INTERNATIONAL JOURNAL · ENGINEERING RESEARCH ·

International Journal for Innovative Engineering and Management Research

*A Peer Revieved Open Access International Journal*

www.ijiemr.org

## COPY RIGHT

IJIEMR Transactions, online available on 10th Febraury 2018. Link :
http://www.ijiemr.org/downloads.php?vol=Volume-7&issue=ISSUE-02

Title: Efficiently Answering Xml Keyword Queries Using Lllist.

Volume 07, Issue 02, Page No: 180 – 189.

Paper Authors

*** SK.NOORBASHA KAREEM HUSSAIN, S.S.N ANJANEYULU.**

* Eswar College of Engineering.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

# EFFICIENTLY ANSWERING XML KEYWORD QUERIES USING LLLIST

*SK.NOORBASHA KAREEM HUSSAIN, **S.S.N ANJANEYULU
*PG Student, Eswar College of Engineering, Narasaraopet, Guntur, AP, India.
**Assistant Professor, Eswar College of Engineering, Narasaraopet, Guntur, AP, India.

**ABSTRACT:**

From the past many years there has been much scope in efficiently replying to the XML (eXtended Markup Language) keyword queries. This paper focuses on wide-ranging XML keyword search supported on its completely different contexts within the XML information. We would be showing that for the purpose of faster document retrieval, the usage of tree is better than usage of array.Algorithms like LList based, Hash search based are performed for improved performance. XML inquiries are assessed by making an interpretation of them into SQL questions over the wide meagerly populated table. Mapping settled components to smoothed tables is the key issue for supporting XML on SQL databases. we introduce a scalable P2P framework for distributed data management applications using mutant query plans: XML serializations of algebraic query plan graphs that can include verbatim XML data, references to resource locations (URLs), and abstract resource names (URNs). We show how we can build distributed catalogs based on multihierarchic namespaces that can efficiently handle content indexing and query routing. Two selection criteria are targeted: the k selected query candidates are most relevant to the given query, while they have to cover maximal number of distinct results. At last, a comprehensive evaluation on real and synthetic data sets demonstrates the effectiveness of our proposed diversification model and the efficiency of our algorithms.

**Index Terms:** Search, XML, Baseline, Anchor based pruning , indexing ,diversification, query,disk based index, XML keyword.Data Mining, Search Engine Optimization, XML Dataset, Baseline Algorithm

## 1. INTRODUCTION

The different techniques and algorithms of top down strategy for XML keyword query processing of this survey is to make sure that answering the given query in faster and efficient way [1]. Many applications in the business and scientific domains XML has been widely used for storing, exchanging and publishing data [2].Keyword search on structured and semi-structured information has attracted a lots of analysis interest recently, because it permits users to retrieve with no requirements to learn query languages and database structure [3]. Compared with keyword search ways in data Retrieval (IR) that favor to realize an inventory of relevant documents, keyword search approaches in structured and semi-structured information (denoted as DB&IR) concentrate a lot of on specific data contents [4].To solve searching problem and to store XML data various indexing techniques are used. For reduction of memory overload of the processing queries indexing approach is introduced which is use of disk based indexing approach. In this approach a specific data structure is used for storing the index values [5]. We propose a methodology that consequently expands XML catchphrase seek in view of its diverse settings in the XML information [6]. Given a short and unclear watchword question and XML information to be sought, we first infer catchphrase look hopefuls of the inquiry by a straightforward element choice model. And after that we outline a powerful XML catchphrase look broadening model to quantify the nature of every hopeful [7]. After that, two effective calculations are proposed to incrementally figure top-k qualified inquiry hopefuls as the expanded pursuit expectations [8]. Current P2P systems offer very limited querying functionality: simple selection on a predefined set of index attributes, IR-style string matching or containment, no manipulation of content [9]. These limitations are acceptable for file-sharing applications, since people find ways to encode metadata about a file in the filename, but more general P2P applications will require a richer query model [10]. The notion of keyword proximity is more complex in the hierarchical XML data model. In this paper, we present the XRANK system that is designed to handle these novel features of XML keyword search [11].

