andhra Loyola Institute Of Engineering & Technology, Vijayawada.

Abstract*—while keyword query permits ordinary users to search large amount of data, the uncertainty of keyword query makes it difficult to adept answer keyword queries, especially for short and indefinite keyword queries. To address this challenging problem, in this paper we come up with an approach that automatically diversifies XML keyword search based on its dissimilar contexts in the XML data. Given a short and indefinite keyword query and XML data to be searched, we first attain keyword search candidates of the query by a easy feature selection model. And then, we design an productive XML keyword search diversification model to measure the quality of each candidate. After that, two efficient algorithms are estimated to incrementally calculate top-k qualified query candidates as the diversified search intentions. Two selection criteria are focused: the k selected query candidates are most relevant to the given query while they have to cover large number of different results. At last, a comprehensive verification on real and fake data sets demonstrates the productiveness of our proposed diversification model and the productiveness of our algorithms.*

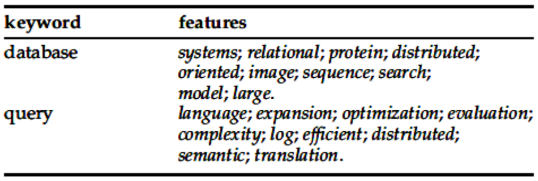
Index Terms—XML keyword search, context-based diversification

INTRODUCTION

As we know that keyword query search on structured and semi-structured data showed much research interest recently, as it is easy to retrieve information without learning of sophisticated query languages and database structure [1]. When we Compared with keyword search methods in information retrieval (IR) they mainly prefer to find a list of relevant documents, keyword search approaches in structured and semi structured data concentrate more on specific information contents, e.g., fragments rooted at the smallest lowest common ancestor nodes of a given keyword query in XML. Given a keyword query, if 1) the sub tree which is rooted at the node v contains total keywords, and 2) there does not exist a descendant node v0 of v such that the sub tree rooted at v0 contains total keywords. In other words, if a node is an SLCA, then its ancestors will be definitely excluded from being SLCAs, by which the minimal information content with SLCA semantics can be used to represent the specific results in XML keyword search. In this paper, we adopt the well-accepted SLCA semantics [2], [3], [4], [5] as a result metric of keyword query over XML data. In general, the more number of keywords for user's query contains, then it is easy to find the user's intention with regards to the query can be identified. However, when the given Keyword query only contains a small and vague keywords, it would become a very difficult problem to find the user's search intention due to the high ambiguity of this type of keyword queries. Although sometimes user involvement is required to find search intentions of keyword queries, when the user involvement is taken it is a time consuming process when the size of relevant result set is large. To address this type of problems we will develop a method which provides diverse keyword query suggestions to the user based on the context of the given keywords in the data which is to be searched. By doing this, users may choose their preferred queries or modify their original queries based on the returned diverse query suggestions.

Example 1. First Consider a query q ={database, query} over the DBLP data set. There are 21,260 publications or venues having the keyword "database", and 9,896 publications or venues containing the keyword "query", which contributes 2,040 results that contain the two given keywords together. When we directly try to read the keyword search results, it would become time consuming and not user friendly due to the large number of results. It takes 54.22 s for just calculating all the Smallest Lowest Common Ancestor results of query q by using XRank [2]. Even if the system processing time is acceptable by accelerating the keyword query evaluation using efficient algorithms [3], [4], the unclear and frequent search intentions in the large set of retrieved results will make users frustrated. To overcome from this problem, we will find different search semantics of the original query from the different contexts of the XML data, which can be used to examine different search intentions of the original query. In this study, the contexts can be modelled by extracting some relevant feature terms of the query q keywords from the XML data, as shown in the below Table 1.Then, we can calculate the keyword search results for each search intention. Table 2 shows that part of statistic information of the answers which are related to the keyword query, which classifies each ambiguous keyword query into different search intentions. The problem of diversifying keyword search is firstly studied in Information Retrieval (IR) community [6], [7], [8], [9], [10]. Most of them perform diversification as a reranking or post-processing step of document retrieval based on the analysis of result set and/or the query
logs

TABLE 1
Top 10 Selected Feature Terms of *q*



keyword	features
database	*systems; relational; protein; distributed; oriented; image; sequence; search; model; large.*
query	*language; expansion; optimization; evaluation; complexity; log; efficient; distributed; semantic; translation.*

.

