International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

An Improved Web Mining Technique to Fetch Web Data Using Apriori and Decision Tree

Rupinder Kaur[1], Kamaljit Kaur[2]

[1]SGGSW University, Department of Computer Science and Engineering,
Fatehgarh Sahib, Punjab, India 140406

[2]Assistant Professor, Department of Computer Science and Engineering, SGGSW University,
Fatehgarh Sahib, Punjab, India 140406

Abstract: ***World Wide Web is the largest source of information. Most of the data on the web is dynamic and is in unstructured form. It is becoming difficult to get the relevant data from the web. Data Mining is the field of computer science which is used to extract knowledge from very large amount of data. Web mining is the application of data mining, which implements various techniques of data mining to get the efficient knowledge from the web data. In past time, most of the websites were developed using HTML but HTML has many limitations like limited tags, not case sensitive and designed to display data only, Web developers has now started to develop Web pages on emerging Web Technologies like XML, Flash etc. XML was designed to describe data and to focus on what the data is. XML also plays the role of a meta- language and allows authors to create customized markup language for different types of documents, making it a standard data format for online data exchange. To date, famous algorithms like Apriori and FP- Growth algorithms are used to fetch the web data for XML contents. In the proposed paper, a hybrid approach is used to fetch HTML as well as XML contents from a web page. In the hybrid approach, Apriori algorithm is used to remove the unimportant information from the contents and Decision tree is used to fetch the contents from a web page. Various factors like execution time, precision, recall and f-measure and g-measure are calculated.***

Keywords: Web Mining, XML, Apriori, Decision Tree.

1. Introduction

Web is the largest source of information. As the number of documents grows, searching for information is turning into a cumbersome and time consuming operation. Data Mining is the field of computer science which is used to extract information from the large amount of data. Web Mining is the application of data mining which is used to generate patterns from the web. Patterns must be such that they are easily understandable, useful and novel. Various techniques of data mining are used to extract the information from the web. Not only data mining but also other tools from fields of artificial intelligence, machine learning, natural language processing can also be used efficiently to fetch web data. It is very wide area of research for the researchers because of the growing use of the web. Web Mining on the basis of type of data to be explore can be divided into three main categories-

1.1 Web Usage Mining

It is the mining of the user preferences while user is navigating through the websites. This is done by applying the mining process on the log files repository. [6] By web usage mining, commercial websites take advantage of knowing the usage pattern of customer, their behaviour and frequency of their visits.

1.2 Web Structure Mining

The Web page structure consists of a Web page as a node and hyperlinks as edges connecting to other pages. [6] In other words, it works in the form of graphs. It focuses on the connectivity of the web site to other sites that are called as hyperlinks.

1.3 Web content Mining

Web content mining is the process of mining of contents of a web page. Contents of a web page may include text, image, audio, video and semi-structured records in the form of html and xml contents which are either embedded in the web page or having links to other pages.

Most of the data on the web is in unstructured form i.e. in the form of free text, images, audio, video and semi-structured form like HTML and XML etc. Since HTML has many limitations like limited tags, not case sensitive and designed to display data only, Web developers has started to develop Web pages on emerging Web Technologies like XML, Flash etc. [4] XML was designed to describe data and to focus on what the data is. XML also plays the role of a meta-language and allows document authors to create customized mark-up language for limitless different types of documents, making it a standard data format for online data exchange. This growing use has raised need for better tools and techniques to perform mining on XML too. In the proposed paper, a hybrid approach is used to fetch XML contents from XML file. Rest of the paper is organised as follows. Section 2 presents the work related to the topic. Section 3 covers the proposed work. Section 4 gives results of the work and Section 5 concludes the paper.

2. Related Work

Qin Ding et al. [3] has studied two algorithms for association rule mining from the XML data. They studied the performance of implementation on three transactional datasets created randomly. A Java-based implementation of the Apriori and the FP-Growth algorithms for this task was

done and performances were compared. They also compare the performance of java based implementation of Apriori with an XQuery-based implementation. XQuery is an XML query language as SQL for relational databases. This query language addresses the need for the ability to intelligently query XML data sources.