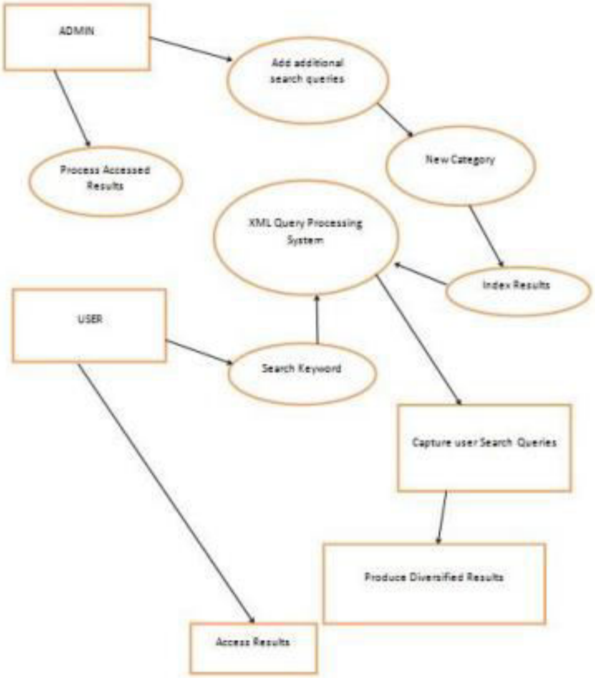
ADMIN
Add additional search queries
Process Accessed Results
New Category
XML Query Processing System
Index Results
USER
Search Keyword
Capture user Search Queries
Produce Diversified Results
Access Results

Fig1-Design of XML Context Diversified Search

## 2. RELATED WORK

You can find a general introduction to mutant query plans, our prototype implementation, and a preliminary performance comparison with traditional pipelined distributed query execution in previous work [12]. The key factors which results in the inefficiency for the

XML keyword search algorithms were CAR and VUN problems is proposed genetic top down processing strategy for visiting all common ancestor nodes only once which avoided CAR problem. An independent query semantic approach proved to avoid VUN problem. They proposed two algorithms namely LList to improve performance and hash search based method for reducing time complexity [13]. To start with, XML catchphrase seek questions don't generally return whole reports, yet can return profoundly settled XML components that contain the fancied watchwords. Second, the settled structure of XML suggests that the thought of positioning is no more at the granularity of a report, yet at the granularity of a XML component. At long last the thought of watchword nearness is more unpredictable in the progressive XML information model. We will assume that sellers export these data bundles in XML. Notice that our data are more structured and varied than the typical file description, and support much more meaningful queries; our query language therefore should be more powerful than the typical IR-based string matching interfaces found in most P2P systems [14]. A seller can run his or her own server to publish items for sale, or can post them to a server run by a consignment shop. Many queries will combine data residing in multiple peers. Transferring all relevant data to a central location wastes time and bandwidth. For-sale data is likely to have locality in terms of geographic location or category of merchandise [15]. We need a distributed query execution mechanism, so that we can run our queries “closer” to the relevant data. We propose keyword search in XML documents, modeled as labeled trees, and describe corresponding efficient algorithms. The proposed keyword search returns the set of smallest trees containing all keywords, where a tree is designated as "smallest" if it contains no tree that also contains all keywords [16].

## 3. SYSTEM MODEL

Changing over a XML encoded dataset connected with each recognized progressive structure, wherein for each distinguished various leveled structure said changing over step incorporates the further strides of: deciding a hub component set for said recognized progressive structure of said XML encoded dataset, wherein every hub component in said hub component set is a discrete level of said distinguished various leveled structure of said dataset; deciding one or more hubs of said XML encoded dataset every hub being a case of a hub component assigning to every hub a one of a kind hub identifier; and creating a SQL hub table containing one or more records, every record relating to a particular one of said distributed hub identifiers [17]. By second part of the creation, there is given a device to changing over a XML encoded dataset into a negligible arrangement of SQL tables, the mechanical assembly including:a gadget for distinguishing no less than one various leveled structure in the XML encoded dataset; anda gadget for changing over a XML encoded dataset connected with each recognized progressive structure gadget [18].