In IR(Information Retrieval), keyword search diversification is done at the topic level or document level. For e.g., Agrawal et al. [7] model user intents at the topical level of the taxonomy and Radzinski and Dumas [11] to obtain the possible number of query intents by mining of query logs. However, it is not always easy to get these useful taxonomy and query logs. In addition, the diversified results in Information Retrieval are often designed at the document level. To improve the precision of query diversification on both structured databases and semi structured data, it is desirable to consider both structure and content of data in diversification model. So the problem of keyword search diversification is needed to be reconsidered in structured databases or semi structured data. Liu et al. [12] is the first work to measure the difference of XML keyword search results by comparing their feature sets. However, the selection of feature set in [12] is limited to metadata in XML and it is also a method of post-process search result analysis. Different from the above post-process methods, another type of works addresses the problem of intent-based keyword query diversification through constructing structured query candidates [13], [14]. Their brief idea is to first map each keyword to a set of attributes (metadata), and then construct a large number of structured query candidates by merging the attribute-keyword pairs. They assume that each structured query candidate represents a type of search intention, i.e., a query interpretation. However, these works are not easy to be applied in real application due to the following three limitations:

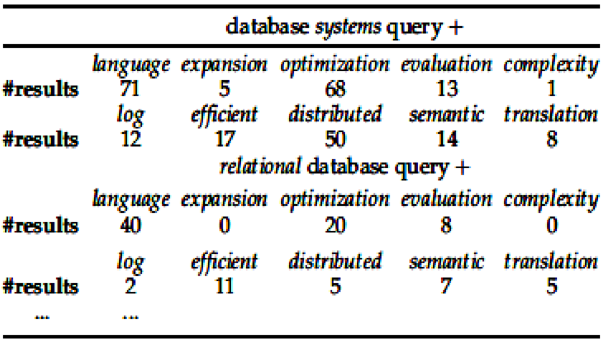
- A huge number of structured XML queries may be created and evaluated;
- There is no guarantee that the structured queries to be evaluated can find matched results due to the structural constraints;
- Similar to [12], the process of constructing structured queries has to depend on the metadata information present in the XML data.

To overcome from all the above problems, we implement a diversification problem in XML keyword search so, that it will directly find the diversified results without retrieving the relevant results .to reach this goal we will take a keyword query based on MI(Mutual Information) score using Simple Feature Selection model[15],[16].in this model we will find correlated feature terms in the probability theory the selection of co-related feature terms are not limited to the labels of XML elements . It also finds the feature terms for each combination of feature terms on original query keyword may represent one of diversified contexts. In this we will find the context of the keyword based on the relevance of the original query and the novelty of its results. To efficiently calculate the diversified results for keyword search we will us1e 2 algorithms 1)Baseline algorithm 2)Anchor based pruning algorithm which is improved algorithm based on observed results.

2 PROBLEM DEFINITION

Consider a keyword query q and XML data T, and over goal is to find the top-k query candidates based on high relevance and maximal diversification for q in T. Finally, each query candidates denotes a context or a search intention of query q in an XML data T.

TABLE 2
Part of Statistic Information for q



		database *systems* query +			
	language	*expansion*	*optimization*	*evaluation*	*complexity*
#results	71	5	68	13	1
	log	*efficient*	*distributed*	*semantic*	*translation*
#results	12	17	50	14	8
		relational database query +			
	language	*expansion*	*optimization*	*evaluation*	*complexity*
#results	40	0	20	8	0
	log	*efficient*	*distributed*	*semantic*	*translation*
#results	2	11	5	7	5
...	...				

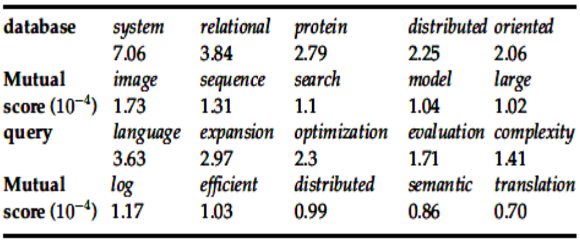
2.1 Feature Selection Model

In this model using of distinct term pairs we will find the MI(Mutual Information) score [15],[16].to find this MI Score Consider an XML data T and their relevance based on term pairs using the dictionary W mainly depends on the application context and it will not affect an subsequent discussion.

MI score can also use to characterize both the relevance and redundancy of the variables, such as the minimum redundancy feature selection

Now assume that we have an XML tree T and sample result R(T) and Prob(x,T) is the probability of terms x which is appearing in R(T).Here Prob(x,T)=R(x,T)/R(T) where R(x,T) is the number of results containing x in T and Prob(x,y,T) represents probability of x and y co-occurring terms in R(T) which means Prob(x,y,T)=R(x,y,T)/R(T)

