IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 10, Issue 2 (Mar. - Apr. 2013), PP 60-64
www.iosrjournals.org

Optimization of Mining Association Rule from XML Documents

P.Jothi lakshmi[1],D.Sasikala[2]
[1](PG Scholar/CSE, Bannari Amman Institute of Technology / Anna University-Chennai, India)
[2](Associate Professor/CSE, Bannari Amman Institute of Technology / Anna University-Chennai, India)

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. The 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. On the basis of the association rule mining and Apriori algorithm, this paper optimizes the result generated by Apriori algorithm using Ant Colony Optimization (ACO) algorithm by choosing confidence value as pheromone update value. ACO is a meta-heuristic inspired by the foraging behaviour of ant colonies and it was introduced by Dorigo.*

Keywords *- Ancestor-descendant relation, Ant Colony Optimization (ACO), Association rule mining, Index Table, XML.*

I. Introduction

Data mining or Knowledge discovery in databases (KDD) is the process of extracting interesting knowledge from a huge volume 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.

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 are also different from traditional transactions and items. The definition of transaction and item is given in XML context. 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.

Some of the rules which are generated in this technique are not believable. So optimization of result is needed. To optimize these rules, propose Ant Colony Optimization (ACO) algorithm for association rule optimization. An Ant Colony Optimization algorithm is essentially a system based on agents that simulate the natural behavior of ants, admit the mechanisms of cooperation and adaptation. In solving the optimization problems with ACO consists of three major functions. By choosing any one of these functions appropriately helps the algorithm to get faster and better results.

In Section 2, we discuss related work. In Section 3, present the basic measures of association rule mining. In Section 4, the work presents XML association rules. In Section 5, Introduction of ACO algorithm. In Section 6, present proposed work and concludes in Section 7.

II. Related Work

Cornelis (2006) introduced a method on mining positive and negative association rules from large databases. An Apriori based algorithm is proposed to find all valid Positive and Negative Association Rules in a support confidence framework. Efficiency is guaranteed by taking advantage of an upward closure property that holds for the support of negative association rules under the definition of validity.

Xin-Ye Li et al. (2007) focused on mining association rules from xml data with index table. In XML context, a novel definition of transaction and item to make mining XML association rule efficient, then build transaction database using an index table. Based on the definition and the index table used for XML searching, check the include relation between a transaction and an item quickly. A higher adaptive mining technique is also described. By using it, process mining rules with no guidance of interest associations given by users and mining

unknown rules. While mining association rule from XML structure is important in most XML application, the problem of mining XML association rules from content of XML documents. There are many XML documents with same or similar structure, such as those XML documents integrated from different enterprise database or those remote clinical XML documents with same structure and so on. Mining association rule from content of these XML documents is necessary to decision making. Since 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. A new definition of transaction, item and association rule in XML context, and use an index table to select transactions and items. By using node encoding, the include relation decision easily and quickly.

Yiwu Xie et al. (2008) introduced the optimization and improvement of the apriori algorithm. There are two aspect of Apriori algorithm that affects the algorithm's efficiency. One is the frequently scanning database and other is large scale of the candidate item sets. In this paper, Apriori algorithm is proposed to reduce the times of scanning database, so the join procedure of frequent item sets is optimize in order to reduce the size of the candidate item sets. In this paper, not only decrease the times of scanning database but also optimize the process that generates candidate item sets.

Badri Patel et al. (2011) proposed optimization of association rule mining Apriori algorithm using ACO presents an ACO algorithm for the specific problem of minimizing the number of association rules. Apriori algorithm uses a user interested support and confidence value on the transaction dataset then produces the association rule set. These association rule set are in the form of discrete and continuous therefore weak rule set are required to prune. Optimization of result is needed. To optimize these rules, propose Ant Colony Optimization (ACO) algorithm for association rule optimization. Admitting the confidence value as the pheromone value and compute the path updating value then optimize association rule set is generated.

