Association rule pruning for XML document using modified index table

Shweta Desai
Department of CE,
L.J. Institute of Engineering & Technology, Ahmedabad, Gujarat

Abstract- XML is well-known for storing and transferring the information. XML provides flexibility to design as per requirement of the application or organization. . The design goals of XML emphasize simplicity, generality, and usability over the Internet. XML databases are increasing continuously. To take the benefit of utilizing interesting information from the XML document, it is very complex and difficult because of the inherent flexibility and semi-structure data. Traditional method to get the information from XML document is by designing schema in relational database and store/retrieve information from there. To avoid this overhead of designing complex schema for individual XML documents, there are various methods and techniques available for mining XML document. These data extracted from XML document can be used to analyse the future trend by using historical data. To find out interesting information, association rule mining is one of the known ways. Various interestingness measures are available to find hidden information from the huge amount of data. In this paper, lift interestingness measure is used to prune the association rules set on modified index table approach.

I. INTRODUCTION

Data mining

Mining is a vivid term characterizing the process that finds a small set of precious information from a huge amount of data stored in different formats and files. Data mining is one step in the knowledge discovery process, an essential one because it uncovers hidden patterns for evaluation [2].

Fig. 1. Data mining process [11]

As shown in above figure, it shows step by step process of data mining. Data mining process is to extract information from a data set and transform it into an understandable structure for further use.

As explained in [3], author said that while generating association rules from large datasets, it's possible that redundant rules will generate. These generated rules cannot directly used by an application. By pruning and clustering the rules, it will remove redundant rules and provide better output.

XML

XML (eXtensible Markup Language) is gaining popularity as new standard for data representation and exchange on the internet. XML uses custom-defined tags to describe the data and the structural relationships of data within a document. XML is a subset of SGML and is defined by the World Wide Web Consortium (W3C).

XML is used for data representation, storage, and exchange in many different arenas [12]. Mining of XML documents differs significantly from other structured data. XML mining includes mining of structures as well as content from XML documents. Fig 2 shows the classification of XML mining. In XML mining, association rules will generate based on frequent sub-structure. Association rule generation in XML data mining is differing from traditional data mining applied on relational database. [5]

XML data mining can be categorize in 3 different ways as given below.

- XML Structure Mining: Mining XML data has its roots in problems which originally arose from several applications in semi structured data management, such as integration of data sources and query processing. Such applications were initially focused on solutions for structurally comparing semi structured data. Important research contributions have especially regarded *pattern matching, change detection, similarity search and detection and summarization*, for XML schema as well as document collections.
- XML Structure and Content Mining: The need for discovering knowledge from XML data according to both structure and content features has become challenging, due to the increase in application contexts for which handling both structure and content information in XML data is essential. XML Structure and Content Mining also represent a point of convergence for research works in semi-structured data and text mining.
- Semantics-aware XML Mining: The increase in volume and heterogeneity of XML-based application scenarios makes data sources exhibit not only different structures and contents but also different ways to semantically annotate the data. The inherent difficulty of devising suitable notions of semantic features and semantic relatedness among XML data leads to one of the hardest challenges in contexts of data management and knowledge discovery.[12]

Association Rules

In association rule mining, we first extract association patterns, which are co-occurring binary risk factors. The frequent co-occurrence of these two conditions may indicate that they are associated with each other. [13] Association rules are one of the most popular ways of representing discovered knowledge and describing a close correlation between frequent items in a database. An X⇒Y type association rule expresses a close correlation between items (attribute-value) in a database. There are many association rule discovery algorithms but Apriori is the first and foremost among them. [14]

To find the support and confidence of the association rules are defined as

$$\text{Support } (X \rightarrow Y) = |Txy|/|T| \quad (1)$$

$$\text{Confidence } (X \rightarrow Y) = |Txy|/|Tx| \quad (2)$$

transaction. Txy={frag| ⊥ | I_ |
and Tx={frag|I_ |

Where T is a set of XML fragments in the database, |T| is the number of total XML fragments as ∀ ∈ (X∪Y)(I IN frag)} ∈ X(I IN frag)}.I IN frag denotes that an XML document contains terminal element or terminal-elements I. [1]

II. LITERATURE SURVEY

In [9], R Agarwal and R Srikant has found out fastest algorithm for mining association rules. They have compared their algorithm with the existing AIS and SETM. They proposed two new algorithms, Apriori and AprioriTid. The difference between Apriori and AprioriTid is, every time, Apriori will use the datasets for generating the candidate sets, where AprioriTid will use previous candidate set to generate next set. Below is the execution time of it.

Author has compared the algorithm on itemsets with the number of passes. For pass 1,2 and 3 Apriori works faster but after 4th pass, AprioriTid is faster.

