ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 9, March 2014

Intensional Query Processing for XML Documents

Neha Manvendra shroff ,Ganesh V. Gujar

Abstract*— The increasing amount of XML datasets available to users increases the necessity of investigating techniques to extract knowledge from these data. Data mining is widely applied in the database research area in order to extract frequent correlations of values from both structured and semistructured datasets. In this work we describe an approach to mine Tree-based association rules(TAR) from XML documents. Such rules provide information on both the structure and the content of XML documents; moreover, they can be stored in XML format to be queried later on. The mined knowledge is approximate, intentional knowledge used to provide: (i) Quick, approximate answers to queries and (ii) Information about structural regularities that can be used as data guides for document querying.

***Index Terms*—Data Mining, Intensional Knowledge, TAR Querying, XML,**

I. INTRODUCTION

XML is a standard for Web data, XML query processing takes a particular importance. The efficient exploitation of XML documents has attracted significant attention since the publication of the XML standard in 1998. More recently the problem has been investigated in the XML context [2], [3], [4], [5], [6], [7], [8]. In [9] authors use XQuery [10] to extract association rules from simple XML documents. XML queries can now be evaluated by mainstream relational database engines extended to support the XML data type and the XQuery language, as well as by in-memory processors. In order for query formulation to be effective users need to know this structure in advance, which is often not the case. In fact, it is not mandatory for an XML document to have a defined schema: 50% of the documents on the web do not possess one [1]. When users specify queries without knowing the document structure, they may fail to retrieve information which was there, This paper addresses the need of getting the gist of the document before querying it, both in terms of content and structure. Discovering recurrent patterns inside XML documents provides high-quality knowledge about the document content: frequent patterns are in fact intensional information about the data contained in the document itself, that is, they specify the document in terms of a set of properties rather than by means of data

A. Fundamental Concepts

Here we are proposing a method for mining and storing TARs (Tree-based Association Rules) as a means to represent intensional knowledge in native XML. Intuitively, a TAR represents intensional knowledge in the form SB ⇒ SH, where SB is the body tree and SH the head tree of the rule and SB is a sub tree of SH. procedure is characterized by the following key aspects: a) it works directly on the XML documents, without transforming the data into any intermediate format, b) it looks for general association rules, without the need to impose what should be contained in the antecedent and consequent of the rule, c) it stores association rules in XML format, and d) it translates the queries on the original dataset into queries on the TARs set

II. TAR EXTRACTION

TAR mining is a process composed of two steps: 1) mining frequent sub trees, that is, sub trees with a support above a user-defined threshold, from the XML document; 2) computing interesting rules, that is, rules with a confidence above a user-defined threshold, from the frequent sub trees. Once the mining process has finished and frequent TARs have been extracted, they are stored in XML format. This decision has been taken to allow the use of the same language (XQuery) for querying both the original dataset and the mined rules. One of the (obvious) reasons for using TARs instead of the original document is that processing iTARs for query answering is faster than processing the document. To take full advantage of this, we introduce indexes on TARs to further speed up the access to mined trees – and in general of intentional query answering. In the literature the problem of making XML query-answering faster by means of path-based indexes has been investigated. In general, path indexes are proposed to quickly answer queries that follow some frequent path template, and are built by indexing only those paths having highly frequent queries. We start from a different perspective: we want to provide a quick, and often approximate, answer also to casual queries.

III. INTENSIONAL PROCEDURE

iTARs provide an approximate intentional view of the content of an XML document, which is in general more concise than the extensional one because it describes the data in terms of its properties, and because only the properties that are verified by a high number of items are extracted. A user query over the original dataset can be automatically transformed into a query over the extracted iTARs. The answer will be intentional, because, rather than providing the set of data satisfying the query, the system will answer with a set of properties that these data “frequently satisfy”, along

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 9, March 2014

with support and confidence. There are two major advantages: i) querying iTARs requires less time than querying the original XML document; ii) approximate, intentional answers are in some cases more useful than the Extensional ones. The classes of queries that can be managed with our approach have been introduced in and further analyzed in the relational database context in. Here are some simple examples for four classes of queries, discussed in the following.

Class 1: This kind of query is used to impose a simple, or complex (containing AND and OR operators), restriction on the value of an attribute or the content of a leaf node.

Class 2: This kind of query is used to retrieve some properties described in the subtrees rooted in a specified element, possibly ordering the result

Class 3: This kind of query is used to count the number of elements with a specific content. This paper uses an association rule whose body matches the query conditions, and obtain as answer.

Class 4: This kind of query is used to select the best k answers satisfying a counting an grouping condition, for example Retrieve the k authors who wrote the highest number of articles".