TABLE 3
Mutual Information Score w.r.t. Terms in q



database	*system*	*relational*	*protein*	*distributed*	*oriented*
	7.06	3.84	2.79	2.25	2.06
Mutual	*image*	*sequence*	*search*	*model*	*large*
score (10^{-4})	1.73	1.31	1.1	1.04	1.02
query	*language*	*expansion*	*optimization*	*evaluation*	*complexity*
	3.63	2.97	2.3	1.71	1.41
Mutual	*log*	*efficient*	*distributed*	*semantic*	*translation*
score (10^{-4})	1.17	1.03	0.99	0.86	0.70

If terms x and y are said to be independent, if x does not give any information about y and y does not give any information about x,at that case their mutual information is said to zero. ifterms x and y are similar, then x determines the value of y and y determines x. This shows, the simple measure can be used to quantify how much the produced word co-occurrences maximize the dependency of the feature terms while reduce the redundancy of feature terms. In this work, we use the following method to measure mutual information score:

$$MI(x,y,T) = Prob(x,y,T) * log\frac{Prob(x,y,T)}{Prob(x,T) * Prob(y,T)}. \qquad (1)$$

For each and every term in the XML data, we have to find a set of feature terms where the feature terms can be selected in any way, for e.g., top-m feature terms or their feature terms with mutual values greater than a given threshold based on domain applications or data administrators. The feature terms can be computed before and stored the procedure of query expansion. So that, for a given keyword query, we can get a matrix of features for the query keywords q using the term-pairs in dictionary W. The matrix represents a place to search intentions (query candidates) of the real query w.r.t. the XML data T. Therefore, our problem is to select a subset of query candidates, which has the highest probability of interpreting the contexts of real query. In this, finding of the query candidates are depends on an approximate sample at the entity level of XML data T.

2.2 Keyword Search Diversification Model

In this model, we not only give importance to new generated queries but also gives importance to both relevance and novelty. relevance indicates about new results whereas novelty indicates distinct results . two criteria are targeted for both relevence and novelty 1) generated query qnew has maximal probability to interpret the contexts of original query with regards to the data to be searched 2) the generated query qnew should have maximal difference from the previously generated query set Q. Such that, we have the qnew scoring function

$$score(q_{new}) = Prob(q_{new} \mid q, T) * DIF(q_{new}, Q, T), \qquad (2)$$

Where Prob(qnew/q,T) indicates probability that qnew is the search intention when the real query is issued over the data; DIF (qnew,Q,T)indicates percentage of results that are produced usingqnew, but not by any before generated query in Q.

2.1 Evaluating the Probabilistic Relevance of an Intended Query Suggestion w.r.t. the Original Query

From Bayes Theorem, we have

$$Prob(q_{new} \mid q, T) = \frac{Prob(q \mid q_{new}, T) * Prob(q_{new} \mid T)}{Prob(q \mid T)}, \qquad (3)$$

WhereProb(q/qnew,T)models the generatingthe observed query q,during the intended query is qnew, and Prob(qnew /T)indicates the query generation probability in thegiven XML data T.

To deal with multiple keyword queries, we will make the independence consideration based on the probability that fiji ,is the intended feature of the query keyword ki. That is,

$$Prob(q \mid q_{new}, T) = \prod_{k_i \in q, f_{ij_i} \in q_{new}} Prob(k_i \mid f_{ij_i}, T). \qquad (4)$$

From the statistical sample information, the intent of a keyword can be inferred from the occurrences of the keyword and its correlated terms in the data T. Thus, we can compute the value ofProb (ki / fiji , T) of interpreting a keyword ki into a search intent fijias follows:

$$\begin{aligned} Prob(k_i \mid f_{ij_i}, T) &= \frac{Prob(f_{ij_i} \mid k_i, T) * Prob(k_i, T)}{Prob(f_{ij_i}, T)} \\ &= \frac{|R(\{k_i, f_{ij_i}\}, T)| / |R(T)|}{|R(f_{ij_i}, T)| / |R(T)|} \qquad (5) \\ &= \frac{|R(s_i, T)|}{|R(f_{ij_i}, T)|}, \end{aligned}$$

where $s_i = \{k_i, f_{ij_i}\}$.

For example Consider a query q ={database, query} and a querycandidate qnew={database system; query expansion}.Prob(q/qnew,T)represents the probability of a publicationthat shows that problem of "database query" regarding the context of "system and expansion", which can becalculated by using

$$\frac{|R(\{\text{database system}\}, T)|}{|R(\text{system}, T)|} * \frac{|R(\{\text{query expansion}\}, T)|}{|R(\text{expansion}, T)|}.$$

. Here,R({database system},T)indicates the number of keywordsearch results of query q over thedata T. |R(system,T)|indicates the number of keywordsearch results of current query system on the data T,but the number can be obtained without presence of current querysystem because it is equal to the size of keyword nodelist of "system" over data T. Similarly, we can also find thevalues of |R({query expansion},T)and |R(expansion , T)|. In this work, we take the huge number of acceptedsemantics—SLCA to design XML keyword search results.Consider an XML data T, the query generation probabilityof qnew can be calculated by using of the following equation:

$$Prob(q_{new} \mid T) = \frac{|R(q_{new}, T)|}{|R(T)|} = \frac{\left|\bigcap_{s_i \in q_{new}} R(s_i, T)\right|}{|R(T)|}, \qquad (6)$$

where: $\bigcap_{s_i \in q_{new}} R(s_i, T)$ is the set of SLCA results by combining the node lists $R(s_i, T)$ for $s_i \in q_{new}$ using the XRank algorithm in [4] which is a popular method to computing the SLCA results by visiting of the XML tree only once.