T.karthikeyan et al. (2012) focused on a study on Ant Colony Optimization with association rule. Ant miner algorithms are mainly for discovery rule for optimization based on Ant Colony Optimization (ACO). Ant miner+ algorithm uses MAX-MIN ant system for discover rules. Soil classification deals with the systematic categorization of soils based on distinguished characteristics as well as criteria. The proposed model delivers to Ant miner and Ant miner+ algorithm were applied to both training and soil dataset to Association rule and found that Ant miner+ performs better than Ant miner. This is mainly used to discover the rules for optimization. In this paper, the algorithms generate the optimized rule based on the measures of specificity and sensitivity. It is implemented using MATLAB R2010a and also explains the quality of the rule and performance evaluation based on 4 training datasets.

Rahman AliMohammadzadeh et al. proposed "Template Guided Association Rule Mining from XML Documents" In this paper, a model is based on XML-enabled association rule framework that was introduced by Feng. XML-AR framework extends the notion of associated items to XML fragments to present associations among trees rather than simple-structured items of atomic values. And also explain the practical model for mining association rules from XML documents.

III. Basic Measures of Association Rule Mining

XML document is in tree structure and in order to mine association rules from XML is different from the traditional well-structured world. An element which does not have any sub element is called terminal element. The sub tree of the xml document is said to be Transaction and leaf node of the tree is said to be item. Transaction of can be identified by the root node of the sub tree and item can be identified by using leaf node in that sub tree. 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). The two basic measures of association rule mining are support and confidence.The rules of support and confidence are defined as follows:

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

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

```
<purchase>
 <person>
      <name>lily</name>
      <gender>female</gender>
 </personal>
 <item>
              <p1>brush</p1>
              <p2> oil</p2>
  </item>
  <item>
              <p1>brush</p1>
              <p2>toothpaste</p2>
```

</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 <p1>...</p1>,<p2>..</p2>etc. The support count is measured by the percentage of XML fragments that contain the items. The confidence is measured by the percentages of XML fragments that contain an item X also contain the item Y. The association rules are in the form <p1>brush</p1> → <p2>toothpaste</p2>.

IV. XML Association Rules

The mining process includes the following steps.1) Extraction of XML transactions and items from index table.2) Generation of relational table.3) Generation and Mining of XML association using Apriori algorithm.

1. Modified Index table

All element nodes in XML documents are indexed by using Modified index table.

Modified Index_table=(docID, nodeEncode, tag, value)

Where docID represents unique id number for each XML document, nodeEncode is the encoding of a node in a document tree and its parent with a dot between them. Tag is the element tag and value is the element content. This table is efficient for the extraction of transactions, items and then include relation checking, especially from many XML documents.

2. Encoding the Node

NodeEncode is the process of encoding of a node *n* in a document tree, it involves the encoding of its parent is augmented by index of n and is separated by a dot in its siblings. A node is encoded on the basis of its occurrences in the same level. If the node occurs more than one time in a document it is encoded by using underscore (_). An XML document is in tree form and it is depicted in Fig 2.

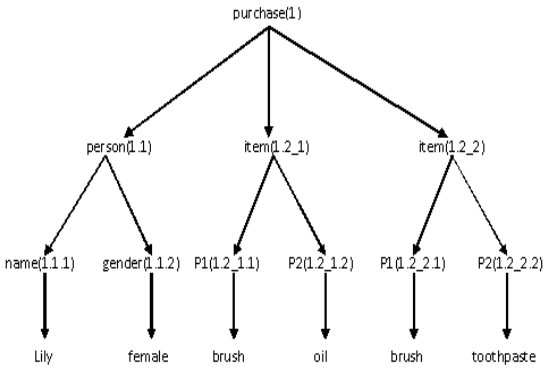

Fig.2 Node Encoding in the tree of XML document .

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 for the first occurrence in the document, it is encoded as 1.1_1 and for the second occurrence, it is 1.1_2 and so on.