Buhwan Jeong et al. [11] explained the use of kernel methods for XML mining. The kernel methods for structured data are applicable to XML mining in various ways. The string kernels with minor modifications are used to compare two pieces of context information (i.e., successive element labels in a path). Undoubtedly, the VSM also provides the same state of-the-art performance in XML content mining as in traditional text mining. The tree kernels are used to measure the structural similarity between XML schema documents and between XML instance documents. By transforming the tree structure into a string (e.g., by a depth-first traversal), the string kernels and the VSM are also used to compute the structural similarity. The kernels are used to measure the structural similarity between XML schema documents and between XML instance documents by transforming the tree structure into a string. Since kernels are mainly developed for bio/chemistry -informatics, new variant kernels for texts (and XML contents) are required.

Nassem et al. [5] constructs an FP-tree structure and mines frequent patterns by traversing the constructed FP-tree. The FP-tree structure is an extended prefix-tree structure involving crucial condensed information of frequent patterns. In addition, some features have been suggested that need to be added into XQuery in order to make the implementation of the First Frequent Pattern growth more efficient. In future work of paper it was planned to implement other standard data mining algorithms which can be expressed in XQuery to improve the performance of the results. This method is advantageous because, it doesn't generate any candidate items. It is disadvantageous because, it suffers from the issues of special and temporal locality issues.

Mishra et al. [9] developed the improved FP-tree. The FP-tree structure has sufficient information to mine complete frequent patterns. It consists of a prefix tree of frequent 1-itemset and a frequent-item header table. FP-growth has to scan the *TDB* twice to construct an FP-tree. The first scan of *TDB* retrieves a set of frequent items from the *TDB*. Then, the retrieved frequent items are ordered by descending order of their supports. The ordered list is called an F-list. In the second scan, a tree *T* whose root node *R* labelled with "null" is created. It will increases the mining efficiency and also takes less memory.

D. Jayalatchumy et al. [12] have compared various algorithms that have been used for HTML contents like images, audio, video etc. Also pros and cons of each algorithm is given.

K. Nethra et al. [13] provides a hybrid approach for web content extraction. A hybrid approach is proposed to extract main content from Web pages. A HTML Web page is converted to DOM tree and features are extracted and with the extracted features, rules are generated. Decision tree classification and Naïve Bayes classification are machine learning methods used for rules generation. By using the rules, noisy part in the Web page is discarded and informative content in the Web page is extracted. The performance of both decision tree classification and Naïve Bayes classification are measured with metrics like precision, recall, F-measure and accuracy.

3. Proposed Approach

In this hybrid approach, two well-known data mining algorithms are used. Our approach is applied in two steps-
Step 1: In the first step Decision tree is implemented to fetch the data from the web.
Step 2: In the second step well-known Apriori algorithm is used to remove redundancy and unimportant data. These algorithms are explained as under:-

Step 1: Decision Tree

A decision tree is used for decision making purpose. Decision tree has root and branch node. From the root node, users split each node recursively based on decision tree learning algorithm. The final result of decision tree consists of branches and each branch represents a possible scenario of decision and its consequences. [13]

Pseudocode of a Decision Tree Induction Algorithm [16]

Input:
- Data partition, D, which is a set of training tuples and their associated class labels.
- attribute_list, the set of candidate attributes.
- Attribute selection method, a procedure to determine the splitting criterion that best partitions the data tuples into individual classes. This criterion includes a splitting_attribute and either a splitting point or splitting subset.

Output:
A Decision Tree

Method:
1. create a node N;
2. if tuples in D are all of the same class, C then return N as leaf node labeled with class C;
3. if attribute_list is empty then return N as leaf node with labeled with majority class in D;|| majority voting
4. apply attribute_selection_method (D, attribute_list) to find the best splitting_criterion;
5. label node N with splitting_criterion;
6. if splitting_attribute is discrete-valued and multiway splits allowed then // no restricted to binary trees
7. attribute_list = splitting attribute; // remove splitting attribute
8. for each outcome j of splitting criterion // partition the tuples and grow subtrees for each partition. let Dj be the set of data tuples in D satisfying outcome j; // a partition

If
Dj is empty then attach a leaf labeled with the majority

class in D to node N; Else
attach the node returned by Generate decision tree (Dj, attribute list) to node N;
end for
9. return N;