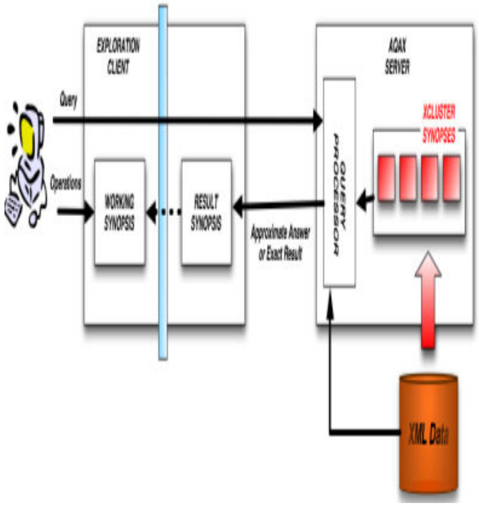

Fig 2.**System Architecture**

## 4. PROPOSED SYSTEM

To design a system on XML keyword query processing which is used for faster retrieval of relevant document by using disk based index approach. In a tree-based representation, nodes can represent an element tag, an attribute or a value [19]. Hierarchical relationships such as ancestor descendant and parent-child among XML elements can be represented with the help of edges. XML query processing depends upon the traditional top-down or bottom up traversing of tree on the XML document. It is highly inefficient, because it produces large collection of documents [20]. To reduce the overhead of processing queries, indexing or labeling methods are efficiently used. At the time of search the distinct term-pairs are selected based on their mutual information .Mutual information is to be used as principle or standard for feature selection and feature transformation in machine learning. It can be used to distinguish both the relevance and redundancy of variables, such as the minimum redundancy feature selection [21].

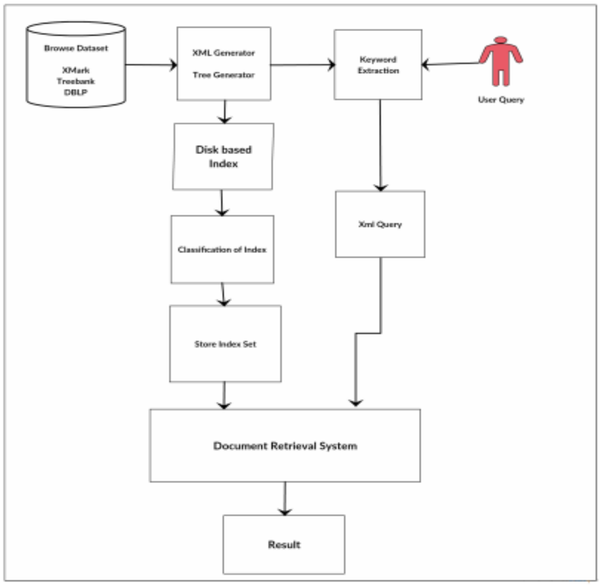

Fig. 3. Architecture of XML Keyword Query Processing

## 5. BASELINE SOLUTION

As mentioned in [17] baseline solution can be implemented as follows:

1. Given keyword query q with n keywords, we first load its pre-computed relevant feature terms from the term co-related graph G of xml data T
2. We generate a new query candidate qnew from the matrix by calling the function GenerateNewQuery ().
3. Generation of new query candidates is in the descending order of their mutual information score.
4. Compute the SLCA results of qnew we need to retrieve the recomputed node lists of the keyword-feature term pairs in qnew from T by getNodeList
5. Compare the SLCA result of the current query and the previous queries in order to obtain the distinct and diversified SLCA result.
6. We compute the final score of qnew as diversified query candidate with respect to previously generated query candidate in Q.
7. At last we compare the new query and previously generated and replace the unqualified once in Q.we can return the top k generated queries with their SLCA result[22].

**Algorithm: The Baseline Hash Search Algorithm**

**Input:**

• key (index value of an array)

**Output:**

• tree (contents of root node)

**Steps:**

• initialize root node

• assign index value to root node

• processCANode

• call function CA – add to root() or add to parent() or add to child()

To improve overall performance and to reduce memory overload hash index are used. The algorithm is NP-Hard because input given to it is a key value from the array of index [23]. This key is processed by an algorithm and contents of that key are displayed. With the indexing and top down approach retrieval of contents is faster than traditional searching

## 6. DISTRIBUTED CATALOGS

In the previous section we glossed over an important issue with mutant query plans the P2P network to maintain distributed catalogs that can efficiently route queries to peers with relevant data. This index structure cannot scale unless it is itself distributed or partitioned among peers. We believe that the main obstacle for building such distributed catalogs for file-sharing systems is the flat filename namespace where any peer can potentially serve any file. In many applications, such as the P2P garage sale we have much richer, structured metadata about our content data providers use multi-hierarchic namespaces to describe the kind of data they serve and data consumers use them to formulate queries [24].