3. Check the Include Relationship between Transaction and Item

A XML document is in the form of tree, a transaction is a sub tree and an item is a leaf node in the sub tree. A transaction is identified by the root node and it is the ascendant node and item is identified by its parent node with dot between them and it is descendant node. Checking include relation between a transaction and an item is used to check the ancestor-descendant relation between two element nodes.

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. According to the definition of transaction and item, if a transaction includes an item, the transaction node is the ascendant of the item node. A descendant node (item node) encoding begins with the encoding of its 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 and it is encoded as 1.2, item <p1>brush</p1>must be included in this transaction, because the encoding of this item node is "1.2.1", which begins with "1.2." and a dot between them.

4. Extraction of XML Transactions and Items

A transaction is a XML fragment and use the root node of its tree is used to indicate a transaction. Thus, records (*docID, nodeEncode*) is to represent a transaction that is selected from index table and this nodeEncode is used to check the include relationship between the transaction and items. The descendant of transaction nodes from modified index table is selected by checking the include relation between transaction and item. The value of (*docID, nodeEncode, tag, value*) is used to represent an item node.

Here, same tag and value are extracted by item node; Imagine, if they are the same item then two item nodes have same value and same tag. The duplicate items are removed (same value and same tag) and record all the different items in relational table. The values that stored in the relational table are in the form of 0's and 1's.With the transaction sets and item sets, the relational table *R* is generated as follows:

1) Columns are used to store XML items; Rows are used to store XML transactions.
2) If the ith transaction includes the jth item, then R (i, j) =1 else R (i, j) =0.

The interruption technique is used to skip many searching steps by comparing times of the include relation in the extraction step. 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.

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

5. Generation and Mining of XML Association Rules

The XML association mining algorithm is based on the plain Apriori algorithm. Sample XML documents are pre-processed and store in the modified index table. This pre-processing is efficient for the extraction of transactions and items.Modified index table stores the docID, nodeEncode, tag and value. So the plain Apriori algorithm takes input as modified index table and generates the output as XML association rules on the basis of the two measures of association rule mining. In the final, association rules extracted from index table are mapped into the XML representation. Here some of the weak rules are generated.

Example by considering many transactions of purchase details as in Fig 2.

<p1>brush</p1> → <p2>toothpaste</p2>

Customers who buy brush also buy toothpaste with a 90% confidence. The combination of toothpaste and brush has a support of 30%.In practical; Customers can buy both brush and toothpaste together. So it considered as promising rule.

<p1>brush</p1> → <p2>oil</p2>

Customers who buy brush also buy oil with a 75% confidence. The combination of oil and brush has a support of 30%. In practical, customers cannot buy both brush and oil together. It appears together in rare case. Therefore, it considered as weak rule. So the weak rules generated by Apriori algorithm are required to prune. Optimization of result is needed. To optimize these rules using Ant Colony Optimization (ACO) algorithm for association rule optimization.

V. Introduction of ACO Algorithm

ACO algorithms were inspired from natural behavior of ant colonies. ACO has been applied successfully to numerous hard optimization problems including the traveling salesman problem. Artificial ants are simple agents implementing constructive heuristics. The basic idea of constructive heuristics is incrementally construct solutions by adding, in each step, a solution component to a partial solution until to a complete solution is formed. The cooperation is the key element of ACO algorithms once good solutions are resulted of the cooperative interaction of several artificial ants during the construction of solutions.ACO has been applied on hard problems to find an optimal object from a finite set. Problems are defined in terms of states and components, which are sequences of components. ACO iteratively generates solutions paths in the space of such components, adding new components to a state.

ACO contains two rules:

1. Local pheromone update rule, which applied while constructing solutions.
2. Global pheromone updating rule, which applied after constructed the solution of all ants.

Furthermore, an ACO algorithm includes two mechanisms: trail evaporation and daemon actions. All trail values are decreases over time in trail evaporation, to avoid unlimited accumulation of trails over some component. The centralized actions are implemented by using daemon actions which cannot be performed by single ants. From a non-local perspective whether to bias the search process can be decided by the invocation of a local optimization procedure, or the update of global information. At each step, each ant computes a set of possible expansions, to its current state and moves to one of the probability.

