Efficient Keyword Query Search over XML Data based on Context-Based Diversification Technique

Venkatesh Vadakapuram, Balaji Halavath, Varanasi Aruna

ABSTRACT - Empowering customers to get entry to databases by using easy keywords can relieve the users from the steep studying curve of mastering a structured query language and expertise complicated and possibly fast evolving data schemas. While key-word query empowers general customers to search huge quantity of data, the paradox of keyword query makes it tough to correctly answer key-word queries, especially for small and vague keyword queries. To deal with this tough hassle, in this paper we suggest an approach that systematically diversifies XML key-word search based on its distinctive contexts within the XML records. The proposed keyword search returns the set of smallest trees containing all keywords, where a tree is specified as “smallest” if it consists of no tree that also includes all keywords. Given a shortest and indistinct key-word query and XML records to be searched, we first derive keyword seek candidates of the query through an easy feature selection model. and then, we design an powerful XML keyword search diversification version to find and calculate the first-class of each candidate. After that, efficient algorithms are proposed to incrementally compute top-k qualified query candidates as the assorted search intentions. At last, a complete evaluation on real and artificial information units demonstrates the effectiveness of our proposed diversification model and the efficiency of our algorithms.

Index Terms - XML keyword search, context-based diversification and Smallest Lowest Common Ancestor

I. INTRODUCTION

Looking for records is a crucial aspect of our lives. Web search engines are broadly used for looking textual documents, images, and motion pictures. There are also extensive collections of structured and semi-structured statistics each on the web and in organizations which includes relational databases, XML, information extracted from textual content file.

Historically, to access those assets, customers must analyze dependent query languages, which includes square and X Query; they additionally want to access information schemes of every character software domain, which might be maximum probably complicated, speedy-evolving, or maybe unavailable in internet packages. A natural question to ask is whether or not we are able to empower customers to efficaciously get right of entry to based information the usage of key-word queries, workflows, and so forth.

Preferably the end result of a key-word seek over dependent information will mechanically gather applicable portions of records which might be in unique locations however are inter-related and together applicable to the question. There are numerous benefits of such an approach. First, it may relieve informal customers from the steep mastering curve.
of analyzing dependent query languages and statistics schemas while getting access to based records. Second, it permits customers to effortlessly access heterogeneous databases. For example, for web sites with database back-ends, this approach provides the extra flexible searching approach than the prevailing answer that makes use of a fixed set of pre-constructed template queries.

Moreover, this technique allows exposing exciting or surprising relationships amongst entities. Making database searchable will extensively increase the records volume that a person can access, have capability to provide search consequences with higher quality as compared with keyword searching on textual files, and for this reason increase the database usability and make huge effect to human’s lives.

As compared with key-word search techniques in records retrieval (IR) that choose to discover a list of relevant files, key-word search procedures in structured and semi structured information (denoted as DB and IR) pay attention more on particular records contents, e.g., fragments rooted at the smallest lowest not unusual ancestor (SLCA) nodes of a given key-word query in XML. Given a keyword query, a node \( v \) is appeared as an SLCA if 1) the sub tree rooted at thenode \( v \) consists of all the key phrases, and a pair of) there does no longer exist a descendant node \( v_0 \) of \( v \) such that the sub tree rooted at \( v_0 \) consists of all of the key phrases. In different words, if a node is an SLCA, then its ancestors may be genuinely excluded from being SLCA, through which the minimum records content with SLCA semantics can be used to represent the precise results in XML key-word search. In this paper, we undertake the properly-accepted SLCA semantics as a end result metric of key-word question over XML records.

In trendy, the extra key phrases a user’s question contains, the less complicated the user’s seek aim with reference to the question may be recognized. but, while the given key-word question simplest carries a small range of indistinct keywords, it might emerge as a totally tough hassle to derive the user’s seek intention due to the excessive ambiguity of this sort of key-word queries. Even though sometimes user involvement is beneficial to identify seek intentions of key-word queries, a user’s interactive process can be time-ingesting while the size of applicable end result set is massive. To deal with this, we are able to expand a way of supplying various key-words.

