Algorithm on Top-k Keyword Search of Uncertain XML

Zhou Li-Yong[1, a], Zhang Xiao-Lin[2, b]

[1] School of Information Engineering Inner Mongolia University of Science and Technology, Bao Tou, Inner Mongolia, 014010, China

[2] School of Information Engineering Inner Mongolia University of Science and Technology, Bao Tou, Inner Mongolia, 014010, China

[a]email:imustc@126.com, [b]email: zhangxl@imust.cn

Keywords: uncertain XML; LRCT; Top-k; keyword search

Abstract. Currently, the Top-k keyword search of uncertain XML returns only the top k probability value of the root node. We need further processing to constructed the sub-tree that it meet some certain conditions. To solve this problem, this paper defines a new Top-k query semantics SRRT-Top-k that based on the minimum correlation Unicom subtree, LRCT-Top-k query returns the minimum correlation Unicom subtree of top probability value k, and presents the PLTop-k algorithm that it based on dynamic data warehouse of Keyword to process LRCT-Top-k queries. PLTop-k algorithm is only scanned once Dynamic Keyword data warehouse can be constructed to meet the sub-tree under specific conditions, and developed a filtering policy to reduce the intermediate results. The theoretical analysis and experimental results show, PLTop-k is a highly Top-k query algorithms of uncertain XML.

Introduction

XML queries has become a hot research, depending on the query mode.In-depth study of the underlying XML keyword search algorithm, this paper Proposed a new Top-k query semantics SRRT-Top-k that based on the minimum correlation Unicom subtree to return query results, and then design a dynamic Keyword data warehouse and a Keyword information transmission method of Information node in dynamic Keyword data warehouse, to facilitate the handling of LRCT-Top-k query, and design a PLTop-k algorithm that based on the dynamic Keyword data warehouse, the algorithm is only scanned once Dynamic Keyword data warehouse can be constructed to meet the sub-tree under specific conditions, and use filtering policy to filter out a lot of useless intermediate results, finally, we experimentally tested under different conditions to verify the algorithm is efficient.

Related research

Literature [1] gives three kinds of algorithms for solving SLCA node: Stack algorithm, ILE algorithm and the SE algorithms, these three algorithms are focused on how to calculate meet specific semantic root node, but the keyword search on XML documents not only need to return to meet specific semantic root node, but also need to build and returns XML document sub-tree of the root node, XML document sub-trees constructed largely determines the efficiency of XML documents keyword search. Literature [2] proposed to build XML fruit tree algorithm—Max Match algorithm, but Max Match algorithm when building sub-tree, it needs to repeat the scan keywords inverted table and frequently compare the string to get the node name, so it affects the efficiency. Literature [3] proposed Fast Match algorithm, Fast Match algorithm merger get results child root node and Construction of fruit trees in the process, do not repeat the scanning inverted table, so efficiency is very high.

LRCT-TOP-k query semantics

A fundamental problem of Uncertain XML's Top-k query keyword was to determine the semantics of Top-k keyword query.

Uncertain XML document model

Uncertain XML document model into probabilistic tree model [5], p- document model [6], in this article, we use the p- document model. We can use a tree structure to represent XML documents, among them, the probability of node are attached to the edge of the XML document tree by P-document mode. In p- document model, the distribution of nodes divided into two types: IND type and MUX type, the children of the IND type nodes are independent existence, the children of the MUX type nodes are mutually exclusive, that is, the node only has one child node exists. An example of p- document model is shown in Figure 1.