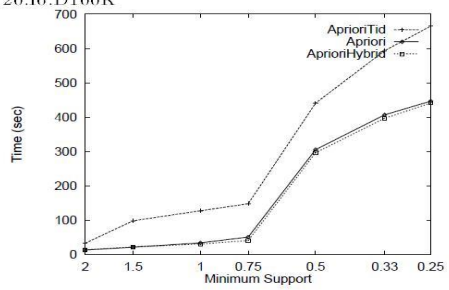

Fig. 2. Execution time of AprioriHybrid, Apriori and AprioriTid

They propose a hybrid algorithm which is the combination of both the algorithms. At K's pass, Apriori will swich to AprioriTid. As shown in Fig 2, Dj is Number of transactions; Tj is Average size of the transactions; Ij is Average size of the maximal potentially large itemsets. If the average size of transaction is high then Apriori and AprioriHybrid gives the same time. So, the performance of the algorithm depends on the mention 3 parameters, T, I and D. As per the available data, one can use any of the mention 3 algorithms.

In [4] authors have compared apriori algorithm with fp- growth. Also implement apriori in Xquery as well in Java. They have proved that apriori works better in Java rather than in Xquery.

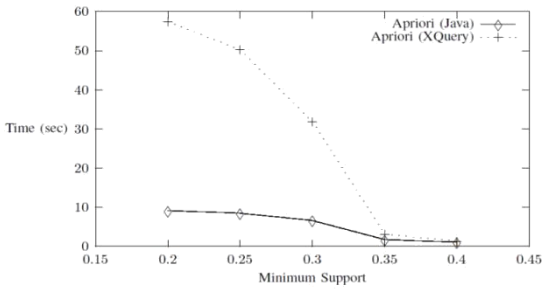

Fig. 3. Java-based Apriori Vs XQuery-based Apriori

But the gap between the two narrows as the number of transactions increases. It is their opinion that the data structure overhead in the XQuery implementation is what led to the performance difference between the Java-based Apriori and XQuery-based Apriori. In Fig 3, it shows the performance analysis of Apriori algorithm, implemented in java and XQuery.

Leena A Deshpande, R.S. Prasad [8] has survey some existing methods for efficiently mining frequent pattern from semi-structure data. Users not only query the data to find a particular piece of information, but he is also keen in knowing better understanding of the query. Because of this variety, semi-structured DBs do not come with a conceptual schema. To make these databases more accessible to users a rich conceptual model is needed.

Traditional retrieving techniques are not directly applied on these databases. XML is mainly used for exchanging wide variety of data on the web. The increasing popularity of XML is partly due to the limitations of the other two technologies: Hypertext Markup Language (HTML) and Standard Generalized Markup Language (SGML) for representing structured and semi-structured documents.

Their Survey is characterized on the following key Challenges:

[1] Design an efficient algorithm to identify patterns from varied and huge semi structured data.

[2] Retrieving frequent graph pattern from a given set of graph for Mining.

There are many techniques and methods available for XML mining. XML document can be mined for association rules using only the query language XQuery without any pre-processing or post-processing. In "Extracting association rules from XML documents using XQuery" author have implemented Apriori using XQuery and demonstrated the process of mining association rules from native XML data. Many issues remain open; one of the issues concerns the structure of the XML data. Since the structure of the XML data can be very complex and irregular, identifying the mining context on such XML data becomes difficult. Therefore, to simplify the task of identifying the context, a set of transformations of the XML data might be required [6]. To overcome the limitation of Xquery, Tree-based association rules from XML documents method has been designed. Such rules provide information on both the structure and the content of XML documents [7]. The problem in tree based method is, it require more time to processing the result.

In [3] "Association Rule Pruning based on Interestingness Measures with Clustering", author says association rule mining plays vital part in knowledge mining. The difficult task is discovering knowledge or useful rules from the large number of rules generated for reduced support. For pruning or grouping rules, several techniques are used such as rule structure cover methods, informative cover methods, rule clustering, etc.

In Association rule mining, large number of Association rules or patterns or knowledge is generated from the large volume of dataset. But most of the association rules have redundant information and thus all of them cannot be used directly for an application. So pruning or grouping rules by some means is necessary to get very important rules or knowledge.

Discovered rules with the given confidence and support thresholds are large in number. All these rules are not useful, since they are heavily redundant in information. There are several ways of grouping rules such as methods based on clustering techniques, 1. Rule structure, 2. Rule instance cover and so on. In their work, rules are grouped based on rule consequent information. So groups of rules are in the form Xi -> Y for i=1, 2… n. That is, different rule antecedents Xi's are collected into one group for a same rule consequent Y.

Since each group has large number of rules, next step is to select small set of representative rules from each group. Representative rules are selected based on rule instance cover as follows. Let **R**y={ Xi Y | i=1,2,…,n } be a set of n rules for some item-set Y and m(Xi Y) be rule cover, which is the set of tuples/records covered by the rule Xi -> Y in the dataset **D**. Let **C**y be the cluster rule cover for a group or cluster of rules **R**y. i.e.,

$C_Y = m(\mathbf{R}y) = \cup i=1,2,\ldots n\, m(Xi Y)$

Next, from cluster rule set **R**y, find a small set of k rules **r** called representative rule set such that m(**r**y) is almost equal to m(**R**y).