IV. RELATED WORK

The problem of association rule mining was initially proposed in Agrawal (R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases) and successively many implementations of the algorithms, downloadable from B.Goethalsand M.J.Zaki. Advances in frequent item set mining, were developed and described in the database literature, Weka 1 being a known framework. More recently the problem has been investigated also in the XML context "Discovering interesting information in xml data with association rules", "Extracting association rules from xml documents using XQuery" and "A new method for mining association rules from a collection of xml documents". In "Discovering interesting information in xml data with association rules" to extract association rules from simple XML documents. They propose a set of functions written only in XQuery which implement together the Apriori algorithm. It is show that their approach performs well on simple XML documents; however it is very difficult to apply this proposal to complex XML documents with an irregular structure. This limitation has been overcome in "Extracting association rules from xml documents usingXQuery", where the authors introduce a proposal to enrich XQuery with data mining and knowledge discovery capabilities, by introducing XMINE RULE, a specific operator for mining association rules for native XML documents. They formalize the syntax and an intuitive semantics for the operator and propose some examples of complex association rules. However, the operator proposed uses the MINE RULE operator, which works on relational data only. This means that, after a step of pruning of unnecessary information, the XML document is translated into the relational format. Moreover, both "Discovering interesting information in xml data with association rules" and "Extracting association rules from xml documents using XQuery" force the designer to specify the structure of the rule to be extracted and then to mine it, if possible. This means that the designer has to specify what should be contained in the body and head of the rule, i.e. the designer has to know the structure of the XML document in advance, and this is an unreasonable requirement when the document has not an explicit DTD. Another limitation of these approaches is that the extracted rules have a fixed root, thus once the root node of the rules to mine has been fixed, only its descendants are analyzed. Consider the dataset given in figure 1

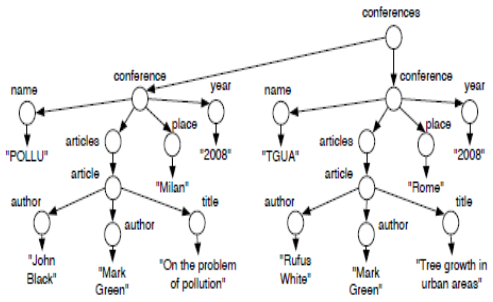

Fig 1:-confernce.xml

In order to infer the co-author relationship among authors of conferences it is necessary to x the root node of the rules in the article element, the body and head in author. In such way it is possible to learn that "John Black" and "Mark Green" frequently write papers together. However, it is not possible to mine item sets stating that frequently, during "2008" conferences have been held in "Milan". Indeed, to mine such property the body of the rules should be fixed in the year element, which is not contained in the sub-tree of the article node, and the head in place. Our idea is to take a more general approach to the problem of extracting association rules from XML documents, i.e. to mine all frequent rules, without having any apriori knowledge of the XML dataset. A similar idea was presented in "A new method for mining association rules from a collection of xml documents" where the authors introduced HoPS, an algorithm for extracting association rules in a set of XML documents. Such rules are called XML association rules and are implications of the form X. Y, where X and Y are fragments of an XML document. In particular the trees X and Y have to be disjunct. The limitation of this proposal is that it does not contemplate the possibility to mine general association rules within a single XML dataset, while achieving this feature is one of our goals. The idea of using association rules as summarized representations of XML documents was also introduced where the XML summary is based on the extraction of association rules both on the structure (schema patterns) and on content values (instance patterns) from XML datasets. The limitation of such an approach is that the so called schema patterns, used to describe general properties of the schema applying to all instances, are not mined, but derived as an abstraction of

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 9, March 2014

similar instance patterns. In our work, XML association rules are mined starting from frequent sub trees of the tree-based representation of a document. In the database literature it is possible to and many proposals of algorithms to extract frequent structures from tree/graph-based data structures. Just to cite some of them, Tree Miner, Path Join, Close Graph propose algorithms to directly mine frequent item sets not association rules-from XML documents. Tree Miner and Close Graph do not preserve the exact structure of the item sets extracted -only the "descendant-of" (and not the "child-of") relationship between nodes is preserved -whereas Path Join does. In this work we propose an algorithm that extends Path Join to mine generic tree-based association rules directly from XML documents.