Query pointers to users primarily based on the context of the given keywords within the information to be searched. With the aid of doing this, users may also choose their preferred queries or adjust their unique queries primarily based on the returned various query recommendations.

II. RELATED WORK

The problem of diversifying key-word search is firstly studied in IR network most often carry out diversification as a publish-processing or re-ranking step of report retrieval based totally on the evaluation of end result set and/or the query logs. In IR, key-word search diversification is designed at the subject or file degree. For e.g., Agrawal et al user intents at the topical stage of the taxonomy and Radlinski and Dumais acquire the viable query intents through mining query logs but, it is not always easy to get those beneficial taxonomy and query logs. Further, the different effects in IR are frequently modeled at file levels. To enhance the precision of query diversification in based databases or semi structured data, it’s far acceptable to recollect both structure and content material of information in diversification version. So the problem of key-word search diversification is essential to be reconsidered in structured databases or semi structured data. Liu et al is the primary work to measure the distinction of XML key-word search outcomes by comparing their characteristic sets. But, the selection of
characteristic set is restricted to metadata in XML and it is also a method of put up-process search result analysis.

One of a kind from the above put up-process techniques, every other form of works addresses the problem of purpose-based totally key-word query diversification via constructing dependent query applicants. Their brief concept is to first map every key-word to a hard and fast of attributes (metadata), and then construct a huge variety of structured query applicants through merging the attribute-key-word pairs. They assume that each dependent query candidate represents a kind of search purpose, i.e., a query interpretation. But, these works aren’t easy to be implemented in real application because of the subsequent three limitations:

A huge range of structured XML queries can be generated and evaluated; there may be no assurance that the structured queries to be evaluated can discover matched outcomes because of the structural constraints much like, the process of building structured queries has to rely upon the metadata data in XML information.

To deal with the above boundaries and challenges, we provoke a proper learn of the diversification main issue in XMLKeyword search process, which can straightly compute the varied results without retrieving all the relevant candidates. Towards this purpose, given a keyword question, we first derive the co-associated characteristic phrases for every query keyword from XML data founded on mutual information within the likelihood theory, which has been used as a criterion for characteristic selection. The choice of our characteristic terms is just not confined to the labels of XML factors.

III. FRAMEWORK
A. Problem Definition:

Given a key-word query q and an XML facts denoted through T, we don’t forget a set of feasible search intentions Q which are generated through bounding each queryKey-word to a context using its applicable feature terms in T. Here, search intentions are also represented in the layout of keyword query. Obviously, we want gift to the users the top ok certified queries in terms of excessive relevance and maximal diversification.

B. Feature Selection Model:

Consider an XML information T and a fixed of time period-pairs W that could appear in T. The composition method of W depends on the software context and could not affect our subsequent discussion. as an instance, it is able to definitely be the entire or a subset of the terms comprising the text in T, the contents of a dictionary, or a wellspecified set of time period-pairs applicable to some programs.on this work, the distinct time period-pairs are selected primarily basedon their mutual information as Mutual factshas been used as a criterion for characteristic selection andfeature ameliorations in device studying. it can be used to represent both the Vance and redundancy of variables, which includes the minimum redundancy feature choice. Assume we have an XML tree T and its pattern end result set R (T). Allow P rob(x, T) be the possibility.

C. Extracting Feature Terms: Although we can pre-compute and control the correlated terms up to
any size, the use of two-term co-occurrences gives the maximum affordable alternative in most applications. Similarly, one-time period co-occurrences can be computed and saved successfully as described in co-occurrences of better order can be applied on the cost of space and, most importantly, time. For the scale of the packages we envision, materializing co-occurrences of length better than two might be infeasible. Consequently, in this work, we materialize -time period co-occurrences, which includes the computation of a sorted list.

D. Modules:

1. Admin: Admin maintains the total information about the complete application. Admin maintains the data in XML format only.

2. User: User search queries and he got the reply in XML format.

3. XML Query Answering: In this project user search the information in semi-structured document. He got reply in XML format only.