Given a q(query) and the XML data T, the value $\frac{1}{Prob(q \mid T)}$ Is notchanged value w.r.t different generatedquery candidates. Therefore, from Equation (3)we can be rewritten as :

$$Prob(q_{new} \mid q, T) = \gamma * \left(\prod \frac{R(s_i, T)}{R(f_{ii_i}, T)} \right) * \frac{\left|\bigcap R(s_i, T)\right|}{|R(T)|}, \qquad (7)$$

where

$k_i \in q,\ s_i \in q_{new},\ f_{ij_i} \in s_i$ and $\gamma = \frac{1}{Prob(q \mid T)}$

can be ranged in (0, 1] because it does not affect the expanded query candidates w.r.t. an original keywordq(query)and data T. Though the above equation can model the probabilities of generated query candidates (i.e., the relevance amongthese query candidates and the original query w.r.t. the data), different query candidates may have overlapped result sets. So, we should also take into account the novelty of results of the query candidates.

2.2.2 Evaluating the Probabilistic Novelty of an Intended Query w.r.t. the Original Query

As we know, the important property of SLCA semantics is exclusivity, i.e., if a node is taken as an SLCA result, then the ancestor nodes cannot become SLCA results. Because of this exclusive property, the process of evaluating the novelty for a newly generated query candidate qnew depends on the evaluation of the other previously generated query candidates Q. Hereby, the novelty DIF(qnew,Q,T)of qnew against Q can be calculated as follows:

$$DIF(q_{new}, Q, T) = \frac{|\{v_x \mid v_x \in R(q_{new}, T) \wedge \nexists v_y \in \{\bigcup_{q' \in Q} R(q', T)\} \wedge v_x \leq v_y\}|}{|R(q_{new}, T) \bigcup \{\bigcup_{q' \in Q} R(q', T)\}|}, \qquad (8)$$

hereR(qnew,T)represents the set of SLCA results generated by qnew; $\bigcup_{q' \in Q} R(q', T)$ repress the set of SLCAresults generated by queries in Q, which donot include the duplicate and ancestor nodes $v_x \leq v_y$ thatmeansvx is aduplicate of vy for “=”, or vx is an ancestor of vy for “<”; $R(q_{new}, T) \bigcup \{ \bigcup_{q' \in Q} R(q', T) \}$ is an SLCA result set that satisfieswith the exclusive property. By performing this, we can avoid presenting overlappedSLCA results to users. In other means, the considerationof novelty allows us to incrementally refine thediversified results into more specific ones when we incrementallydeal with more query candidates. Our main problem is to find top k qualified querycandidates and their relevant SLCA results. To do this, wecan compute the exact score of the search intention for each generated query candidate.However,to reduce thecalculating cost, an alternative way is to calculate the relativescores of queries. Therefore, we have the followingequation transformation. After we use the Equations (7)and (8) into Equation (2), we have the final equation

$$\begin{aligned} score(q_{new}) &= \gamma * \prod \left(\frac{R(s_i, T)}{R(f_{ij_i}, T)} \right) * \frac{|\bigcap R(s_i, T)|}{|R(T)|} \\ &* \frac{\left| \left\{ v_x \,|\, v_x \in R(q_{new}, T) \wedge \nexists v_y \in \left\{ \bigcup_{q' \in Q} R(q', T) \right\} \wedge v_x \leq v_y \right\} \right|}{\left| R(q_{new}, T) \bigcup \left\{ \bigcup_{q' \in Q} R(q', T) \right\} \right|} \\ &= \frac{\gamma}{|R(T)|} * \prod \left(\frac{R(s_i, T)}{R(f_{ij_i}, T)} \right) * |\bigcap R(s_i, T)| \\ &* \frac{\left| \left\{ v_x \,|\, v_x \in R(q_{new}, T) \wedge \nexists v_y \in \left\{ \bigcup_{q' \in Q} R(q', T) \right\} \wedge v_x \leq v_y \right\} \right|}{\left| R(q_{new}, T) \bigcup \left\{ \bigcup_{q' \in Q} R(q', T) \right\} \right|} \quad (9) \\ &\mapsto \prod \left(\frac{R(s_i, T)}{R(f_{ij_i}, T)} \right) * |\bigcap R(s_i, T)| \\ &* \frac{\left| \left\{ v_x \,|\, v_x \in R(q_{new}, T) \wedge \nexists v_y \in \left\{ \bigcup_{q' \in Q} R(q', T) \right\} \wedge v_x \leq v_y \right\} \right|}{\left| R(q_{new}, T) \bigcup \left\{ \bigcup_{q' \in Q} R(q', T) \right\} \right|}, \end{aligned}$$