Step 2: Apriori Algorithm

As the name implies, this algorithm uses prior knowledge about frequent itemset properties. It employs an iterative approach where k-itemsets are used to explore (k +1)-itemsets. To improve the efficiency of the generation of frequent itemsets, it uses an important property called the Apriori property which says that all nonempty subsets of a frequent itemset must also be frequent. The Apriori algorithm first computes the frequent 1- itemsets, L1. To find frequent 2-itemsets L2, a set of candidate 2-itemsets, C2, is generated by joining L1 with itself. [17]

Pseudocode of Apriori algorithm [17]

Procedure Apriori (T, minSupport)// T is the database and minSupport is the minimum Support.

1. L1= {frequent items};
2. for (k=2; L_{k-1}!=Ø; k++) {
3. C_k = candidate generated from L_{k-1} // that is product of L_{k-1} * L_{k-1} and eliminating any k-1 size itemset that is frequent
4. for each transaction t in database do {
   # increment the count of all candidates in C_k that are contained in t
5. L_k = candidates in C_k with minSupport
   } // end for each
   } // end for
6. Return $U_k L_k$;

}

In this approach, Apriori algorithm is used to remove the unimportant data from the web pages.

4. Methodology

It describes the process of fetching web contents like images and hyperlinks using the hybrid approach. At the end parameters like execution time in milliseconds, precision, recall, and f-measure are calculated.

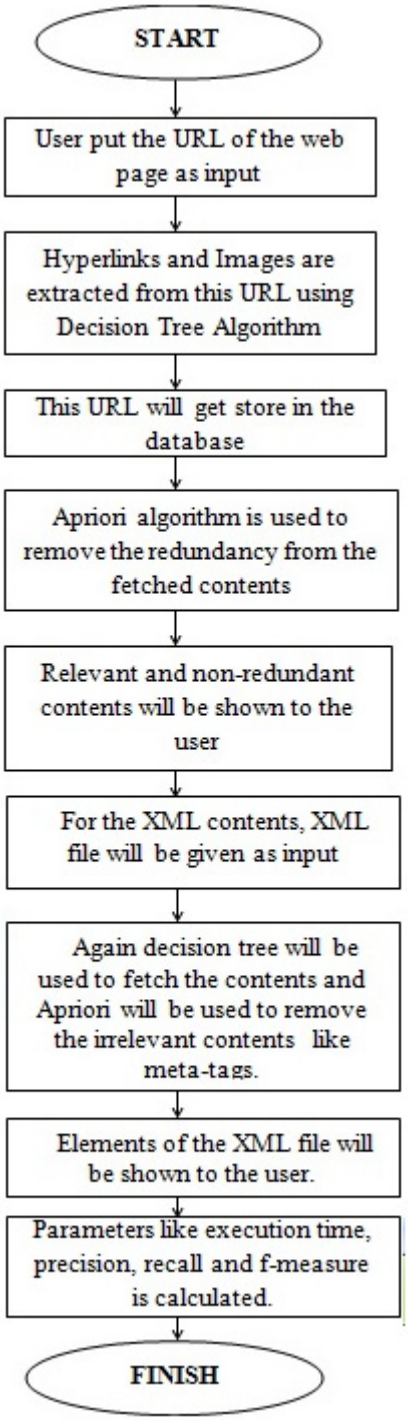

Figure 1: Methodology of proposed work

5. Results

We run the project on some of the top visited URLs like Yahoo, eBay, Facebook, Amazon etc. The output is shown as below. Results of Apriori algorithm are also shown independently. All the hyperlinks and images are dynamically fetched. XML elements are also fetched and execution time is calculated.