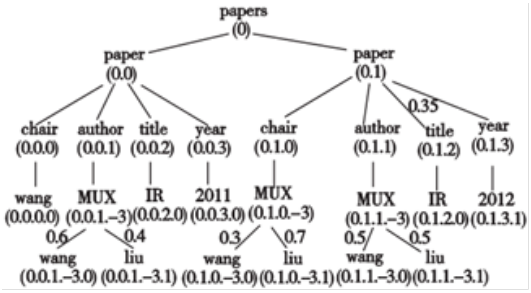

Fig.1. Document model instance

Least related connected sub-tree

In general XML document, suppose the query keyword set is Q = $\{k_1, k_2,..., k_n\}$, the most kid tree of the connection key $k_1, k_2,..., k_n$ instance called least related connected sub-tree LRCT[8]. If a sub-tree rooted SLCA which contains a keyword more than once, still as a minimum correlation communication sub-tree returns. As shown in Figure 2(a), Figure 2(b) documents shown, if the query keyword is "wang, IR", the results are returned respectively as shown in 3(a), Figure 3(b) and 4(a), Figure 4(b). Of course this case, the probability of Figure 3(a) and 3(b) minimum correlation communication sub-tree shown are 0.14, the probability of Figure 4(a) and Figure 4(b) minimum correlation communication sub-tree shown are 0.09.

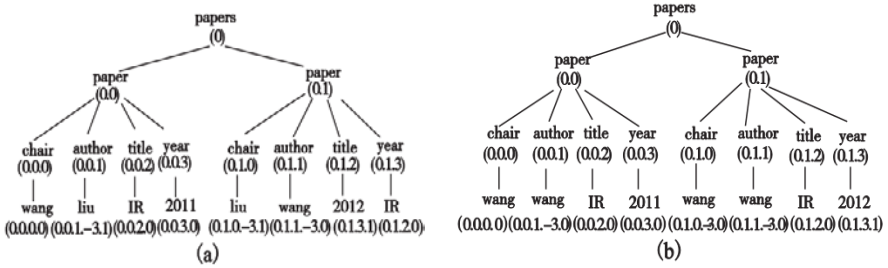

Fig.2. Two corresponding possible world examples

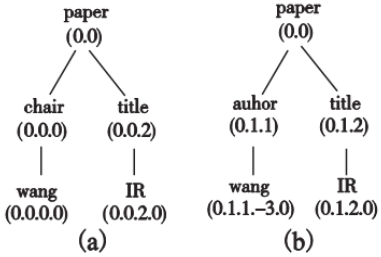

Fig.3. LRCT returned by figure 2(a)

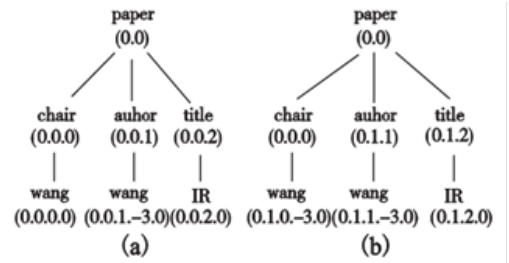

Fig.4. LRCT returned by figure 2(b)

LRCT-Top-k query semantics

Let uncertainty XML document D, W = $\{w_1, w_2, w_3, ... , w_n\}$ is D's possible worlds, discover set of keywords Q = $\{k_1, k_2, ..., k_m\}$, for examples w_i of every possible world in W, satisfy the query conditions set of least related connected sub-tree is LRCTi= $\{s_{ri1}, s_{ri2}, \cdots, s_{rih}i\}$ (h_i is a size

of collection $LRCT_i$), all probability of least relater connected sub-tree in $LRCT_i$ was p_i, LRCT for the set of all the least related connected sub-tree in all $LRCT_i$($1 \leqslant i \leqslant n$). At this point if s_{rik} and s_{rjt} same, that same structure, and the corresponding label corresponding coding is the same, then remove s_{rjt}, s_{rik} probability update is $p_i + p_j$, LRCT-Top-k to return the probability of the top k of least related connected sub-tree in LRCT.

PLTop-k algorithm

Dynamic Keyword data warehouse

Dynamic Keyword data warehouse's data structure shown in Figure 5, consists of a dimensional array of n * 1 and a list of n elements composition, each array element storing uncertain XML documents level number and a indicator to a pointer to the list; Node in the linked list is Information node, the same level in the same list of all the nodes corresponding uncertainty XML document node, and the nodes in a linked list in order of priority according to the order of the corresponding preorder of uncertain XML document nodes. If the list corresponding to the first element of uncertainty XML document node level is i, the pointer of the bit I data of the array is pointed to the list, the list is also known as the i layer list.