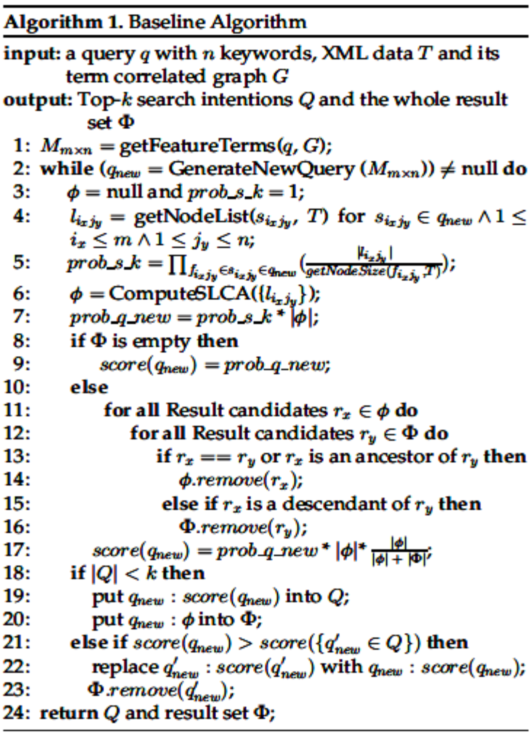
where $s_i \in q_{new}, s_i = k_i \cup f_{ij_i}, k_i \in q, q' \in Q$ a nd the symbolrepresents the left side of the equation depends on the right side of the equation because the value is not changed for calculating the diversification scores of different searchintentions.

3 EXTRACTING FEATURE TERMS

To address the problem of deriving meaningful featureterms w.r.t. an original keyword query, there are two relevant works [17], [18]. In [17], Sarkas et al. proposed a solution of producing top-k interesting and meaningful expansions to a keyword query by derivingk otherwords with high “interestingness” values. The expanded queries can be used to search more specific documents. The interestingness is formalized with the notion of surprise [19],[20], [21]. In [18], Bansal et al. proposed efficient algorithms to identify keyword clusters in large quantity of blog posts for specific temporal intervals. Our work combinesboth of their ideas together: we first measure the correlation of each pair of terms using our mutual information model in Equation (1), which is a simple surprise metric; and then we build terms correlated graph that contains all the terms and their correlation values. Different from [17], [18], our work makes use of entity-based sample information to build a correlated graph with high accurate for XML data. In order to efficiently measure the correlation of a pair of terms, we use a statistic method to measure how much the co-occurrences of a pair of terms deviate from the independentassumption where the entity nodes (e.g., the nodes with the “*” node types in XML DTD) are taken as a sample space. For instance, given a pair of terms x and y, their mutual information score can be evaluated based on Equation (1) where Prob(x; T)(or Prob(y,T) is the value of dividing the number of entities consistingx (or y) by the total entity size of the sample space; Prob(x,y,T)is the value of dividing the number of entities consisting both xand y by the total entity size of the sample space. In this work, we build a term correlated graph offline i.e we precompute it before processing queries. The correlation values among the terms are also recorded in the graph, which is used to generate the term-feature dictionary W. During this XML data tree traversal, we first obtain the meaningful text information from the entity nodes in XML data. Here, we would like to filter out the stop words. And then, we produce a set of term-pairs by scanning the obtained text. After that, all the generated term-pairs will be recorded in the terms of correlated graph. In this procedure of building correlation graph, we also record the count of each term-pair to be generated from various entity nodes. As such, after the XML data tree is traversed completely, we can calculate the mutual information score for each termpair based on Equation (1). To decrease the size of correlation graph, the term-pairs with their correlation less than athreshold can be filtered out. Based on the offline builtgraph, we can on-the-fly select the top-m different terms as its features for each given query keyword.

