Mining Association rules from XML Documents using Index table

Sasikala D[1], Premalatha K[2]
[1]*Professor, Department of CSE, Bannari Amman Institute of Technology, Sathyamangalam.*
[2]*Professor, Department of CSE, Bannari Amman Institute of Technology, Sathyamangalam.*

Abstract: Association rule mining finds the interesting correlation among a large set of data items. With a large amount of data being collected and stored continuously in databases, it has become mandatory to mine interesting relationship between the attributes. Semi-structured data refers to set of data with some implicit structure but not enough of a regular. Mining association rule from semi-structured data is confronted with more challenges due to the inherent flexibilities of it in both structure and semantics.

The eXtensible Markup Language (XML) is a major standard for storing and exchanging information. This paper presents an index based scheme to index all the elements in a group of XML document. This table is used to check the ancestor-descendant relation between an item and transaction efficiently and a relational. Apriori algorithm is used to mine association rules from the XML documents with no guidance of the user.

Key words:- index table, XML, Association rule mining, ancestor-descendant relation

I. INTRODUCTION

Data mining or Knowledge discovery in databases (KDD) is the process of extracting interesting knowledge from a large amounts of data, stored in large relational databases. The extracted knowledge can be represented in many forms like clusters, decision trees, decision rules, association rules etc. Among these association rules discovers the interesting relationships among data items in the relational database. The recent success of XML as a standard for storing and exchanging information in the web poses new challenges in the data mining community. The flexibility of XML in both structure and semantics leads to more challenges in mining association rules from XML data[5].

An XML document is in tree structure with each element as nodes in it. The form of XML association rule is different from that of traditional one. The transactions and items in an XML document is also different from traditional transactions and items. So a new definition of transaction and item is given in XML context. In this paper an index table is used to retrieve transactions and items from the XML document. Index table is also used to check the include relation between a transaction and an item. The unknown association rules are mined from XML document by using this technique[6].

The remainder of the paper is organized as follows. The research directions are introduced in Section 2. A new definition of XML transaction, item and association rule is given in Section 3. Techniques for association mining from XML data based on index table are described in Section 4. The performance data is reported in Section 5. Section 6 concludes the paper.

II. RESEARCH ISSUES

There are three main research directions in XML association rule mining. First is to mine association rule between XML document tags. For example if a XML document has a tag<author>, there is 90% possibility for it to have a tag<book> at the same time. Second is to mine association rule from the element contents. There are several methods like XMINE operator, XMINE RULE operator etc.For example if a XML document has element text like <item>milk</item>,then there is a possibility of it to have <item>bread</item> at the same time[4].

The above methods are suitable only for a single document with known structure and for mining association rules with the guidance of interest associations given by users.

The method used to check include relation between a transaction and an item is to check if the textual representation of item is in the textual representation of a transaction. This method is computationally expensive. Third is to discover similar structures among a collection of XML documents. The similar structure is called frequent substructure. TREE MINER algorithm is used to mine association rules from frequent substructure.

The structure of XML document is a tree structure, the form of XML association rule is different from traditional association rule. The transaction and item of XML documents are also different from traditional transaction and item. An index table is used to extract items and transactions from an XML document.

Compared to the above methods, the method reported in this study aims to mine XML content association rules with no guidance of interesting associations and aims to mine unknown association rules from XML documents with similar structure.

III. MEASURES OF ASSOCIATION RULE MINING

The structure of an XML document is a tree and mining XML association rules is different from that in the traditional well-structured world. A new definition of XML association rule which is given in this method. Terminal-element is the element without sub element. The transaction is a sub tree, and the items are the leaf nodes in the sub tree. The root node of sub tree is used to identify a transaction and leaf node in the sub tree to identify an item. An XML association rule is an implication of the form $X \rightarrow Y$, $X \subseteq I$, $Y \subseteq I$, and $X \subseteq Y$, where I is a set of terminal-elements (tree-structured items) as in [1] and [10].

The support and confidence of the rule are defined as:

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

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

```
<purchase>
  <Personal_details>
      <name>stella</name>
      <income>average</income>
  </personal_details>
   <item>
 <hardware>desktop</hardware>
<software>antivirus</antivirus>
   </item>
</purchase>
```

Fig.1 sample XML document

Let us consider the XML document in Fig 1,the XML fragment <item>..</item> is a transaction. The items are the terminal elements in the <item> element like <hardware>..</hardware>,<software>…</software> etc. The support count is computed as the percentage of XML fragments that contain the items. The confidence is calculated as the percentages of XML fragments that contain an item X also contain the item Y. The association rules are in the form <hardware>desktop</hardware>→ <software>antivirus </software>as in [7].

IV. XML ASSOCIATION RULE

The mining process consists of the following steps 1) extracting XML transactions and items from index table 2) generating a relational table made up of transactions and items 3) mining XML association rules using Apriori algorithm[2].