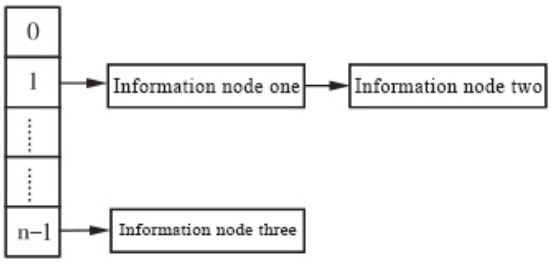

Fig.5. The basic structure of Dynamic Keyword data warehouse

The method for Keyword information transfer of Information node

In the process of solving the PLTop-k, a fundamental question is how the Keyword information of child nodes gets the Keyword information of father node, therefore, the design of the method of Keyword information transmission of Information junction.

Let query set of keywords $Q = \{k_1, k_2, ..., k_n\}$, father node is P, article i ($0 \leqslant i \leqslant 2^n-1$) items total probability of node P's Keyword information is p_i, string is z_i, the number of sub probability list are N_i, the sub probability of each sub probability list is p_{ik} ($0 \leqslant k \leqslant N_i-1$). The number of node P`s child nodes are m, for each child node Cc ($1 \leqslant c \leqslant m$), it`s conditional probability is fc, article j ($0 \leqslant j \leqslant 2^n-1$) items`s total probability of the node Cc of the Keyword information is p_j, string is z_j, the number of the sub probability list of J-item is N_j, the sub probability of each sub probability list is p_{js} ($0 \leqslant s \leqslant N_j-1$).

1) If the node P is MUX type node, the first use (1) and (2) to update all child nodes Cc of Keyword information.

$$p_j = p_j * f_c \ (0 \leqslant j \leqslant 2^n - 1) \qquad (1)$$

$$p_{js} = p_{js} * f_c \ (1 \leqslant j \leqslant 2^n - 1, 0 \leqslant s \leqslant N_j - 1) \qquad (2)$$

Then follow these rules in order to obtain the total probability p_i and sub probability information of the second term i ($1 \leqslant i \leqslant 2^n-1$) of a parent node P`s Keyword information: p_i is the total probability of the i-th item of all child nodes' Keyword information; For each child node Cc, if the divide probability information of item i of Keyword information in node Cc is not empty, then, we put all points probability list of i-item sequentially added to the divide probability information of the i-th item of node P`s Keyword information; Finally, (3) updating the node P's Keyword information of the overall probability of first 0.

$$p_0 = f_0 + (\sum_{c=2}^{m} f_c) \qquad (3)$$

2) If node P is the type of IND, Firstly, (4) and (5) to update each child node Cc's Keyword information.

$$p_j = \begin{cases} p_j * f_c \ (1 \leqslant j \leqslant 2^n - 1) \\ 1 - f_c * \ (1 - p_j) \ (j = 0) \end{cases} \quad (4)$$

$$p_{js} = p_{js} * f_c \ (1 \leqslant j \leqslant 2^n - 1, 0 \leqslant s \leqslant N_j - 1) \quad (5)$$

Then follow these rules, we get the overall probability and sub probability of the first i ($0 \leqslant i \leqslant 2^n-1$) items of node P`s Keyword information. Each one of a string of the node P`s Keyword information with each one of a string of all children nodes Cc make and operating. If the results obtained with the same string z_i, we will get involved in this string of results corresponding to minute probability information permutations and combinations of the operation, the results of permutations and combinations is placed in the divide probability information of item I in Keyword information of node P, In this case the item`s overall probability pi equal participation in the results obtained in this string corresponding to the total probability of the product, every one of sub probability is equal to the product of list stars probability of the participation in the formation of this sub probability list.