4 KEYWORD SEARCH DIVERSIFICATION ALGORITHMS

In this, we first propose a baseline algorithm toretrieve the diversified keyword search results. Then, two anchor-based pruning algorithms are designed to improve the efficiency of the keyword search diversification by making use of the intermediate



```
Algorithm 1. Baseline Algorithm
input: a query $q$ with $n$ keywords, XML data $T$ and its
        term correlated graph $G$
output: Top-$k$ search intentions $Q$ and the whole result
        set $\Phi$
$M_{m \times n}$ = getFeatureTerms($q$, $G$);
while ($q_{new}$ = GenerateNewQuery ($M_{m \times n}$)) $\neq$ null do
    $\phi$ = null and $prob\_s\_k = 1$;
    $l_{i_x j_y}$ = getNodeList($s_{i_x j_y}$, $T$) for $s_{i_x j_y} \in q_{new} \wedge 1 \leq i_x \leq m \wedge 1 \leq j_y \leq n$;
    $prob\_s\_k = \prod_{f_{i_x j_y} \in s_{i_x j_y} \in q_{new}} (\frac{|l_{i_x j_y}|}{getNodeSize(f_{i_x j_y}, T)})$;
    $\phi$ = ComputeSLCA($\{l_{i_x j_y}\}$);
    $prob\_q\_new = prob\_s\_k * |\phi|$;
    if $\Phi$ is empty then
        $score(q_{new}) = prob\_q\_new$;
    else
        for all Result candidates $r_x \in \phi$ do
            for all Result candidates $r_y \in \Phi$ do
                if $r_x == r_y$ or $r_x$ is an ancestor of $r_y$ then
                    $\phi.remove(r_x)$;
                else if $r_x$ is a descendant of $r_y$ then
                    $\Phi.remove(r_y)$;
        $score(q_{new}) = prob\_q\_new * |\phi| * \frac{|\phi|}{|\phi| + |\Phi|}$;
    if $|Q| < k$ then
        put $q_{new}$ : $score(q_{new})$ into $Q$;
        put $q_{new}$ : $\phi$ into $\Phi$;
    else if $score(q_{new}) > score(\{q'_{new} \in Q\})$ then
        replace $q'_{new}$ : $score(q'_{new})$ with $q_{new}$ : $score(q_{new})$;
        $\Phi.remove(q'_{new})$;
return $Q$ and result set $\Phi$;
```

4.2 Anchor-Based Pruning Solution

By using of this baseline solution, we can find that the main cost of this solution is spent for computing SLCA results by removing unqualified SLCA results from the newly and before generated result sets. To decrease the computing cost, we are mapping to design an anchor based pruning algorithm, which will avoid the unnecessary computing cost of unqualified results (i.e., duplicates and ancestors). In this, we first find the each term-pair to be created from various entity nodes. As after the XML data tree nodes are visited completely, we can compute the mutual information score for each and every term pair from Equation (1). To decrease the size of correlation graph,of the term-pairs with their correlation lower than a given threshold can be separated. Based on the offline built graph, we can on-the-fly select the top-m distinct terms as its features for every query keyword.

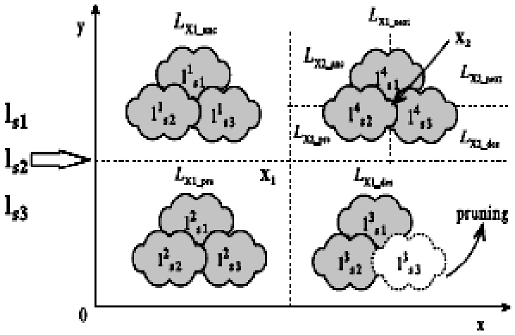
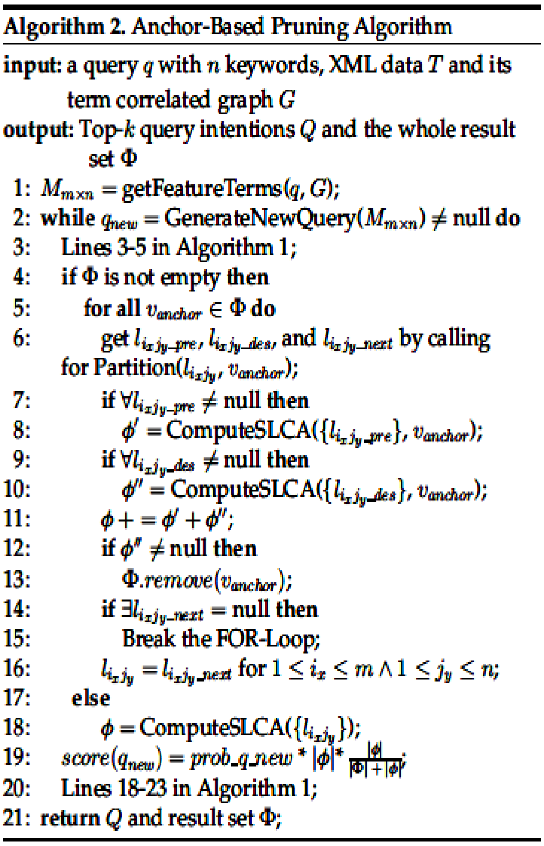