Figure 2: User input the URL

Figure 3: Hyperlinks are shown

Figure 4: Images are shown

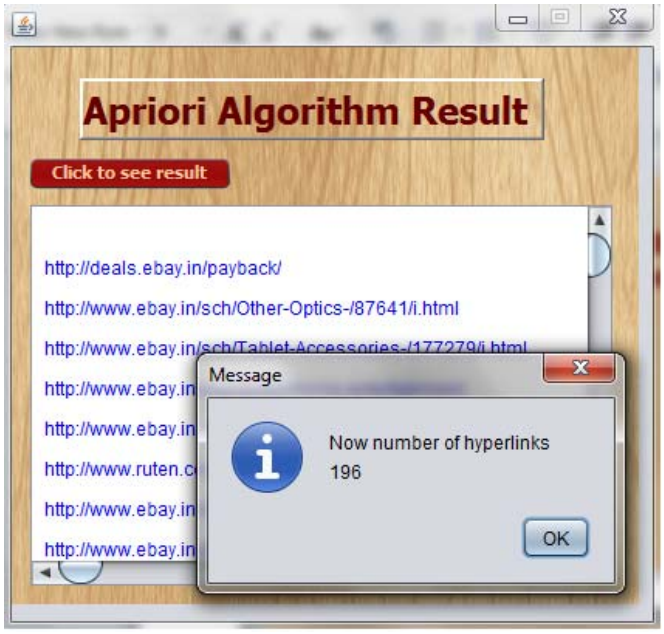

Figure 5: Result of Apriori

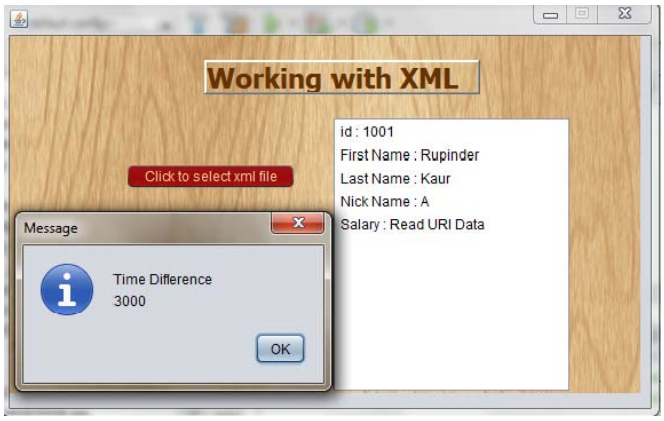

Figure 6: Elements of XML file is fetched and execution time is calculated

5.1 Parameter Analysis

Following is the table showing parameters like execution time, precision, recall, F-measure and G-measure. These parameters are defined are below-

Execution Time: is defined as the difference between the time when project starts its execution and time up to which hyperlinks and image links are fetched. It is calculated in milliseconds.

Precision: It is defined as the fraction of retrieved instances that are relevant. [18]

Recall: (also known as sensitivity) is the fraction of relevant instances that are retrieved. [18]

F-Measure: is the harmonic mean of precision and recall.
F-Measure= (2*Precision*Recall)/ (Precision + Recall)

G-Measure: is the geometric mean of precision and recall.
G-Measure=√ (Precision*Recall)

Table 1: Parameter's Result

ID	*URLs*	*Time Taken (MS)*	*Precision*	*Recall*	*F-Measure*	*G-Measure*
1	https://in.yahoo.com/?p=us	8192	0.5083	0.1311	0.2085	0.2581
2	http://www.jabong.com/	9916	0.2256	0.4594	0.3026	0.3219
3	https://www.facebook.com/	7908	0.1923	0.6	0.2912	0.3396
4	http://www.ebay.in/	7295	0.205	0.825	0.3285	0.4113
5	http://www.amazon.in/	7734	0.1746	0.5322	0.2629	0.3048
6	http://www.homeshop18.com/	7690	0.5652	0.8461	0.6777	0.6915
7	http://www.naturewallpaper.net/	7007	0.3548	0.8181	0.495	0.5388
8	http://rontalk.com/	8651	0.1941	0.9696	0.3234	0.4338
9	http://tanishq.co.in/Home	9253	0.1466	0.8181	0.495	0.5388
10	http://www.zigwheels.com/	8694	0.4043	0.7702	0.5303	0.5580

6. Conclusion

In this proposed approach, hyperlinks and images are fetched from the web using two well know data mining algorithms named Apriori and Decision Tree. These algorithms when applied individually previously gave more reliable results. So, that's why these algorithms are chosen and a combined approach is proposed. Here Decision Tree is used to retrieve the web contents and Apriori is used to remove the unimportant data like redundancy, tags etc. XML elements are also fetched from XML file. Execution time of this fetching is calculated. Step-wise methodology is explained and screen shots of running project are also provided. At the end, parameters for this hybrid approach are calculated which are execution time, precision, recall, F-measure and G-measure.