3) If the node P as an ordinary node, the same processing method with formula (2).

PLTop-k algorithm description

The major steps of PLTop-k algorithm are as follows:

The first step, dynamic Keyword data warehouse to be initialized. Uncertainty XML document depth-first traversal, in the ergodic process, if the node point contains the query keywords, then use the information of the node to initialize the Information node, the Information node is then stored in the Dynamic Keyword data compartment.

The second step, output the Minimum relevant communication sub-tree and its probability. From the Dynamic Keyword data warehouse of the bottom list beginning, from the bottom up, from left to right traversal the node in the Dynamic Keyword data warehouse, ergodic process, according to the father nodes of the different types, using the method mentioned in section B obtained Keyword information of the father node. If there is no father node of the current node in the on one list of Dynamic Keyword data warehouse, from Property one, we can use the current node represents its parent node, of course, at this time, the Keyword information of the current node needs to be updated. If the node is SLCA node, then use the node`s PEDewey coding and the PEDewey coding of sub probability list structure minimum correlation communication sub-tree.

Experiment

Experiments using the Java language to achieve the proposed PLTop-k algorithm, and the algorithm compare with the algorithm PrListTop-kN(PrListTop-k-N algorithm is PLTop-k algorithm does not use filtering policy) under different experimental conditions. Experimental hardware environment: CPU Intel (R) Core (TM) 2 (2.93GHz), RAM to 2.00G, experimental tool is MyEclipse8.5, JDK6.0. Experimental use of synthetic data sets, that is in the classic DBLP XML document data sets, and add some uncertain information, the combined uncertainty of XML document data sets. Use three sets of experimental contrast PLTop-k algorithm and PrListTop-kN algorithm, the experimental conditions of the first set of experiments is the same query as use cases, the numbers of returned results (the value of K) are same, but the query document is different. Experimental conditions for the second set of experiments is the same document queries, the numbers of returned results (the value of K) are same, but the query in different cases. Experimental conditions for the third set of experiments are the same query document, query use cases are the same, but the numbers of returned results (the value of K) are different. Each experiment was repeated ten times, remove the maximum and minimum values of the experimental data obtained in, using the averaging method for recording and finishing. Query by example in Table 1.

Tab.1. Experimental use query by example

Name	Keyword
Q1	{author, title}
Q2	{author, year}
Q3	{ year, pages }
Q4	{ pages, cdrom }

The first set of tests used in the query use cases Q1, k = 3, select five different sizes of documents for testing, Query results shown in Figure 6. The second set of test documents size is 56.4M, k = 3, Table 5.1 respectively executes a query with four cases, the query results shown in Figure 7. The third set of test selected document size is 56.4M, query in cases of Q1, and change the k value, the query results shown in Figure 8.

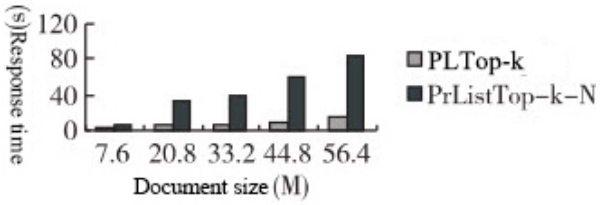

Fig.6. Both algorithm performance when compared to the size of the document changed