4.1 Node encode

Index table indexes all element nodes in XML documents. An index table

Index_table=(docID, nodeEncode, tag, value)

Where docID represents XML document number, nodeEncode is the encoding of a node n in a document tree and it is the encoding of its parent, augmented by the index of n among its siblings and adding a dot to separate them. If the node occurs more than one time in a document it is encoded on the basis of its number of occurrence. For example, the root node is encoded as 1, its first child node is encoded as 1.1, and its second child node is encoded as 1.2 and so on and if the occurs for the first time in the document it is encoded as 1.1_1 and for the second time it is 1.1_2 and so on.

Where docid represents XML document number, nodeEncode is the encoding of a node in a document tree and its parent with a dot between them. An XML document in tree form is depicted in Fig 2

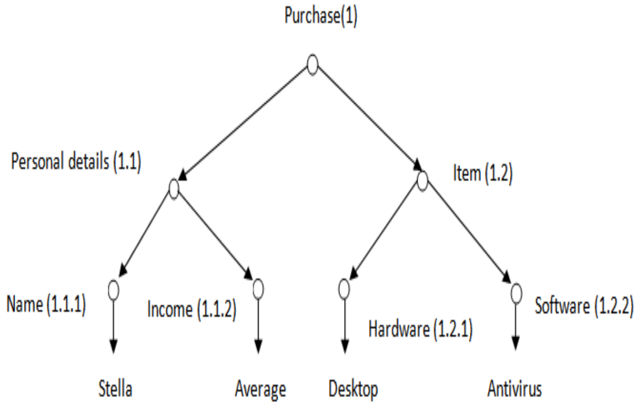

Fig.2 Node Encoding in the tree of XML document

4.2 Include Relation Checking

A transaction is a sub tree in the XML tree, an item is a leaf node in the sub tree. The root node is used to identify a transaction. Checking include relation between a transaction and an item becomes to check the ancestor-descendant relation between two element nodes. According to the encoding schema, the following lemma is used,

Lemma 1: A transaction includes an item if and only if the encoding of the item node begins with the encoding of the transaction node adding with a dot and they belong to the same document.

At same time, if a transaction includes an item, according to the above definition of transaction and item, the transaction node is the ascendant of the item node. A descendant node (item node) encoding begins with the encoding of its ascendant node (transaction node) adding with a dot and they belong to the same document.

For example, in Figure 2, XML fragment <item>….</item> is a transaction, the root node of this transaction is encoded with “1.2.1”; item <hardware>desktop</hardware> is included in this transaction, because the encoding of this item node is “1.2.1”, which begins with “1.2.”.

4.3 Extraction of Item and Transaction

A transaction is a XML fragment and use the root node of its tree is used to denote a transaction. Thus, records (docID, nodeEncode) is selected from index table to represent a transaction, the nodeEncode here aims to check the include relation between this transaction and items as in [8].

The leaf nodes that are descendant of transaction nodes from index table is selected by checking the include relation according to lemma 1. The value of (docID, nodeEncode, tag, value) is used to represent an item node. In this step, the item nodes extracted may have same tag and value; assume that if two item nodes have same value and same tag, then they are the same item. Those duplicate items are removed and generate an item table to record all different items in this step.

With the transaction sets and item sets, the relational table R is generated as follows:

1) Column is made up of XML items; row is made up of XML transactions.

2) If the ith transaction includes the jth item, then R (i, j) =1, otherwise, R (i, j)=0.

An example XML document is given as follows and the corresponding index table is depicted in Table 3.1

```
<order>
<person>
<name>martin louis</name>
<gender>male</gender>
</person>
<item>
<vcd>pop music</vcd>
<vcd>starwar I</vcd>
<book> starwar II</book>
</item>
<item>
```

```
<vcd>starwar I</vcd>
<book>starwar I</book>
</item>
</order>
```

DocID	NodeEncode	Tag	Value
1	1	order	NULL
1	1.1	person	NULL
1	1.1.1	name	martin louis
1	1.1.2	gender	male
1	1.2_1	item	NULL
1	1.2_1.1	vcd	pop music
1	1.2_1.2	vcd	starwar I
1	1.2_1.3	book	starwar II
1	1.2_2	item	NULL
1	1.2_2.1	vcd	starwar I
1	1.2_2.2	book	starwar I

The index table is efficient for the extraction of transactions, items and to check include relation, especially for many XML documents. The modified index table is as shown above.

During the extraction step, the interruption technique is used to skip many searching steps and comparing times of the include relation according to the lemma 2. That is, the nodes that following after a transaction node is extracted and compare the include relation among them. But it needn’t extract all item nodes and compare with all transactions one by one. This will reduce the execution time greatly.

Lemma 2: All items that are included in a transaction are always following with the transaction in index table derived by a depth-first traversal of XML document tree.

4.4 Mining XML association rules

The XML association mining algorithm here is based on the plain Apriori algorithm.

