Rules Extraction in XML Using Correlation

Sheenu Toms[*]**, Deepa John**[**]

[*] Department Of Computer Science & Engg, Rajagiri School of Engineering & Technology
[**] Department Of Computer Science & Engg, Rajagiri School of Engineering & Technology

Abstract- As the amount of digital information on the internet grows it is more and more critical to retrieve information from semistructured documents. Also the data returned as answers to queries may not give interpretable knowledge. Therefore the concept of association rule mining is introduced in XML datastructure. Even if it is effective occasionally it give unexpected results. This paper propose a new method based on correlation which will more effectively return the rules. Correlation is used to find out relationship between different data items.

Index Terms- Approximate query answering, Association Rule Mining, correlation, XML

I. Introduction

Now a day's XML [1] is used in all areas of internet applications as a storage unit because of its flexibility and portability. Increasing semistructured[2] and unstructured[3] data in internet made the database research field to concentrate on XML since conventional database can only be used for storing structured data.

Despite of its popularity, XML still lack appropriate techniques to retrieve data effectively due to the semistructured or unstructured nature. Since query answering should be effective for only structured data there is a possibility to return unexpected results in the case of XML, because 50 percent of the documents in internet do not have a specific structure.

The man problems arise while querying the XML data are:

- Information Overload [4].
- Information Deprivation [5].

So it is appropriate to know about the structure and semantics characteristics of dataset. Thus researchers tried to incorporate a data mining technique called association rule mining [6] for XML data for this purpose. The idea of using association rules can be seen in many papers and here we are going to propose a new data mining technique which is more effective than association rule called correlation. Correlation is an alternative method to find the relationships between data in XML and it can be seen as a lift of an association rule.

This paper is structured as follows: In section II we give brief outline of the works done in XML mining. Section III contains the proposed method and section IV will conclude this paper.

II. Research

XML store data in a tree like structure. So mining data from the XML document is more difficult than in the case of traditional database.

Association rule mining was first proposed for conventional database and then it came to XML. Association rule mining is proposed in [6] and it has many applications such as Basket data analysis, cross-marketing, catalog design, loss-leader analysis, clustering, classification, etc.

The main steps in association rule mining are:

1. Find the frequent items greater than a minimum support from the data given.
2. Find the rules from the frequent items which satisfies a minimum confidence.

Before going on to the support and confidence we can see how the association rule look like. Consider supermarket in which lots of transactions taking place. The transaction Database is given in Table 1.

Table 1: Example for a Transaction Database

Transaction id	Items purchased
1	Shampoo, Conditioner
2	Hair Oil
3	Hair Oil, Shampoo ,Conditioner
4	Shampoo
5	Shampoo, Conditioner

From the analysis of Fig 1 we can conclude the result as follows:

Table 2: Analyzed Data from Table 1

Items	No. Of occurrences
Shampoo	4
Shampoo, Conditioner	3
Shampoo, Hair Oil	1
Shampoo, Hair Oil, Conditioner	1
Hair Oil	2

From Table 2 we can formulate the following association rules:

Rule 1: Buys(Shampoo) → Buys(Conditioner) [3/5,0.75]
Rule 2: Buys(Shampoo) → Buys(Hair Oil) [1/5,0.25]
Rule 3: Buys(Hair Oil) → Buys(Conditioner) [1/5,0.5]

We can generate many rules from Table 2 and the above rules are just some of them. In these rules we are given some numerical values which are the calculated support and confidence of the given data.

Support corresponds to the occurrence of an element or set of elements and confidence corresponds to the conditional probability of finding an element having found another element.

For *Rule 1* the support is 4 which means that out of the 5 transactions, 4 of them contain Shampoo. Confidence for this rule is 0.6 indicates the probability of buying conditioner with shampoo.

So we can generalize the association rule as follows:

$A \Rightarrow B$ (If A then B)

Support = P(AUB) = No. of tuples containing both A & B / total number of tuples.
Confidence = P(B/A) = P(AUB)/P(A).

These are the general concepts of finding association rules in transaction databases.

In order to find the frequent items and create association rules researchers developed many mining algoritms. The main basic algorithm in association rule mining is Apriori Algorithm [7]. This algoritm is based on the property that “any subset of frequent itemset must be frequent” called Apriori property. Other main algoritms used in association rule mining are FP tree algorithm [8], Predictive Apriori [9], Tertius [10], and GSP [11]. Above explanations are all based on the database concern. When it comes to XML all the basic concepts still remains same, but we have to work on a tree like structure ie we have to find out the frequent sub trees instead of frequent items, and the support or confidence are calculated in sub trees. Main algoritms used for mining in XML are TreeMiner [12], PathJoin [13], FREQT [14], DRYADE [15], DRYADEPARENT [16], CMTTreeMiner [17] and POTMiner [18]. These algorithms extract the frequent subtrees in the XML document. Experiments shows that DRYADEPARENT is currently the fastest algorithm and CMTTreeMiner the second with respect to efficiency.

In the case of XML the rule can be as follows:

$A \Rightarrow B$ (If A then B), where A and B are trees.