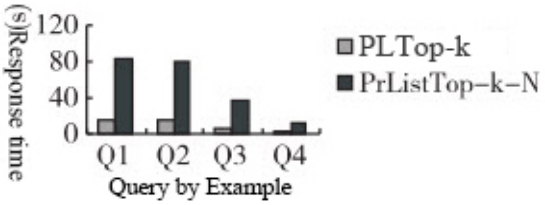

Fig.7. Both algorithm performance when compared to the query by example changed

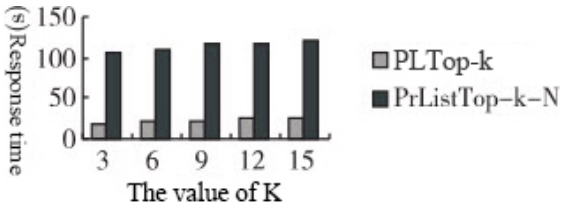

Fig.8. Performance comparison of k value changes

In the first set of experiments can be seen to increase with the size of the document, the running times of PLTop-k algorithm and PrListTop-k-N algorithm are increased, the reason is that with the increase of the document, the junction of these two algorithms process will increase, but we can clearly see that the performance of the PLTop-k algorithm is superior to PrListTop-k-N algorithm, the reason is that algorithm PLTop-k in the process of implementation, due to the filtering policy, a large number of sub-probability lists that they are not involved in the end result are filtered out, this will significantly reduce the operation that it is the divide probability list of child node passing to father node, improve efficiency. The second set of experiments can be seen at the times of different lengths are used in the query use cases, this is because the query use case is different, in the uncertain XML document, the number of Keyword query matching node is different, Dynamic Keyword data warehouse so that the number of nodes is different, thus making the number of the node of these two algorithms to process is different. However, this test conditions, the performance of the algorithm PLTop-k's also better than the algorithm PrListTop-k-N, the reason is the same with the teat conditions of the first group. As can be seen from the third set of experiments with increasing k values, both algorithms are also used to increase the time, this is because the increase in the value of k, so that the junction`s comparison operation of these two algorithms has increased, but it can be seen that the performance of the algorithm PLTop-k is superior to algorithm PrListTop-k-N.

Conclusion

In this paper, proposes a Top-k Keyword query semantics LRCT-Top-k of uncertain XML that it based on the minimum correlation communication sub-tree. And designed for solving LRCT-Top-k algorithm: PLTop-k algorithm, this algorithm traverses the node in the dynamic Keyword data warehouse from left to right, and to develop different filtering policies for different situations, will not participate in the final result as soon as the list of sub-probability filter out, reduce intermediate results, to improve the time and space efficiency, the test results show that the PLTop-k algorithm efficient query efficiency.

Acknowledgement

In this paper, the research was sponsored by the National Natural Science Foundation of China (Project No. 61163015) and Inner Mongolia Natural Science Fund Project (Project No. 2013MS0909).

References

[1] CuiJian, ZhouJun-feng, GuoJing-feng. FastMatch: anefficiental gorithm for XML keyword search[J]. Application Research of Computers, 2012, 29(6):2184-2187.

[2] XuY, PapakonstantinouY. Efficient keyword search for smallest LCAsin XML databases[C]. In: Proceedings of the ACMSIGMOD International Conference on Management of Data, NewYork,2005:527-538.

[3] LiuZi-yang, ChenYi. Reasoning and identify in grelevant matches for XML keyword search [J]. Proceedings of the VLDBEndowment, 2008, 1(1):921-932.

[4] LiJiang-xin, LiuCheng-fei, ZhouRui, etal. Top-k keyword search over probabilistic XML data[C]. In: Proceedings of the 27th International Conference on Data Engineering. LosAlamjtos, 2011:673-684.

[5] AbiteboulS, SenellartP. Querying and updating probabilistic in for Mation in XML[C]. In: Proceedings of the 10th International Conference on Extending Database Technology. Munich, 2006:1059-1068.

[6] AbiteboulS, SenellartP. Querying and updating probabilistic in for Mation in XML[C]. In: Proceedings of the 10th International Conference on Extending Database Technology. Munich, 2006:1059-1068.

[7] NingBo, LiuCheng-fei, JeffreyXuYu, etal. Matching Top-k answers of twig patterns in probabilistic XML[C]. In: Proceedings of 15th International Conference on Database Systems for Advanced Applications. Tsukuba, 2010:125-139.

[8] WangXiao-feng, MengXiao-feng, ZhouJun-feng, etal. Keyword search on XML streams[J]. Journal of Computer Research and Development, 2006, 43(Sup.): 484-489.