Input: index_table

Output: XML association rule

Algorithm:

```
n=0; transsum=0;
currec=the first record of index_table;
while (not to the end of index_table){
    if(currec.value is NULL){
        transsum++;
        nextrec= the next record of index_table;
        while (has nextrec){
            if((nextrec.encode).startsWith(currec.encode+".")and (nextrec and nextrec have same document ID)){
                if (nextrec.value isn’t NULL) {
                    itemnum=findnum(n,nextrec.tag,nextrec.value);
                    bb[transsum][itemnum]=1;
                }
            else{
                currec=nextrec; // skip iter
                break ;
            end if
        end while
    end if
end while
```

CALL aprior(bb[][]);
In the final, association rules extracted from index table are mapped into the XML representation.

V. EXPERIMENTAL RESULTS

A set of XML document were taken for experiment. To improve the performance of the system, modified index table is used to extract association rules. The rules are generated based on the user given tag. The execution time for the index table and modified index table based methods is shown in Fig.3. The execution time for the modified index table is less than that of the index table method. While varying support and confidence values, the number of rules generated varies accordingly and it is depicted in the Fig 3

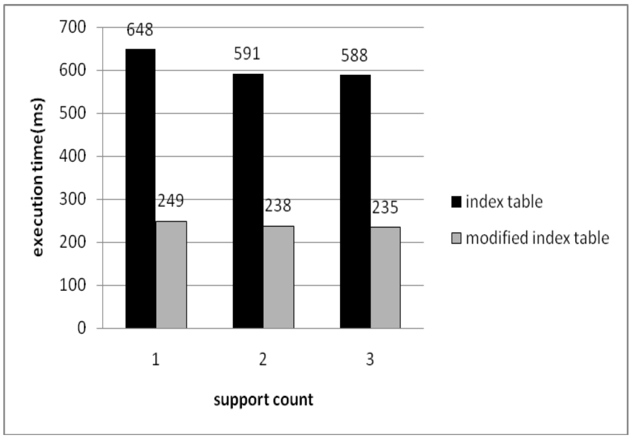

Fig.3 Comparison of execurtion times for index table and modified index table based methods

VI. CONCLUSION

Association rule mining discovers interesting relationship among the data items. Apriori algorithm and FP growth are generally used to mine association rules. Association rules are mined from XML document with the help of index table in the existing system. Index table is used to check the include relationship between a transaction and an item. To improve the performance of the system, modified index table is used to extract association rules. The rules are generated based on the user given tag. The other measures like lift, conviction and specificity can be used along with the basic measures to improve the performance. Further the measures can be optimized by applying optimization algorithms.

VII. REFERENCES

[1] Pan Youneng, Deng Sanhong,(2004) “Web Mining Research Based on XML and Association Rules”, in the journal of New Technology of Library and Information Service, Vol 112, No.7, pp.30-34.

[2] Guo Lin, (2005) “Research of Data Mining Techniques for XML Documents” [Master Thesis], Dalian University of technology.

[3] G.A.Potamias, V.S.Moustakis, (2001) “Knowledge Discovery from Distributed Clinical Data Source: The Era For Internet-Based Epidemiology”, in the Proceedings of the 23rd annual EMBS international conference, pp. 3638-3641.

[4] Daniele Braga, Alessandro Campi, Stefano Ceri, Mika Klemettinen, Pier Luca Lanzi,(2002)“A Tool for Extracting XML Association Rules from XML Documents”, in the proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence, pp.57-64.

[5] Daniele Braga, Alessandro Campi, Stefano Ceri, Mika Klemettinen, Pier Luca Lanzi,(2003)“Discovering Interesting Information in XML Data with Association Rules”, in the proceedings of ACM-SAC.

[6] Ling Feng, Tharam Dillon, (2004) “Mining XML-Enabled Association Rules With Templates”, in the Proceedings of the 3rd International Workshop on Knowledge Discovery in Inductive Databases pp.66-88.

[7] Mohammed J. Zaki, (2002) “Efficiently Mining Frequent Trees in a Forest”, in the 8th international conference ACM SIGKDD on Knowledge Discovery and Data Mining.

[8] Alexandre Termier, Marie-Christine Rousset, Michèl Sebag, (2002) “Treefinder: A First Step towards XML Data Mining”,in the Proceedings of the IEEE International Conference on Data Mining, pp.450-457.

[9] Henry Tan, Tharam S. Dillon, Fedja Hadzic, Ling Feng and Elizabeth Chang, (2005) “MB3-Miner: mining embedded sub Trees using Tree Model Guided candidate generation”, in the first International Workshop on Mining Complex Data (MCD) in conjunction with ICDM'05, pp.103-110.

[10] Li, XY, Yuan, JS & Kong, YH 2007, ‘Mining Association Rules From XML Data with Index Table’, Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, pp. 3905 – 3910.