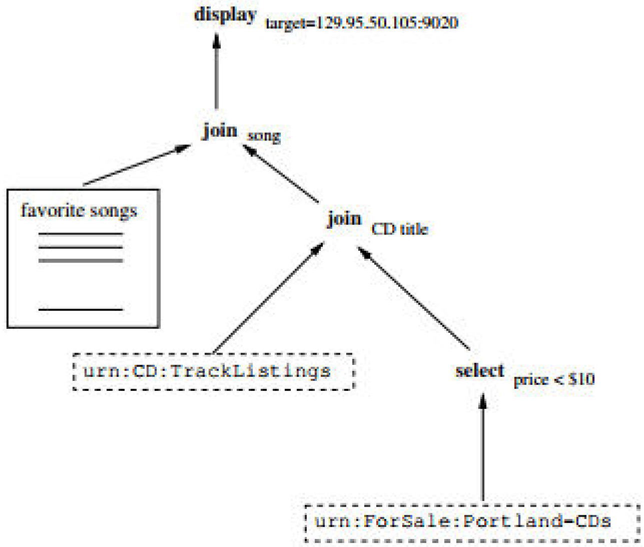

Figure 3: A mutant query plan.

### A. DOM Tree Construction

Get the Input Query Result Page from the User. Given a query result page, the DOM Tree Construction module first constructs a DOM tree for the page rooted in the tag. Each node represents a tag in the HTML page and its children are tags enclosed inside it. Each internal node n of the tag tree has a tag string tsn, which includes the tags of n and all tags of n’s descendants, and a tag path tpn which includes the tags from the root to n [25].

### B. Data Region Extraction

The Data Region Extraction module recognizes all conceivable information areas, which for the most part contain progressively created information, top down beginning from the root hub. We first expect that some youngster sub trees of the same guardian hub structure comparative information records, which amass an information area. Numerous inquiry result pages some extra thing that clarifies the information records, for example, a suggestion or remark, frequently isolates comparable information records. We propose another strategy to handle non-adjoining information locales with the goal that it can be connected to more web databases [26].

## 7. RESULT & DISCUSSION

The output of Hash search algorithm which is tree structure of given input value i.e key value. The experiments carried out on XMark dataset each key value given input to the algorithm the tree structure generated is different we are getting structure for root node which is ’site’. Parent node ’regions’ and its child node ’Africa’ and so on. For each node the structure of tree varies. Documents which are retrieved after experiments are relevant or not has measured. Relevant documents in this scenario are the documents which contain the keyword which are asks by the user in the form of query.

- Precision: The number of documents retrieved which are relevant to the query fired by user.
- Recall: The number of documents which are relevant to the query that are successfully retrieved.
- Accuracy: It is the closeness of a measurement to the true value.
- Throughput: It is the maximum rate at which query can be processed.

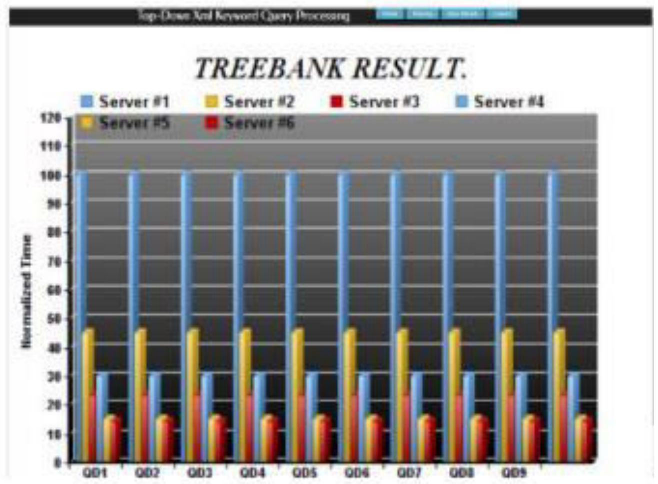

Fig. 4. Time Calculation of query processing on Treebank

## 8. CONCLUSION

We propose a methodology that naturally enhances XML watchword seek taking into account its diverse settings in the XML information. Given a short and dubious watchword inquiry and XML information to be sought, we first infer catchphrase seek competitors of the question by a basic element determination model. And successful XML catchphrase seek expansion model to gauge the nature of every applicant. We presented our framework for distributed data management based on mutant query plans and multihierarchic namespaces. Mutant query plans enable peers to independently optimize and partially evaluate queries without global knowledge, and with a minimum of coordination overhead. we verified the effectiveness of our diversification model by analyzing the returned search intentions for the given keyword queries over DBLP data set based on the DCG measure and the possibility of diversified query suggestions. Meanwhile, we also demonstrated the efficiency of our proposed algorithms by running substantial number of queries over both DBLP and XMark data sets.