$\cup j=1,2,\ldots k\, m(XiY) \approx \cup i=1,2,\ldots n\, m(XiY)$, where $k << n$

III. EXISTING SYSTEM

In the existing system, author has used modified index table. Modified index table is a better option than using tree for mining XML document. But the problem in mining XML document using index table

is, there are drawbacks in the existing method, such as node encoding from the XML document is a quite a tough and ineffective. The node encoding used before are based on numbering the tags, which affect the differentiation of nodes. Also it takes more time for execution. Author has proposed an efficient way of collecting information from XML document by modifying the index table. The method is having following steps:

Step 1: Encode the items in XML with a unique id (UID)

It will assign UID to each item in the XML, if item will repeat it will use the previous UID.

Step 2: Create 2 tables

5) UID values (UID, PATH, VALUE)
6) Index (DocID, UID)

UID	PATH	VALUE
1	/course/subjectcode	CE01
2	/course/courseid	211
3	/course/title	Introduction to DI

Fig. 4. UID table

Here, the path will contain the path from root to item tag.

DocID	UID
1	1,2,3,4,5,6,7,8,9,10
2	11,2,12,13,14,6,15,16,17,18
3	19,20,21,4,22,23,24,25,26,27

Fig. 5. index table

Step 3: Mining association rules by using Apriori algorithm. By using support and confidence, the rules will be generated based on the minimum values.

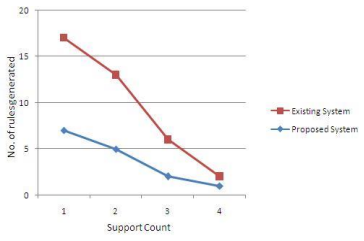

Fig. 6. Association rules generated for confidence 60%

Create a table which will contain support value of each UID.

Apriori algorithm is used to mine the association rules which are of the form, UID3 -> UID4. It will converted into XML representation using the values that are given in the

UID value table and it will be of the form,
<Subjectcode>CE01</Subjectcode> ⟶
<title>IntroductiontoADMS</title>

Fig. 6. is showing result of no of rules generated by using index table method and by using modified index table.

IV. PROPOSED SYSTEM

Association rules which will generate, it's not necessary that all the rules with high confidence and support value will have strong association between the items. By using other interestingness measures like lift, specificity, conviction, one can reduce the no of rules generated.

V. CONCLUSION

As in this paper, many algorithms related to association rule mining and XML data mining have been discussed. Apriori and FP growth are fast algorithm for association rule mining, but it varies depends on the data structure and no of transactions. For XML document mining, Xquery is used to mine XML document without applying any pre or post processing. Java supports XML and provide in-built methods to process it, which will help for faster processing of XML files. By comparing the XML mining using JAVA and Xquery, Java provides faster output. Using tree structure for mining XML is an easy method but it's time consuming process.

Using of index table is efficient way of pre-processing of XML files. Hence mining effective patterns play important role in providing the optimized solution.

VI. REFERNCES

[1] D. Sasikala, K. Premalatha "Mining Association Rules from XML Document using Modified Index Table", International Conference on Computer Communication and Informatics (ICCCI - 2013), 2013,

[2] Data Mining: Concepts and Techniques: Concepts and Techniques
By Jiawei Han, Micheline Kamber, Jian Pe

[3] S.Kannan, R.Bhaskaran "Association Rule Pruning based on Interestingness Measures with Clustering", IJCSI International Journal of Computer Science Issues, Vol. 6, No. 1, 2009

[4] Ding, Qin, and Gnanasekaran Sundarraj. "Association Rule Mining from XML Data" DMIN. 2006.

[5] Nayak, Richi. "XML data mining: Process and applications." *The Process and Applications of XML Data Mining* (2008)

[6] Wan, Jacky WW, and Gillian Dobbie. "Extracting association rules from XML documents using XQuery." Proceedings of the 5th ACM international workshop on Web information and data management, ACM, 2003

[7] Saranya T.J., "Mining Tree based association rules from XML documents", International Journal of Advanced Technology & Engineering Research (IJATER),Mar 2012 Volume 2

[8] Deshpande, Leena A., and R. S. Prasad. "Efficient Frequent Pattern Mining Techniques of Semi Structured data: a Survey" International Journal of Advanced Computer Research (IJACR) Volume-3 Number-1 Issue-8 March-2013 (2013)

[9] Agrawal, Rakesh, and Ramakrishnan Srikant. "Fast algorithms for mining association rules" *Proc. 20th Int. Conf. Very Large Data Bases, VLDB* Vol. 1215 1994

[10] Soumadip Ghosh, Amitava Nag, Debasish Biswas, Arindrajit Pal, Sushanta Biswas, Debasree Sarkar, Partha Pratim Sarkar "XML mining using Genetic algorithm", Journal of Global Research in Computer Science, Volume 2, 2011

[11] Online article, http://www.netlineindia.com/datamining.html

[12] Online article, https://sites.google.com/site/xmlmining/

[13] Simon, Gyorgy J., et al. "Survival Association Rule Mining Towards Type 2 Diabetes Risk Assessment." *AMIA Annual Symposium Proceedings*. Vol. 2013. American Medical Informatics Association, 2013.

[14] García, Enrique, et al. "A collaborative educational association rule mining tool."*The Internet and Higher Education* 14.2 (2011): 77-88.