Fig. 1. Usability of anchor nodes.



```
Algorithm 2. Anchor-Based Pruning Algorithm
input: a query $q$ with $n$ keywords, XML data $T$ and its
        term correlated graph $G$
output: Top-$k$ query intentions $Q$ and the whole result
        set $\Phi$
$M_{m \times n}$ = getFeatureTerms($q$, $G$);
while $q_{new}$ = GenerateNewQuery($M_{m \times n}$) $\neq$ null do
    Lines 3-5 in Algorithm 1;
    if $\Phi$ is not empty then
        for all $v_{anchor} \in \Phi$ do
            get $l_{i_x j_y\_pre}$, $l_{i_x j_y\_des}$, and $l_{i_x j_y\_next}$ by calling
    for Partition($l_{i_x j_y}$, $v_{anchor}$);
            if $\forall l_{i_x j_y\_pre} \neq$ null then
                $\phi' =$ ComputeSLCA($\{l_{i_x j_y\_pre}\}$, $v_{anchor}$);
            if $\forall l_{i_x j_y\_des} \neq$ null then
                $\phi'' =$ ComputeSLCA($\{l_{i_x j_y\_des}\}$, $v_{anchor}$);
            $\phi += \phi' + \phi''$;
            if $\phi'' \neq$ null then
                $\Phi.remove(v_{anchor})$;
            if $\exists l_{i_x j_y\_next}$ = null then
                Break the FOR-Loop;
            $l_{i_x j_y} = l_{i_x j_y\_next}$ for $1 \leq i_x \leq m \wedge 1 \leq j_y \leq n$;
      else
            $\phi =$ ComputeSLCA($\{l_{i_x j_y}\}$);
    $score(q_{new}) = prob\_q\_new * |\phi| * \frac{|\phi|}{|\Phi| + |\phi|}$;
    Lines 18-23 in Algorithm 1;
return $Q$ and result set $\Phi$;
```

6 RELATED WORK

Diversifying results of document retrieval has been implemented[6], [7], [8], [9]. Most of them perform diversification as a re-ranking or post-processing step of document retrieval. Related work on result diversification in IR also includes [22], [23], [24]. Santos et al. [22] used probabilistic model to diversify document ranking, by which web search result diversification is introduced. They also used the similar model to discuss search result diversification through sub-queries which are in [23]. Gollapudi and Sharma [24] proposed a set of natural axioms that a diversification system is expected to satisfy, by which it will Improve user satisfaction with diversified results. Different From all the above relevant works, in this paper, our diversification model was created to process keyword queries over structured data. We have to consider the structures of data in our model and algorithms, not limited to pure text data like the above methods. Moreover, our algorithms can generate query suggestions and evaluate them. The diversified search results can be returned with the passed query suggestions not depending on the whole result set of the real keyword query. Recently, they also introduced some relevant work to communicate the problem of result diversification in structured data. For instance, [25]they also conducted clear experimental evaluation of the many diversification techniques implemented in a common framework and proposed a method based on threshold value to control the tradeoff between relevance and diversity features in their diversification metric. But it is a huge challenge for users to set the threshold value. Hasan et al. [26] developed efficient algorithms to find top-k most distant set of results for well organized queries over semi-structured data. As we know, a organized query can be used to express much more clear search intention of a user. Hence, diversifying structured query results is less significant than that of keyword search results. In [27], Panigrahi et al. focused on the selection of diverse item set, not considering structural relationships of the items to be selected. The most relevant work to ours is the approach DivQ in [13] where Demidova et al. first identified the attribute-keyword pairs for an original keyword query and then constructed a large number of structured queries by connecting the attribute-keyword pairs using the data schema (the attributes can be mapped to corresponding labels in the schema). The challenging problem in [13] is that to generated structured queries with slightly different structures may still be considered as different types of search intentions, which may hurt the effectiveness of diversification as shown in our experiments. However, our diversification model in this work utilized mutually co-occurring feature term sets as contexts to represent different query suggestions and the feature terms are selected based on their mutual correlation and the distinct result sets together. The structure of data are considered by satisfying the exclusive property of SLCA semantics.

7 CONCLUSIONS

In this paper , we used Feature Selection Model to find the diversification results of Keyword query q from xml data and then based on relevance and novelty we will measure the context of original query and results , After that we used two algorithms 1)Baseline Algorithm 2)Anchor-based Pruning algorithm to observe the properties of XML search results. Finally, we will perform comprehensive evaluation of real data and synthetic data is performed for the effectiveness of our diversification model by analyzing the returned search intentions for the given keyword queries over DBLP data set based on the nDCG measure and the possibility of diversified query suggestions. After that, we also finds that efficiency of our proposed algorithms by running of substantial number of queries over both XMark and DBLP data sets. By observing this results, we will get that our proposed algorithms can return qualified search intentions and results to users with in a short amount of time.