V. TREE RULER PROTOTYPE

Tree Ruler is a prototype tool that integrates all the functionalities proposed in our approach. Given an XML document, the tool is able to extract intensional knowledge, and allows the user to compose traditional queries as well as queries. In particular, given an XML document, it is possible to extract Tree-based rules and the corresponding index file. The user formulates XQuery expressions on the data, and these queries are automatically translated in order to be executed on the intensional knowledge. The answer is given in terms of the set of Tree-based rules which react the search criteria. Integrates the functionalities proposed in our approach. Given an XML document, it enables users to extract intensional knowledge and compose traditional queries as well as queries over the intensional knowledge, receiving both extensional and intensional answers. Users formulate XQueries over the original data, and queries are automatically translated and executed on the intensional knowledge. The tool is composed by several tabs for performing different tasks. In particular, there are three tabs:

a) Get the Gist allows intensional information extraction from an XML document, given the support, confidence and the files where the extracted TARs and their index are to be stored.

b) Get the Idea allows showing the intensional information as well as the original document, to give users the possibility to compare the two kinds of information. Get the Answers allows querying the intensional knowledge and the original XML document. Users have to write an extensional query.

VI. RESULTS

After Applying association rules mining on XML document the result that we get is as follows:

```
<itemset>
  <item>pen</item>
  <support>4</support>
</itemset>
<itemset>
  <item>notebook</item>
  <support>4</support>
</itemset>
<itemset>
  <item>ink</item>
  <support>4</support>
</itemset>
<itemset>
  <item>pencile box</item>
  <support>3</support>
</itemset>
.....
<itemset>
  <item>pen</item>
  <item>notebook</item>
  <support>4</support>
</itemset>
<itemset>
  <item>notebook</item>
  <item>ink</item>
  <support>3</support>
</itemset>
<itemset>
  <item>pen</item>
  <item>ink</item>
  <support>3</support>
</itemset>
<itemset>
  <item>pencile box</item>
  <item>notebook</item>
  <support>3</support>
</itemset>
<itemset>
  <item>pencile box</item>
  <item>pen</item>
  <support>3</support>
</itemset>
```

VII. CONCLUSION

The main goals to achieve in this work are: 1) mine all frequent association rules without imposing any a-priori restriction on the structure and the content of the rules; 2) store mined information in XML format; 3) use extracted knowledge to gain information about the original datasets. performed four types of experiments: 1) time required for the extraction of the intensional knowledge from an XML database; 2) time needed to answer intensional and extensional queries over an XML file; 3) a use case scenario on the XML database, in order to monitor extraction time given a specific support or confidence; 4) a study of the accuracy of intensional answers with TreeRuler.

REFERENCES

[1] D. Barbosa ,L. Mignet, and P. Veltri. Studying the xml web: Gathering statistics from an xml sample. World Wide Web, 8(4):413–438, 2005.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In Proc. of the 20th Int.

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 9, March 2014

Conf. on Very Large Data Bases, pages 487–499. Morgan Kaufmann Publishers Inc., 1994

[3] D. Braga, A. Campi, S. Ceri, M. Klemettinen, and P. Lanzi. Discovering interesting information in xml data with association rules. In Proc. Of the ACM Symposium on Applied Computing, pages 450–454, 2003.

[4] J. W. W. Wan and G. Dobbie. Extracting association rules from xml documents using xquery. In Proc. of the 5th ACM Int. Workshop on Web Information and Data Management, pages 94–97. ACM Press, 2003.

[5] J. Paik, H. Y. Youn, and U. M. Kim. A new method for mining association rules from a collection of xml documents. In Proc. of Int. Conf. on Computational Science and Its Applications, pages 936–945, 2005.

[6] L. Feng, T. S. Dillon, H. Weigand, and E. Chang. An xml-enabled association rule framework. In Proc. of the 14th Int. Conf. on Database and Expert Systems Applications, pages 88–97, 2003.

[7] H. C. Liu and J. Zeleznikow. Relational computation for mining association rules from xml data. In Proc. of the 14th ACM Conf. on Information and Knowledge Management, pages 253–254, 2005.

[8] K. Wang and H. Liu. Discovering structural association of semi structured data. IEEE Transactions on Knowledge and Data Engineering, 12(3):353–371, 2000.

[9] J. W. W. Wan and G. Dobbie. Extracting association rules from xml documents using xquery. In Proc. of the 5th ACM Int. Workshop on Web Information and Data Management, pages 94–97. ACM Press, 2003.

[10] World Wide Web Consortium. XQuery 1.0: An XML query language, 2007. http://www.w3C.org/TR/xquery.

AUTHOR BIOGRAPHY

Neha shroff a final year student of MTECH CSE .Working in Savitribai Phule Women's Engineering College in CSE department. Area of interest is Data Mining

Mr.Ganesh Gujar is a final year student of ME (CSE) in govt. Engg. College Aurangabad. . Working in Hi-Tech Institute of Technology Aurangabad Maharashtra in CSE department. His area of interest is Data structure, Computer network, Operating System, Mainframe system.