This is called Tree Association Rule(TAR). TAR describes the co-occurrence of two trees A and B. Always A should be a subtree of B.

The support and confidence of TAR is:
Support = Count(B, T) / Cardinality(T).

Confidence = Count(B, T) / Count(A, T) where T is the tree representation of the corresponding XML document. Cardinality(T) represent the total no. of nodes in XML document, Count(B,T) denotes the no. of occurrence of subtree B in tree T.

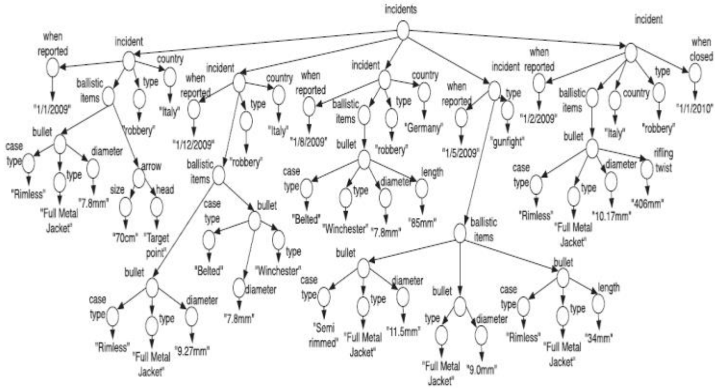

Figure 1: incidents.xml [19]

The rules extracted and stored should satisfy a minimum support and confidence. The minimum support and confidence can be given prior to rules extraction. Greater the minimum values of support and confidence greater the effectiveness of the rule.

Let us consider the example of “incidents.xml” in the odyssey dataset in Fig 1 [19]. It consists of crime scene information across Europe. Experts have used association rule mining in it.

Some of the rules mined from this xml document are given in Fig 2 [19].

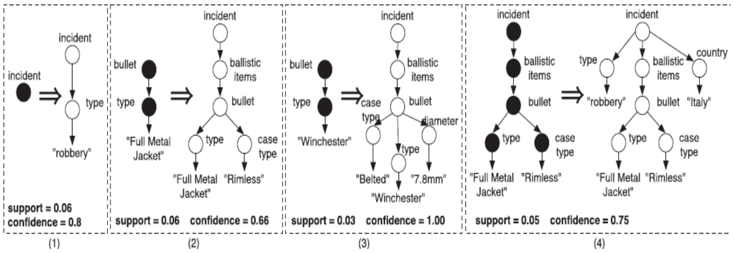

Figure 2: Some rules extracted from Fig 1 [19]

Rule 1 in Fig 2 says that 80 percent of the incidents are robberies. Rule 2 states that 66 percent of Full Metal Jacket Bullets have rimless cases. Rule 3 states that 100 percent of the Winchester bullets have 7.8mm diameter and a belted case. Finally rule 4 states that 75 percent of the incidents involving rimless Full Metal Jacket bullets are robberies happening in Italy. So by analysing these rules we can have a different view of an XML document that may give many useful information’s to ordinary users.

The rules extracted are now stored in the XML format for the convenience of querying, ie the same language that we are using for querying XML document can also used with the rules for information retrieval.

III. PROPOSED METHOD

Support-confidence framework is not so efficient in rule mining. Because it identifies a rule (A=>B) as strong even if the occurrence A might not imply the occurrence of B. Correlation is an alternative to this framework which is more effective in rule mining.

A. Correlation Concepts

Let A and B are two subtrees. The occurrence of A is independent of the occurrence of B iff

$$P(A \cup B) = P(A) \cdot P(B)$$

Otherwise A and B are dependent and correlated. The measure of correlation, or correlation between A and B is given by the formula:

$$Corr(A,B)= P(A \cup B) / P(A) . P(B)$$

- corr(A,B) >1 means that A and B are positively correlated i.e. the occurrence of one implies the occurrence of the other.
- corr(A,B) < 1 means that the occurrence of A is negatively correlated with (or discourages) the occurrence of B.
- corr(A,B) =1 means that A and B are independent and there is no correlation between them.

B. Association and correlation

The correlation formula can be re-written as:

$$Corr(A,B) = P(B|A) / P(B)$$

We know that

$$Support(A \rightarrow B)= P(AUB)$$
$$Confidence(A \rightarrow B)= P(B|A)$$

Confidence can be rewritten as

$$Confidence(A \rightarrow B)= corr(A,B)\ P(B)$$

Correlation, support and confidence are all different. Correlation provides an extra information about the association rule (A → B).We can say that the correlation corr(A,B) provides the LIFT of the association rule (A=>B), i.e. A is said to increase (or LIFT) the likelihood of B by the factor of the value returned by the formula for corr(A,B).

C. Correlation rules

Correlation concepts & rules can be used to further support our derived association rules**.** Consider the rule 1 in Fig 2. The correlation values for this rule is

Rule 1: Corr(A,B)=P(B/A)/P(B)
P(B) = 4/62 = 0.06 (from Fig 1)

Corr(A,B) = 0.8/0.06 = 13.33 > 1, means that in rule 1 the two subtrees are positively correlated and this rule is strong.