REFERENCES

[1] 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.

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

[3] 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.

[4] Y. Xu and Y. Papakonstantinou, "Efficient keyword search for smallest lcas in xml databases," in Proc. SIGMOD Conf., 2005, pp. 537–538.

[5] J. Li, C. Liu, R. Zhou, and W. Wang, "Top-k keyword search over probabilistic xml data," in Proc. IEEE 27th Int. Conf. Data Eng., 2011, pp. 673–684.

[6] J. G. Carbonell and J. Goldstein, "The use of MMR, diversity based reranking for reordering documents and producing summaries,"inProc. SIGIR, 1998, pp. 335–336.

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

[8] H. Chen and D. R. Karger, "Less is more: Probabilistic models forretrieving fewer relevant documents," in Proc. SIGIR, 2006,pp. 429–436.

[9] C. L. A. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A.Ashkan, S. B€uttcher, and I. MacKinnon, "Novelty and diversity in information retrieval evaluation," in Proc. SIGIR, 2008,pp. 659–666.

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

[11] F. Radlinski and S. T. Dumais, "Improving personalized web search using result diversification," in Proc. SIGIR, 2006, pp. 691–692.

[12] Z. Liu, P. Sun, and Y. Chen, "Structured search result differentiation,"J. Proc. VLDB Endowment, vol. 2, no. 1, pp. 313–324,2009.

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

[14] 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.

[15] 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.

[16] C. O. Sakar and O. Kursun, "A hybrid method for feature selectionbased on mutual information and canonical correlation analysis,"inProc. 20th Int. Conf. Pattern Recognit., 2010, pp. 4360–4363.

[17] N. Sarkas, N. Bansal, G. Das, and N. Koudas, "Measure-drivenkeyword-query expansion," J. Proc. VLDB Endowment, vol. 2,no. 1, pp. 121–132, 2009.

[18] N. Bansal, F. Chiang, N. Koudas, and F. W. Tompa, "Seeking stable clusters in the blogosphere," in Proc. 33rd Int. Conf. Very Large Data Bases, 2007, pp. 806–817.

[19] S. Brin, R. Motwani, and C. Silverstein, "Beyond market baskets:Generalizing association rules to correlations," in Proc. SIGMOD Conf., 1997, pp. 265–276.

[20] W. DuMouchel and D. Pregibon, "Empirical bayes screening for multi-item associations," in Proc. 7th ACM SIGKDD Int. Conf.Knowl. Discovery Data Mining, 2001, pp. 67–76.

[21] A. Silberschatz and A. Tuzhilin, "On subjective measures of interestingness in knowledge discovery," in Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 1995, pp. 275–281.

[22] R. L. T. Santos, C. Macdonald, and I. Ounis, "Exploiting query reformulations for web search result diversification," in Proc. 16th Int. Conf. World Wide Web, 2010, pp. 881–890.

[23] R. L. T. Santos, J. Peng, C. Macdonald, and I. Ounis, "Explicit search result diversification through sub-queries," in Proc. 32nd Eur. Conf. Adv. Inf. Retrieval, 2010, pp. 87–99.

[24] S. Gollapudi and A. Sharma, "An axiomatic approach for result diversification," in Proc. 16th Int. Conf. World Wide Web, 2009, pp. 381–390.

[25] 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., 2011,pp. 1163–1174.

[26] 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., 2012, pp. 2099–2103.

[27] D. Panigrahi, A. D. Sarma, G. Aggarwal, and A. Tomkins, "Online selection of diverse results," in Proc. 5th ACM Int. Conf. Web Search Data Mining, 2012, pp. 263–272.

Author Information

Mr.Ravi Kumar Tenali received M.Tech (CSE) form Swarandhra college of engineering and Technology (JNTUK).Working as Asst.Professor in dept of CSE in Andhra Loyola Institute of Engineering and Technology.He has 11yrs Teaching experience.He has published many papers in the international journal and international conferences.His area of interest includes computer networks,data mining and cloud computing.

Ms.Chandana Swetha Vengalareceived the B.Tech degree in Computer Science and Engineering from Andhra Loyola Institute of Engineering &Technology Vijayawada, India, in 2017.Her research interests include data mining.

Ms.Harika Kowtharapu received the B.Tech degree in Computer Science and Engineering from Andhra Loyola Institute of Engineering&Technology Vijayawada, India, in 2017.Her research interests include data mining.

.