## 9. REFERENCES

[1] 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,page 660-672

[2]Luis M. de Campos, Juan M. Fernández-Luna, Juan F. Huete, and Eduardo Vicente-López "Using Personalization to Improve XML

Retrieval" IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 5, MAY 2014

[3] YouqiangGuo ,Guixiu Tao, Yuqing Liang, Lei Wang, Honghao Zhu, " XML Keyword Search Based on Node Classification and Hierarchical Semantics" Communications in Information Science and Management Engineering Jan. 2014, Vol. 4 Iss. 1, PP. 6-12.

[4] Junfeng Zhou, Wei Wang, Ziyang Chen and Jeffrey Xu Yu, "Top-Down XML Keyword Query Processing",in IEEE Transactions on Knowledge and Data Engineening,Volume:28,Issue: 5, May 1 2016,pp. 1340-1353.

[5] L. Kircher, M. Grossniklaus, C. Grun, and M. H. Scholl, "Efficient structural bulk updates on the pre/dist/size XML encoding", in Proc. IEEE 31st Int. Conf. Data Eng., 2015, pp. 447-458.

[6] M. K. Agarwal and K. Ramamritham, "Enabling generic keyword search over raw XML data", in Proc. 31st Int. Conf. Data Eng., 2015, pp. 1496- 1499.

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

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

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

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

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

[12] 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.

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

[14] 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.

[15] LI Guoliang ,FENG Jianhua And ZHOU Lizhu," Keyword Searches in Data-Centric XML Documents Using Tree Partitioning" in TSINGHUA SCIENCE AND TECHNOLOGY

ISSNl 11007-0214l l02/ 211 lpp7-18 Volume 14, Number 1, February 2009

[16] Jianxin Li, Chengfei Liu, Rui Zhou and Bo Ning ," Processing XML Keyword Search by Constructing Effective Structured Queries" Advances in Data and Web Management Lecture Notes in Computer Science Volume 5446 , 2009, pp 88-9

[17] Simon Farande,Dr.,D.S.Bhosale,"Survey Paper on XML Context Diversified Search" in International Journal of Advance Research in Science and Engineering Vol. No.5,Special Issue No.01,March 2016,IJARSE,ISSN 2319-8354,page 1-4.

[18] Jing Wang, Nikos Ntarmos and Peter Triantafillou, “Indexing Query Graphs to Speedup Graph Query Processing”, in Proc. 19th International Conference on Extending Database Technology (EDBT), March 15-18, 2016, ISBN 978-3-89318-070-7.

[19] Ajay B. Gadicha, A. S. Alvi, Vijay B. Gadicha and S. M. Zaki, “Top-Down Approach Process Built on Conceptual Design to Physical Design Using LIS, GCS Schema”, in International Journal of Engineering Sciences & Emerging Technologies, Volume 3, Issue 1, August 2012, pp. 90-96.

[20] Jeremy Barbay, Alejandro Lopez-Ortiz and Tyler Lu, “Faster Adaptive Set Intersections for Text Searching”, in Proceeding WEA’06 Proceedings of the 5th international conference on Experimental Algorithms, May 24-27, 2006, pp. 146-157.

[21] Yi Chen, Wei Wang and Ziyang Liu, “Keyword-based search and exploration on databases”, in 2011 IEEE 27th International Conference on Data Engineering, DOI. 10.1109/ICDE.2011.5767958, 16 May 2011, pp. 1380-1383.

[22] "The use of MMR, diversity-based reranking for reordering documents and producing summaries", Proc. SIGIR, pp. 335-336, 1998

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

[24] H. Chen and D. R. Karger, "Less is more: Probabilistic models for retrieving fewer relevant documents", Proc. SIGIR, pp. 429-436, 2006

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

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

**Authors profile:**

**SK.N KAREEM HUSSAIN** is a student pursuing M.Tech (CSE) in Eswar College of Engineering, Narasaraopet, Guntur, India..

**S.S.N ANJANEYULU** is having 9 years of experience in the field of teaching in various Engineering Colleges. At present he is working as Asst. Prof. in Eswar College of Engineering, Narasaraopet, Guntur, India.