Similarly we can find whether the rules extracted are strong or not. If the correlation value for a rule is found to be equal or less than 1 then it is discarded.

IV. CONCLUSION

Correlation Analysis provides an alternative framework for finding interesting relationships, or to improve understanding of meaning of some association rules ie it is a lift of an association rule. There has been many researches going on in the area of finding more effective mining algorithms for XML. The proposed method will effectively mine XML.

REFERENCES

[1] World Wide Web Consortium, Extensible Markup Language (XML) 1.0, http://www.w3C.org/TR/REC-xml/, 1998.

[2] Peter Buneman, "Semistructured Data", Department of Computer and Information Science University of Pennsylvania.

[3] Abidin, S.Z.Z. ; Idris, N.M. ; Husain, A.H. "Extraction and classification of unstructured data in WebPages for structured multimedia database via XML", Information Retrieval & Knowledge Management, (CAMP), 2010**.**

[4] Yongming Guo Dehua Chen Liangxu Liu, Jiajin Le , " A Frame of Per Personalized Information Filtering System Based on XML" Networked Computing and Information Management, Volume: 1, 2008. NCM '08.

[5] R.Sree Lekshmi, B. Sasi kumar, " Extracting Information from Semistructured XML using TARs",International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-1, October 2012.

[6] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," Proc. 20th Int'l Conf. Very Large Data Bases, pp. 478-499, 1994.

[7] Rakesh Agrawal, RamaKrishnan Srikant, "Fast Algorithms for mining Association Rules", Proceedings of the 20th VLDB Conference Santiago, Chile, 1994.

[8] Florian Verhein,"FP growth Algorithm an Introduction", Copyright 2008 Florian verhein January 10 2008.

[9] Bo Wu , Defu Zhang , Qihua Lan, Jiemin Zheng, "An Efficient Frequent Patterns Mining Algorithm Based on Apriori Algorithm and the FP- Tree Structure" Convergence and Hybrid Information Technology, 2008. ICCIT '08.

[10] Sunitha B.Aher, Labo L.S.D.R.S, "A Comparative Study of Association Rule Algoritms for Course recommender System in E-learning", International Journal of Computer Applications (0975-8887), volume 39-No 1,Feb 2012.

[11] Chenngguan Xiang, Guizhou Normall Coll, Guiyang, "The GSP Algorithm in Dynamic Cost Prediction of Enterprise", NaturalComputation (ICNC), Volume 4, Seventh International Conference on 2011.

[12] M.J. Zaki, "Efficiently Mining Frequent Trees in a Forest: Algorithms and Applications," IEEE Trans. Knowledge and Data Eng., vol. 17, no. 8, pp. 1021-1035, Aug. 2005.

[13] Y. Xiao, J.F. Yao, Z. Li, and M.H. Dunham, "Efficient Data Mining for Maximal Frequent Subtrees," Proc. IEEE Third Int'l Conf.Data Mining, pp. 379-386, 2003.

[14] T. Asai, K. Abe, S. Kawasoe, H. Arimura, H. Sakamoto, and S. Arikawa, "Efficient Substructure Discovery from Large Semi- Structured Structured Data," Proc. SIAM Int'l Conf.Data Mining, 2002.

[15] A.Termier, M. Rousset, and M. Sebag, "Dryade: A New Approach for Discovering Closed Frequent Trees in Heterogeneous Tree Databases," Proc. IEEE Fourth Int'l Conf. Data Mining, pp. 543-546, 2004.

[16] A.Termier, M. Rousset, M. Sebag, K. Ohara, T. Washio, and Hotoda, "DryadeParent, an Efficient and Robust Closed Attribute Tree Mining Algorithm," IEEE Trans. Knowledge and Data Eng., vol. 20, no. 3,pp.300-320, Mar. 2008.

[17] Y. Chi, Y. Yang, Y. Xia, and R.R. Muntz, "CMTreeMiner: Mining both Closed and Maximal Frequent Subtrees," Proc. Eighth Pacific-Asia Conf. Knowledge Discovery and Data Mining, pp. 63-73, 2004.

[18] A. Jime´nez, F. Berzal, and J.C. Cubero, "Mining Induced and Embedded Subtrees in Ordered, Unordered, and Partially- Ordered Trees," Proc. 17th Int'l Symp. Methodologies for Intelligent Systems, pp. 111-120, 2008.

[19] Mirjana Mazuran, Elisa Quintarelli, and Letizia Tanca, "Data Mining for XML Query-Answering Support", Ieee Transactions On Knowledge And Data Engineering, Vol. 24, No. 8, August 2012 .

AUTHORS

First Author – Sheenu Toms, doing M.Tech, Rajagiri School of Engineering & Tech, Kakkanad, Kerala, India, sheenutoms@gmail.com

Second Author – Deepa John, Asst. Professor, Rajagiri School of Engineering & Tech, Kakkanad, Kerala, India, sheenutoms@gmail.com

Correspondence Author – Sheenu Toms, sheenutoms@gmail.com, sheenutoms@yahoo.com