E. Keyword Search Diversification Algorithms:

In this phase, we first introduce the technique of producing a new query from the matrix of the unique keyword query w.r.t. the information to be searched. And then based on the matrix, we advise a baseline algorithm to retrieve the diversified keyword search consequences. At remaining, anchor-based pruning algorithms are designed to enhance the performance of the key-wordsearch diversification by utilizing intermediate results.

F. Generate Search Intentions:

Given a key-word query q, we first retrieve the corresponding characteristic phrases for each query keyword and then construct a matrix of search intentions. In the matrix, the characteristic phrases in each column are taken care of primarily based on their mutual records rankings. Each combination of the characteristic phrases (one term consistent with column) represents a search aim. We iteratively pick out the combination with the maximal aggregated mutual statistics rating as the next first-rate seek goal until the terminal necessities are reached.

G. Baseline Solution:

Given a key-word query, the intuitive concept of baseline algorithm is that we first retrieve the pre-computed characteristic terms of the given key-word query from the XML records T after which we generate all the viable supposed queries primarily based on the retrieved feature terms; at last, we compute the SLCAs as key-word search outcomes for each query and measure its diversification rating. As such, the top -ok diversified queries and their corresponding effects may be returned to users.

H. Anchor-Based Pruning

By studying the baseline answer, we will find that the principle cost of this solution is spent on computing SLCA results and disposing of unqualified SLCA results from the newly and previously generated end result sets. To lessen the computational price, we’re motivated to layout an anchor-based totally pruning solution, which can avoid the needless computational value of unqualified SLCA effects (i.e., duplicates and ancestors). On this subsection, we first analyze the interrelationships between the intermediate SLCA candidates.

I. Further Enhancement:

In this work, we presented a framework which improves the performance for XML keyword queries by a context-based diversification with baseline solution and anchor-based pruning and anchor based parallel sharing with SLCA semantics. In future it can be motivated to extend
the context-based diversification search with the aim to process multiple parallel request in order to improve the response time with multiple XML datasets while this is only on the single dataset.

IV. EXPERIMENTAL RESULTS

In this segment, we display the extensive experimental results for comparing the overall performance of our baseline solution (denoted as baseline evaluation BE) and anchor-based algorithm (denoted as anchor-based totally assessment AE), which have been applied in Java and run on a 3.0GHz Intel Pentium 4 device with 2GB RAM working home windows 7. For our anchor-based parallel sharing algorithm, it was applied on single PC computers. The experimental results are shown below.

We use a actual dataset, DBLP and a synthetic XML bench-mark dataset XMark for trying out the proposed XML key-word search diversification version and our designed algorithms. the scale of DBLP dataset is 971MB and the size of generated XMark dataset is 697MB. in comparison with DBLP dataset, the artificial XMark.

V. CONCLUSION

In context of paper, we first present an approach to search diversified outcomes of keyword queries from XML records primarily based on the contexts of the query keywords within the data. The diversification of the contexts have been measured with the aid of exploring their relevance to the original query and the newness in their consequences. Moreover, we designed three efficient algorithms based totally at the observed residences of XML key-word search outcomes. Ultimately, we proven the efficiency y of our proposed algorithms through walk of extensive wide variety of queries over both DBLP. in the meantime, we additionally demonstrated the effectiveness of our diversification model by means of analyzing the back search intentions for the given key-word queries over DBLP dataset. From the experimental outcomes, we get that our proposed diversification algorithms can go back qualified seek intentions an d consequences to customers in a quick time.

References:


Author Details:

Venkatesh Vadakapuram, pursuing M.Tech in Computer Science and Engineering, from Sreenidhi Institute of Science and Technology of Autonomous Affiliation and Approved by UGC, located at Yammnampet, Ghatkesar, Telangana State, India.

Email: venky.chary3@gmail.com

Phone: 9491366934

Balaji Halavath, Associate Professor, Sreenidhi Institute of Science and Technology of Autonomous Affiliation and Approved by UGC, located at Yammnampet, Ghatkesar, Telangana State, India.

Email: balajimitk@gmail.com

Varanasi Aruna, Head of Department, Sreenidhi Institute of Science and Technology of Autonomous Affiliation and Approved by UGC, located at Yammnampet, Ghatkesar, Telangana State, India.