7. Future Scope

Though, our proposed approach implements basic classic algorithms which are Apriori and Decision Tree Induction algorithm. There are several modifications proposed to these algorithms like Apriori-TID, Hybrid Apriori [12] and for Decision Tree certain other algorithms are there like C4.5, CART which can be implemented in future to increase the efficiency of this hybrid approach. Also this approach runs best while using high speed internet, some methods can be proposed in future so that this approach can give its best at slow speed internet too.

References

[1] O. Etzioni, "The world wide Web: Quagmire or gold mine." Communications of the ACM, Vol. 39 No. 11, pp. 65-68, Nov. 1996.
[2] R. Kosala, H. Blockeel "Web mining research: A survey," ACM SIGKDD Explorations, Vol. 2 No. 1, pp. 1-15, June 2000.
[3] Qin Ding and Gnanasekaran Sundarraj, "Association Rule Mining from XML Data", Conference on Data Mining | DMIN'06|.
[4] Krishna Murthy. A, Suresha, "XML: URL Data Set Creation for Future Web Mining Research Avenues", International Journal of Computer Applications (0975 – 8887), 2011.
[5] Mohammad. Nassem, Prof. Sirish Mohan Dubey, "First Frequent Pattern-tree based XML pattern fragment growth method for Web Contents", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 1, January 2012.
[6] Ganesh Dhar, Govind Murari Upadhyay, "Web Mining: Concepts and Decision Making Aid". Volume 2, Issue 7, July 2012.
[7] Darshna Navadiya, Roshni Patel, "Web Content Mining Techniques- A Comprehensive Survey", IJERT, Vol. 1 Issue 10, December- 2012.
[8] Amit Kumar Mishra, Hitesh Gupta, "A Recent Review on XML data mining and FFP", International Journal of Engineering Research, Volume No.2, Issue No.1, pp : 07-12.
[9] Amit Kumar Mishra, Hitesh Gupta, "A Novel FP-Tree Algorithm for Large XML Data Mining", International Journal of Engineering Sciences and Research Technology, [Mishra, 2(3): March, 2013].
[10] Abdelhakim Herrouz, Chabane Khentout Mahieddine Djoudi, "Overview of Web Content Mining Tools", The International Journal of Engineering And Science, Volume 2, Issue 6, pp. 106-110, June 2013.
[11] Buhwan Jeong, Daewon Lee, Jaewook Lee, and Hyunbo Cho, "Towards XML Mining: The Role of Kernel Methods".
[12] D. Jayalatchumy, Dr. P.Thambidurai, "Web Mining Research Issues and Futur Directions – A Survey", IOSR Journal of ComputerEngineering (IOSR-JCE), Volume 14, Issue 3 (Sep. - Oct. 2013), pp. 20-27.
[13] K. Nethra, J. Anitha G. Thilagavathi, "Web Content
[14] Extraction Using Hybrid Approach", ICTACT Journal On Soft Computing, JAN 2014, VOLUME: 04, ISSUE: 02.
[15] http://en.wikipedia.org/wiki/Web_content.
[16] http://webdesign.about.com/od/content/qt/what-is-web-content.htm.
[17] http://www.tutorialspoint.com/data_mining/dm_dti.htm.
[18] http://en.wikipedia.org/wiki/Apriori_algorithm.
[19] http://en.wikipedia.org/wiki/Precision_and_recall.

Author Profile

Rupinder Kaur is currently completing Master of Technology from Sri Guru Granth Sahib World University, Fatehgarh Sahib. She has received the Bachelor of Technology degree in Information Technology from Baba Banda Singh Bahadur Engineering Technology, Fatehgarh Sahib in the year of 2008. She is currently writing thesis on "Web Mining".

Kamaljit Kaur is currently pursuing PhD in Data and Web Mining. She obtained her Master of Technology and Bachelor of Technology degrees in CSE. She has over 10 years of experience. Currently she is working as Assistant Professor at SGGSW University, Fatehgarh Sahib and guiding various M.Tech thesis in area of database security and data mining.