The probability distribution is as follows.

The probability of moving k ants from state t to state n depends on the combination of two moves:

1. The attractiveness of the move, as computed by some heuristic denoting the prior desirability of that move;
2. The trail level of the move, denoting how proficient it has been in the past to make that particular move: it represents therefore a posterior indication of the desirability of that move.

VI. Proposed Work

This work presents an ACO algorithm for the specific problem of minimizing the number of association rules. Apriori algorithm uses support and confidence value based on user interest then produces the association rules for the dataset. These association rule set are in the form of discrete and continues therefore weak rule set are required to prune. Optimization of result is needed.

To optimize these we are going to propose ACO algorithm for association rule optimization. An Ant Colony Optimization (ACO) algorithm is essentially a system based on agents that simulate the natural behavior of ants, admits the mechanisms of cooperation and adaptation. The optimization problems with ACO System can be solved by three major functions. Choosing any of the below functions appropriately helps the algorithm to get better and faster results.

The problem-dependent heuristic function (η) is a first function which measures the quality of items that can be added to the current partial solution. During the algorithm, the heuristic function remains unchanged. A rule for pheromone updating, which specifies how to adapt the pheromone trail (τ), and probabilistic transition rule based on the value of the heuristic function and on the contents of the pheromone trail that is used to recursively construct a solution.

The steps of the proposed work includes

1. Random candidate subset selection
2. Initialize support and confidence
3. Generate frequent item set-1
4. Update support value using Updated Selection Measure (USM)
5. Generate frequent item set-2
6. Update Confidence value
7. Association rules will be generated using Apriori algorithm

By using the confidence value as the pheromone (p) value and compute the path updating value (P) $=P+\Delta t$, Where $\Delta t = (2 +1-1)/d$, (d-Number of transaction set), then optimize association rule set is generated.

VII. Conclusion

In this paper, propose an ACO algorithm for optimization association rule generated using Apriori algorithm. This work describes a method for the problem of association rule mining. An ACO algorithm is proposed in order to minimize number of association rules and make the rule as a promising rule.

References

Journal Papers:

[1] Al-Ani Ahmed, "Feature Subset Selection Using Ant Colony Optimization", International Journal of Computational Intelligence (IJCI), vol. 2, No. 1, pp. 53-58.

[2] Nada M. A. AL-salami, Saad Ghaleb Yaseen, "Ant Colony Optimization", International Journal of Computer Science and Network Security (IJCSNS), vol.8 No.6, pp 351-357, June,(2008).

[3] Yiwu Xie, Yutong Li, Chunli Wang, MingyuLu, "The Optimization and Improvement of the Apriori Algorithm", International Symposium on Intelligent Information Technology Application Workshops, IEEE, (2008).

[4] Yan-hua Wang, Xia Feng, "The Optimization of Apriori Algorithm Based on Directed Network", Third International Symposium on Intelligent Information Technology Application, IEEE, (2009).

Books:

[5] M Dorigo, T Stutzle, Ant Colony Optimization(The MIT press Cambridge, MA).

Proceedings Papers:

[6] Rakesh Agrawal, Ramakrishnan, Fast Algorithms for Mining Association Rules, Proceedings of the 20th VLDB Conference Santiago, (1994).

[7] Marco Dorigo, Christian Blum,An Efficient Algorithm for Mining Association Rules in Large Databases, Proceedings of the 21st VLDB Conference, Switzerland, (1995).

[8] Charu C. Aggarwal, Philip S. Yu, Online Generation of Association Rules, Proceedings of 14th International Conference, IEEE, February,(1998).

[9] Gao, Shao-jun Li,A method of Improvement and Optimization on Association Rule Apriori algorithm, Proceedings of the 6th conference on intelligent control and automation, (2006).